

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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



Lt. Shree Chimanbhai Shukla

MSCIT– SEM-1 - LARAVEL

Website: www.hnsgroupofcolleges.org

Email : hnsinfo@hnshukla.com

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Sr No.	Topic	Details	Marks	Min Lec.
1	Object Oriented Programming in PHP	<ul style="list-style-type: none"> • The Basics • Class Constants • Autoloading Classes • Constructors and Destructors • Visibility • Object Inheritance • Scope Resolution Operator (::) • Static Keyword • Class Abstraction • Object Interfaces • Anonymous classes • Overloading • Object Iteration • Magic Methods • Final Keyword • Object Cloning • Comparing Objects • Type Hinting • Late Static Bindings • Objects and references 		
	Bootstrap Basics	<ul style="list-style-type: none"> • Introduction: File Structure • Basic HTML Template • Default Grid System • Basic Grid HTML • Offsetting Columns • Nesting Columns • What Is Responsive Design? • Code • Tables • Forms • Buttons • Images • Icons 		

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

		<ul style="list-style-type: none"> • Glyphicons • Dropdown Menus • Button Groups • Navigations • Navbar • Breadcrumb • Pagination • Label • Badges • Thumbnails • Alerts • progress bar • wells 		
2	Introduction to Laravel, Installation & Configuration	<ul style="list-style-type: none"> • What is Laravel • Features • MVC architecture • structure of laravel application (laravel directory structure) • Basic requirements for Laravel • Using Laravel Installer • Using Composer • Finding and installing new packages 		
3	Artisan	<ul style="list-style-type: none"> • Artisan Command Line Tool • database creation • artisan migration • migration structure • creation migration • Database seeding 		
	Routing in Laravel	<ul style="list-style-type: none"> • Basic Routing • Route Parameters • Route Filters • Named Routes • Route Groups • Sub-Domain Routing • Route Prefixing • Route Model Binding • Throwing 404 Errors 		

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

4	Blade Template	<ul style="list-style-type: none"> • Template inheritance • Master layout • Extending the master layout • display variables • Blade conditional statements • Blade Loops • Executing PHP functions in blade 		
	SQL Interaction	<ul style="list-style-type: none"> • Introduction • Running Raw SQL Queries • Database Transactions 		
5	Eloquent ORM	<ul style="list-style-type: none"> • Eloquent ORM Models: Naming conventions • table name & primary keys • timestamps Basic Operations: Create, Retrieve, Update, Delete Using Models, displaying data from models in views. 		
	Validation	<ul style="list-style-type: none"> • Defining The Routes • Creating The Controller • Writing The Validation Logic • Displaying The Validation Errors, Array validations, creating new validators, Error messages & custom errors • Available Validators: Accepted, After (Date), Alpha, Alpha Dash, Alpha Numeric, Array, Before (Date), Between, Boolean, Date, Date Format, Different, Digits, Digits Between, E-Mail, Exists (Database), Image (File), In, Integer, Max, Min, Not In, Numeric, Regular Expression, Required, String Custom validation rules. 		

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

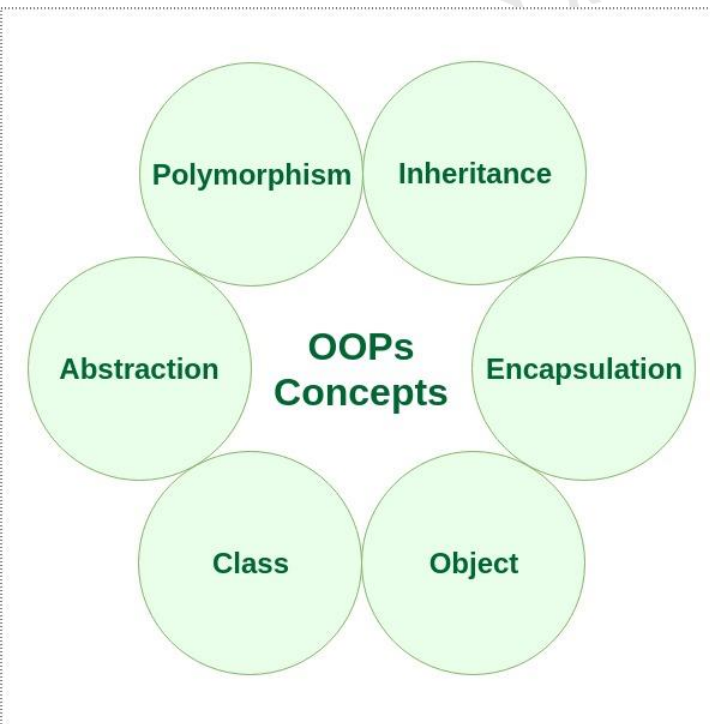
UNIT – 1

Object Oriented Programming in PHP

Topic 1 : Explain concept of OOP.

