



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645



Lt. Shree Chimanbhai Shukla

B.C.A./B.SC.I.T.-SEMESTER-6

ANDROID

Shree H.N.Shukla College
2, Vaishali Nagar,
Nr. Amrapali Under Bridge,
Rajkot
Ph. (0281)2440478



Shree H.N.Shukla College
3, Vaishali Nagar,
Nr. Amrapali Under
Bridge,Rajkot.
Ph. (0281)2471645



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

UNIT – 1 INTRODUCTION TO KOTLIN PROGRAMMING

Q- 1 Explain Basics of Kotlin.

❖ What is Kotlin?

- Kotlin is a new open source programming language like Java, JavaScript, Python etc.
- It is a high level strongly statically typed language that combines functional and technical part in a same place.
- Currently, Kotlin mainly targets the Java Virtual Machine (JVM), but also compiles to JavaScript.
- Kotlin is influenced by other popular programming languages such as Java, C#, JavaScript.
- The syntax of Kotlin may not be exactly similar to Java Programming Language, however, internally Kotlin is dependent on the existing Java Class library to produce wonderful results for the programmers.
- Kotlin provides interoperability, code safety, and clarity to the developers around the world.
- Kotlin was developed and released by JetBrains in 2011.
- Kotlin is free, has been free and will remain free. It is developed under the Apache 2.0 license and the source code is available on GitHub

❖ Why Kotlin?

- Kotlin is getting high popularity among all level of programmers and it is used for:
 - ✓ Cross-platform Mobile applications.
 - ✓ Android Application Development.
 - ✓ Web Application Development
 - ✓ Server Side Applications
 - ✓ Desktop Application Development
 - ✓ Data science based applications



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- Kotlin works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.) and it's 100% compatible with Java.
- Kotlin is used by many large companies like Google, Netflix, Slack, Uber etc to develop their Android based applications.
- The most importantly, there are many companies actively looking for Kotlin developers, especially in the Android development space.

Q- 2 Explain Kotlin Operators.

- An operator is a symbol that tells the compiler to perform specific mathematical or logical manipulations.
 - Kotlin is rich in built-in operators and provide the following types of operators:
 1. Arithmetic Operators
 2. Relational Operators
 3. Assignment Operators
 4. Unary Operators
 5. Logical Operators
 6. Bitwise Operations
1. Arithmetic Operators
- Kotlin arithmetic operators are used to perform basic mathematical operations such as addition, subtraction, multiplication and division etc.

Operator	Name	Description	Example
+	Addition	Adds together two values	$x + y$
-	Subtraction	Subtracts one value from another	$x - y$
*	Multiplication	Multiplies two values	$x * y$
/	Division	Divides one value by another	x / y
%	Modulus	Returns the division remainder	$x \% y$



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

2. Relational Operators

- Kotlin relational (comparison) operators are used to compare two values, and returns a **Boolean** value: either **true** or **false**.

Operator	Name	Example
>	greater than	$x > y$
<	less than	$x < y$
>=	greater than or equal to	$x >= y$
<=	less than or equal to	$x <= y$
==	is equal to	$x == y$
!=	not equal to	$x != y$

3. Assignment Operators

- Kotlin assignment operators are used to assign values to variables.
- Following is an example where we used assignment operator = to assign a values into two variables:

Operator	Example	Expanded Form
=	$x = 10$	$x = 10$
+=	$x += 10$	$x = x + 10$
-=	$x -= 10$	$x = x - 10$
*=	$x *= 10$	$x = x * 10$
/=	$x /= 10$	$x = x / 10$
%=	$x \% = 10$	$x = x \% 10$



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

4. Unary Operators

- The unary operators require only one operand; they perform various operations such as incrementing/decrementing a value by one, negating an expression, or inverting the value of a boolean.

Operator	Name	Example
+	unary plus	+x
-	unary minus	-x
++	increment by 1	++x
--	decrement by 1	--x
!	inverts the value of a Boolean	!x

5. Logical Operators

- Kotlin logical operators are used to determine the logic between two variables or values:
- Following is the list of Kotlin Logical Operators:

Operator	Name	Description	Example
&&	Logical and	Returns true if both operands are true	x && y
	Logical or	Returns true if either of the operands is true	x y
!	Logical not	Reverse the result, returns false if the operand is true	!x

6. Bitwise Operations



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- Kotlin does not have any bitwise operators but Kotlin provides a list of helper functions to perform bitwise operations.
- Following is the list of Kotlin Bitwise Functions:

Function	Description	Example
shl (bits)	signed shift left	x.shl(y)
shr (bits)	signed shift right	x.shr(y)
ushr (bits)	unsigned shift right	x.ushr(y)
and (bits)	bitwise and	x.and(y)
or (bits)	bitwise or	x.or(y)
xor (bits)	bitwise xor	x.xor(y)
inv()	bitwise inverse	x.inv()

Q- 3 Explain Kotlin Elvis Operator Priority.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

Precedence

Precedence	Title	Symbols
Highest	Postfix	++, --, ., ?., ?
	Prefix	-, +, ++, --, !, labelDefinition
	Type RHS	:, as, as?
	Multiplicative	*, /, %
	Additive	+, -
	Range	..
	Infix function	SimpleName
	Elvis	?:
	Named checks	in, !in, is, !is
	Comparison	<, >, <=, >=
	Equality	==, \!==
	Conjunction	&&
	Disjunction	
Lowest	Assignment	=, +=, -=, *=, /=, %=

- Given the table above, it's apparent that + and - operators have higher precedence than ?: . So, `x ?: 0 + y` is evaluated as `x ?: (0 + y)`. That's why, the code snippet above prints out 1.

Q- 4 Explain Kotlin Decision Making.

- There are case where we want a block of code to be executed when some condition is satisfied.
- In Kotlin, we have Decision Making Statement that enable computer to decide which block of code to be execute based on some conditional choices. Decision making statement statements is also referred to as selection statements.

STATEMENT	DESCRIPTION
if Statements	Block of statement executed only when specified test expression is true.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

if else Statements	When we want to execute some block of code if a condition is true and another block of code if a condition is false, In such a case we use if...else statement.
if..else..if Ladder	It tests set of conditions in sequence.
Nested Statements	When there is an if statement inside another if statement then it is known as nested if else.

✓ KOTLIN - IF...ELSE EXPRESSION

- Kotlin **if...else** expressions works like an **if...else** expression in any other Modern Computer Programming. So let's start with our traditional if...else statement available in Kotlin.
- **Syntax**
- The syntax of a traditional **if...else** expression is as follows:

```
if (condition) {  
    // code block A to be executed if condition is true  
} else {  
    // code block B to be executed if condition is false  
}
```

- Here **if** statement is executed and the given **condition** is checked. If this condition is evaluated to **true** then code block A is executed, otherwise program goes into else part and code block B is executed.

- **Syntax**

```
val result = if (condition) {  
    // code block A to be executed if condition is true  
} else {  
    // code block B to be executed if condition is false  
}
```

- If you're using if as an expression, for example, for returning its value or assigning it to a variable, the else branch is mandatory.

➤ Kotlin if...else...if Ladder



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- You can use **else if** condition to specify a new condition if the first condition is false.
- **Syntax**

```
if (condition1) {  
    // code block A to be executed if condition1 is true  
} else if (condition2) {  
    // code block B to be executed if condition2 is true  
} else {  
    // code block C to be executed if condition1 and condition2 are false  
}
```

➤ Kotlin Nested if Expression

- Kotlin allows to put an if expression inside another if expression. This is called nested if expression
- **Syntax**

```
if(condition1) {  
    // code block A to be executed if condition1 is true  
    if( condition2) {  
        // code block B to be executed if condition2 is true  
    }else{  
        // code block C to be executed if condition2 is fals  
    }  
} else {  
    // code block D to be executed if condition1 is false  
}
```

➤ When Expression

- Consider a situation when you have large number of conditions to check. Though you can use if..else if expression to handle the situation, but Kotlin provides when expression to handle the situation in nicer way. Using when expression is far easy and more clean in comparison to writing many if...else if expressions. Kotlin when expression evaluates a section of code among many alternatives as explained in below example.
- Kotlin when matches its argument against all branches sequentially until some branch condition is satisfied.
- Example :

```
fun main(args: Array<String>) {
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
val day = 2
```

```
when (day) {  
    1 -> println("Monday")  
    2 -> println("Tuesday")  
    3 -> println("Wednesday")  
    4 -> println("Thursday")  
    5 -> println("Friday")  
    6 -> println("Saturday")  
    7 -> println("Sunday")  
    else -> println("Invalid day.")  
}
```

Q- 5 Explain Kotlin Loop Control.

1. Kotlin while Loop

- The **while loop** is used to iterate a part of program several time. Loop executed the block of code until the condition has true.
- Kotlin while loop is similar to Java while loop.
- **Syntax**

```
while(condition)  
{  
    //body of loop  
}
```

2. Kotlin do-while Loop

- The do-while loop is similar to while loop except one key difference. A do-while loop first execute the body of do block after that it check the condition of while.
- As a do block of do-while loop executed first before checking the condition, do-while loop execute at least once even the condition within while is false.
- The while statement of do-while loop end with ";" (semicolon).

- **Syntax**

```
do
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
}  
    //body of do block  
}  
while(condition);
```

3. Kotlin for Loop

- Kotlin for loop is used to iterate a part of program several times.
- It iterates through arrays, ranges, collections, or anything that provides for iterate.
- Kotlin for loop is equivalent to the foreach loop in languages like C#.
- Syntax:

```
for (item in collection)  
{  
    //body of loop  
}
```

Q- 5 Explain Kotlin Collections.

- Collections are a common concept for most programming languages. A collection usually contains a number of objects of the same type and Objects in a collection are called elements or items.
- The Kotlin Standard Library provides a comprehensive set of tools for managing collections.
- The following collection types are relevant for Kotlin:
 1. **Kotlin List** - List is an ordered collection with access to elements by directories. Elements can occur more than once in a list.
 2. **Kotlin Set** - Set is a collection of unique elements which means a group of objects without repetitions.
 3. **Kotlin Map** - Map (or dictionary) is a set of key-value pairs. Keys are unique, and each of them maps to exactly one value.

Kotlin Collection Types

- Kotlin provides the following types of collection:
 1. Collection or Immutable Collection
 2. Mutable Collection



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

1. Kotlin Immutable Collection

- Immutable Collection or simply calling a Collection interface provides read-only methods which means once a collection is created, we can not change it because there is no method available to change the object created.

Collection Types	Methods of Immutable Collection
List	listOf()
Map	mapOf()
Set	setOf()

Example

```
fun main()
{
    val numbers = listOf("one", "two", "three", "four")
    println(numbers)
}
```

2. Kotlin Mutable Collection

- Mutable collections provides both read and write methods.

Collection Types	Methods of Immutable Collection
List	ArrayList<T>() arrayListOf() mutableListOf()



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

Map	HashMap hashMapOf() mutableMapOf()
Set	hashSetOf() mutableSetOf()

Example

```
fun main() {  
    val numbers = mutableListOf("one", "two", "three", "four")  
    numbers.add("five")  
    println(numbers)  
}
```

Q- 6 Explain Kotlin Functions.

- A function is a block of code which only runs when it is called.
- You can pass data, known as parameters, into a function.
- Functions are used to perform certain actions, and they are also known as methods.
- There are two types of functions depending on whether it is available in standard library or defined by user.
 1. Standard library function
 2. User defined function

1. Standard library function

- Kotlin Standard library function is built-in library functions which are implicitly present in library and available for use.

2. User defined function

- User defined function is a function which is created by user. User defined function takes the parameter(s), perform an action and return the result of that action as a value.
- Kotlin functions are declared using the fun keyword. For example:



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
fun functionName()  
{  
    // body of function  
}
```

- We have to call the function to run codes inside the body of the function.

```
functionName()
```

➤ Kotlin Recursion Function

- Recursion function is a function which calls itself continuously. This technique is called recursion.

• Syntax

```
fun functionName()  
{  
    .. . . .  
    functionName() //calling same function  
}
```

➤ Kotlin Tail Recursion

- Tail recursion is a recursion which performs the calculation first, then makes the recursive call. The result of current step is passed into the next recursive call.
- Tail recursion follows one rule for implementation. This rule is as follow:
- The recursive call must be the last call of the method. To declare a recursion as tail recursion we need to use tailrec modifier before the recursive function.

➤ Kotlin Default Argument

- Kotlin provides a facility to assign *default argument* (parameter) in a function definition.
- If a function is called without passing any argument than default argument are used as parameter of function definition.
- And when a function is called using argument, than the passing argument is used as parameter in function definition.

➤ Kotlin Named Argument



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- A named argument is an argument in which we define the name of argument in the function call.
- The name defined to argument of function call checks the name in the function definition and assign to it.

➤ Kotlin Lambda Function

- Lambda is a function which has no name. Lambda is defined with a curly braces `{ }` which takes *variable as a parameter* (if any) and body of function. The *body of function* is written after variable (if any) followed by `->` operator.

Syntax of lambda

```
{ variable -> body_of_function }
```

➤ Higher order function

- **High order function (Higher level function)** is a function which accepts function as a parameter or returns a function or can do both.
- Means, instead of passing Int, String, or other types as a parameter in a function we can pass a function as a parameter in other function.

➤ Inline Function

- An **inline function** is declare with a keyword inline. The use of inline function enhances the performance of higher order function. The inline function tells the compiler to copy parameters and functions to the call site.
- The virtual function or local function cannot be declared as **inline**. Following are some expressions and declarations which are not supported anywhere inside the inline functions:
 - Declaration of local classes
 - Declaration of inner nested classes
 - Function expressions
 - Declarations of local function
 - Default value for optional parameters

Q- 7 Kotlin – What is OOP?

- OOP stands for Object-Oriented Programming.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- Procedural programming is about writing procedures or methods that perform operations on the data, while object-oriented programming is about creating objects that contain both data and methods.
- Object-oriented programming has several advantages over procedural programming:
 - OOP is faster and easier to execute.
 - OOP provides a clear structure for the programs.
 - OOP helps to keep the Kotlin code DRY "Don't Repeat Yourself", and makes the code easier to maintain, modify and debug.
 - OOP makes it possible to create full reusable applications with less code and shorter development time.

➤ Kotlin Classes

- A class is a blueprint for the objects which defines a template to be used to create the required objects.
- Classes are the main building blocks of any Object Oriented Programming language. A Kotlin class is defined using the class keyword. Following is the syntax to create a Kotlin Class:

```
class ClassName
{
    // Class Header
    // Variables or data members
    // Member functions or Methods
    ...
}
```

- By default, Kotlin classes are public and we can control the visibility of the class members using different modifiers that we will learn in Visibility Modifiers.

➤ Kotlin Objects

- The objects are created from the Kotlin class and they share the common properties and behaviours defined by a class in form of data members (properties) and member functions (behaviours) respectively.
- The syntax to declare an object of a class is:

```
var varName = ClassName()
```




SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

Q- 8 Explain Kotlin visibility Modifier.

- **Visibility modifiers** are the keywords which are used to restrict the use of class, interface, methods, and property of Kotlin in the application. These modifiers are used at multiple places such as class header or method body.
- In Kotlin, visibility modifiers are categorized into four different types:
 1. public
 2. protected
 3. internal
 4. private

1. public modifier

- A **public** modifier is accessible from everywhere in the project. It is a default modifier in Kotlin.
- If any class, interface etc. are not specified with any access modifier then that class, interface etc. are used in public scope.

2. protected modifier

- A **protected** modifier with class or interface allows visibility to its class or subclass only.
- A protected declaration (when overridden) in its subclass is also protected modifier unless it is explicitly changed.

3. internal modifier

- The internal modifiers are newly added in Kotlin, it is not available in Java. Declaring anything makes that field marked as internal field.
- The internal modifier makes the field visible only inside the module in which it is implemented.

4. private modifier

- A **private** modifier allows the declaration to be accessible only within the block in which properties, fields, etc. are declare.
- The private modifier declaration does not allow to access the outside the scope.
- A private package can be accessible within that specific file.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Q- 9 Kotlin – What is Inheritance?

- Inheritance is an important feature of object oriented programming language. Inheritance allows to inherit the feature of existing class (or base or parent class) to new class (or derived class or child class).
- The main class is called super class (or parent class) and the class which inherits the superclass is called subclass (or child class). The subclass contains features of superclass as well as its own.
- The concept of inheritance is allowed when two or more classes have same properties. It allows code reusability. A derived class has only one base class but may have multiple interfaces whereas a base class may have one or more derived classes.
- In Kotlin, the derived class inherits a base class using: operator in the class header (after the derive class name or constructor)

```
open class Base(p: Int)
{
    ----
}
class Derived(p: Int) : Base(p)
{
    ----
}
```

➤ Kotlin open keyword

- As Kotlin classes are **final** by default, they cannot be inherited simply. We use the **open** keyword before the class to inherit a class and make it to non-final.

Q- 10 Kotlin – What is abstract class?

- A class which is declared with abstract keyword is known as abstract class.
- An abstract class cannot be instantiated.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- Means, we cannot create object of abstract class.
- The method and properties of abstract class are non-abstract unless they are explicitly declared as abstract.

Declaration of abstract class

```
abstract class A
{
    var x = 0
    abstract fun doSomething()
}
```

- Abstract classes are partially defined classes, methods and properties which are no implementation but must be implemented into derived class.
- If the derived class does not implement the properties of base class then is also meant to be an abstract class.
- Abstract class or abstract function does not need to annotate with open keyword as they are open by default.
- Abstract member function does not contain its body.
- The member function cannot be declared as abstract if it contains in body in abstract class.

Q- 11 Kotlin – What is Interface?

- An interface can be implemented by a class in order to use its defined functionality.
- The keyword “interface” is used to define an interface in Kotlin as shown in the following piece of code.

```
interface ExampleInterface {
    var myVar: String // abstract property
    fun absMethod() // abstract method
    fun sayHello() = "Hello there" // method with default implementation
}
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- In the above example, we have created one interface named as “ExampleInterface” and inside that we have a couple of abstract properties and methods all together.
- We can implement multiple abstract methods of different interfaces in same class. All the abstract methods must be implemented in subclass.
- The other non-abstract methods of interface can be called from derived class.
- For example, creating two interface MyInterface1 and MyInterface2 with abstract methods doSomething() and absMethod() respectively.
- These abstract methods are overridden in derive class MyClass.

```
interface MyInterface1 {
    fun doSomething()
}
interface MyInterface2 {
    fun absMethod()
}
class MyClass : MyInterface1, MyInterface2 {
    override fun doSomething() {
        println("overriding doSomething() of MyInterface1")
    }

    override fun absMethod() {
        println("overriding absMethod() of MyInterface2")
    }
}
fun main(args: Array<String>) {
    val myClass = MyClass()
    myClass.doSomething()
    myClass.absMethod()
}
```

Q- 11 Explain super and this keyword in details.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

➤ Super Keyword

- The super keyword in kotlin is a reference variable which is used to refer immediate parent class object.
- Whenever you create the instance of subclass, an instance of parent class is created implicitly which is referred by super reference variable.
- Usage of kotlin super Keyword
 1. super can be used to refer immediate parent class instance variable.
 2. super can be used to invoke immediate parent class method.
 3. super() can be used to invoke immediate parent class constructor.

➤ this Keyword

- The this keyword refers to the current object in a method or constructor.
- The most common use of the this keyword is to eliminate the confusion between class attributes and parameters with the same name (because a class attribute is shadowed by a method or constructor parameter).
- this can also be used to:
 1. Invoke current class constructor
 2. Invoke current class method
 3. Return the current class object
 4. Pass an argument in the method call
 5. Pass an argument in the constructor call

UNIT – 2 INTRODUCTION TO ANDROID & ANDROID APPLICATION DESIGN

Importance of Mobile

- Commonplace problem solved with a one-button speed dial or a sample text message like “WRU?”
- Our mobile keep us safe and connected. Now We roam around freely and connected to friends, family.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Handset Manufacturers

- Before Android, mobile developers faced many roadblocks when it came to writing application. Building the better application, the unique application, the competing application, the hybrid application and incorporating many common tasks such as messaging and calling in a familiar way were often unrealistic goals.

First Generation Mobile

- These early phones were flawed, but they did something important – They changed the way people thought about communication. As mobile phone prices dropped, batteries improved, and reception areas grew, more and more people began carrying these handy devices.

WAP(Wireless Application Protocol)

- Commercializing WAP application was difficult and there was no built-in billing mechanism. Some of the most popular commercial WAP application that emerged during this time were simple wallpaper and ringtone catalogues that enabled users to personalize their phones for the first time. For example, a user browsed a WAP site and requested a specific item.
- He filled out a simple order form with his phone number and his handset model. It was up to the content provider to deliver an image or audio file compatible with the given phone.
- Payment and verification were handled through various premium priced delivery mechanisms such as Short Message Service (SMS), Enhanced Messaging Service (EMS), and Multimedia Messaging Service (MMS).

Different Platforms

- A variety of different proprietary platforms emerged—and developers are still actively creating applications for them.
 - Some smartphone devices ran Palm OS (now Garnet OS) and RIM BlackBerry OS.
 - Sun Microsystems took its popular Java platform and J2ME emerged (now known as Java Micro Edition [Java ME]).
 - Chipset maker Qualcomm developed and licensed its Binary Runtime Environment for Wireless (BREW).
 - **Other platforms, such as Symbian OS, were developed by handset manufacturers such as Nokia, Sony Ericsson, Motorola, and Samsung.**
 - **The Apple iPhone OS (OS X iPhone) joined the ranks in 2008.**



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Forming the Open Handset Alliance

- The Open Handset Alliance (OHA) was formed in November 2007, to answer that very question. The OHA is a business alliance comprised of many of the largest and most successful mobile companies on the planet. Its members include chip makers, handset manufacturers, software developers, and service providers. The entire mobile supply chain is well represented.
- **Andy Rubin** has been credited as the father of the Android platform. His company, Android Inc., was **acquired (Purchased) by Google in 2005**.
- Working together, OHA members, including Google, began developing a non-proprietary open standard platform based upon technology developed at Android Inc.
- that would aim to alleviate(improve) the aforementioned problems hindering the mobile community.
- The result is the Android project. To this day, most android platform development is completed by Rubin's team at Google, where he acts as VP of Engineering and manages the Android platform roadmap
- Google hosts the Android open source project and provides online Android documentation, tools, forums, and the Software Development Kit (SDK) for developers. All major Android news originates at Google.