Marks-2

Ans.:



Detailing :

As the name suggests, Object-Oriented Programming or OOPs refers to languages that uses objects in programming. OOP first came into picture in 1970's by Alan Kay and his team.

Object-oriented programming aims to implement real-world entities like inheritance, hiding, polymorphism etc in programming.

OOPs Concepts:

- Polymorphism
- Inheritance
- Encapsulation
- Abstraction
- Class
- Object

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

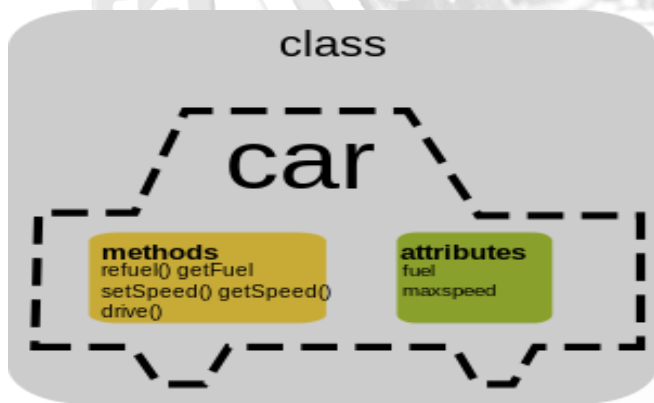
1 Word Question Answer

Sr. No.	Question	Answer
1.	OOP stands for?	Object Oriented Programming
2.	Who invented OOP?	Alan Kay & Team
3.	When OOP came into picture?	1970's
4.	OOP implements..	Real World Entities

Topic 2 : What is Class in oop ?

Marks-2

Ans.:



Detailing :

A class is a user defined blueprint or prototype from which objects are created. It represents the set of properties or methods that are common to all objects of one type. In general, class declarations can include these components, in order:

1. Modifiers: A class can be public or has default access (Refer this for details).
2. Class name: The name should begin with a initial letter (capitalized by convention).

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

3. Body: The class body surrounded by braces, { }.

Example of class :

```
<?php  
class Fruit {  
    // code goes here...  
}  
?>
```

1 Word Question Answer

Sr. No.	Question	Answer
1.	What is class?	User Defined blueprint
2.	Structure of class & subclasses called?	Class hierarchy
3.	Class name should begin with..	Letter
4.	The class body surrounded by..	Curly Braces

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



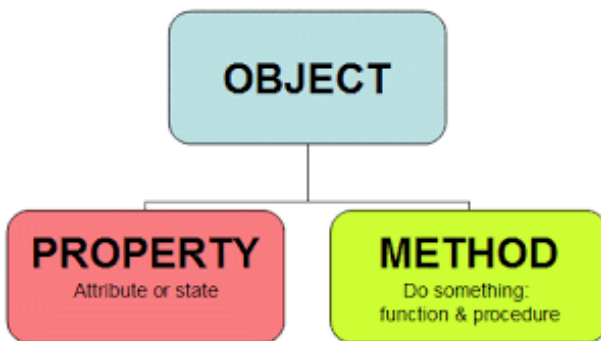
2 – Vaishalinagar
Nr. Amrapali Railway Crossing
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

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

Topic 3 : What is Object and Property?

Marks-2

Ans :



Detailing :

It is a basic unit of Object Oriented Programming and represents the real life entities. An object consists of :

1. State : It is represented by attributes of an object. It also reflects the properties of an object.
2. Behaviour : It is represented by methods of an object. It also reflects the response of an object with other objects.
3. Identity : It gives a unique name to an object and enables one object to interact with other objects.