Sr.no	Questions	Answer
1	OHA full form is ____	Open Handset Alliance
2	OHA formed in	November 2007
3	WAP full form is_	Wireless Application Protocol



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Android Platform Differences

Android is hailed as “the first complete, open, and free mobile platform”:

- ✓ **Complete:** The designers took a comprehensive approach when they developed the Android platform. They began with a secure operating system and built a robust software framework on top that allows for rich application development opportunities.
- ✓ **Open:** The Android platform is provided through open source licensing. Developers have unprecedented access to the handset features when developing applications.
- ✓ **Free:** Android applications are free to develop. There are no licensing or royalty fees to develop on the platform. No required membership fees. No required testing fees. No required signing or certification fees. Android applications can be distributed and commercialized in a variety of ways.

Latest version of Android



Figure 1.5 The Android mascot and logo.



Figure 1.6 Some Android SDKs and their codenames.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Free and Open Source

Android is an open source platform. Neither developers nor handset manufacturers pay royalties or license fees to develop for the platform.

Familiar and Inexpensive Development Tools

Unlike some proprietary platforms that require developer registration fees, vetting, and expensive compilers, there are no upfront costs to developing Android applications.

Freely Available Software Development Kit

The Android SDK and tools are freely available. Developers can download the Android SDK from the Android website after agreeing to the terms of the Android Software Development Kit License Agreement.

Familiar Language, Familiar Development Environments

Many developers choose the popular and freely available Eclipse IDE to design and develop Android applications. Android applications can be developed on the following operating systems:

- ✓ Windows XP (32-bit) or Vista (32-bit or 64-bit)
- ✓ Mac OS X 10.5.8 or later (x86 only)
- ✓ Linux (tested on Linux Ubuntu 8.04 LTS, Hardy Heron)

Reasonable Learning Curve for Developers

Android applications are written in a well-respected programming language: **Java**.

The Android application framework includes traditional programming constructs, such as threads and processes and specially designed data structures to encapsulate objects commonly used in mobile applications.

A “Free Market” for Applications

Android developers can distribute their applications to users in a variety of ways:

- ✓ Google developed the Android Market (see Figure 1.7), a generic Android application store with a revenue-sharing model.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645



- ✓ Handango.com added Android applications to its existing catalogue using their billing models and revenue-sharing model.
- ✓ Developers can come up with their own delivery and payment mechanisms.

The Android Platform

- ✓ Android is an operating system and a software platform upon which applications are developed.
- ✓ A core set of applications for everyday tasks, such as web browsing and email, are included on Android handsets.
- ✓ As a product of the OHA's vision for a robust and open source development environment for wireless, Android is an emerging mobile development platform.
- ✓ The platform was designed for the sole purpose of encouraging a free and open market that all mobile applications phone users might want to have and software developers might want to develop.

Android's Underlying Architecture

- ✓ The Android platform is designed to be more fault-tolerant than many of its predecessors.
- ✓ The handset runs a Linux operating system upon which Android applications are executed in a secure fashion. Each Android application runs in its own virtual machine

The Linux Operating System

The Linux 2.6 kernel handles core system services and acts as a hardware abstraction layer (HAL) between the physical hardware of the handset and the Android software stack.

Some of the core functions the kernel handles include

- ✓ Enforcement of application permissions and security
- ✓ Low-level memory management
- ✓ Process management and threading
- ✓ The network stack
- ✓ Display, keypad input, camera, Wi-Fi, Flash memory, audio, and binder (IPC) driver access

Sr.no Questions

Answer



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

1	Android is ____	Framework
2	Android work based on	Linux
3	Full form of DVM	Dalvik Virtual Machine



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

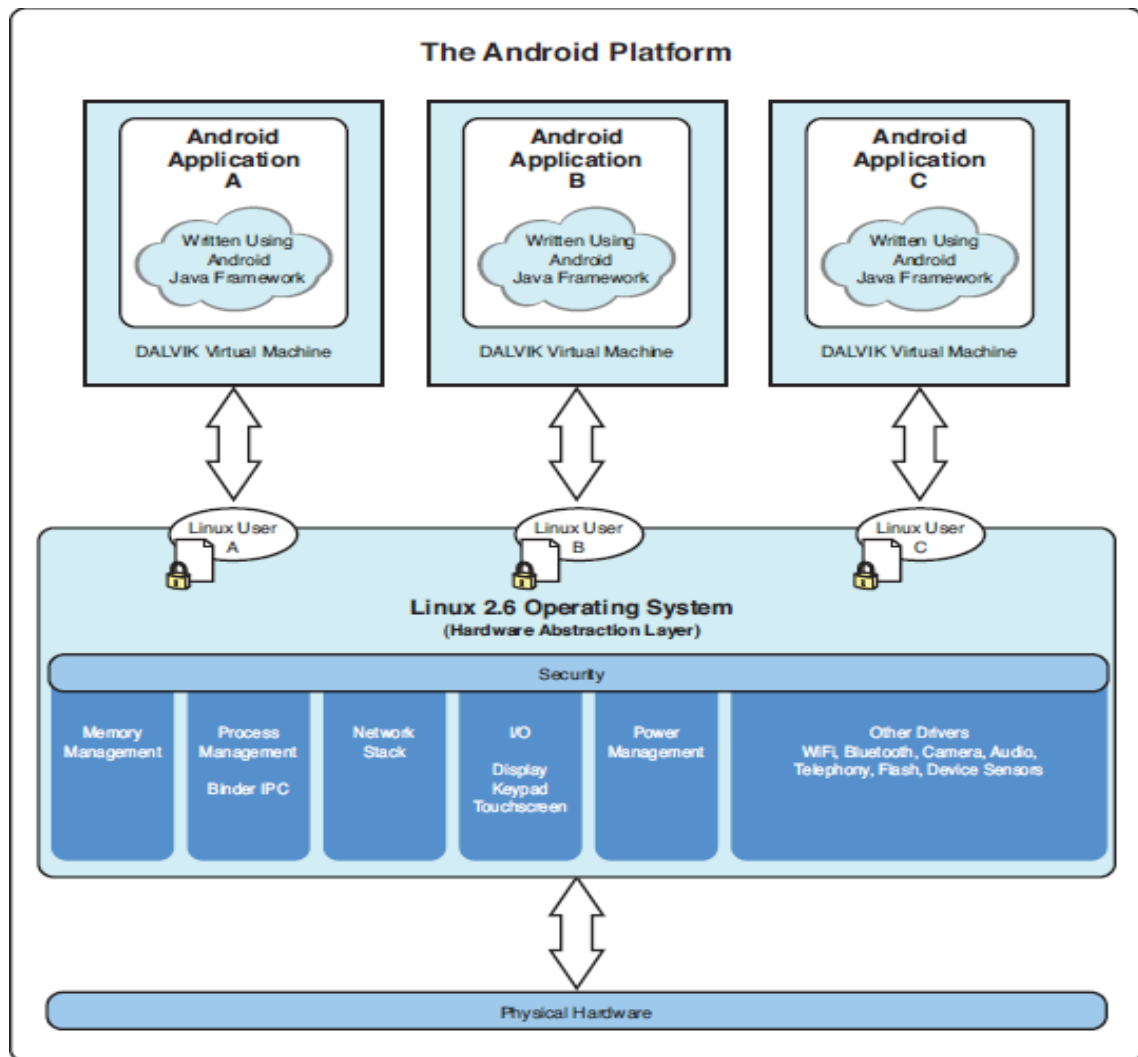


Diagram of Android platform architecture

Android Application Runtime Environment

- ✓ Each Android application runs in a separate process, with its own instance of the Dalvik virtual machine (VM). Based on the Java VM, the Dalvik design has been optimized for mobile devices.
- ✓ The Dalvik VM has a small memory footprint, and multiple instances of the Dalvik VM can run concurrently on the handset.

Security and Permissions

- ✓ The integrity of the Android platform is maintained through a variety of



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

security measures.

- ✓ These measures help ensure that the user's data is secure



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Applications as Operating System Users

- ✓ When an application is installed, the operating system creates a new user profile associated with the application. Each application runs as a different user, with its own private files on the file system, a user ID, and a secure operating environment.

Explicitly Defined Application Permissions

- ✓ Applications might also enforce their own permissions by declaring them for other applications to use. The application can declare any number of different permission types, such as read-only or read-write permissions

Application Signing for Trust Relationships

- ✓ All Android applications packages are signed with a certificate, so users know that the application is authentic.
- ✓ The private key for the certificate is held by the developer.

Marketplace Developer Registration

- ✓ To publish applications on the popular Android Market, developers must create a developer account. The Android Market is managed closely and no malware is tolerated.

Developing Android Applications

- ✓ The Android SDK provides an extensive set of application programming interfaces (APIs) that is both modern and robust. Android handset core system services are exposed and accessible to all applications.

Android Programming Language Choices

- ✓ Android applications are written in Java (see Figure 1.9). For now, the Java language is the developer's only choice on the Android platform.
- ✓ There has been some speculation that other programming languages, such as C++, might be added in future versions of Android.

Commonly Used Packages

- ✓ Developers use familiar class libraries exposed through Android's Java packages to perform common tasks such as graphics, database access, network access, secure communications, and utilities (such as XML parsing).
- ✓ The Android packages include support for



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

-
- ❖ Common user interface widgets (Buttons, Spin Controls, Text Input)
 - ❖ User interface layout
 - ❖ Secure networking and web browsing features (SSL, WebKit)
 - ❖ Structured storage and relational databases (SQLite)
 - ❖ Powerful 2D and 3D graphics (including SGL and OpenGL ES)
 - ❖ Audio and visual media formats (MPEG4, MP3, Still Images)
 - ❖ Access to optional hardware such as location-based services (LBS), Wi-Fi, Bluetooth, and hardware sensors

Android Application Framework

- ✓ The Android application framework provides everything necessary to implement your average application.
- ✓ The Android application lifecycle involves the following key components:
 - Activities are functions the application performs.
 - Groups of views define the application's layout.
 - Intents inform the system about an application's plans.
 - Services allow for background processing without user interaction.
 - Notifications alert the user when something interesting happens.
- ✓ Android applications can interact with the operating system and underlying hardware using a collection of managers.

What is Android?

1.1. The Android operating system

- ✓ Android is an operating system based on the Linux kernel. The project responsible for developing the Android system is called the *Android Open Source Project (AOSP)* and is primarily lead by Google.
- ✓ The Android system supports background processing, provides a rich user interface library, supports 2-D and 3-D graphics using the OpenGL-ES (short OpenGL) standard and grants access to the file system as well as an embedded SQLite database.
- ✓ An Android application typically consists of different visual and non visual components and can reuse components of other applications.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

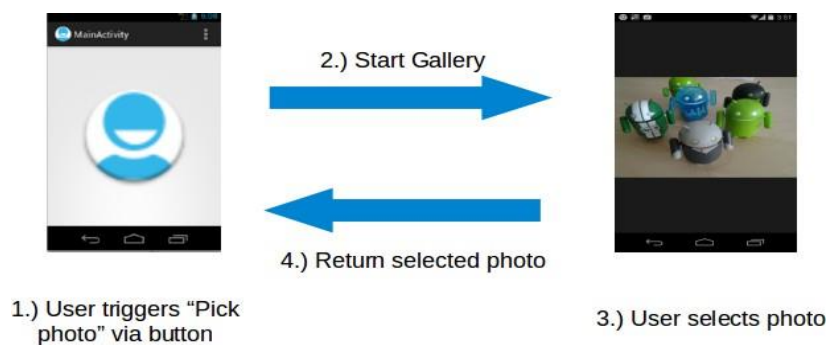
(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

1.2. Task

- ✓ In Android the reuse of other application components is a concept known as *task*. An application can access other Android components to achieve a task. For example, from a component of your application you can trigger another component in the Android system, which manages photos, even if this component is not part of your application.
- ✓ In this component you select a photo and return to your application to use the selected photo.
- ✓ Such a flow of events is depicted in the following graphic.



1.3. Android platform components

- ✓ The Android system is a full software stack, which is typically divided into the four areas as depicted in the following graphic



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Applications

Home, Contacts, Phone, Browser, ...

Application Framework

Managers for Activity, Window, Package, ...

Libraries

SQLite, OpenGL, SSL, ...

Runtime

Dalvik VM, Core libs

Linux Kernel

Display, camera, flash, wifi, audio, IPC (binder), ...



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- ✓ The levels can be described as:
 - Applications - The Android Open Source Project contains several default application, like the Browser, Camera, Gallery, Music, Phone and more.
 - Application framework - An API which allows high-level interactions with the Android system from Android applications.
 - Libraries and runtime - The libraries for many common functions (e.g.: graphic rendering, data storage, web browsing, etc.) of the Application Framework and the Dalvik runtime, as well as the core Java libraries for running Android applications.
 - Linux kernel - Communication layer for the underlying hardware.
- ✓ The Linux kernel, the libraries and the runtime are encapsulated by the application framework. The Android application developer typically works with the two layers ontop to create new Android applications.

1.4. Google Play

- ✓ Google offers the Google Play service, a marketplace in which programmers can offer their Android applications to Android users. Customers use the *Google Play* application which allows them to buy and install applications from the Google Play service.
- ✓ Google Play also offers an update service. If a programmer uploads a new version of his application to Google Play, this service notifies existing users that an update is available and allows them to install the update.
- ✓ Google Play provides access to services and libraries for Android application programmers, too. For example, it provides a service to use and display Google Maps and another to synchronize the application state between different Android installations.
- ✓ Providing these services via Google Play has the advantage that they are available for older Android releases and can be updated by Google without the need for an update of the Android release on the phone.

Sr.no	Questions	Answer
1	Is android support java programming?	Yes
2	Is android support xml?	Yes
3	From where user can download applications?	Play Store



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

2. Android Development Tools

2.1. Android SDK

- ✓ The Android Software Development Kit (Android SDK) contains the necessary tools to create, compile and package Android applications.
- ✓ Most of these tools are command line based. The primary way to develop Android applications is based on the Java programming language.

2.2. Android debug bridge (adb)

- ✓ The Android SDK contains the Android debug bridge (adb), which is a tool that allows you to connect to a virtual or real android device, for the purpose of managing the device or debugging your application.

2.3. Android Developer Tools and Android Studio

- ✓ Google provides two integrated development environments (IDEs) to develop new applications.
- ✓ The *Android Developer Tools* (ADT) are based on the Eclipse IDE. ADT is a set of components (plug-ins), which extend the Eclipse IDE with Android development capabilities.
- ✓ Google also supports an IDE called *Android Studio* for creating Android applications. This IDE is based on the IntelliJ IDE.
- ✓ Both IDEs contain all required functionality to create, compile, debug and deploy Android applications. They also allow the developer to create and start virtual Android devices for testing

2.4. Dalvik Virtual Machine

- ✓ Currently Android versions use the Dalvik virtual machine. The latest Android versions introduced a new runtime the Android RunTime.

2.5. Android RunTime (ART)

- ✓ With Android 4.4, Google introduced the *Android RunTime* (ART) as optional runtime for Android 4.4. It is used as default runtime for all Android versions after 4.4.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

-
- ✓ ART uses Ahead of Time compilation. During the deployment process of an application on an Android device, the application code is translated into machine code.
 - ✓ This results in approx. 30% larger compile code, but allows faster execution from the beginning of the application.
 - ✓ This also saves battery life, as the compilation is only done once, during the first start of the application.
 - ✓ The dex2oat tool takes the .dex file created by the Android tool change and compile that into an Executable and Linkable Format (ELF file).
 - ✓ This file contains the dex code, compiled native code and meta-data. Keeping the .dex code allows that existing tools still work.
 - ✓ The garbage collection in ART has been optimized to reduce times in which the application freezes.

2.6. How to develop Android applications

- ✓ Android applications are primarily written in the Java programming language.
- ✓ During development the developer creates the Android specific configuration files and writes the application logic in the Java programming language.
- ✓ The ADT or the Android Studio tools convert these application files, transparently to the user, into an Android application.
- ✓ When developers trigger the deployment in their IDE, the whole Android application is compiled, packaged, deployed and started.

2.7. Conversion process from source code to Android application

- ✓ The Java source files are converted to Java class files by the Java compiler.
- ✓ The Android SDK contains a tool called dx which converts Java class files into a .dex (Dalvik Executable) file.
- ✓ All class files of the application are placed in this .dex file. During this conversion process redundant information in the class files are optimized in the .dex file.
- ✓ For example, if the same String is found in different class files, the .dex file contains only one reference of this String.
- ✓ These .dex files are therefore much smaller in size than the corresponding class files.
- ✓ The .dex file and the resources of an Android project, e.g., the images and XML files, are packed into an .apk (Android Package) file. The program aapt (Android Asset Packaging Tool) performs this step.
- ✓ The resulting .apk file contains all necessary data to run the Android application and can be deployed to an Android device via the adb tool.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Sr.no	Questions	Answer
1	Full form of adb ____	Android Debugging Bridge
2	Full form of apk	Application package
3	Full form of XML	Extensible Markup Language

3. Security and permissions

3.1. Security concept in Android

- ✓ The Android system installs every Android application with a unique user and group ID. Each application file is private to this generated user, e.g., other applications cannot access these files. In addition each Android application is started in its own process.
- ✓ Therefore, by means of the underlying Linux kernel, every Android application is isolated from other running applications.
- ✓ If data should be shared, the application must do this explicitly via an Android component which handles the sharing of the data, e.g., via a service or a content provider.

3.2. Permission concept in Android

- ✓ Android contains a permission system and predefines permissions for certain tasks. Every application can request required permissions and also define new permissions. For example, an application may declare that it requires access to the Internet.
- ✓ Permissions have different levels. Some permissions are automatically granted by the Android system, some are automatically rejected. In most cases the requested permissions are presented to the user before installing the application. The user needs to decide if these permissions shall be given to the application.
- ✓ If the user denies a required permission, the related application cannot be installed. The check of the permission is only performed during installation, permissions cannot be denied or granted after the installation.
- ✓ An Android application declares the required permissions in its AndroidManifest.xml configuration file. It can also define additional permissions which it can use to restrict access to certain components.

Prog-1



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Bulding a Simple HelloWorld Application

- 1) New ->Project ->Android -> Android Project -> Next
- 2) Project Name = HelloWorld

Package Name = com.androidbook.HelloWorld

Target Name = Android 1.6

- 3) HelloWorldActivity

```
System.out.println("Hello World!!!");
```

AndroidManifest.xml	Global application description file. It defines your application's capabilities and permissions and how it runs.
default.properties	Automatically created project file. It defines your application's build target and other build system options, as required.
src Folder	Required folder where all source code for the application resides
src/com.androidbook.myfirstandroidapp/ MyFirstAndroidApp- Activity.java	Core source file that defines the entry point of your Android application.
gen Folder	Required folder where auto-generated resource files for the application reside
gen/com.androidbook.myfirstandroidapp/R.java	Application resource management source file generated for you; it should not be edited



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Nr. Amrapali Under Bridge

Raiya Road

Raiya Road

Rajkot – 360001

Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

res Folder	Required folder where all application resources are managed. Application resources include animations, drawable image assets, layout files, XML files, data resources such as strings, and raw files.
res/drawable-*/icon.png	Resource folders that store different resolutions of the application icon.
res/layout/main.xml	Single screen layout file.
res/values/strings.xml	Application string resources.
assets Folder	Folder where all application assets are stored. Application assets are pieces of application data (files, directories) that you do not want managed as application resources

Prog-2

Step-1 Create Android Application

- 1) Follow the option **File -> New -> Project** and finally select **Android New Application** wizard from the wizard list.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

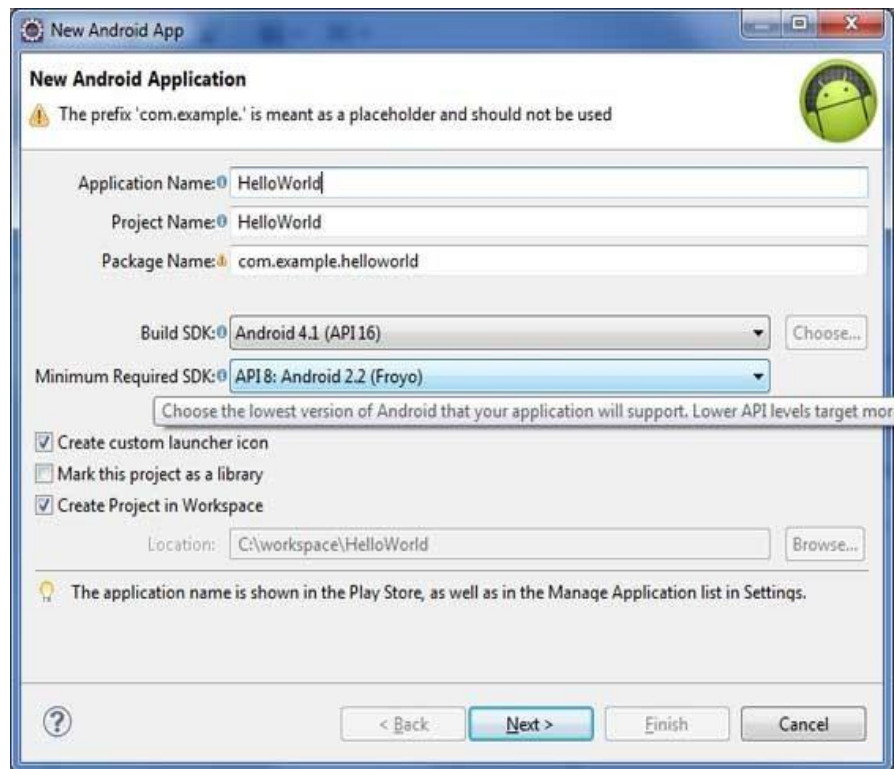
3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645



1) src

This contains the **.java** source files for your project. By default, it includes an *MainActivity.java* source file having an activity class that runs when your app is launched using the app icon.

2) gen

This contains the **.R** file, a compiler-generated file that references all the resources found in your project. You should not modify this file.

3) bin

This folder contains the Android package files **.apk** built by the ADT during the build process and everything else needed to run an Android application.

4) res/drawable-hdpi

This is a directory for drawable objects that are designed for high-density screens.

5) res/layout

This is a directory for files that define your app's user interface.

6) es/values

This is a directory for other various XML files that contain a collection of resources, such as strings and colors definitions.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

7) [AndroidManifest.xml](#)

This is the manifest file which describes the fundamental characteristics of the app and defines each of its components.

The Main Activity File

```
package com.example.helloworld;
import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;
import android.view.MenuItem;
import android.support.v4.app.NavUtils;

public class MainActivity extends Activity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.activity_main,
menu);
        return tr
    }
}
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

The Manifest File

```
<manifest
xmlns:android="http://schemas.android.com/apk/res/android
"
    package="com.example.helloworld"
    android:versionCode="1"
    android:versionName="1.0" >
<uses-sdk
    android:minSdkVersion="8"
    android:targetSdkVersion="15" />
<application
    android:icon="@drawable/ic_launcher"
    android:label="@string/app_name"
    android:theme="@style/AppTheme" >
    <activity

        android:name=".MainActivity"
        android:label="@string/title_activity_main" >
        <intent-filter>
            <action
android:name="android.intent.action.MAIN" />
            <category
android:name="android.intent.category.LAUNCHER"/>
        </intent-filter>
    </activity>
</application>
</manifest>
```

The Strings File

```
<resources>
    <string name="app_name">HelloWorld</string>
    <string name="hello_world">Hello world!</string>
    <string name="menu_settings">Settings</string>
    <string
name="title_activity_main">MainActivity</string>
</resources>
```

The R File

```
package com.example.helloworld;
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
public final class R {
    public static final class attr {
    }
    public static final class dimen {
        public static final int padding_large=0x7f040002;
        public static final int
padding_medium=0x7f040001;
        public static final int padding_small=0x7f040000;
    }
    public static final class drawable {
        public static final int
ic_action_search=0x7f020000;
        public static final int ic_launcher=0x7f020001;
    }
    public static final class id {
        public static final int menu_settings=0x7f080000;
    }
    public static final class layout {
        public static final int activity_main=0x7f030000;
    }
    public static final class menu {

        public static final int activity_main=0x7f070000;
    }
    public static final class string {
        public static final int app_name=0x7f050000;
        public static final int hello_world=0x7f050001;
        public static final int menu_settings=0x7f050002;
        public static final int
title_activity_main=0x7f050003;
    }
    public static final class style {
        public static final int AppTheme=0x7f060000;
    }
}}
```

[The Layout File](#)

```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android
"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >

<TextView
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

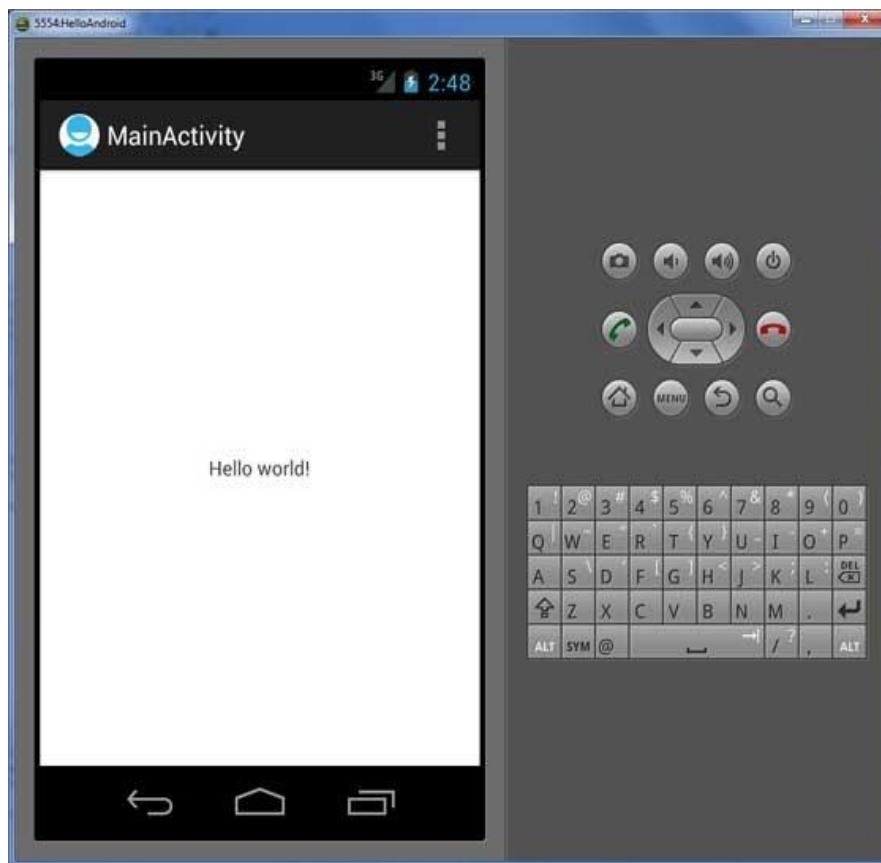
Rajkot - 360001

Ph.No–(0281)2471645

```
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_centerHorizontal="true"
android:layout_centerVertical="true"
android:padding="@dimen/padding_medium"
android:text="@string/hello_world"
tools:context=".MainActivity" />
```

</RelativeLayout>

Running the Application



Sr.no	Questions	Answer
1	Is android support Emulaor?	Yes
2	What is android studio?	Editor
3	Is android support layout?	Yes



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

UNIT – 3 ANDROID USER INTERFACE DESIGN

Introducing Android Views and Layouts

- ✓ Before we go any further, we need to define a few terms. This gives you a better understanding of certain capabilities provided by the Android SDK before they are fully introduced.

Introducing the Android View

- ✓ The Android SDK has a Java package named `android.view`. This package contains a number of interfaces and classes related to drawing on the screen. However, when we refer to the View object, we actually refer to only one of the classes within this package: the `android.view.View` class.
- ✓ The View class is the basic user interface building block within Android. It represents a rectangular portion of the screen. The View class serves as the base class for nearly all the user interface controls and layouts within the Android SDK.

Introducing the Android Control

- ✓ The Android SDK contains a Java package named `android.widget`. When we refer to controls, we are typically referring to a class within this package. The Android SDK includes classes to draw most common objects, including `ImageView`, `FrameLayout`, `EditText`, and `Button` classes. As mentioned previously, all controls are typically derived from the View class.

TextView

- ✓ The `TextView` control is a child control within other screen elements and controls. As with most of the user interface elements, it is derived from `View` and is within the `android.widget` package.
- ✓ Because it is a `View`, all the standard attributes such as width, height, padding, and visibility can be applied to the object. However, as a text-displaying control, you can apply many other `TextView`-specific attributes to control behavior and how the text is viewed in a variety of situations.
- ✓ `<TextView>` is the XML layout file tag used to display text on the screen. You can set the `android:text` property of the `TextView` to be either a raw text string in the layout file or a reference to a string resource.
- ✓ Here are examples of both methods you can use to set the `android:text` attribute of a `TextView`. The first method sets the text attribute to a raw string; the second method uses a string resource called `sample_text`, which must be defined in the `strings.xml` resource file.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

**Nr. Amrapali Under Bridge
Raiya Road**

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

**Nr. Amrapali Under Bridge
Raiya Road**

Rajkot - 360001

Ph.No–(0281)2471645

```
<TextView
```

```
android:id="@+id/TextView01"
```

```
android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"
```

```
android:text="Some sample text here" />
```

```
<TextView
```

```
android:id="@+id/TextView02"
```

```
android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"
```

```
android:text="@string/sample_text" />
```

- ✓ To display this TextView on the screen, all your Activity needs to do is call the setContentView() method :- with the layout resource identifier in which you defined the preceding XML shown.
- ✓ setText() method :- change the text displayed programmatically by calling the on the TextView object.
- ✓ getText() :- Retrieving the text is done with the method.
- ✓

Configuring Layout and Sizing

- ✓ The width of a TextView can be controlled in terms of the ems measurement rather than in pixels. An em is a term used in typography that is defined in terms of the point size of a particular font.
- ✓ Through the ems attribute, you can set the width of the TextView. Additionally,
- ✓ you can use the maxEms and minEms attributes to set the maximum width and minimum width, respectively, of the TextView in terms of ems.
- ✓ The height of a TextView can be set in terms of lines of text rather than pixels. Again, this is useful for controlling how much text can be viewed regardless of the font size. The lines attribute sets the number of lines that the TextView can display. You can also use maxLines and minLines to control the maximum height and minimum height, respectively, that the TextView displays.
- ✓ Here is an example that combines these two types of sizing attributes. This TextView is two lines of text high and 12 ems of text wide. The layout width and height are specified to the size of the TextView and are required attributes in the XML schema:

```
<TextView
```

```
android:id="@+id/TextView04"
```

```
android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"
```

```
android:lines="2"
```

```
android:ems="12"
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Nr. Amrapali Under Bridge

Raiya Road

Raiya Road

Rajkot – 360001

Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

```
android:text="@string/autolink_test"/>
```

EditText

- ✓ The Android SDK provides a convenient control called EditText to handle text input from a user. The EditText class is derived from TextView. In fact, most of its functionality is contained within TextView but enabled when created as an EditText.
- ✓
- ✓ Define an EditText control in an XML layout file:

```
<EditText  
android:id="@+id/EditText01"  
android:layout_height="wrap_content"  
android:hint="type here"  
android:lines="4" android:layout_width  
"fill_parent"/>
```

- ✓ hint :- attribute puts some text in the edit box that goes away when the user starts entering text.
- ✓ Lines :- which defines how many lines tall the input box is. If this is not set, the entry field grows as the user enters text.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

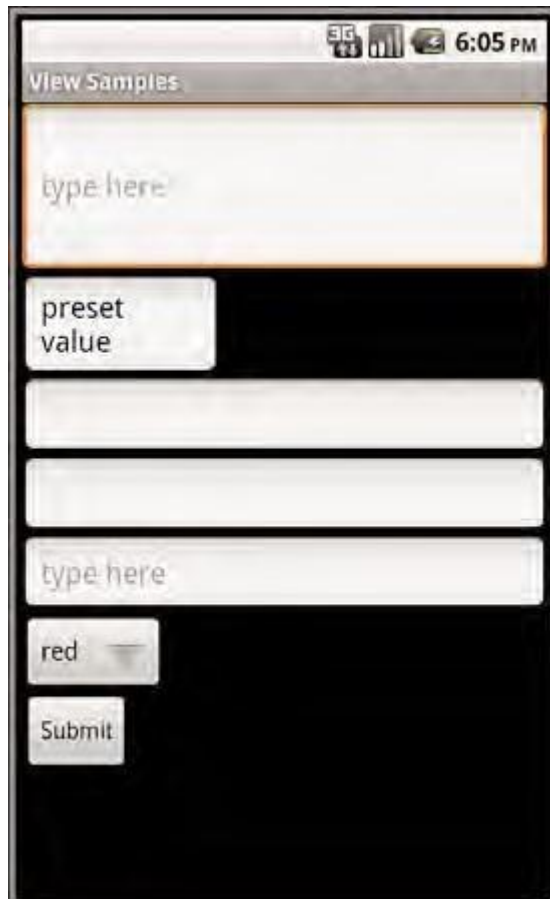
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645



- ✓ The EditText object is essentially an editable TextView. This means that you can read text from it in the same way as you did with TextView
- ✓
- ✓ getText() :- read text from it
- ✓ setText() :- set initial text to draw in the text entry area
- ✓

Helping the User with Auto Completion

- 1) autoCompleteTextView :- There are two forms of auto-complete. One is the more standard style of filling in the entire text entry based on what the user types. If the user begins typing a string that matches a word in a developer-provided list, the user can choose to complete the word with just a tap
- 2) MultiAutoCompleteTextView :- allows the user to enter a list of items, each of which has auto-complete functionality. These items must be separated in some way by providing a Tokenizer. A common Tokenizer implementation is provided for comma-separated lists and is used by specifying the



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Nr. Amrapali Under Bridge

Raiya Road

Raiya Road

Rajkot – 360001

Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

MultiAutoComplete TextView.CommaTokenizer object.

This can be helpful for lists of specifying common tags and the like.





SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Nr. Amrapali Under Bridge

Raiya Road

Raiya Road

Rajkot – 360001

Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645



- ✓ Using autoCompleteTextView (left) and MultiAutoCompleteTextView (right).
- ✓
- ✓ Both of the auto-complete text editors use an adapter to get the list of text that they use to provide completions to the user. This example shows how to provide an autoCompleteTextView for the user

```
final String[] COLORS {  
    "red", "green", "orange", "blue", "purple",  
    "black", "yellow", "cyan", "magenta" };
```

```
ArrayAdapter<String> adapter new  
ArrayAdapter<String>(this,  
    android.R.layout.simple_dropdown_item_1line, COLORS);  
AutoCompleteTextView text (AutoCompleteTextView)  
findViewById(R.id.AutoCompleteTextView01);  
text.setAdapter(adapter);
```

Spinner Controls

- ✓ Sometimes you want to limit the choices available for users to type. For instance, if users are going to enter the name of a state, you might as well limit them to only the valid states because this is a known set.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- ✓ Although you could do this by letting them type something and then blocking invalid entries, you can also provide similar functionality with a Spinner control. As with the auto-complete method, the possible choices for a spinner can come from an Adapter.

You can also set the available choices in the layout definition by using the entries attribute with an array resource (specifically a string-array that is referenced as something such as @array/state-list). The Spinner control isn't actually an EditText, although it is frequently used in a similar fashion. Here is an example of the XML layout definition for a Spinner control for choosing a color:

```
<Spinner  
android:id="@+id/Spinner01"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:entries="@array/colors"  
android:prompt="@string/spin_prompt" />
```



- ✓ When the user selects it, a pop-up shows the prompt text followed by a list of the possible choices. This list allows only a single item to be selected at a time, and



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Nr. Amrapali Under Bridge

Raiya Road

Raiya Road

Rajkot – 360001

Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

when one is selected, the pop-up goes away

- ✓ There are a couple of things to notice here.
- ✓ First, the entries attribute is set to the values that shows by assigning it to an array resource, referred to here as @array/colors.
- ✓ Second, the prompt attribute is defined to a string resource. Unlike some other string attributes, this one is required to be a string resource. The prompt displays

when the popup comes up and can be used to tell the user what kinds of values that can be selected from.

Button

- ✓ The android.widget.Button class provides a basic button implementation in the Android SDK. Within the XML layout resources, buttons are specified using the Button element.
- ✓ The primary attribute for a basic button is the text field. This is the label that appears on the middle of the button's face. You often use basic Button controls for buttons with text such as "Ok," "Cancel," or "Submit."





SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Various types of button controls

The following XML layout resource file shows a typical Button control definition:

```
<Button
android:id "@+id/basic_button"
android:layout_width "wrap_content"
android:layout_height "wrap_content"
android:text "Basic Button" />
```

A button won't do anything, other than animate, without some code to handle the click event. Here is an example of some code that handles a click for a basic button and displays a Toast message on the screen:

```
setContentView(R.layout.buttons);
final Button basic_button (Button) findViewById(R.id.basic_button);
basic_button.setOnClickListener(new View.OnClickListener() { public
void onClick(View v) {
Toast.makeText(ButtonsActivity.this,
"Button clicked", Toast.LENGTH_SHORT).show();
}
});
```

- ✓ To handle the click event for when a button is pressed, we first get a reference to the Button by its resource identifier. Next, the `setOnClickListener()` method is called. It requires a valid instance of the class `View.OnClickListener`. A simple way to provide this is to define the instance right in the method call. This requires implementing the `onClick()` method. Within the `onClick()` method, you are free to carry out whatever actions you need.
- ✓
- ✓ A button with its primary label as an image is an `ImageButton`. An `ImageButton` is, for most purposes, almost exactly like a basic button. Click actions are handled in the same way. The primary difference is that you can set its `src` attribute to be an image.
- ✓
- ✓ example of an `ImageButton` definition in an XML layout resource file:

```
<ImageButton
android:layout_width "wrap_content"
android:layout_height "wrap_content"
android:id "@+id/image_button"
android:src "@drawable/droid" />
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Check Boxes

- ✓ The check box button is often used in lists of items where the user can select multiple
- ✓ items. The Android check box contains a text attribute that appears to the side of the check box. This is used in a similar way to the label of a basic button. An XML layout resource definition for a CheckBox control:

<CheckBox

android:id="@+id/checkbox"

android:layout_width="wrap_content"

android:layout_height="wrap_content"

android:text="Check me?" />

The following example shows how to check for the state of the button programmatically and change the text label to reflect the change:

```
final CheckBox check_button (CheckBox) findViewById(R.id.checkbox);
```

```
check_button.setOnClickListener(new View.OnClickListener() {
```

```
public void onClick (View v) {
```

```
TextView tv (TextView) findViewById(R.id.checkbox);
```

```
tv.setText(check_button.isChecked() ?
```

```
“This option is checked” :
```

```
“This option is not checked”);
```

```
}
```

```
});
```

- ✓ A check box automatically shows the check as enabled or disabled. This enables us to deal with behavior in our application rather than worrying about how the button should behave. The layout shows that the text starts out one way but, after the user presses the button, the text changes to one of two different things depending
- ✓ on the checked state
- ✓ A Toggle Button is similar to a check box in behavior but is usually used to show or alter the on or off state of something. Like the CheckBox, it has a state (checked or not).
- ✓ Also like the check box, the act of changing what displays on the button is handled for us.
- ✓ Unlike the CheckBox, it does not show text next to it. Instead, it has two text fields. The first attribute is textOn, which is the text that displays on the button when its checked state is on. The second attribute is textOff, which is the text that displays on the button when its checked state is off. The default text for these is “ON” and “OFF,” respectively. The layout code shows a definition for a toggle button that shows “Enabled” or “Disabled” based on the state of the button:



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
<ToggleButton  
android:id "@+id/toggle_button"  
android:layout_width "wrap_content"  
android:layout_height "wrap_content"  
android:text "Toggle"  
android:textOff "Disabled"  
android:textOn "Enabled" />
```

Using RadioGroups and RadioButtons

- ✓ use radio buttons when a user should be allowed to only select one item from a small group of items. For instance, a question asking for gender can give three options:
- ✓ male, female, and unspecified. Only one of these options should be checked at a time.
- ✓ The RadioButton objects are similar to CheckBox objects. They have a text label next to them, set via the text attribute, and they have a state (checked or unchecked).
- ✓ group RadioButton objects inside a RadioGroup that handles enforcing their combined states so that only one RadioButton can be checked at a time. If the user selects a RadioButton that is already checked, it does not become unchecked. However, you can provide the user with an action to clear the state of the entire RadioGroup so that none of the buttons are checked.

XML layout resource

```
<RadioGroup  
android:id "@+id/RadioGroup01"  
android:layout_width "wrap_content"  
android:layout_height "wrap_content">  
<RadioButton  
android:id "@+id/RadioButton01"  
android:layout_width "wrap_content"  
android:layout_height "wrap_content"  
android:text "Option 1"></RadioButton>  
<RadioButton  
android:id "@+id/RadioButton02"  
android:layout_width "wrap_content"  
android:layout_height "wrap_content"  
android:text "Option 2"></RadioButton>  
<RadioButton  
android:id "@+id/RadioButton03"  
android:layout_width "wrap_content"  
android:layout_height "wrap_content"
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
android:text "Option 3"></RadioButton>
```

```
<RadioButton
```

```
android:id "@+id/RadioButton04"
```

```
android:layout_width "wrap_content"
```

```
android:layout_height "wrap_content"
```

```
android:text "Option 4"></RadioButton>
```

```
</RadioGroup>
```

actions on these RadioButton objects through the RadioGroup

```
final RadioGroup group (RadioGroup)findViewById(R.id.RadioGroup01);final
```

```
TextView tv (TextView)
```

```
findViewById(R.id.TextView01);
```

```
group.setOnCheckedChangeListener(new
```

```
RadioGroup.OnCheckedChangeListener() {
```

```
public void onCheckedChanged( RadioGroup
```

```
group, int checkedId) {
```

```
if (checkedId != -1) { RadioButton
```

```
rb (RadioButton)
```

```
findViewById(checkedId);
```

```
if (rb != null) {
```

```
tv.setText("You chose: " + rb.getText());
```

```
}
```

```
} else {
```

```
tv.setText("Choose 1");
```

```
}
```

```
}
```

```
});
```

Date and Times from users

- ✓ The Android SDK provides a couple controls for getting date and time input from the user. The first is the DatePicker control (Figure 7.8, top). It can be used to get a month,
- ✓ day, and year from the user.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645



Date and time controls

XML layout resource definition for a DatePicker follows:

```
<DatePicker
android:id="@+id/DatePicker01"
android:layout_width="wrap_content"
android:layout_height="wrap_content" />
```

onDateChanged() call when the date changes

```
final DatePicker date (DatePicker)findViewById(R.id.DatePicker01);
date.init(date.getYear(), date.getMonth(), date.getDayOfMonth(), new
DatePicker.OnDateChangeListener() {
public void onDateChanged(DatePicker view, int year,int
monthOfYear, int dayOfMonth) {
Date dt new Date(year-1900, monthOfYear, dayOfMonth, time.getCurrentHour(),
time.getCurrentMinute());
```

```
text.setText(dt.toString());
}
});
```

- ✓ The preceding code sets the DatePicker.OnDateChangeListener by a call to the DatePicker.init() method. A DatePicker control is initialized with the current date. A TextView is set with the date value that the user entered into the DatePicker control. The value of 1900 is subtracted from the year parameter to make it compatible with the java.util.Date class.

✓



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Nr. Amrapali Under Bridge

Raiya Road

Raiya Road

Rajkot – 360001

Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

-
- ✓ A TimePicker control (also shown in Figure 7.8, bottom) is similar to the DatePicker control. It also doesn't have any unique attributes. However, to register for a method call when the values change, you call the more traditional method of