Example :

```
<?php
class Fruit {
    public $name;
    function set_name($name) {
        $this->name = $name;
    }
}
$apple = new Fruit();
$apple->set_name("Apple");
?>
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

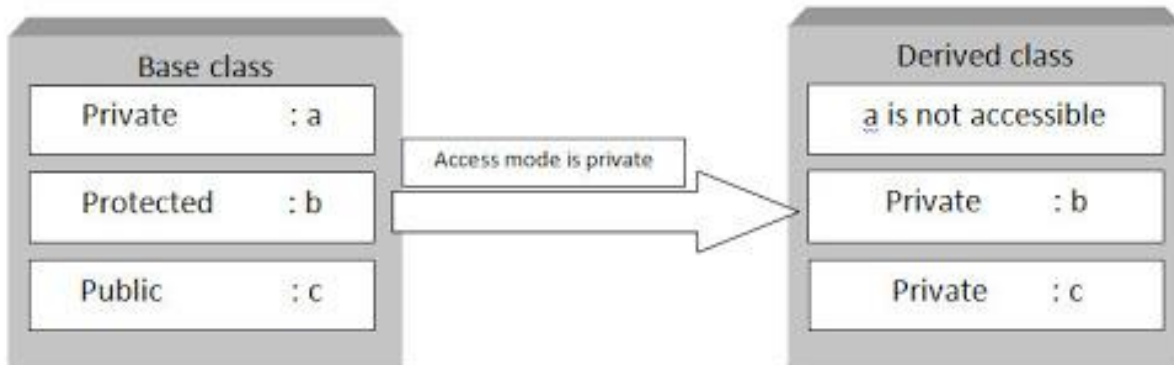
1 Word Question Answer

Sr. No.	Question	Answer
1.	What is object?	Instance of class
2.	What is property?	Data member
3.	What is behaviour?	Method / Function
4.	What is identity?	Name of Object

Topic 4 : What are visibility levels in oop.

Marks-3

Ans. :



Detailed :

Encapsulation not only provides a convenient way of treating an object as a single entity, but it also provides protection of the object by controlling what parts of the object are visible outside of the object.

This visibility is specifically controlled by access specifiers that define the level of visibility:

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

→ Public

- Public members are accessible any place the object is visible
- Public methods should be the only way to change an object's attributes
- Public members make up an object's public interface.

→ Private

- Private members are accessible only from other members of the same class
 - i.e. once instantiated, only that object.

→ Protected

- Protected members are accessible from other members of the same class and to members of derived classes derived
 - i.e. a child class can also access the protected members of the parent.

Example :

```
<?php
class MyClass
{
    public $a = 'Public';
    protected $b = 'Protected';
    private $c = 'Private';

    function display()
    {
        echo $this->a;
        echo $this->b;
        echo $this->c;
    }
}
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
$obj = new MyClass();  
echo $obj->a; // Works  
echo $obj->b; // Fatal Error  
echo $obj->c; // Fatal Error  
$obj->display(); // Shows Public, Protected and Private
```

1 Word Question Answer

Sr. No.	Question	Answer
1.	What is visibility levels in oop?	Way to protect data
2.	Public keyword is known as	Visibility level
3.	Protected keyword is known as	Visibility level
4.	Private keyword is known as	Visibility level
5.	Which access specifier is used globally.	Public

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



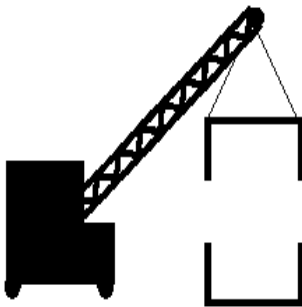
2 – Vaishalinagar
Nr. Amrapali Railway Crossing
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

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

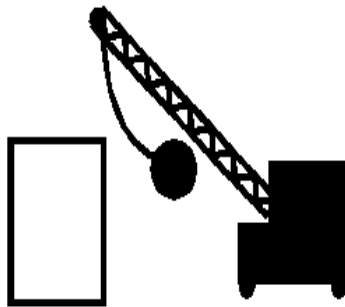
Topic 5 : What is constructor and destructor in oop?

Marks-5

Ans :



Constructor



Destructor

```
MyClass *MyObjPtr = new MyClass();
```

```
delete MyObjPtr;
```

Detailed :

→ Constructor

If a class name and function name will be similar in that case function is known as user defined constructor. Constructor is special type of method because its name is similar to class name. Constructor automatically calls when object will be initializing. PHP introduce a new functionality to define a constructor using `__construct()`.

By using this function we can define a constructor. It is known as predefined constructor. Its better than user defined constructor because if we change class name then user defined constructor treated as normal method. If predefined constructor and user defined constructor, both define in the same class, then predefined constructor treat like a Constructor while user defined constructor treated as normal method.

For Example :