```
TimePicker.setOnTimeChangeListener(). time.setOnTimeChangeListener(new  
TimePicker.OnTimeChangeListener() {public void onTimeChanged(TimePicker  
view,  
int hourOfDay, int minute) {  
Date dt new Date(date.getYear()-1900, date.getMonth(),  
date.getDayOfMonth(), hourOfDay, minute);  
text.setText(dt.toString());  
}  
});
```

ProgressBar

- ✓ The standard progress bar is a circular indicator that only animates. It does not show how complete an action is. It can, however, show that something is taking place. This is useful when an action is indeterminate in length. There are three sizes of this type of
- ✓ progress indicator



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645



- ✓ Various types of progress and rating indicators.
- ✓
- ✓ The second type is a horizontal progress bar that shows the completeness of an action.
- ✓ (For example, you can see how much of a file is downloading.) This horizontal progress bar can also have a secondary progress indicator on it. This can be used, for instance, to show the completion of a downloading media file while that file plays.

This is an XML layout resource definition for a basic indeterminate progress bar:

```
<ProgressBar  
android:id="@+id/progress_bar"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content" />
```

[Time Passage with the Chronometer](#)

- ✓ Sometimes you want to show time passing instead of incremental progress. In this case, you can use the Chronometer control as a timer. This might be



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Nr. Amrapali Under Bridge

Raiya Road

Raiya Road

Rajkot – 360001

Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

- ✓ useful if it's the user who is taking time doing some task or in a game where some action needs to be timed. The Chronometer control can be formatted with text, as shown in this

XML layout resource definition:

```
<Chronometer
android:id="@+id/Chronometer01"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:format="Timer: %s" />
```

- ✓ You can use the Chronometer object's format attribute to put text around the time that
- ✓ displays. A Chronometer won't show the passage of time until its start() method is called. To stop it, simply call its stop() method. Finally, you can change the time from which the timer is counting. That is, you can set it to count from a particular time in the past instead of from the time it's started. You call the setBase() method to do this. In this next example code, the timer is retrieved from the View by its resource identifier. We then check its base value and set it to 0. Finally, we start the timer counting up from there.

✓

```
final Chronometer timer
(Chronometer)findViewById(R.id.Chronometer01);long
base timer.getBase(); Log.d(ViewsMenu.debugTag, "base
"+ base); timer.setBase(0);
timer.start();
```

Menu

two special application menus for use within your Android applications: the options menu and the context menu.

1) Options Menus

The Android SDK provides a method for users to bring up a menu by pressing the menu key from within the application. You can use options menus within your application to bring up help, to navigate, to provide additional controls, or to configure options. The OptionsMenu control can contain icons, submenus, and keyboard shortcuts.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645



An options menu.

- ✓ For an options menu to show when a user presses the Menu button on their device, you need to override the implementation of `onCreateOptionsMenu()` in your Activity.
- ✓
- ✓ Here is a sample implementation that gives the user three menu items to choose from:

```
public boolean onCreateOptionsMenu( android.view.Menu menu) {
super.onCreateOptionsMenu(menu);
menu.add("Forms")
.setIcon(android.R.drawable.ic_menu_edit)
.setIntent(new Intent(this, FormsActivity.class));
menu.add("Indicators")
.setIntent(new Intent(this, IndicatorsActivity.class))
.setIcon(android.R.drawable.ic_menu_info_details);
menu.add("Containers")
.setIcon(android.R.drawable.ic_menu_view)
.setIntent(new Intent(this, ContainersActivity.class));
return true;
}
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

ii) ContextMenu

- ✓ The ContextMenu is a subtype of Menu that you can configure to display when a long press is performed on a View. As the name implies, the ContextMenu provides for contextual menus to display to the user for performing additional actions on selected items.
- ✓ ContextMenu objects are slightly more complex than OptionsMenu objects. You need to implement the onCreateContextMenu() method of your Activity for one to display.
- ✓ However, before that is called, you must call the registerForContextMenu() method and pass in the View for which you want to have a context menu. This means each View on your screen can have a different context menu, which is appropriate as the menus are designed to be highly contextual.
- ✓ Recall that any View control can register to trigger a call to the onCreateContextMenu() method when the user performs a long press. That means we have to check which View control it was for which the user tried to get a context menu. Next, we inflate the appropriate menu from a menu resource that we defined with XML. Because we can't define header information in the menu resource file, we set a stock Android SDK resource to it and add a title. Here is the menu resource that is inflated:

```
<menu
xmlns:android="http://schemas.android.com/apk/res/android">
<item
android:id="@+id/start_timer"
android:title "Start" />
<item
android:id="@+id/stop_timer"
android:title "Stop" />
<item
android:id="@+id/reset_timer"
android:title "Reset" />
</menu>
```

- ✓ Now we need to handle the ContextMenu clicks by implementing the onOptionsItemSelected() method in our Activity.
- ✓
- ✓ Here's an example:

```
public boolean onOptionsItemSelected(MenuItem item) {
super.onOptionsItemSelected(item);
boolean result false;
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
Chronometer timer (Chronometer) findViewById(R.id.Chronometer01);
switch (item.getItemId()){
case R.id.stop_timer:
timer.stop();
result true;
break;
case R.id.start_timer:
timer.start();
result true;
break;
case R.id.reset_timer:
timer.setBase(SystemClock.elapsedRealtime());result
true;
break;
}
return result;
}
```

Dialogs

- ✓ An Activity can use dialogs to organize information and react to user-driven events. For example, an activity might display a dialog informing the user of a problem or ask the user to confirm an action such as deleting a data record

Exploring the Different Types of Dialogs

- ✓ There are a number of different dialog types available within the Android SDK. Each has a special function that most users should be somewhat familiar with. The dialog types available include

- 1) **Dialog:** The basic class for all Dialog types. A basic Dialog is shown in the top left in below figure.
- 2) **AlertDialog:** A Dialog with one, two, or three Button controls. An AlertDialog is shown in the top center in below figure.
- 3) **CharacterPickerDialog:** A Dialog for choosing an accented character associated with a base character. A CharacterPickerDialog is shown in the top right in below figure.
- 4) **DatePickerDialog:** A Dialog with a DatePicker control. A DatePickerDialog is shown in the bottom left of in below figure.
- 5) **ProgressDialog:** A Dialog with a determinate or indeterminate ProgressBar control. An indeterminate ProgressDialog is shown in the bottom center of in following figure.
- 1) **TimePickerDialog:** A Dialog with a TimePicker control. A TimePickerDialog is shown in the bottom right in below figure



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645



The different dialog types available in Android.

Defining a Dialog

- ✓ Each Dialog must be defined within the Activity in which it is used. A Dialog may be launched once, or used repeatedly. Understanding how an Activity manages the Dialog lifecycle is important to implementing a Dialog correctly. Let's look at the key methods that an Activity must use to manage a Dialog:
 - 1) The `showDialog()` method is used to display a Dialog.
 - 2) The `dismissDialog()` method is used to stop showing a Dialog. The Dialog is kept around in the Activity's Dialog pool. If the Dialog is shown again using `showDialog()`, the cached version is displayed once more.
 - 3) The `removeDialog()` method is used to remove a Dialog from the Activity objects Dialog pool. The Dialog is no longer kept around for future use. If you call `showDialog()` again, the Dialog must be re-created.
- ✓ Adding the Dialog to an Activity involves several steps:
 - 1) Define a unique identifier for the Dialog within the Activity.
 - 2) Implement the `onCreateDialog()` method of the Activity to return a Dialog of the appropriate type, when supplied the unique identifier
 - 3) Implement the `onPrepareDialog()` method of the Activity to initialize the Dialog as appropriate.
 - 4) Launch the Dialog using the `showDialog()` method with the unique identifier.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Defining a Dialog

- ✓ A Dialog used by an Activity must be defined in advance. Each Dialog has a special identifier (an integer). When the `showDialog()` method is called, you pass in this identifier. At this point, the `onCreateDialog()` method is called and must return a Dialog of the appropriate type.
- ✓ It is up to the developer to override the `onCreateDialog()` method of the Activity and return the appropriate Dialog for a given identifier. If an Activity has multiple
- ✓ Dialog windows, the `onCreateDialog()` method generally contains a switch statement to return the appropriate Dialog based on the incoming parameter—the Dialog identifier.

Initializing a Dialog

- ✓ Because a Dialog is often kept around by the Activity in its Dialog pool, it might be important to re-initialize a Dialog each time it is shown, instead of just when it is created the first time. For this purpose, you can override the `onPrepareDialog()` method of the Activity.
- ✓ Although the `onCreateDialog()` method may only be called once for initial Dialog creation, the `onPrepareDialog()` method is called each time the `showDialog()` method is called, giving the Activity a chance to modify the Dialog before it is shown to the user.
- ✓

Launching a Dialog

- ✓ You can display any Dialog defined within an Activity by calling its `showDialog()` method and passing it a valid Dialog identifier—one that will be recognized by the `onCreateDialog()` method.

Dismissing a Dialog

- ✓ Most types of dialogs have automatic dismissal circumstances. However, if you want to force a Dialog to be dismissed, simply call the `dismissDialog()` method and pass in the Dialog identifier.

Removing a Dialog from Use

- ✓ Dismissing a Dialog does not destroy it. If the Dialog is shown again, its cached contents are redisplayed. If you want to force an Activity to remove a Dialog from its pool and not use it again, you can call the `removeDialog()` method, passing in the valid Dialog identifier.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

3.2) Layout Classes

- ✓ The types of layouts built-in to the Android SDK framework include
 - a) `FrameLayout`
 - b) `LinearLayout`
 - c) `RelativeLayout`
 - d) `TableLayout`

- ✓ All layouts, regardless of their type, have basic layout attributes. Layout attributes apply to any child View within that layout. You can set layout attributes at runtime programmatically, but ideally you set them in the XML layout files using the following syntax: `android:layout_attribute_name "value"`

- ✓ There are several layout attributes that all ViewGroup objects share. These include size attributes and margin attributes. You can find basic layout attributes in the `ViewGroup.LayoutParams` class. The margin attributes enable each child View within a layout to have padding on each side. Find these attributes in the `ViewGroup.MarginLayoutParams` classes. There are also a number of ViewGroup attributes for handling child View drawing bounds and animation settings.

Attribute Name	Applies To	Description	Value
<code>android:layout_height</code>	Parent view Child view	Height of the view. Required attribute for child view controls in layouts.	Specific dimension value, <code>fill_parent</code> , or <code>wrap_content</code> . The <code>match_parent</code> option is available in API Level 8+.
<code>android:layout_width</code>	Parent view Child view	Width of the view. Required attribute for child view controls in layouts.	Specific dimension value, <code>fill_parent</code> , or <code>wrap_content</code> . The <code>match_parent</code> option is available in API Level 8+.
<code>android:layout_margin</code>	Child view	Extra space on all sides of the view.	Specific dimension value.

- ✓ an XML layout resource example of a `LinearLayout` set to the size of the screen, containing one `TextView` that is set to its full height and the width of the `LinearLayout` (and therefore the screen):

```
<LinearLayout xmlns:android
“http://schemas.android.com/apk/res/android”
android:layout_width ”fill_parent” android:layout_height
”fill_parent”>
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

```
<TextView  
android:id="@+id/TextView01"  
android:layout_height="fill_parent"  
android:layout_width="fill_parent" />  
</LinearLayout>
```

- ✓ an example of a Button object with some margins set via XML used in a layoutresource file:

```
<Button  
android:id="@+id/Button01"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:text="Press Me"  
android:layout_marginRight="20px"  
android:layout_marginTop="60px" />
```

- ✓ layout elements can cover any rectangular space on the screen; it doesn't need to be the entire screen. Layouts can be nested within one another.

Frame layout

- ✓ A FrameLayout view is designed to display a stack of child View items. You can add multiple views to this layout, but each View is drawn from the top-left corner of the layout. find the layout attributes available for FrameLayout child View objects in android.control.FrameLayout.LayoutParams



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

Attribute Name	Applies To	Description	Value
android:foreground	Parent view	Drawable to draw over the content.	Drawable resource.
android:foreground-Gravity	Parent view	Gravity of foreground drawable.	One or more constants separated by " ". The constants available are top, bottom, left, right, center_vertical, fill_vertical, center_horizontal, fill_horizontal, center, and fill.



- ✓ an XML layout resource with a `FrameLayout` and two child `View` objects, both `ImageView` objects. The green rectangle is drawn first and the red oval is drawn on top of it. The green rectangle is larger, so it defines the bounds of the

`FrameLayout`:

```
<FrameLayout xmlns:android
http://schemas.android.com/apk/res/android
android:id="@+id/FrameLayout01"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_gravity="center">
<ImageView
android:id="@+id/ImageView01"
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
android:layout_width "wrap_content"
android:layout_height "wrap_content"
android:src "@drawable/green_rect"
android:minHeight "200px"
android:minWidth "200px" />
<ImageView
android:id "@+id/ImageView02"
android:layout_width "wrap_content"
android:layout_height "wrap_content"
android:src "@drawable/red_oval"
android:minHeight "100px"
android:minWidth "100px"
android:layout_gravity "center" />
</FrameLayout>
```

Sr.no	Questions	Answer
1	TextBox in Android is called	EditText
2	Label in Android is called	TextView
3	Is Android support control properties?	Yes



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

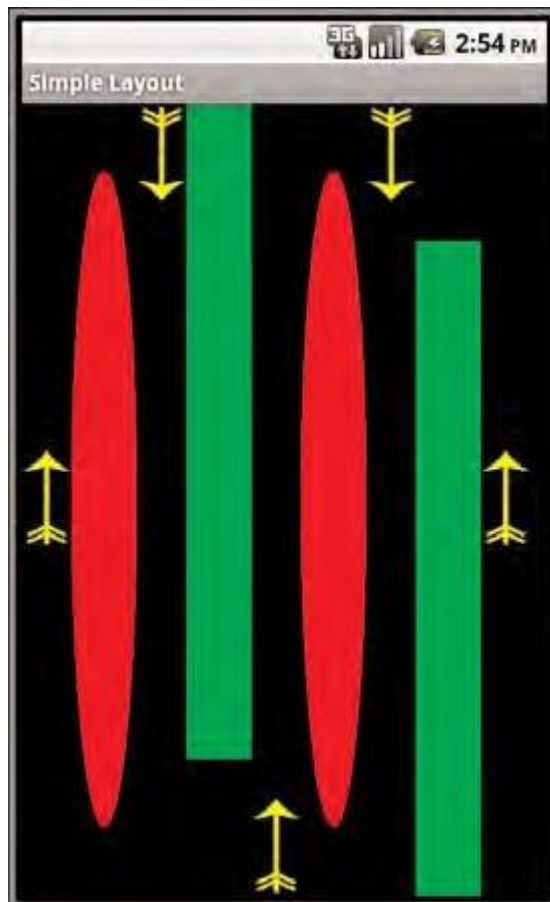
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

LinearLayout

- ✓ A LinearLayout view organizes its child View objects in a single row or column, depending on whether its orientation attribute is set to horizontal or vertical. This is a very handy layout method for creating forms.



- ✓ You can find the layout attributes available for LinearLayout child View objects in `android.control.LinearLayout.LayoutParams`. Table 8.3 describes some of the important attributes specific to LinearLayout views.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

Attribute Name	Applies To	Description	Value
android:orientation	Parent view	Layout is a single row (horizontal) or single column (vertical).	Horizontal or vertical.
android:gravity	Parent view	Gravity of child views within layout.	One or more constants separated by " ". The constants available are top, bottom, left, right, center_vertical, fill_vertical, center_horizontal, fill_horizontal, center, and fill.
android:layout_gravity	Child view	The gravity for a specific child view. Used for positioning of views.	One or more constants separated by " ". The constants available are top, bottom, left, right, center_vertical, fill_vertical, center_horizontal, fill_horizontal, center, and fill.
android:layout_weight	Child view	The weight for a specific child view. Used to provide ratio of screen space used within the parent control.	The sum of values across all child views in a parent view must equal 1. For example, one child control might have a value of .3 and another have a value of .7.

RelativeLayout

- ✓ The RelativeLayout view enables you to specify where the child view controls are in relation to each other.
- ✓ For instance, you can set a child View to be positioned “above” or “below” or “to the left of” or “to the right of” another View, referred to by its unique identifier. You can also align child View objects relative to one another or the parent layout edges.
- ✓ Combining RelativeLayout attributes can simplify creating interesting user interfaces without resorting to multiple layout groups to achieve a desired effect.
- ✓ You can find the layout attributes available for RelativeLayout child View objects in android.control.RelativeLayout.LayoutParams.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645



Attribute Name	Applies To	Description	Value
android:gravity	Parent view	Gravity of child views within layout.	One or more constants separated by " ". The constants available are top, bottom, left, right, center_vertical, fill_vertical, center_horizontal, fill_horizontal, center, and fill.
android:layout_centerInParent	Child view	Centers child view horizontally and vertically within parent view.	True or false.

- ✓ XML layout resource with a RelativeLayout and two child View objects, a Button



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

object aligned relative to its parent, and an ImageView aligned and positioned relative to the Button



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
<?xml version "1.0" encoding "utf-8"?>
<RelativeLayout xmlns:android
"http://schemas.android.com/apk/res/android"android:id
"@+id/RelativeLayout01" android:layout_height
"fill_parent" android:layout_width "fill_parent">
<Button
android:id "@+id/ButtonCenter"
android:text "Center" android:layout_width
"wrap_content" android:layout_height
"wrap_content"
android:layout_centerInParent "true" />
<ImageView
android:id "@+id/ImageView01"
android:layout_width "wrap_content"
android:layout_height "wrap_content"
android:layout_above "@id/ButtonCenter"
android:layout_centerHorizontal "true"
android:src "@drawable/arrow" />
</RelativeLayout>
```

TableLayout

- ✓ A TableLayout view organizes children into rows, as shown in Figure 8.7. You add individual View objects within each row of the table using a TableRow layout View (which is basically a horizontally oriented LinearLayout) for each row of the table. Each column of the TableRow can contain one View (or layout with child View objects).
- ✓ You place View items added to a TableRow in columns in the order they are added. You can specify the column number (zero-based) to skip columns as necessary; otherwise, the View object is put in the next column to the right. Columns scale to the size of the largest View of that column.
- ✓ You can also include normal View objects instead of TableRow elements, if you want the View to take up an entire row.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645



- ✓ You can find the layout attributes available for `TableLayout` child View objects in `android.control.TableLayout.LayoutParams`. You can find the layout attributes available for `TableRow` child View objects in `android.control.TableRow.LayoutParams`.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

Attribute Name	Applies To	Description	Value
android:collapseColumns	TableLayout	A comma-delimited list of column indices to collapse (0-based)	String or string resource. For example, "0,1,3,5"
android:shrinkColumns	TableLayout	A comma-delimited list of column indices to shrink (0-based)	String or string resource. Use "*" for all columns. For example, "0,1,3,5"
android:stretchColumns	TableLayout	A comma-delimited list of column indices to stretch (0-based)	String or string resource. Use "*" for all columns. For example, "0,1,3,5"
android:layout_column	TableRow child view	Index of column this child view should be displayed in (0-based)	Integer or integer resource. For example, 1
android:layout_span	TableRow child view	Number of columns this child view should span across	Integer or integer resource greater than or equal to 1. For example, 3

- ✓ An XML layout resource with a TableLayout with two rows (two TableRow child objects).The TableLayout is set to stretch the columns to the size of the screen width.The first TableRow has three columns; each cell has a Button object.
- ✓ The second TableRow puts only one Button view into the second column explicitly:

```
<TableLayout xmlns:android
"http://schemas.android.com/apk/res/android"android:id
"@+id/TableLayout01" android:layout_width "fill_parent"
android:layout_height "fill_parent"
android:stretchColumns "*">
<TableRow
android:id "@+id/TableRow01">
<Button
android:id "@+id/ButtonLeft"
android:text "Left Door" />
<Button
android:id "@+id/ButtonMiddle"
android:text "Middle Door" />
<Button
android:id "@+id/ButtonRight"
android:text "Right Door" />
</TableRow>
<TableRow
android:id "@+id/TableRow02">
<Button
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

android:id="@+id/ButtonBack"

android:text "Go Back"



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
android:layout_column ”1” />
```

```
</TableRow>
```

```
</TableLayout>
```

Data-Driven Containers

- ✓ Some of the View container controls are designed for displaying repetitive View objects in a particular way. Examples of this type of View container control include ListView, GridView, and GalleryView:
- ✓ **ListView**: Contains a vertically scrolling, horizontally filled list of View objects, each of which typically contains a row of data; the user can choose an item to perform some action upon.
- ✓ **GridView**: Contains a grid of View objects, with a specific number of columns; this container is often used with image icons; the user can choose an item to perform some action upon.
- ✓ **GalleryView**: Contains a horizontally scrolling list of View objects, also often used with image icons; the user can select an item to perform some action upon.
- ✓ These containers are all types of AdapterView controls. An AdapterView control contains a set of child View controls to display data from some data source.
- ✓ An Adapter generates these child View controls from a data source. As this is an important part of all these container controls, we talk about the Adapter objects first.

ArrayAdapter

- ✓ An ArrayAdapter binds each element of the array to a single View object within the layout resource.
- ✓ Here is an example of creating an ArrayAdapter:

```
private String[] items {  
    “Item 1”, “Item 2”, “Item 3” };
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
ArrayAdapter adapt  
new ArrayAdapter<String>  
(this, R.layout.textview,  
items);
```

- ✓ a String array called items. This is the array used by the ArrayAdapter as the source data. We also use a layout resource, which is the View that is repeated for each item in the array.

This is defined as follows:

```
<TextView xmlns:android  
“http://schemas.android.com/apk/res/android”  
android:layout_width “fill_parent”  
android:layout_height “wrap_content”  
android:textSize “20px” />
```

Sr.no	Questions	Answer
1	Is Android support layout?	Yes
2	Full form of px	Pixel
3	Is Layout support Table Layout?	Yes

```
timer.start  
();result  
true;  
break;  
case R.id.reset_timer:  
timer.setBase(SystemClock.elapsedRealtime()  
);result true;  
break;  
}  
return result;  
}
```

Dialogs

- ✓ An Activity can use dialogs to organize information and react to user-driven events. For example, an activity might display a dialog informing the user of a problem or ask the user to confirm an action such as deleting a data record

Exploring the Different Types of Dialogs

- ✓ There are a number of different dialog types available within the Android SDK. Each has a special function that most users should be somewhat familiar with. The dialog types available include



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- 6) **Dialog:** The basic class for all Dialog types. A basic Dialog is shown in the top left in below figure.
- 7) **AlertDialog:** A Dialog with one, two, or three Button controls. An AlertDialog is shown in the top center in below figure.
- 8) **CharacterPickerDialog:** A Dialog for choosing an accented character associated with a base character. A CharacterPickerDialog is shown in the top right in below figure.
- 9) **DatePickerDialog:** A Dialog with a DatePicker control. A DatePickerDialog is shown in the bottom left of in below figure.
- 10) **ProgressDialog:** A Dialog with a determinate or indeterminate ProgressBar control. An indeterminate ProgressDialog is shown in the bottom center of in following figure.
- 1) **TimePickerDialog:** A Dialog with a TimePicker control. A TimePickerDialog is shown in the bottom right in below figure

UNIT – 4 DATABASE CONNECTIVITY USING SQLITE AND CONTENT PROVIDER

➤ [Working with Application Preferences](#)

- ✓ Many applications need a lightweight data storage mechanism called shared preferences for storing application state, simple user information, configuration options, and other such information.
- ✓ Android provides a simple preferences system for storing primitive application data at the Activity level and preferences shared across all of an application's activities. You cannot share preferences outside of the package. Preferences are stored as groups of key/value pairs. The following data types are supported as preference settings:
 1. Boolean values
 2. Float values
 3. Integer values
 4. Long values
 5. String values
- ✓ Preference functionality can be found in the SharedPreferences interface of the android.content package. To add preferences support to your application, you must take the following steps:
 1. Retrieve an instance of a SharedPreferences object.
 2. Create a SharedPreferences.Editor to modify preference content.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

3. Make changes to the preferences using the Editor.

4. Commit your changes.

➤ Creating Private and Shared Preferences

- ✓ Individual activities can have their own private preferences. These preferences are for the specific Activity only and are not shared with other activities within the application. The activity gets only one group of private preferences.
- ✓ The following code retrieves the activity's private preferences:


```
import
android.content.SharedPreferences;
```
- ✓ ...
- ✓ `SharedPreferences settingsActivity getPreferences(MODE_PRIVATE);`

➤ Searching and Reading Preferences

- ✓ Reading preferences is straightforward. Simply retrieve the `SharedPreferences` instance you want to read. You can check for a preference by name, retrieve strongly typed preferences, and register to listen for changes to the preferences. Following method describes some helpful methods in the `SharedPreferences` interface.

Method	Purpose
<code>SharedPreferences.contains()</code>	Sees whether a specific preference exists by name
<code>SharedPreferences.edit()</code>	Retrieves the editor to change these preferences
<code>SharedPreferences.getAll()</code>	Retrieves a map of all preference key/value pairs
<code>SharedPreferences.getBoolean()</code>	Retrieves a specific Boolean-type preference by name
<code>SharedPreferences.getFloat()</code>	Retrieves a specific Float-type preference by name
<code>SharedPreferences.getInt()</code>	Retrieves a specific Integer-type preference by name
<code>SharedPreferences.getLong()</code>	Retrieves a specific Long-type preference by name
<code>SharedPreferences.getString()</code>	Retrieves a specific String-type preference by name



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

➤ [Adding, Updating, and Deleting Preferences](#)

- ✓ To change preferences, you need to open the preference Editor, make your changes, and commit them. Table 10.2 describes some helpful methods in the SharedPreferences.Editor interface

Method	Purpose
SharedPreferences.Editor.clear()	Removes all preferences. This operation happens first, regardless of when it is called within an editing session; then all other changes are made and
SharedPreferences.Editor.remove()	Removes a specific preference by name. This operation happens first, regardless of when it is called within an editing session; then all other changes are made and committed.
SharedPreferences.Editor.putBoolean()	Sets a specific Boolean-type preference
	by name.
SharedPreferences.Editor.putFloat()	Sets a specific Float-type preference by name.
SharedPreferences.Editor.putInt()	Sets a specific Integer-type preference by name.

➤ [Storing Structured Data Using SQLite Databases](#)

- ✓ The Android SDK includes a number of useful SQLite database management classes. Many of these classes are found in the android.database.sqlite package. Here you can find utility classes for managing database creation and versioning, databasemanagement, and query builder helper classes to help you format proper SQL statements and queries.
- ✓ The package also includes specialized Cursor objects for iterating query results. You can also find all the specialized exceptions associated with SQLite.
- ✓ Here we focus on creating databases within our Android applications. For that, we use the built-in SQLite support to programmatically create and use a SQLite database to store.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- ✓ SQLite is a opensource SQL database that stores data to a text file on a device. Android comes in with built in SQLite database implementation.
- ✓ SQLite supports all the relational database features. In order to access this database, you don't need to establish any kind of connections for it like JDBC,ODBC e.t.c

➤ [Creating a SQLite Database Instance Using the Application Context](#)

- ✓ The simplest way to create a new SQLiteDatabase instance for your application is to use the openOrCreateDatabase() method of your application Context, like this:

```
1)
import android.database.sqlite.SQLiteDatabase;
...
SQLiteDatabase mDatabase;
mDatabase openOrCreateDatabase(
“my_sqlite_database.db”,
SQLiteDatabase.CREATE_IF_NECESSARY, null);
```

```
2)SQLiteDatabase mydatabase =
openOrCreateDatabase("your database
name",MODE_PRIVATE,null);
```

➤ [Finding the Application's Database File on the Device File System](#)

```
/data/data/com.androidbook.SimpleDatabase/databases/my_sqlite_database.db
```

➤ [Configuring the SQLite Database Properties](#)

- ✓ Now that you have a valid SQLiteDatabase instance, it's time to configure it. Some important database configuration options include version, locale, and the thread- safe locking feature.

```
import java.util.Locale;
...
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
mDatabase.setLocale(Locale.getDefault());  
mDatabase.setLockingEnabled(true);
```

```
mDatabase.setVersion(1);
```

➤ [Creating Tables and Other SQLite Schema Objects](#)

- ✓ Creating tables and other SQLite schema objects is as simple as forming proper SQLite statements and executing them. The following is a valid CREATE TABLE SQL statement.
- ✓ This statement creates a table called tbl_authors. The table has three fields: a uniqueid
- ✓ number, which auto-increments with each record and acts as our primary key, and firstname and lastname text fields:

1)

```
CREATE TABLE tbl_authors (  
id INTEGER PRIMARY KEY AUTOINCREMENT,  
firstname TEXT, lastname TEXT);
```

encapsulate this CREATE TABLE SQL statement in a static final String variable (called CREATE_AUTHOR_TABLE) and then execute it on your database using the execSQL() method:

```
mDatabase.execSQL(CREATE_AUTHOR_TABLE);
```

2)

```
mydatabase.execSQL("CREATE TABLE IF NOT EXISTS TutorialPoint(Username  
VARCHAR,Password VARCHAR);");  
mydatabase.execSQL("INSERT INTO TutorialPoint VALUES('admin','admin');");
```

execSQL(String sql Object[] bindArgs)

This method not only insert data but also used to update or modify already existing data in database using bind arguments

➤ [Creating and updating database with SQLiteOpenHelper](#)

- ✓ To create and upgrade a database in your Android application you create a subclass of the SQLiteOpenHelper class. In the constructor of your subclass you call the



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

super() method of SQLiteOpenHelper, specifying the database name and the current database version.

- ✓ In this class you need to override the following methods to create and update your database.
 - onCreate() - is called by the framework, if the database is accessed but not yet created.
 - onUpgrade() - called, if the database version is increased in your application code. This method allows you to update an existing database schema or to drop the existing database and recreate it via the onCreate() method.
- ✓ Both methods receive an SQLiteDatabase object as parameter which is the Java representation of the database.
- ✓ The SQLiteOpenHelper class provides the getReadableDatabase() and getWritableDatabase() methods to get access to an SQLiteDatabase object; either in read or write mode.
- ✓ The database tables should use the identifier `_id` for the primary key of the table. Several Android functions rely on this standard.

➤ [SQLiteDatabase](#)

- ✓ SQLiteDatabase is the base class for working with a SQLite database in Android and provides methods to open, query, update and close the database.
- ✓ More specifically SQLiteDatabase provides the insert(), update() and delete() methods.
- ✓ In addition it provides the execSQL() method, which allows to execute an SQL statement directly.
- ✓ The object ContentValues allows to define key/values. The *key* represents the table column identifier and the *value* represents the content for the table record in this column. ContentValues can be used for inserts and updates of database entries.
- ✓ Queries can be created via the rawQuery() and query() methods or via the SQLiteQueryBuilder class.
- ✓ rawQuery() directly accepts an SQL select statement as input. query() provides a structured interface for specifying the SQL query.
- ✓ SQLiteQueryBuilder is a convenience class that helps to build SQL queries.

➤ [Inserting Records](#)



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

✓ We use the insert() method to add new data to our tables. We use the ContentValues object to pair the column names to the column values for the record we want to insert.

✓ For example, here we insert a record into tbl_authors for J.K. Rowling:
import android.content.ContentValues;

...

```
ContentValues values = new ContentValues();
values.put("firstname", "J.K.");
values.put("lastname", "Rowling");
long newAuthorID = mDatabase.insert("tbl_authors", null, values);
```

✓ The insert() method returns the id of the newly created record. We use this author id to create book records for this author.

✓ The addContact() method accepts Contact object as parameter. We need to build ContentValues parameters using Contact object. Once we inserted data in database we need to close the database connection.

```
addContact()
// Adding new contact
public void addContact(Contact contact) {
    SQLiteDatabase db = this.getWritableDatabase();

    ContentValues values = new ContentValues();
    values.put(KEY_NAME, contact.getName()); // Contact
    Name
    values.put(KEY_PH_NO, contact.getPhoneNumber()); //
    Contact Phone Number

    // Inserting Row
    db.insert(TABLE_CONTACTS, null, values);
    db.close(); // Closing database connection
}
```

➤ [Updating Records](#)

You can modify records in the database using the update() method. The update() method takes four arguments:

- ✓ The table to update records
- ✓ A ContentValues object with the modified fields to update
- ✓ An optional WHERE clause, in which ? identifies a WHERE clause argument
- ✓ An array of WHERE clause arguments, each of which is substituted in place



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

of the '?'s from the second parameter

- ✓ Passing null to the WHERE clause modifies all records within the table, which can be useful for making sweeping changes to your database.
- ✓ Most of the time, we want to modify individual records by their unique identifier. The following function takes two parameters: an updated book title and a bookId. We find the record in the table called tbl_books that corresponds with the id and update that book's title. Again, we use the ContentValues object to bind our column names to our data values:

```
public void updateBookTitle(Integer bookId, String newtitle) {
    ContentValues values = new ContentValues();
    values.put("title", newtitle);
    mDatabase.update("tbl_books",
        values, "id=?", new String[] { bookId.toString() });
}
```

Because we are not updating the other fields, we do not need to include them in the

- ✓ ContentValues object. We include only the title field because it is the only field we change.

updateContact() will update single contact in database. This method accepts Contact class object as parameter.

```
updateContact()
    // Updating single contact
public int updateContact(Contact contact) {
    SQLiteDatabase db = this.getWritableDatabase();

    ContentValues values = new ContentValues();
    values.put(KEY_NAME, contact.getName());
    values.put(KEY_PH_NO, contact.getPhoneNumber());

    // updating row
    return db.update(TABLE_CONTACTS, values, KEY_ID + " =
    ?",
        new String[] { String.valueOf(contact.getID())
    });
}
```

➤ Deleting Records

- ✓ You can remove records from the database using the remove() method. The remove() method takes three arguments:
 - The table to delete the record from



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

- An optional WHERE clause, in which ? identifies a WHERE clause argument
 - An array of WHERE clause arguments, each of which is substituted in place of the ?'s from the second parameter
- ✓ Passing null to the WHERE clause deletes all records within the table. For example, this function call deletes all records within the table called tbl_authors:
- ✓ `mDatabase.delete("tbl_authors", null, null);`
- ✓ Most of the time, though, we want to delete individual records by their unique identifiers.
- ✓ The following function takes a parameter bookId and deletes the record corresponding to that unique id (primary key) within the table called tbl_books:

```
public void deleteBook(Integer bookId) {  
    mDatabase.delete("tbl_books", "id ?",  
        new String[] { bookId.toString() });  
}
```

`deleteContact()` will delete single contact from database.

```
deleteContact()  
    // Deleting single contact  
public void deleteContact(Contact contact) {  
    SQLiteDatabase db = this.getWritableDatabase();  
    db.delete(TABLE_CONTACTS, KEY_ID + " = ?",  
        new String[] { String.valueOf(contact.getID())  
    });  
    db.close();  
}
```

➤ [Working with Transactions](#)

- ✓ Often you have multiple database operations you want to happen all together or not at all.
- ✓ You can use SQL Transactions to group operations together; if any of the operations fails, you can handle the error and either recover or roll back all operations. If the operations all succeed, you can then commit them. Here we have the basic structure for a transaction:

```
mDatabase.beginTransaction();
```

```
try {  
    // Insert some records, updated others, delete a few  
    // Do whatever you need to do as a unit, then commit it
```




SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
mDatabase.setTransactionSuccessful();
} catch (Exception e) {
// Transaction failed. Failed! Do something here.
// It's up to you.
} finally {
mDatabase.endTransaction();

}
```

➤ Working with Cursors

- ✓ When results are returned from a SQL query, you often access them using a Cursor found in the android.database.Cursor class. Cursor objects are rather like file pointers; they allow random access to query results.
- ✓ You can think of query results as a table, in which each row corresponds to a returned record. The Cursor object includes helpful methods for determining how many results were returned by the query the Cursor represents and methods for determining the column names (fields) for each returned record. The columns in the query results are defined by the query, not necessarily by the database columns. These might include calculated columns, column aliases, and composite columns.
- ✓ Cursor objects are generally kept around for a time. If you do something simple (such as get a count of records or in cases when you know you retrieved only a single simple record), you can execute your query and quickly extract what you need; don't forget to close the Cursor when you're done, as shown here:

```
// SIMPLE QUERY: select * from tbl_books
Cursor c mDatabase.query("tbl_books",null,null,null,null,null);
// Do something quick with the Cursor here...
```

```
c.close();
```

- ✓ The following method getContact() will read single contact row. It accepts id as parameter and will return the matched row from the database.

```
getContact()
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
// Getting single contact
public Contact getContact(int id) {
    SQLiteDatabase db = this.getReadableDatabase();

    Cursor cursor = db.query(TABLE_CONTACTS, new String[]
    { KEY_ID,
        KEY_NAME, KEY_PH_NO }, KEY_ID + "=?",
    new String[] { String.valueOf(id) }, null,
    null, null, null);
    if (cursor != null)
        cursor.moveToFirst();

    Contact contact = new
    Contact(Integer.parseInt(cursor.getString(0)),
        cursor.getString(1), cursor.getString(2));
    // return contact
    return contact;
}
```

➤ [Executing Simple Queries](#)

- ✓ Your first stop for database queries should be the query() methods available in the SQLiteDatabase class. This method queries the database and returns any results as in a

Cursor object

```
return database.query(DATABASE_TABLE,

    new String[] { KEY_ROWID, KEY_CATEGORY, KEY_SUMMARY,
    KEY_DESCRIPTION },

    null, null, null, null, null);
```

[Parameters of the query\(\) method](#)

Parameter	Comment
String dbName	The table name to compile the query against.
String[] columnNames	A list of which table columns to return. Passing "null" will return all columns.
String whereClause	Where-clause, i.e. filter for the selection of data, null will select all data.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

String[] selectionArgs	You may include ?s in the "whereClause". These placeholders will get replaced by the values from the selectionArgs array.
String[] groupBy	A filter declaring how to group rows, null will cause the rows to not be grouped.
String[] having	Filter for the groups, null means no filter.
String[] orderBy	Table columns which will be used to order the data, null means no ordering.

1)

we called the query() method with only one parameter set to the table name.

```
Cursor c mDatabase.query("tbl_books",null,null,null,null,null,null);
```

This is equivalent to the SQL query

2)

```
SELECT * FROM tbl_books;
```

Add a WHERE clause to your query, so you can retrieve one record at a time:

```
Cursor c mDatabase.query("tbl_books", null, "id ?", new String[]{"9"}, null, null, null);
```

This is equivalent to the SQL query

```
SELECT * tbl_books WHERE id 9;
```

3)

For example,

if you need only the titles of each book in the book table, you might use the following

call to the query() method:

```
String asColumnsToReturn[] { "title", "id" };
```

```
String strSortOrder "title ASC";
```

```
Cursor c mDatabase.query("tbl_books", asColumnsToReturn, null, null, null, null, strSortOrder);
```

This is equivalent to the SQL query

```
SELECT title, id FROM tbl_books ORDER BY title ASC;
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

➤ [Deleting Tables and Other SQLite Objects](#)

You delete tables and other SQLite objects in exactly the same way you create them. Format the appropriate SQLite statements and execute them. For example, to drop our tables and triggers, we can execute three SQL statements:

```
mDatabase.execSQL("DROP TABLE tbl_books;");  
mDatabase.execSQL("DROP TABLE tbl_authors;");
```

```
mDatabase.execSQL("DROP TRIGGER IF EXISTS fk_insert_book;");
```

➤ [Keeping Track of Database Field Names](#)

✓ To make a class to encapsulate your database schema in a class, such as `PetDatabase`, shown here:

```
import android.provider.BaseColumns;  
public final class PetDatabase {  
    private PetDatabase() {}  
    public static final class Pets implements BaseColumns {  
        private Pets() {}  
        public static final String PETS_TABLE_NAME "table_pets";  
        public static final String PET_NAME "pet_name";  
        public static final String PET_TYPE_ID "pet_type_id";  
        public static final String DEFAULT_SORT_ORDER "pet_name ASC";  
    }  
    public static final class PetType implements BaseColumns {  
        private PetType() {}  
        public static final String PETTYPE_TABLE_NAME "table_pettypes";  
        public static final String PET_TYPE_NAME "pet_type";  
        public static final String DEFAULT_SORT_ORDER "pet_type ASC";  
    }  
}
```

➤ [Extending the SQLiteOpenHelper Class](#)

✓ To extend the `SQLiteOpenHelper` class, we must implement several important methods, which help manage the database versioning. The methods to



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

override are onCreate() , onUpgrade(), and onOpen().We use our newly defined PetDatabase class to generate appropriate SQL statements, as shown here:

```
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import com.androidbook.PetTracker.PetDatabase.PetType;
import com.androidbook.PetTracker.PetDatabase.Pets;

class PetTrackerDatabaseHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME “pet_tracker.db”;
    private static final int DATABASE_VERSION 1;
    PetTrackerDatabaseHelper(Context context) {

        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL(“CREATE TABLE “ + PetType.PETTYPE_TABLE_NAME+” (“
        + PetType._ID + “ INTEGER PRIMARY KEY AUTOINCREMENT ,”
        + PetType.PET_TYPE_NAME + “ TEXT”
        + “);”);
        db.execSQL(“CREATE TABLE “ + Pets.PETS_TABLE_NAME + “ (“
        + Pets._ID + “ INTEGER PRIMARY KEY AUTOINCREMENT ,”
        + Pets.PET_NAME + “ TEXT,”
        + Pets.PET_TYPE_ID + “ INTEGER” // FK to pet type table
        + “);”);
    }
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion,
    int newVersion){
        // Housekeeping here.
        // Implement how “move” your application data
        // during an upgrade of schema versions
        // Move or delete data as required. Your call.
    }
    @Override
    public void onOpen(SQLiteDatabase db) {
        super.onOpen(db);
    }

}
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

Sr.no	Questions	Answer
1	SQLite used for_____	Store Data in Device
2	onOpen(db) used for_____	Open Database
3	Is Sqlite support query?	Yes



Content provider with example

- ✓ In Android security model, one application cannot directly access (read/write) other application's data. Every application has its own id data directory and own protected memory area.
- ✓ Content provider is the best way to share data across applications. Content provider is a set of data wrapped up in a custom API to read and write. Applications/Processes have to register themselves as a provider of data. Other applications can request Android to read/write that data through a fixed API. Content provider API adheres to CRUD principle.
- ✓ Examples for content provider are Contacts-that exposes user information to other applications, Media store-Allows other applications to access,store media files.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

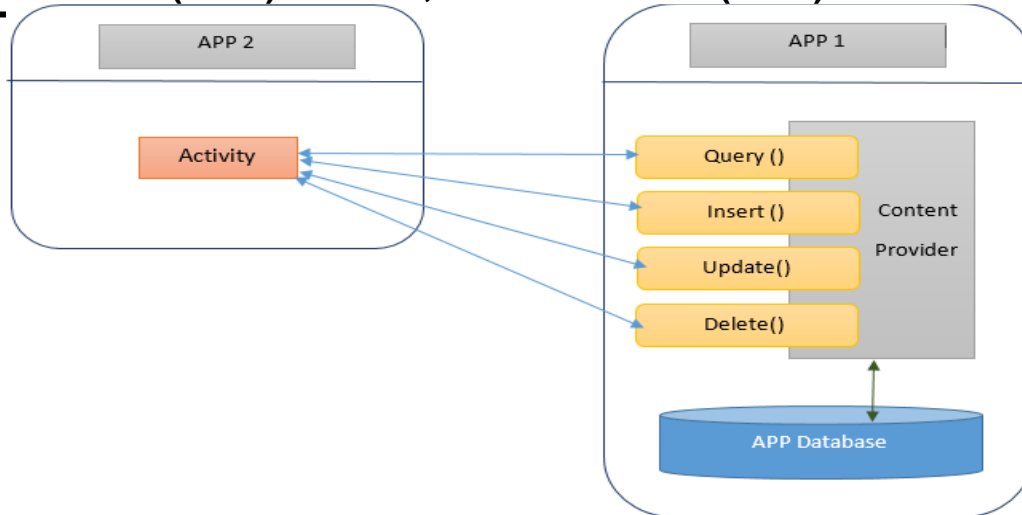
3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645



- ✓ Above diagram shows how content provider works. App 1 stores its data in its own database and provides a provider. App 2 communicates with the provider to access App 1's data.
- ✓ Content providers are simple interfaces which uses standard insert(), query(), update(), delete() methods to access application data. So it is easy to implement a content provider.
- ✓ A special URI starting with **content://** will be assigned to each content providers and that will be recognized across applications.

Writing a content provider:

- ✓ The ContentProvider class is the central component of a content provider. To create content provider we have to
 1. Create sub class for ContentProvider.
 2. Define content URI
 3. Implement all the unimplemented methods. insert(), update(), query(), delete(), getType().
 4. Declare the content provider in AndroidManifest.xml

Defining URI:

Content provider URI consists of four parts.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

[content://authority/path/id](#)

content:// All the content provider URIs should start with this value '**authority**' is Java namespace of the content provider implementation. (fully qualified Java package name)

'path' is the virtual directory within the provider that identifies the kind of data being requested.

'id' is optional part that specifies the primary key of a record being requested. We can omit this part to request all records.

Adding new records:

- ✓ We need to override insert() method of the ContentProvider to insert new record to the database via content provider. The caller method will have to specify the contentprovider URI and the values to be inserted without the ID. Successful insert operation will return the URI value with the newly inserted ID. For example: If we insert a new record to the content provider content://com.example/sample
- ✓ the insert method will return content://com.example/sample/1

Updating records:

- ✓ To update one or more records via content provider we need to specify the content provider URI. update() method of the ContentProvider is used to update records. We have to specify the ID of the record to update a single record. To update multiple records, we have to specify 'selection' parameter to indicate which rows are to be changed.
- ✓ This method will return the number of rows updated.

Deleting records:

- ✓ Deleting one or more records is similar to update process. We need to specify either ID or 'selection' to delete records. delete() method of the ContentProvider will return the number of records deleted.

Querying the content provider:



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- ✓ To query the data via content provider we override query() method of ContentProvider. This method has many parameters. We can specify list columns to
- ✓ put into the result cursor using 'projection' parameter. We can specify 'selection' criteria. We can specify 'sortOrder' too.
- ✓ If we do not specify projection, all the columns will be included in the result cursor. If we do not specify sortOrder the provider will choose its own sort order.

getType() method:

- ✓ This method is used to handle requests for the MIME type of the data at the given URI. We use either vnd.android.cursor.item or vnd.android.cursor.dir/vnd.android.cursor.item is used to represent specific item. Another one is used to specify all items.

Registering the provider in AndroidManifest.xml

- ✓ As with any other major parts of Android application, we have to register the content providers too in the AndroidManifest.xml. <provider> element is used to register the content providers. <application> is its parent element.