```
class a
{
    public function a()
    {
        //This is user defined constructor;
    }
}
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
public function __construct()
{
    //This is predefined constructor;
}
}
```

→ Destructor

PHP destructor allows you to clean up resources before PHP releases the object from the memory. To add a destructor to a class, you just simply add a special method called `__destruct()` as follows:

```
public function __destruct()
{
    // clean up resources here
}
```

Example of constructor & destructor

```
<?php
class Animal {
    public $name = "NoName";
    public function __construct($name) {
        echo "I'm alive! <br>";
        $this->name = $name;
    }
    public function __destruct() {
        echo "<br> I'm dead now :(";
    }
}
$animal = new Animal("Dog");
echo "Name of the animal: " . $animal->name;
?>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

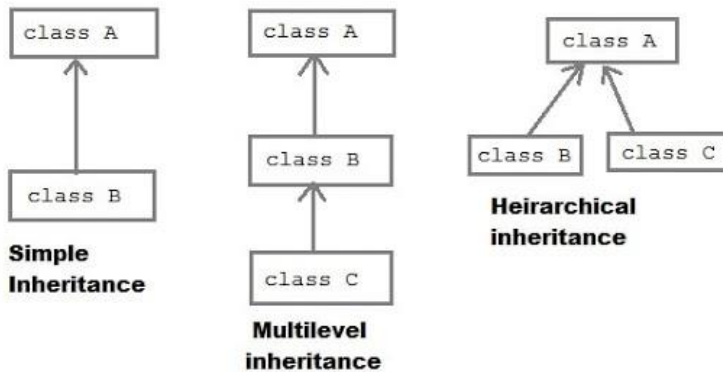
1 Word Question Answers

Sr. No.	Question	Answer
1.	Which function is used for constructor?	__construct()
2.	Which function is used for destructor?	__destruct()
3.	Constructor is a...	Magic Method
4.	Destructor is a...	Magic Method

Topic 6 : Explain Inheritance in detail.

Marks-5

Ans. :



Detailed :

- **Inheritance** is a mechanism of extending an existing class by inheriting a class we create a new class with all functionality of that existing class, and we can add new members to the new class.
- When we inherit one class from another we say that inherited class is a subclass and the class who has inherit is called parent class.
- We declare a new class with additional keyword **extends**.
- Note : PHP only supports multilevel inheritance.

Syntax :

```
class Parent {
// The parent's class code
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
}  
  
class Child extends Parent {  
    // The child can use the parent's class code  
}
```

Example :

```
<?php  
class ParentClass  
{  
    public function ParentFunction()  
    {  
        echo "Shree H.N.Shukla College<br>";  
    }  
}  
class ChildClass extends ParentClass  
{  
    public function ChildFunction()  
    {  
        echo "Rajkot.";  
    }  
}  
$obj=new ChildClass();  
$obj->ParentFunction();  
$obj->ChildFunction();  
?>
```

1 Word Question Answer

Sr. No.	Question	Answer
1.	Which keyword is used for inheritance?	extends
2.	Inherited class known as?	sub class
3.	We can use parent class methods in subclass.	True
4.	PHP supports multiple inheritance.	False

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

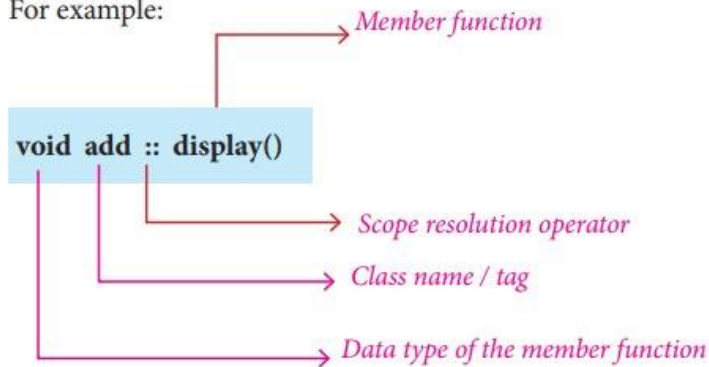
3 – Vaishalinagar
Nr. Amrapali Railway Crossing
Raiya Road
Rajkot - 360001
Ph.No–(0281)2471645

Topic 7 : Explain scope resolution operator in oop.