```
<provider
    android:name=".MyProvider"
    android:authorities="com.example.contentproviderexample.M
yProvider">
</provider>
```

- ✓ Here authorities is the URI authority to access the content provider. Typically this will be the fully qualified name of the content provider class.
- A content provider is implemented as a subclass of **ContentProvider** class and must implement a standard set of APIs that enable other applications to perform transactions.

```
public class MyContentProvider extends ContentProvider {
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- **onCreate()** This method is called when the provider is started.
- **query()** This method receives a request from a client. The result is returned as a Cursor object.
- **insert()** This method inserts a new record into the content provider.
- **delete()** This method deletes an existing record from the content provider.
- **update()** This method updates an existing record from the content provider.
- **getType()** This method returns the MIME type of the data at the given URI.

Content URIs

<prefix>://<authority>/<data_type>/<id>

Part	Description
------	-------------

prefix	This is always set to content://
--------	----------------------------------

authority	This specifies the name of the content provider, for example <i>contacts</i> , <i>browser</i> etc. For third-party content providers, this could be the fully qualified name, such as <i>com.tutorialspoint.statusprovider</i>
-----------	--

data_type	This indicates the type of data that this particular provider provides. For example, if you are getting all the contacts from the <i>Contacts</i> content provider, then the data path would be <i>people</i> and URI would look like this <i>content://contacts/people</i>
-----------	---

id	This specifies the specific record requested. For example, if you are looking for contact number 5 in the <i>Contacts</i> content provider then URI would look like this <i>content://contacts/people/5</i> .
----	---

Sample application:

- ✓ In this sample, we are going to create two application. First application will share its data through content provider and the second application will access the data.

1. Application 'MyProvider'

- ✓ This application will create a content provider and share its data. This uses SQLite database 'mydb'. It has a table called 'names'. The table names have two columns id, name.
- ✓ The URI of this content provider is *content://com.example.contentproviderexample.MyProvider/cte* We have layout to insert new records to the database table.

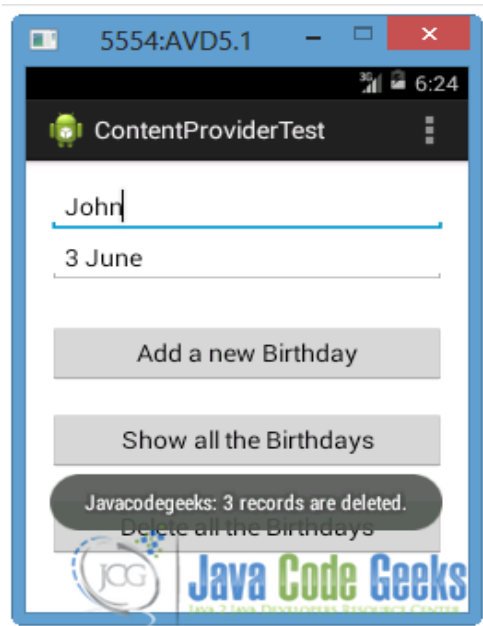
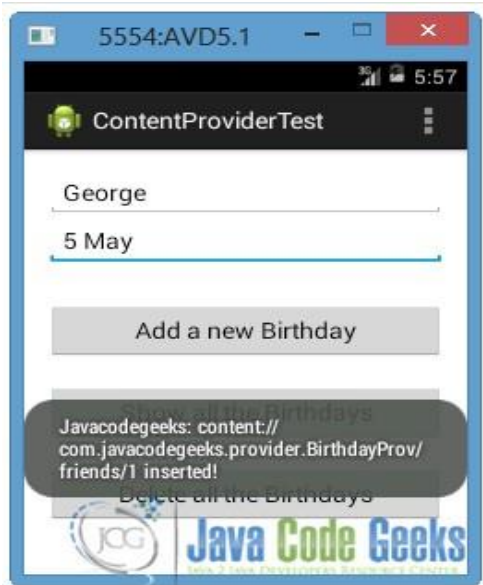


SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645



Code

- 1) Open res/values/strings.xml file to the “strings.xml” tab and paste the following code.

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <string name="hello">Hello World,
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

Contentex3Activity!</string>

```
<string name="app_name">ContentProviderTest</string>

    <string name="action_settings">Settings</string>
    <string name="birthday">Birthday</string>
    <string name="name">Name</string>
    <string name="add">Add a new Birthday</string>
    <string name="show">Show all Birthdays</string>
    <string name="delete">Delete all Birthdays</string>
    <dimen name="activity_vertical_margin">12pt</dimen>
    <dimen name="activity_horizontal_margin">12pt</dimen>
    <string name="BirthProvider">BirthProvider</string>
</resources>
```

2) Main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"

    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >

    <EditText
        android:id="@+id/name"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="@string/name" />

    <EditText
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
android:id="@+id/birthday"  
android:layout_width="fill_parent"  
android:layout_height="wrap_content"  
android:layout_alignLeft="@+id/name"  
android:layout_below="@+id/name"  
android:ems="10"  
android:hint="@string/birthday" />
```

<Button

```
android:id="@+id/btnAdd"  
android:layout_width="fill_parent"  
android:layout_height="wrap_content"  
android:onClick="addBirthday"  
android:layout_alignLeft="@+id/birthday"  
android:layout_below="@+id/birthday"  
android:layout_marginTop="30dp"  
android:text="@string/add" />
```

<Button

```
android:id="@+id/btnShow"  
android:layout_width="fill_parent"  
android:layout_height="wrap_content"  
android:layout_alignLeft="@+id/btnAdd"  
android:layout_below="@+id/btnAdd"  
android:layout_marginTop="20dp"  
android:onClick="showALLBirthdays"  
android:text="@string/show" />
```

<Button

```
android:id="@+id/btnDelete"  
android:layout_width="fill_parent"  
android:layout_height="wrap_content"  
android:layout_below="@+id/btnShow"  
android:layout_alignLeft="@+id/btnShow"  
android:layout_marginTop="20dp"  
android:onClick="deleteALLBirthdays"  
android:text="@string/delete" />
```

</RelativeLayout>

- 3) Right click to com.androidbook.contentex3 package → New → Class.
Specify the name for the new Class and the package you want to put it. We will name it BirthProvider and we will put it in the same package as



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

the MainActivity.java file.

Java Class
Create a new Java class.

Source folder:

Package:

Enclosing type:

Name:


Modifiers: public default private protected
 abstract final static

Superclass:

Interfaces:

Which method stubs would you like to create?
 public static void main(String[] args)
 Constructors from superclass
 Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))
 Generate comments

 **Java Code Geeks**
JAVA 2 JAVA DEVELOPERS RESOURCE CENTER

:2fba00d8bc729d89839f.javacodegeeks.netdna-cdn.com/wp-content/uploads/2013/11/BirthProviderClass.png



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

- ✓ BirthProvider is a subclass of [ContentProvider](#) and in order to identify the data in the provider we should use Content URIs. A [Content URI](#) has the following format:

1 [content://<authority>/<path>](#)

- ✓ At the <authority> part we specify the name of the provider (in our occasion is the com.javacodegeeks.provider.Birthday, as you can see in the code below) and the
- ✓ <path> part points to the table. If a specific record is requested, we will add the <id> of that record at the end of the Content URI.
- ✓ As we already mentioned we are going to use SQLite database, so in order to create and manage it we will use [SQLiteOpenHelper](#). Also, we are going to use the [UriMatcher Class](#), which maps the content URIs with particular patterns. That will help us to choose the desired action for an incoming content URI.
- ✓ In addition, we want our content provider to work properly so we need to override the functions onCreate(), query(), insert(), update(), delete(), getType(). To learn more about these functions, you can visit the [ContentProvider Class Overview](#).
- ✓ Open
src/com.javacodegeeks.android.contentprovidertest/BirthProvider.java and paste the following code:

```
package com.androidbook.contentex3;
```

```
import java.util.HashMap;
```

```
import android.content.ContentProvider;
```

```
import android.content.ContentUris;
```

```
import android.content.ContentValues;
```

```
import android.content.Context;
```

```
import android.content.UriMatcher;
```

```
import android.database.Cursor;
```

```
import android.database.SQLException;
```

```
import android.database.sqlite.SQLiteDatabase;
```

```
import android.database.sqlite.SQLiteDatabase.CursorFactory;
```

```
import android.database.sqlite.SQLiteOpenHelper;
```

```
import android.database.sqlite.SQLiteQueryBuilder;
```

```
import android.net.Uri;
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
import android.text.TextUtils;
import android.util.Log;

public class BirthProvider extends ContentProvider{
    // fields for my content provide
    static final String PROVIDER_NAME =
    "com.androidbook.contentex3.birthday";
    static final String URL = "content://" + PROVIDER_NAME +
    "/friends";
    static final Uri CONTENT_URI = Uri.parse(URL);

    // fields for the database
    static final String ID = "id";
    static final String NAME = "name";
    static final String BIRTHDAY = "birthday";

    // integer values used in content URI
    static final int FRIENDS = 1;
    static final int FRIENDS_ID = 2;

    DBHelper dbHelper;

    // projection map for a query
    private static HashMap<String, String> BirthMap;

    // maps content URI "patterns" to the integer values that were
    set above
    static final UriMatcher uriMatcher;

    static{
        uriMatcher = new UriMatcher(UriMatcher.NO_MATCH);
        uriMatcher.addURI(PROVIDER_NAME, "friends", FRIENDS);
        uriMatcher.addURI(PROVIDER_NAME, "friends/#",
        FRIENDS_ID);
    }

    // database declarations
    private SQLiteDatabase database;
    static final String DATABASE_NAME = "BirthdayReminder";
    static final String TABLE_NAME = "birthTable";
    static final int DATABASE_VERSION = 1;
    static final String CREATE_TABLE =
    " CREATE TABLE " + TABLE_NAME +
    " (id INTEGER PRIMARY KEY AUTOINCREMENT, " +
```




SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
" name TEXT NOT NULL, " +
" birthday TEXT NOT NULL);";

// class that creates and manages the provider's database
private static class DBHelper extends SQLiteOpenHelper{

    public DBHelper(Context context) {
super(context, DATABASE_NAME, null, DATABASE_VERSION);
// TODO Auto-generated constructor stub
    }

@Override
public void onCreate(SQLiteDatabase db) {
// TODO Auto-generated method stub
    db.execSQL(CREATE_TABLE);
}

@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion,
    int newVersion) {
// TODO Auto-generated method stub
    Log.w(DBHelper.class.getName(),
        "Upgrading database from version " + oldVersion +
" to "
        + newVersion + ". Old data will be destroyed");
    db.execSQL("DROP TABLE IF EXISTS " +
TABLE_NAME);
}
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
        onCreate(db);
    }

    }
    @Override
    public int delete(Uri uri, String selection, String[]
selectionArgs) {
        // TODO Auto-generated method stub
        int count = 0;

        switch (uriMatcher.match(uri)){
            case FRIENDS:
                // delete all the records of the table
                count = database.delete(TABLE_NAME, selection,
selectionArgs);
                break;
            case FRIENDS_ID:
                String id = uri.getLastPathSegment(); //gets the id
                count = database.delete( TABLE_NAME, ID + " = " + id +
                (!TextUtils.isEmpty(selection)
                ? " AND (" +selection + ')': "" ),
                selectionArgs);
                break;
            default:
                throw new IllegalArgumentException("Unsupported URI " +
uri);
        }

        getContext().getContentResolver().notifyChange(uri, null);
        return count;
    }

    @Override
    public String getType(Uri uri) {
        // TODO Auto-generated method stub
        switch (uriMatcher.match(uri)){
            // Get all friend-birthday records
            case FRIENDS:
                return "vnd.android.cursor.dir/vnd.example.friends";
                // Get a particular friend
            case FRIENDS_ID:
                return "vnd.android.cursor.item/vnd.example.friends";
        }
    }
}
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

default:

```
throw new IllegalArgumentException("Unsupported URI: " +  
uri);  
    }  
}
```

@Override

```
public Uri insert(Uri uri, ContentValues values) {  
    // TODO Auto-generated method stub
```

```
long row = database.insert(TABLE_NAME, "", values);
```

```
    // If record is added successfully
```

```
    if(row > 0) {
```

```
        Uri newUri =
```

```
ContentUris.withAppendedId(CONTENT_URI, row);
```

```
getContext().getContentResolver().notifyChange(newUri,
```

```
null);
```

```
return newUri;
```

```
    }
```

```
throw new SQLException("Fail to add a new record into " +  
uri);
```

```
}
```

@Override

```
public boolean onCreate() {
```

```
    // TODO Auto-generated method stub
```

```
Context context = getContext();
```

```
dbHelper = new DBHelper(context);
```

```
    // permissions to be writable
```

```
database = dbHelper.getWritableDatabase();
```

```
if(database == null)
```

```
    return false;
```

```
else
```

```
    return true;
```

```
}
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge
Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
public Cursor query(Uri uri, String[] projection, Stringselection,
    String[] selectionArgs, String sortOrder) {
    // TODO Auto-generated method stub
    SQLiteQueryBuilder queryBuilder = new
    SQLiteQueryBuilder();
        // the TABLE_NAME to query on
    queryBuilder.setTables(TABLE_NAME);

    switch (uriMatcher.match(uri)) {
        // maps all database column names
        case FRIENDS:
            queryBuilder.setProjectionMap(BirthMap);

            break;
        case FRIENDS_ID:
            queryBuilder.appendWhere( ID + "=" +
            uri.getLastPathSegment());

            break;
        default:
            throw new IllegalArgumentException("Unknown URI "
+ uri);
    }
    if (sortOrder == null || sortOrder == ""){
        // No sorting-> sort on names by default
        sortOrder = NAME;
    }
    Cursor cursor = queryBuilder.query(database, projection,
    selection,
        selectionArgs, null, null,
    sortOrder);
    /**
     * register to watch a content URI for changes
     */

    cursor.setNotificationUri(getContext().getContentResolver(),
    uri);

    return cursor;
}

@Override
public int update(Uri uri, ContentValues values, String
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Nr. Amrapali Under Bridge

Raiya Road

Raiya Road

Rajkot – 360001

Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

selection,

```
        String[] selectionArgs) {
    // TODO Auto-generated method stub
    int count = 0;

    switch (uriMatcher.match(uri)){
        case FRIENDS:
            count = database.update(TABLE_NAME, values,
selection, selectionArgs);

            break;
        case FRIENDS_ID:
            count = database.update(TABLE_NAME, values, ID +
                " = " + uri.getLastPathSegment() +
                (!TextUtils.isEmpty(selection) ? " AND ("
+
                selection + ')')
                : ""),

            break; default:
selectionArgs
    );
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
throw new IllegalArgumentException("Unsupported URI" + uri );  
}
```

```
getContext().getContentResolver().notifyChange(uri, null);  
return count;  
}  
}
```

4) Register the Content Provider

- ✓ We should register our provider, so the application could read from and write to our provider. To set the permissions, it is needed to add the provider to the AndroidManifest.xml file using <provider> ... </provider> tags.
- ✓ To add our provider, open AndroidManifest.xml file, at the last tab – named “AndroidManifest.xml” – and paste the following code.

```
<provider android:name="BirthProvider"  
android:authorities="com.androidbook.contentex3.birthday"></pr  
ovider>
```

- 5) When an application wants to access the data of a ContentProvider, it makes a request. These requests are handled by the ContentResolver object, which communicates with the ContentProvider as a client.

- ✓ In our example, we add three functions (deleteAllBirthdays (View view), addBirthday(View view), showAllBirthdays(View view)) to have user interaction with the application. The user can add a new record in our provider, delete or show all of them.

Open

src/com.javacodegeeks.android.contentprovidertest/MainAct
ivity.java and paste the following code.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
package com.androidbook.contentex3;
```

```
import android.app.Activity;
```

```
import android.content.ContentValues;
```

```
import android.database.Cursor;
```

```
import android.net.Uri;
```

```
import android.os.Bundle;
```

```
import android.view.Menu;
```

```
import android.view.View;
```

```
import android.widget.EditText;
```

```
import android.widget.Toast;
```

```
public class Contentex3Activity extends Activity {
```

```
    /** Called when the activity is first created. */
```

```
    @Override
```

```
    public void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.main);
```

```
    }
```

```
    public void deleteAllBirthdays (View view) {
```

```
        // delete all the records and the table of the
```

```
database provider
```

```
        String URL =
```

```
"content://com.androidbook.contentex3.birthday/friends";
```

```
        Uri friends = Uri.parse(URL);
```

```
        int count = getContentResolver().delete(friends, null,
```

```
        null);
```

```
        String countNum = "androidbook: " + count + " records are
```

```
deleted.";
```

```
        Toast.makeText(getBaseContext(),
```

```
            countNum, Toast.LENGTH_LONG).show();
```

```
    }
```

```
    public void addBirthday(View view) {
```

```
        // Add a new birthday record
```

```
        ContentValues values = new ContentValues();
```

```
        values.put(BirthProvider.NAME,
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot – 360001

Ph.No–(0281)2440478,2472590

3 – Vaishalinagar

Nr. Amrapali Under Bridge

Raiya Road

Rajkot - 360001

Ph.No–(0281)2471645

```
((EditText)findViewById(R.id.name)).getText().toString();

values.put(BirthProvider.BIRTHDAY,

((EditText)findViewById(R.id.birthday)).getText().toString());

Uri uri = getContentResolver().insert(
    BirthProvider.CONTENT_URI, values);

Toast.makeText(getBaseContext(),

    "androidbook:" + uri.toString() + "
    inserted!", Toast.LENGTH_LONG).show();
}

public void showAllBirthdays(View view) {
    // Show all the birthdays sorted by
friend's name
    String URL =
"content://com.androidbook.contentex3.birthday/friends";
    Uri friends = Uri.parse(URL);
    Cursor c = getContentResolver().query(friends, null, null,
null,
    "name");
    String result = "androidbook Results:";

    if (!c.moveToFirst()) {
        Toast.makeText(this, result+" no content yet!",
            Toast.LENGTH_LONG).show();
    }
}
```




SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
else{
do{
    result = result + "\n" +
c.getString(c.getColumnIndex(BirthProvider.NAME)) +
        " with id " +
    c.getString(c.getColumnIndex(BirthProvider.ID)) +
        " has birthday: " +
    c.getString(c.getColumnIndex(BirthProvider.BIRTHDAY));
    } while (c.moveToNext());
    Toast.makeText(this, result,
Toast.LENGTH_LONG).show();
    }
}
```

Sr.no	Questions	Answer
1	Is android support Toast?	Yes
2	Is Android support cursor?	Yes
3	ContentProvider provides _____	Data through Query
4	Full form of URI _____	Uniform Resource Identifier

UNIT – 5 LOCATION BASED SERVICES (LBS), COMMON ANDROID API, NOTIFICATIONS, SERVICES, DEPLOYMENT OF APPLICATIONS



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

1. Overview of HTTP access on Android

1.1. Available official API's

Android contains the standard Java network java.net package which can be used to access network resources. The base class for HTTP network access in the java.net package is the HttpURLConnection class.

1.2. Required permissions

To access the Internet your application requires the android.permission.INTERNET permission.

To check the network state your application requires the android.permission.ACCESS_NETWORK_STATE permission.

1.3. Open source libraries

Performing network operations on Android can be cumbersome. You have to open and close a connections, enable caches and ensure to perform the network operation in a background thread.

To simplify these operations several popular Open Source libraries are available. The most popular ones are the following:

- Volley
- OkHttp

2. Good practices for network access under Android

Within an Android application you should avoid performing long running operations on the user interface thread. This includes file and network access.

As of Android 3.0 (Honeycomb) the system is configured to crash with a NetworkOnMainThreadException exception, if network is accessed in the user interface thread.

A typical setup for performing network access in a productive Android application is using a service. While it is possible to do network access from an activity or a fragment, using a



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

**2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590**

**3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645**

service typical leads to a better overall design because you code in the activity becomes simpler.

3. Java and HTTP access

Java provides a general-purpose, lightweight HTTP client API to access resources via the HTTP or HTTPS protocol. The main classes to access the Internet are the `java.net.URL` class and the `java.net.HttpURLConnection` class.

The `URL` class can be used to define a pointer to a web resource while the `HttpURLConnection` class can be used to access a web resource.

`HttpURLConnection` allows you to create an `InputStream`.

Once you have accessed an `InputStream` you can read it similarly to an `InputStream` from a local file.

In the latest version `HttpURLConnection` supports the transparent response compression (via the header `Accept-Encoding: gzip`, `Server Name Indication` (extension of `SSL` and `TLS`) and a response cache.

The API is relatively straight forward. For example to retrieve the webpage `www.vogella.com` you can use the following example.

```
try {
    URL url = new URL("http://www.vogella.com");
    HttpURLConnection con = (HttpURLConnection) url
        .openConnection();
    readStream(con.getInputStream());
} catch (Exception e) {
    e.printStackTrace();
}
```

```
private void readStream(InputStream in) {
    BufferedReader reader = null;
    try {
        reader = new BufferedReader(new InputStreamReader(in));
        String line = "";
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

**2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590**

**3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645**

```
while ((line = reader.readLine()) != null) {
    System.out.println(line);
}
} catch (IOException e) {
    e.printStackTrace();
} finally {
    if (reader != null) {
        try {
            reader.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
}
```

4. Check the network availability

Obviously the network on an Android device is not always available. You can check the network is currently available via the following code.

```
public boolean isNetworkAvailable() {
    ConnectivityManager cm = (ConnectivityManager)
        getSystemService(Context.CONNECTIVITY_SERVICE);
    NetworkInfo networkInfo = cm.getActiveNetworkInfo();
    // if no network is available networkInfo will be null
    // otherwise check if we are connected
    if (networkInfo != null && networkInfo.isConnected()) {
        return true;
    }
    return false;
}
```

5. Proxy

This chapter is only relevant for you if you are testing with the Android simulator behind a proxy. You can set the proxy via the Settings class. For example you could add the following line to your onCreate method in your *activity*.

```
Settings.System.putString(getContentResolver(),
    Settings.System.HTTP_PROXY, "myproxy:8080");
```



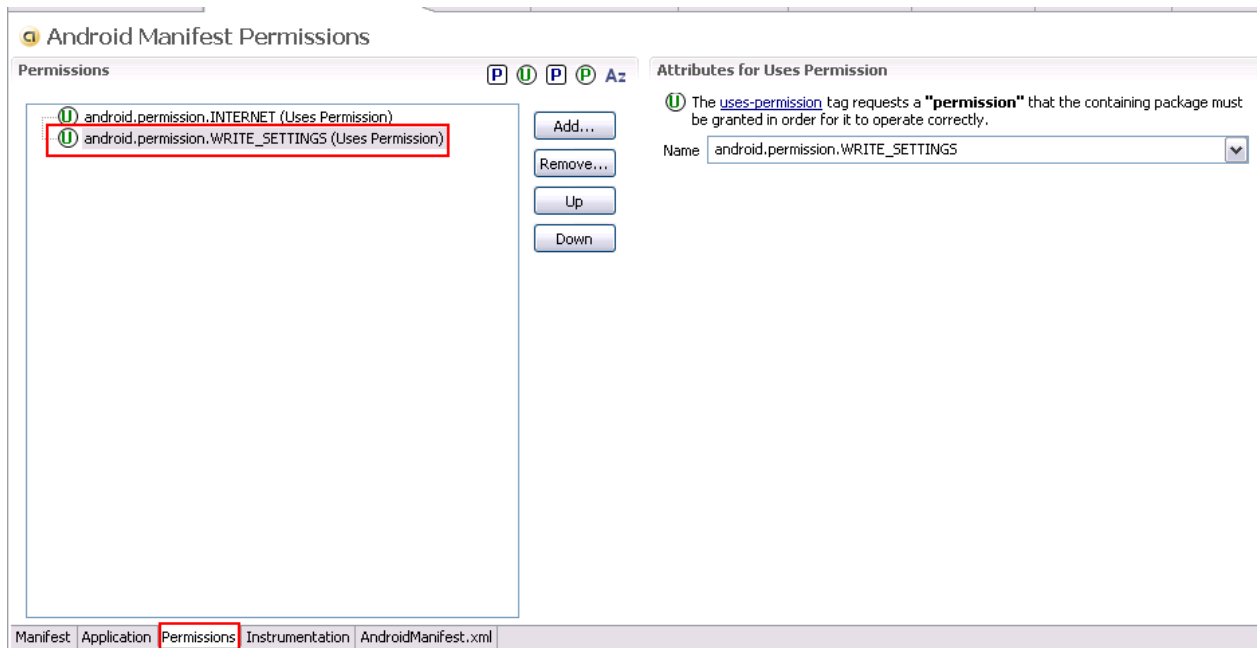
SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

To change the proxy settings you have to have the android.permission.WRITE_SETTINGS permission in your AndroidManifest.xml file.



[Parsing XML Data](#)

Extensible Markup Language (XML) is a set of rules for encoding documents in machine-readable form. XML is a popular format for sharing data on the internet. Websites that frequently update their content, such as news sites or blogs, often provide an XML feed so that external programs can keep abreast of content changes. Uploading and parsing XML data is a common task for network-connected apps. This lesson explains how to parse XML documents and use their data.

[Choose a Parser](#)

We recommend [XmlPullParser](#), which is an efficient and maintainable way to parse XML on Android. Historically Android has had two implementations of this interface:

- [KXmlParser](#) via [XmlPullParserFactory.newPullParser\(\)](#).
- [ExpatPullParser](#), via [Xml.newPullParser\(\)](#).



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.
(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

Either choice is fine. The example in this section uses ExpatPullParser, via [Xml.newPullParser\(\)](#)

Analyze the Feed

The first step in parsing a feed is to decide which fields you're interested in. The parser extracts data for those fields and ignores the rest.

Here is an excerpt from the feed that's being parsed in the sample app. Each post to [StackOverflow.com](#) appears in the feed as an entry tag that contains several nested tags:

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom"
xmlns:creativecommons="http://backend.userland.com/creativeComm
onsRssModule" ...">
<title type="text">newest questions tagged android - Stack
Overflow</title>
...
  <entry>
    ...
  </entry>
  <entry>
    <id>http://stackoverflow.com/q/9439999</id>
    <re:rank scheme="http://stackoverflow.com">0</re:rank>
    <title type="text">Where is my data file?</title>
    <category
scheme="http://stackoverflow.com/feeds/tag?tagnames=android&sort=
newest/tags" term="android"/>
    <category
scheme="http://stackoverflow.com/feeds/tag?tagnames=android&sort=
newest/tags" term="file"/>
    <author>
      <name>cliff2310</name>
      <uri>http://stackoverflow.com/users/1128925</uri>
    </author>
    <link rel="alternate"
href="http://stackoverflow.com/questions/9439999/where-is-my-
data-file" />
    <published>2012-02-25T00:30:54Z</published>
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
<updated>2012-02-25T00:30:54Z</updated>
<summary type="html">
  <p>I have an Application that requires a data
file...</p>
</summary>
</entry>
<entry>
  ...
</entry>
...
</feed>
```

The sample app extracts data for the entry tag and its nested tags title, link, and summary.

Instantiate the Parser

The next step is to instantiate a parser and kick off the parsing process. In this snippet, a parser is initialized to not process namespaces, and to use the provided [InputStream](#) as its input. It starts the parsing process with a call to [nextTag\(\)](#) and invokes the `readFeed()` method, which extracts and processes the data the app is interested in:

```
public class StackOverflowXmlParser {
    // We don't use namespaces
    private static final String ns = null;

    public List parse(InputStream in) throws
    XmlPullParserException, IOException {
        try {
            XmlPullParser parser = Xml.newPullParser();
            parser.setFeature(XmlPullParser.FEATURE_PROCESS_NAMESPACES, false);
            parser.setInput(in, null);
            parser.nextTag();
            return readFeed(parser);
        } finally {
            in.close();
        }
    }
}
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
}  
}  
...  
}
```

Read the Feed

The `readFeed()` method does the actual work of processing the feed. It looks for elements tagged "entry" as a starting point for recursively processing the feed. If a tag isn't an entry tag, it skips it. Once the whole feed has been recursively processed, `readFeed()` returns a [List](#) containing the entries (including nested data members) it extracted from the feed. This [List](#) is then returned by the parser.

```
private List readFeed(XmlPullParser parser) throws  
XmlPullParserException, IOException {  
    List entries = new ArrayList();  
  
    parser.require(XmlPullParser.START_TAG, ns, "feed");  
    while (parser.next() != XmlPullParser.END_TAG) {  
        if (parser.getEventType() != XmlPullParser.START_TAG) {  
            continue;  
        }  
        String name = parser.getName();  
        // Starts by looking for the entry tag  
        if (name.equals("entry")) {  
            entries.add(readEntry(parser));  
        } else {  
            skip(parser);  
        }  
    }  
    return entries;  
}
```

Telephony API

[The Android Telephony package](#)

Programmers will need ways to not just retrieve telephony data, but also to dial a phone number, intercept outgoing phone calls, or send/receive SMS messages. This is achieved with a combination of classes provided in the [android.telephony package](#), along with inbuilt Intents and phone services.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT. (AFFILIATED TO SAURASHTRA UNIVERSITY)

**2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590**

**3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645**

The most important of these is the [TelephonyManager class](#), which can be used to retrieve metadata about the SIM card, the physical device and the network provider. Programmers will need to use the Android Context, through the `getSystemService()` method, with a constant as shown below, to obtain an instance of the TelephonyManager class.

```
// ..creating TelephonyManager from Context
final TelephonyManager mytelMgr =
    (TelephonyManager) this.getSystemService(Context.TELEPHONY_SERVICE);
```

With this handle, telephony data can be retrieved. Sample code to retrieve the phone's call state, using the `getCallState()` method, is shown below.

```
public String getCurrentCallState(final TelephonyManager
mytelMgr) {
    int callState = mytelMgr.getCallState();
    String callStateString = "NOTKNOWN";
    switch (callState) {
        case TelephonyManager.CALL_STATE_IDLE:
            callStateString = "IDLE";
            break;
        case TelephonyManager.CALL_STATE_OFFHOOK:
            callStateString = "OFFHOOK";
            break;
        case TelephonyManager.CALL_STATE_RINGING:
            callStateString = "RINGING";
            break;
    }
}
```

Additionally, it is often important to know the change in the call or service state of the phone, in an app. For example, you may want to mute your application when a call arrives. This is done by attaching a listener, called [PhoneStateListener](#) to the TelephonyManager.

Finally, it is important to note that information retrieval via TelephonyManager is permission-protected. For the application to access this information, the `android.permission.READ_PHONE_STATE` permission has to be set in the app's manifest.xml. Figure 2 shows some metrics retrieved using TelephonyManager.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

The **android.telephony.TelephonyManager** class provides information about the telephony services such as subscriber id, sim serial number, phone network type etc. Moreover, you can determine the phone state etc.

[activity_main.xml](#)

```
<RelativeLayout xmlns:androclass="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >
```

```
<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:layout_marginLeft="38dp"
    android:layout_marginTop="30dp"
    android:text="Phone Details:" />
```

</RelativeLayout>

MainActivity.java

```
package com.javatpoint.telephonymanager;
```

```
import android.os.Bundle;
import android.app.Activity;
import android.content.Context;
import android.telephony.TelephonyManager;
import android.view.Menu;
import android.widget.TextView;
```

```
public class MainActivity extends Activity {
    TextView textView1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT “SKY IS THE LIMIT”



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

**2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590**

**3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645**

```
textView1=(TextView)findViewById(R.id.textView1);
```

```
//Get the instance of TelephonyManager  
TelephonyManager tm=(TelephonyManager) getSystemService(Context.TELEPHONY_  
SERVICE);
```

```
//Calling the methods of TelephonyManager the returns the information  
String IMEINumber=tm.getDeviceId();  
String subscriberID=tm.getDeviceId();  
String SIMSerialNumber=tm.getSimSerialNumber();  
String networkCountryISO=tm.getNetworkCountryIso();  
String SIMCountryISO=tm.getSimCountryIso();  
String softwareVersion=tm.getDeviceSoftwareVersion();  
String voiceMailNumber=tm.getVoiceMailNumber();
```

```
//Get the phone type  
String strphoneType="";
```

```
int phoneType=tm.getPhoneType();
```

```
switch (phoneType)  
{  
    case (TelephonyManager.PHONE_TYPE_CDMA):  
        strphoneType="CDMA";  
        break;  
    case (TelephonyManager.PHONE_TYPE_GSM):  
        strphoneType="GSM";  
        break;  
    case (TelephonyManager.PHONE_TYPE_NONE):  
        strphoneType="NONE";  
        break;  
}
```

```
//getting information if phone is in roaming  
boolean isRoaming=tm.isNetworkRoaming();
```

```
String info="Phone Details:\n";  
info+="\n IMEI Number:"+IMEINumber;  
info+="\n SubscriberID:"+subscriberID;  
info+="\n Sim Serial Number:"+SIMSerialNumber;
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT. (AFFILIATED TO SAURASHTRA UNIVERSITY)

**2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590**

**3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645**

```
info+="\n Network Country ISO:"+networkCountryISO;  
info+="\n SIM Country ISO:"+SIMCountryISO;  
info+="\n Software Version:"+softwareVersion;  
info+="\n Voice Mail Number:"+voiceMailNumber;  
info+="\n Phone Network Type:"+strphoneType;  
info+="\n In Roaming? :"+isRoaming;
```

```
textView1.setText(info);//displaying the information in the textView  
}
```

```
}
```

AndroidManifest.xml

```
<uses-sdk  
    android:minSdkVersion="8"  
    android:targetSdkVersion="17" />  
  
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
```



Web View

If you want to deliver a web application (or just a web page) as a part of a client application, you can do it using [WebView](#). The [WebView](#) class is an extension of Android's [View](#) class that



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

**2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590**

**3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645**

allows you to display web pages as a part of your activity layout. It does *not* include any features of a fully developed web browser, such as navigation controls or an address bar. All that [WebView](#) does, by default, is show a web page.

A common scenario in which using [WebView](#) is helpful is when you want to provide information in your application that you might need to update, such as an end-user agreement or a user guide. Within your Android application, you can create an [Activity](#) that contains a [WebView](#), then use that to display your document that's hosted online. Another scenario in which [WebView](#) can help is if your application provides data to the user that always requires an Internet connection to retrieve data, such as email. In this case, you might find that it's easier to build a Web your Android application that shows a web page with all the user data, rather than performing a network request, then parsing the data and rendering it in an Android layout. Instead, you can design a web page that's tailored for Android devices and then implement a [WebView](#) in your Android application that loads the web page.

This document shows you how to get started with [WebView](#) and how to do some additional things, such as handle page navigation and bind JavaScript from your web page to client-side code in your Android application.

Adding a WebView to your application

To add a [WebView](#) to your Application, simply include the `<WebView>` element in your activity layout. For example, here's a layout file in which the [WebView](#) fills the screen:

```
<?xml version="1.0" encoding="utf-8"?>
<WebView xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/webview"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
/>
```

To load a web page in the [WebView](#), use [loadUrl\(\)](#). For example:

```
WebView myWebView = (WebView) findViewById(R.id.webview);
myWebView.loadUrl("http://www.example.com");
```

To get Internet access, request the [INTERNET](#) permission in your manifest file. For example:

```
<manifest ... >
    <uses-permission android:name="android.permission.INTERNET"
/>
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT. (AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

...
</manifest>

Sr.no	Questions	Answer
1	SQLite used for___	Store Data in Device
2	onOpen(db) used for___	Open Database
3	Is Sqlite support query?	Yes

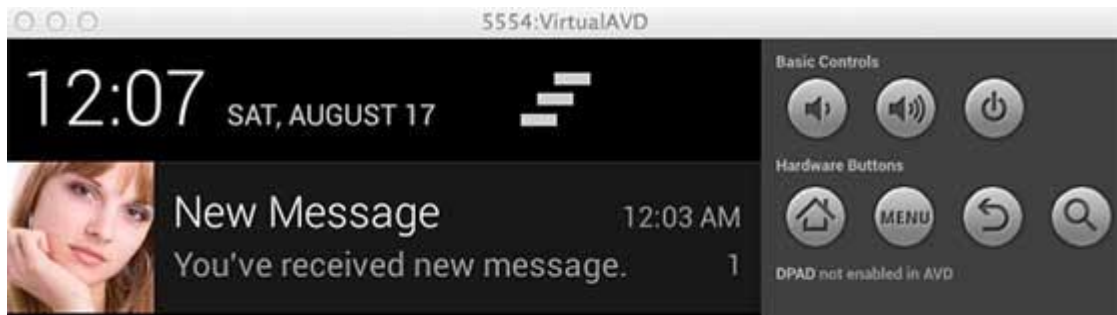
Notifications

Android **Toast** class provides a handy way to show users alerts but problem is that these alerts are not persistent which means alert flashes on the screen for a few seconds and then disappears.

For important messages to be given to the user, it is required to have more persistent method. A **notification** is a message you can display as an icon at the top of the device which we call notification bar or status bar.



To see the details of the notification, you will have to select the icon which will display notification drawer having detail about the notification. While working with emulator with virtual device, you will have to click and drag down the status bar to expand it which will give you detail as follows. This will be just **64 dp** tall and called normal view.





SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

Notification Manager

2.1. Notification Manager

Android allows to put notification into the titlebar of your application. The user can expand the notification bar and by selecting the notification the user can trigger another activity.

An [Activity](#) or [Service](#) can initiate a status bar notification. Because an Activity can perform actions only while it is active and in focus, you should create your status bar notifications from a Service. This way, the notification can be created from the background, while the user is using another application or while the device is asleep. To create a notification, you must use two classes: [Notification](#) and [NotificationManager](#).

Use an instance of the [Notification](#) class to define the properties of your status bar notification, such as the status bar icon, the expanded message, and extra settings such as a sound to play. The [NotificationManager](#) is an Android system service that executes and manages all Notifications. You do not instantiate the NotificationManager. In order to give it your Notification, you must retrieve a reference to the NotificationManager with [getSystemService\(\)](#) and then, when you want to notify the user, pass it your Notification object with [notify\(\)](#).

2.2. Setting up Notifications

Notifications in Android are represented by the Notification class. To create notifications you use the NotificationManager class which can be received from the Context, e.g. an activity or a service, via the getSystemService() method.

```
NotificationManager notificationManager = (NotificationManager)
getSystemService(NOTIFICATION_SERVICE);
```

```
Notification noti = new Notification.Builder(this)
    .setContentTitle("New mail from " + "test@gmail.com")
    .setContentText("Subject").setSmallIcon(R.drawable.icon)
    .setContentIntent(pIntent)
    .addAction(R.drawable.icon, "Call", pIntent)
    .addAction(R.drawable.icon, "More", pIntent)
    .addAction(R.drawable.icon, "And more", pIntent).build();
NotificationManager notificationManager = (NotificationManager)
getSystemService(NOTIFICATION_SERVICE);
// hide the notification after its selected
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
noti.flags |= Notification.FLAG_AUTO_CANCEL;
```

```
notificationManager.notify(0, noti);
```

Introduction to Notifications

Notifications are used to alert users on some events that requires their attention. Notifications alert the users through various forms:

- Display a status bar icon
- Flash the LED
- Vibrate
- Ringtones

Notifying with the status bar

Status bar notifications indicates some ongoing background services such as downloading or playing music or displays an alert/information. To see the notification user needs to open the notification drawer. Notifications are handled by Notification Manager in Android. Notification Manager is a system service used to manage notifications.

To create a status bar notification we need to use two classes, **NotificationManager**, **Notification**.

Lets see an simple example to display notification in status bar.

The NotificationManager is a system Service used to manage Notification. Get a reference to it using the getSystemService() method.

```
NotificationManager notificationManager = (NotificationManager)  
getSystemService(NOTIFICATION_SERVICE);
```

Vibrating the phone

Vibrating Android device is found effective when you wanted to get notified on the incoming connection in various cases like, user cannot hear a ringtone or he wants to get notified silently. Its a very basic feature found in all mobile phones. Lets see how to vibrate an android device programatically. Its just two lines of code.