Marks-3

Ans. :

For example:



Detailed :

- The scope resolution operator or in simpler terms, the double colon (::), is a token that allows access to static, constant methods of a class.
- When referencing these items from outside the class definition, use the name of the class.
- Static variables or methods defined in same class can be access using self keyword.
- Static variables or methods defined in parent class and can be access from the child class using parent keyword.

Example #1 :: from outside the class definition

```
<?php
class demo
{
    const user="admin";
}
echo demo::user;
echo "<br>";
$obj=new demo();
echo $obj::user;
?>
```

Example #2 :: from inside the class definition

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<?php
class demo
{
    const user="admin";
    public function test()
    {
        echo self::user;
    }
}
$obj=new demo();
$obj->test();
?>
```

Example #3 Calling a parent's method

```
<?php
class demo1 {
    const user="admin";
}
class demo2 extends demo1 {
    public function test() {
        echo parent::user;
    }
}
$obj=new demo2();
$obj->test();
?>
```

1 Word Question Answer

Sr. No.	Question	Answer
1.	Which symbol called scope resolution ?	::
2.	How many keywords to access properties?	Three
3.	Scope resolution operators are used to access.	Properties/Methods

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



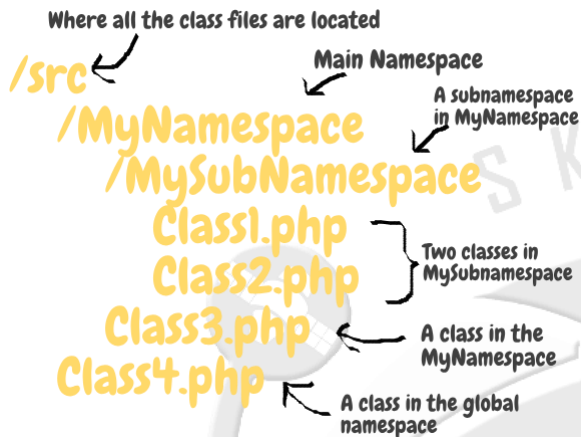
2 – Vaishalinagar
Nr. Amrapali Railway Crossing
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

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

Topic 8 : Explain autoloading classes in oop.

Marks-3

Ans.:



Detailed :

- PHP provides helper functions to include or import PHP files, if it has not been included before.
- If we use PHP autoload then we do not need to write require or include statement for each PHP class files.
- As PHP has evolved using OOPs concept, we create more classes for our PHP project and create a separate PHP file for each single PHP class. Hence we increase the number of require or include statements.
- Therefore we need some standard way to include each files rather specifying include statement by our self every time.

How to create autoload class?

To understand the autoload function, we will see an example.

We will create two different class files and an index file.

First File : test.php

```
<?php
class test
{
    public function __construct()
    {
        echo "Class test working";
    }
}
?>
```

Second File : image.php

```
<?php
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
class image
{
    public function __construct()
    {
        echo "Class image loaded successfully<br>";
    }
}
?>
```

Third File : index.php

```
<?php
function __autoload($classname)
{
    require_once $classname.'.php';
}
$obj1=new image();
$obj2=new test();
?>
```

1 Word Question Answer

Sr. No.	Question	Answer
1.	Which function is used for autoloading?	__autoload()
2.	Autoload method runs..	Automatically
3.	Class name and file name should be same.	True

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Topic 9 : What is class constant?

Marks-2

Ans.:

```
class Program
{
    public const int constNumber = 256;
    static void Main(string[] args)
    {
        constNumber = 235;
    }
}
```

Constant creation

Error, if we try to modify it.

(constant) int Program.constNumber = 256

The left-hand side of an assignment must be a variable, property or indexer

Detailed :

- A constant is, just like the variable that can never be changed.
- When you declare a constant, you assign a value to it, and after that the value will never change.
- Normally simple variables are just easier to use, but in certain cases constant are preferable, because this specific value should not be changed during runtime.
- You can use double colon (::) operator to access a class constant.