Vibrate pattern

```
public abstract void vibrate (long[] pattern, int repeat)
```

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT “SKY IS THE LIMIT”



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

**2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590**

**3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645**

Pattern for vibration is nothing but an array of duration's to turn ON and OFF the vibrator in milliseconds. The first value indicates the number of milliseconds to wait before turning the vibrator ON. The next value indicates the number of milliseconds for which to keep the vibrator on before turning it off. Subsequent values, alternates between ON and OFF.

```
long pattern[]={0,100,200,300,400};
```

If you feel not to have repeats, just pass -1 for 'repeat'. To repeat patterns, just pass the index from where u wanted to start. I wanted to start from 0'th index and hence I am passing 0 to 'repeat'.

```
vibrator.vibrate(pattern, 0);
```

Android manifest permission for Vibration

Last but not the least add permission to Android manifest file.

```
<uses-permission android:name="android.permission.VIBRATE"/>
```

Activity.java

```
import android.os.Bundle;  
import android.os.Vibrator;  
import android.app.Activity;  
import android.content.Context;  
import android.view.View;
```

```
public class MainActivity extends Activity {  
    public Vibrator vibrator;
```

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
}
```

```
public void startVibrate(View v) {  
    long pattern[] = { 0, 100, 200, 300, 400 };  
    vibrator = (Vibrator) getSystemService(Context.VIBRATOR_SERVICE);  
    vibrator.vibrate(pattern, 0);
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

}

```
public void stopVibrate(View v) {  
    vibrator.cancel();
```

```
}
```

```
}
```

➤ Blinking the lights

If you have a notification LED flashing, you know that your phone has something to tell you. But you don't necessarily know what. Light Flow allows you to set different LED colors for different notifications. You could have a yellow light flash when you have a missed call, a green light when you have a text message, a blue light when you have a new email, and a red light when the phone is low on battery and needs to be plugged in. You can control which types of notifications produce a notification light – Light Flow supports notifications from over 550 different apps.

Services

Services in Android is a component that runs in the background without any user interaction or interface. Unlike Activities services does not have any graphical interface. Services run invisibly performing long running tasks - doing Internet look ups, playing music, triggering notifications etc.,

Services run with a higher priority than inactive or invisible activities and therefore it is less likely that the Android system terminates them for resource management. The only reason Android will stop a Service prematurely is to provide additional resources for a foreground component usually an Activity. When this happens, your Service can be configured to restart automatically.

Creating a Service

Create a new class that extends Service. Below is the code snippet.

```
import android.app.Service;  
import android.content.Intent;  
import android.os.IBinder;
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
public class MyService extends Service {  
    public void onCreate() {  
        super.onCreate();  
        // This method is invoked when the service is called.  
    }  
    public IBinder onBind() {  
        super.onCreate();  
        // A client or activity is binded to service  
    }  
}
```

Services takes Two Forms

Started

A Service is started when a application component such as an activity invokes it by calling `startService()`. Once started, a service can run in the background indefinitely, even if the component that started it is destroyed

Bound

A service is "bound" when an application component binds to it by calling `bindService()`. A bound service offers a client-server interface that allows components to interact with the service, send requests, get results, and even do so across processes with inter-process communication.

AndroidManifest.xml

```
<service  
    android:name=".MyService"  
    android:enabled="true" />
```

A service is a component that runs in the background to perform long-running operations without needing to interact with the user. For example, a service might play music in the background while the user is in a different application, or it might fetch data over the network without blocking user interaction with an activity. A service can essentially take two states:

State	Description
Started	A service is started when an application component, such as an activity, starts it by calling <code>startService()</code> . Once started, a service can run in the



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

	background indefinitely, even if the component that started it is destroyed.
Bound	A service is bound when an application component binds to it by calling <code>bindService()</code> . A bound service offers a client-server interface that allows components to interact with the service, send requests, get results, and even do so across processes with interprocess communication (IPC).

A service has lifecycle callback methods that you can implement to monitor changes in the service's state and you can perform work at the appropriate stage. The following diagram on the left shows the lifecycle when the service is created with `startService()` and the diagram on the right shows the lifecycle when the service is created with `bindService()`: (image courtesy : android.com)

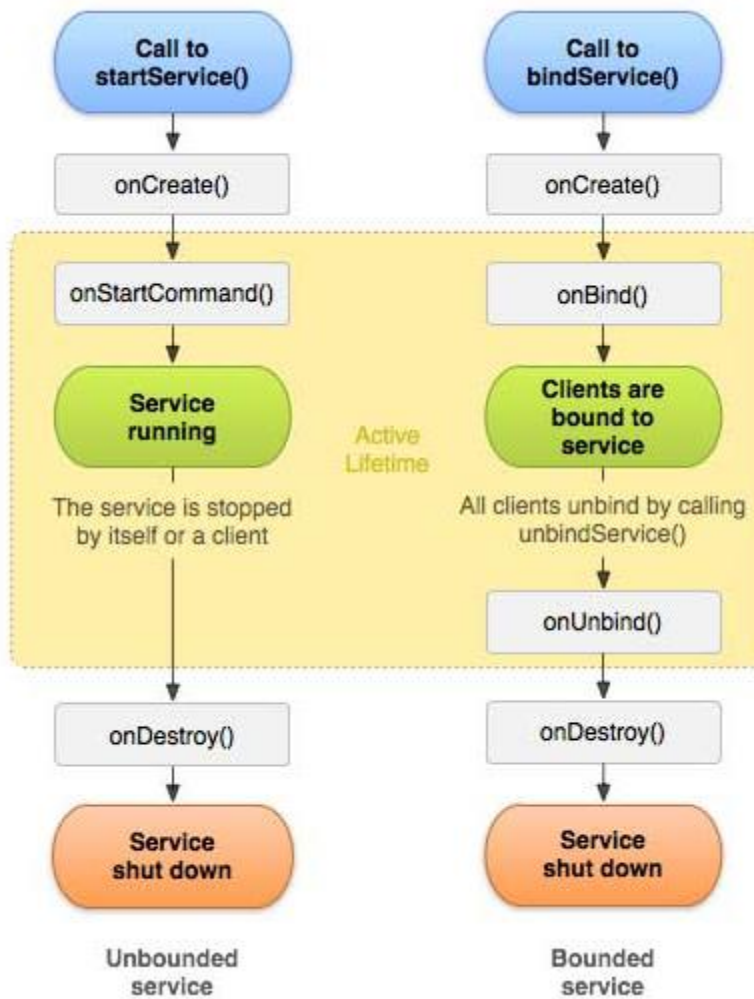


SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
 Nr. Amrapali Under Bridge
 Raiya Road
 Rajkot – 360001
 Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
 Nr. Amrapali Under Bridge
 Raiya Road
 Rajkot - 360001
 Ph.No–(0281)2471645



To create an service, you create a Java class that extends the Service base class or one of its existing subclasses. The **Service** base class defines various callback methods and the most important are given below. You don't need to implement all the callbacks methods. However, it's important that you understand each one and implement those that ensure your app behaves the way users expect.

Callback	Description
onStartCommand()	The system calls this method when another component, such as an activity, requests that the service be started, by calling startService(). If you implement this method, it is your responsibility to stop the service when its work is done, by calling stopSelf() or stopService() methods.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

onBind()	The system calls this method when another component wants to bind with the service by calling <code>bindService()</code> . If you implement this method, you must provide an interface that clients use to communicate with the service, by returning an <i>IBinder</i> object. You must always implement this method, but if you don't want to allow binding, then you should return null.
onUnbind()	The system calls this method when all clients have disconnected from a particular interface published by the service.
onRebind()	The system calls this method when new clients have connected to the service, after it had previously been notified that all had disconnected in its <code>onUnbind(Intent)</code> .
onCreate()	The system calls this method when the service is first created using <code>onStartCommand()</code> or <code>onBind()</code> . This call is required to perform one-time setup.
onDestroy()	The system calls this method when the service is no longer used and is being destroyed. Your service should implement this to clean up any resources such as threads, registered listeners, receivers, etc.

Service Overview

We already know that Android Activity can be started, stopped, destroyed if the system resources become too low and maybe can be recreated, **a Service are designed to have a longer life. A Service in Android can be started from an Activity, from a Broadcast receiver and other services too.**

We have to notice that using Service we don't create automatically new threads, so if we implement a simple logic inside our Service, that doesn't require long time processing we don't need to run it a separate thread, but if we have to implement complex logic, with long time processing, we have to take care of creating a new thread, otherwise the service runs on the main thread and it could cause ANR problem.

In Android a Service is used for two main reason:

- Implement multi-task
- Enable Inter-Process-Communication (IPC)

A typical example of the first case is an app that required to download data from a remote server, in this case we can have Activity that interacts with a user and starts a service that accomplishes



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

the work in background while the user can use the app and maybe when the service finishes sends a message to the user.

In the second case, we want to “share” some common functions so that different app can re-use them. For example, we can suppose we have a service that sends an email and we want to share this service among several apps so that we don't have to rewrite the same code. In this case we can use IPC so that the service exposes a “remote” interface that can be called by other app.

JSON

JSON stands for JavaScript Object Notation. It is an independent data exchange format and is the best alternative for XML. This chapter explains how to parse the JSON file and extract necessary information from it.

Android provides four different classes to manipulate JSON data. These classes are **JSONArray**, **JSONObject**, **JSONStringer** and **JSONTokenizer**.

In the JSON given below we are interested in getting temperature only.

```
{
  "sys":
    {
      "country": "GB",
      "sunrise": 1381107633,
      "sunset": 1381149604
    },
  "weather": [
    {
      "id": 711,
      "main": "Smoke",
      "description": "smoke",
      "icon": "50n"
    }
  ],
  "main":
    {
      "temp": 304.15,
      "pressure": 1009,

```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
}  
}
```

JSON - Elements

An JSON file consist of many components. Here is the table defining the compoents of an JSON file and their description:

Sr.No	Component & description
1	Array() In a JSON file , square bracket ([]) represents a JSON array
2	Objects() In a JSON file, curly bracket ({}) represents a JSON object
3	Key A JSON object contains a key that is just a string. Pairs of key/value make up a JSON object
4	Value Each key has a value that could be string , integer or double e.t.c

JSON - Parsing

For parsing a JSON object, we will create an object of class JSONObject and specify a string containing JSON data to it. Its syntax is:

```
String in;  
JSONObject reader = new JSONObject(in);
```

The last step is to parse the JSON. An JSON file consist of different object with different key/value pair e.t.c. So JSONObject has a seperate function for parsing each of the component of JSON file. Its syntax is given below:

```
JSONObject sys = reader.getJSONObject("sys");  
country = sys.getString("country");
```

```
JSONObject main = reader.getJSONObject("main");  
temperature = main.getString("temp");
```

The method **getJSONObject** returns the JSON object. The method **getString** returns the string value of the specified key.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

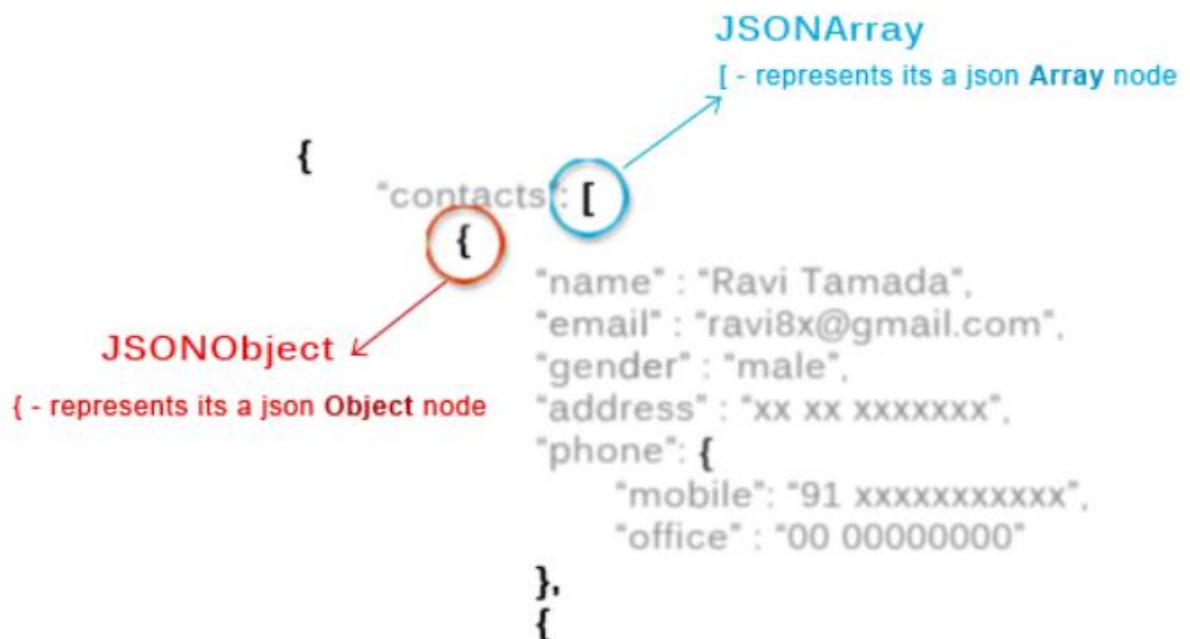
The difference between [and { - (Square brackets and Curly brackets)

In general all the JSON nodes will start with a square bracket or with a curly bracket. The difference between [and { is, the square bracket ([) represents starting of a **JSONArray** node whereas curly bracket ({) represents **JSONObject**. So while accessing these nodes we need to call appropriate method to access the data.

If your JSON node starts with [, then we should use `getJSONArray()` method. Same as if the node starts with {, then we should use `getJSONObject()` method.

JSON Structure

AndroidHiv



```
<uses-permission android:name="android.permission.INTERNET" />
```

JSON (Javascript Object Notation) is a programming language . It is minimal, textual, and a subset of JavaScript. It is an alternative to XML.

Android provides support to parse the JSON object and array.

SHREE H.N.SHUKLA COLLEGE OF I.T & MGMT “SKY IS THE LIMIT”



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

Advantage of JSON over XML

- 1) JSON is faster and easier than xml for AJAX applications.
- 2) Unlike XML, it is shorter and quicker to read and write.
- 3) It uses array.

json object

A JSON object contains key/value pairs like map. The keys are strings and the values are the JSON types. Keys and values are separated by comma. The { (curly brace) represents the json object.

```
{
  "employee": {
    "name": "sachin",
    "salary": 56000,
    "married": true
  }
}
```

json array

The [(square bracket) represents the json array.

```
["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"]
{ "Employee" :
  [
    {"id":"101","name":"Sonoo Jaiswal","salary":"50000"},
    {"id":"102","name":"Vimal Jaiswal","salary":"60000"}
  ]
}
```

activity_main.xml

```
<RelativeLayout xmlns:androclass="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
```



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
tools:context=".MainActivity" >
```

```
<TextView  
    android:id="@+id/textView1"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignParentLeft="true"  
    android:layout_alignParentTop="true"  
    android:layout_marginLeft="75dp"  
    android:layout_marginTop="46dp"  
    android:text="TextView" />
```

```
</RelativeLayout>
```

Activity.java

```
package com.javatpoint.jsonparsing;
```

```
import org.json.JSONException;  
import org.json.JSONObject;  
import android.app.Activity;  
import android.os.Bundle;  
import android.widget.TextView;
```

```
public class MainActivity extends Activity {  
    public static final String JSON_STRING="{\"employee\":{\"name\":\"Sachin\",\"salary\":56000}}";
```

```
@Override
```

```
public void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);
```

```
    TextView textView1=(TextView)findViewById(R.id.textView1);
```

```
    try{  
        JSONObject emp=(new JSONObject(JSON_STRING)).getJSONObject("employee");  
        String empname=emp.getString("name");  
        int empsalary=emp.getInt("salary");
```

```
        String str="Employee Name:"+empname+"\n"+"Employee Salary:"+empsalary;  
        textView1.setText(str);
```

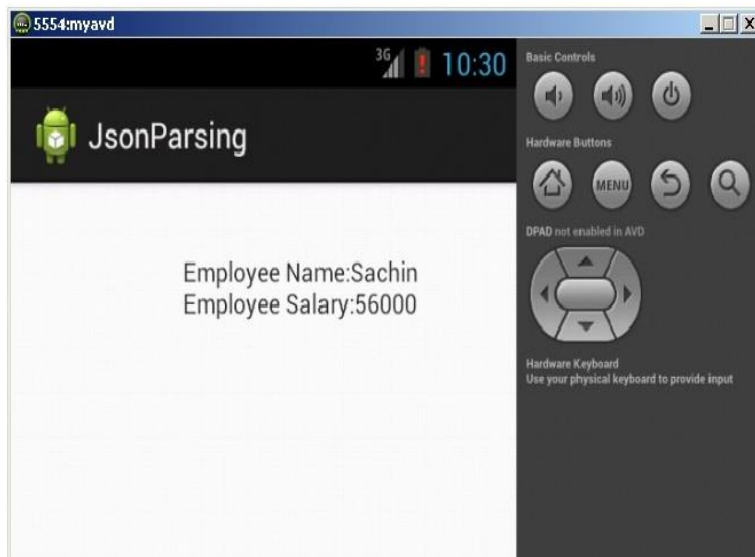


SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.
(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

```
}catch (Exception e) {e.printStackTrace();}  
  
}  
  
}
```



Deployment of applications

➤ [Get Started with Publishing](#)

Start publishing on Google Play in minutes by:

- Registering for a Google Play publisher account
- Setting up a Google Wallet Merchant Account, if you will sell apps or in-app products.
- Exploring the [Google Play Developer Console](#) and publishing tools.

When you're ready, use the Start button to go to the Developer Console.

[Register for a Publisher Account](#)



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645


1. Visit the [Google Play Developer Console](#).
2. Enter basic information about your **developer identity** — name, email address, and so on. You can modify this information later.
3. Read and accept the **Developer Distribution Agreement** for your country or region. Note that apps and store listings that you publish on Google Play must comply with the Developer Program Policies and US export law.
4. Pay a **\$25 USD registration fee** using Google Wallet. If you don't have a Google Wallet account, you can quickly set one up during the process.
5. When your registration is verified, you'll be notified at the email address you entered during registration.

Set Up a Google Wallet Merchant Account



If you want to sell priced apps, in-app products, or subscriptions, you'll need a Google Wallet Merchant Account. You can set one up at any time, but first review the list of [merchant countries](#).

To set up a Google Wallet Merchant Account:

1. **Sign in** to your Google Play Developer Console at
2. <https://play.google.com/apps/publish/>.
3. Open **Financial reports**  on the side navigation.
4. Click **Setup a Merchant Account now**.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

This takes you to the Google Wallet site; you'll need information about your business to complete this step.

Explore the Developer Console

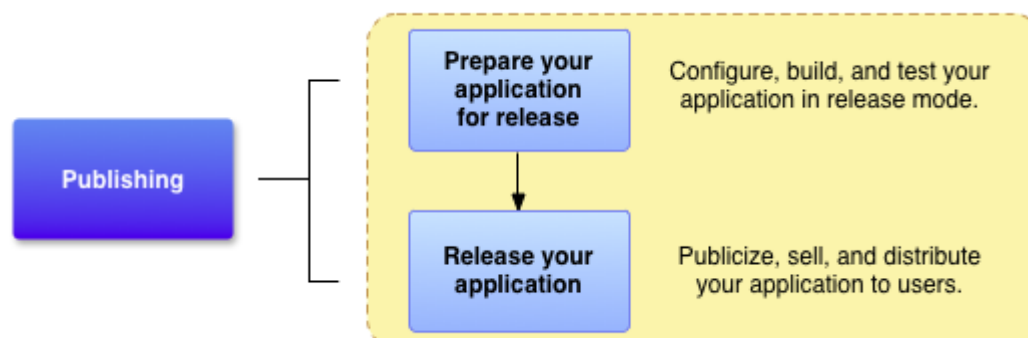
When your registration is verified, you can sign in to your Developer Console, which is the home for your app publishing operations and tools on Google Play.

The screenshot shows the Google Play Developer Console interface. At the top, there's a search bar and user information. The main area displays a table of applications under the heading 'ALL APPLICATIONS'. The table has columns for App Name, Price, Active / Total Installs, Avg. Rating / Total #, Crashes & ANRS, Last Update, and Status. Two applications are listed: 'Animal Translator 1.1' and 'Earthquake! 3.8'. A sidebar on the left contains navigation options like 'All applications', 'Financial reports', 'Settings', and 'Announcements'. A '+ Add new application' button is visible above the table.

APP NAME	PRICE	ACTIVE / TOTAL INSTALLS	AVG. RATING / TOTAL #	CRASHES & ANRS	LAST UPDATE	STATUS
Animal Translator 1.1	Free	12,078 / 185,410	★ 3.29 / 566		Apr 21, 2010	Published
Earthquake! 3.8	Free	71,426 / 785,829	★ 4.15 / 6,212		Feb 1, 2013	Published

➤ Preparing for Release

Before you distribute your Android application to users you need to prepare it for release. The preparation process is a required [development task](#) for all Android applications and is the first step in the publishing process (see figure 1).





SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

**2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590**

**3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645**

When you prepare your application for release, you configure, build, and test a release version of your application. The configuration tasks are straightforward, involving basic code cleanup and code modification tasks that help optimize your application. The build process is similar to the debug build process and can be done using JDK and Android SDK tools. The testing tasks serve as a final check, ensuring that your application performs as expected under real-world conditions. When you are finished preparing your application for release you have a signed .apk file, which you can distribute directly to users or distribute through an application marketplace such as Google Play.

This document summarizes the main tasks you need to perform to prepare your application for release. The tasks that are described in this document apply to all Android applications regardless how they are released or distributed to users. If you are releasing your application through Google Play, you should also read [Publishing Checklist for Google Play](#) to be sure your release-ready application satisfies all Google Play requirements.

Publishing is the general process that makes your Android applications available to users. When you publish an Android application you perform two main tasks:

- You prepare the application for release.

During the preparation step you build a release version of your application, which users can download and install on their Android-powered devices.

- You release the application to users.

During the release step you publicize, sell, and distribute the release version of your application to users.

Usually, you release your application through an application marketplace, such as [Google Play](#). However, you can also release applications by sending them directly to users or by letting users download them from your own website.

Figure 1 shows how the publishing process fits into the overall Android [application development process](#). The publishing process is typically performed after you finish testing your application in a debug environment. Also, as a best practice, your application should meet all of your release criteria for functionality, performance, and stability before you begin the publishing process.



SHREE H. N. SHUKLA COLLEGE OF I.T. & MGMT.

(AFFILIATED TO SAURASHTRA UNIVERSITY)

2 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot – 360001

3 – Vaishalinagar
Nr. Amrapali Under Bridge
Raiya Road
Rajkot - 360001

Ph.No–(0281)2440478,2472590 Ph.No–(0281)2471645