```
<?php
class User
{
    const defaultUserName="admin";
    const minimumPasswordLength=6;
}
echo "The default username is : ".User::defaultUserName;
echo "<br>The minimum length of password is : ".User::minimumPasswordLength;
?>
```

1 Word Question Answer

Sr. No.	Question	Answer
1.	Which keyword is used for constant value?	Const
2.	Constant value can never be changed.	True
3.	We can access constant class using.	:: operator

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Q.11 Explain Static keyword

Marks – 2

Ans.

Detailed :

Declaring class properties or methods as static makes them accessible without creating an object of the class.

In PHP we can create static variable and static method.

Static Method:

- Static methods can be called directly without creating an object of a class.
- Static methods are declared with the static keyword.

Example of static method :

```
<?php
class welcome
{
    public static function Hello()
    {
        echo "Welcome to my class";
    }
}
welcome::Hello();
?>
```

Static Variable:

You can declare a variable to be static simply by placing the keyword STATIC in front of the variable name.

Example of static variable:

```
<?php
function counter()
{
    STATIC $count=0;
    $count++;
    echo $count."<br>";
}
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

}

```
counter();  
counter();  
counter();
```

?>

One Marks Question Answer

No	Question	Answer
1	What is static keyword?	Declaring class properties or methods as static makes them accessible without needing an instantiation of the class
2	Static properties cannot be accessed through the object using _____.	the arrow operator ->

Q.12 Explain Class Abstraction.

Marks – 3

Detailed :

Abstract classes are those classes which cannot be directly initialized or in other word we can say that you cannot create object of abstract classes.

Abstract classes always created for inheritance purpose.

The class must be inherited.

You can create abstract classes in PHP using **abstract** keyword.

Abstract class must have at least one abstract method.

Methods defined as abstract simply declare the method's signature - they cannot define the implementation.

You can only inherit abstract class in your child class.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Example #1 [abstract class]

```
<?php
    abstract class Demo
    {
        function Test(){
            return "CCSIT";}
    }
    $objDemo=new Demo();//fatal error: Cannot instantiate abstract class Demo
?>
```

Example #2 [abstract class (in inheritance)]

```
<?php
    abstract class BaseClass
    {
        function BaseTest(){
            return "College Code :15035";}
    }
    class DerivedClass extends BaseClass
    {
        function DerivedTest() {
            return "<br>Name : CCSIT";}
    }
    $objDerived=new DerivedClass();
    echo $objDerived->BaseTest();
    echo $objDerived->DerivedTest();
?>
```

Implementation of abstract method

As we know that abstract functions are those functions of abstract class which is only defined.

It will be declared in your child class.

You can create any method abstract using keyword **abstract**.

You can only create abstract method either in **abstract class** or **interface**.

When inheriting from an abstract class, all methods marked abstract in the parent's class declaration must be defined by the child; additionally, these methods must be defined with the same (or a less restricted) visibility.

For example, if the abstract method is defined as protected, the function implementation must be defined as either protected or public, but not private.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Furthermore the signatures of the methods must match, i.e. the type hints and the number of required arguments must be the same.

For example, if the child class defines an optional argument, where the abstract method's signature does not, there is no conflict in the signature.

Only an **abstract classes** can have an abstract method normal classes cannot have abstract methods.

You can declare abstract and non - abstract methods within an abstract class.

Example #3 [abstract Method]

```
<?php
abstract class BaseClass
{
    Abstract protected function Test($code,$subject);
}
Class DerivedClass extends BaseClass
{
    protected function Test($code,$subject){
        echo "$code $subject";
    }
}
Class OtherClass extends DerivedClass
{
    function callDerived(){
        $this->Test("CS-16","Web Development Using Joomla!CMS");
    }
}
$objOther=new OtherClass();$objOther->callDerived();
?>
```

In class BaseClass we have defined an abstract function Test().

Now when we have inherited class BaseClass then declared function Test().

Note: If you have an abstract method in your abstract class then once you inherit your abstract class then it is necessary to declare your abstract method. If you will not declare your abstract method then PHP will throw error in that case.

One Marks Question Answer

No	Question	Answer
1	You can create abstract classes in PHP using _____ keyword.	Abstract

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

2	You can only create abstract method either in _____ or _____.	<u>abstract class</u> <u>interface</u>
---	---	---

Q.13 Explain Object Interfaces.

Marks - 5

Object interfaces allow you to create code which specifies which methods a class must implement, without having to define how these methods are handled.

Interfaces are defined in the same way as a class, but with the interface keyword replacing the *class* keyword and without any of the methods having their contents defined.

All methods declared in an interface **must be public**; this is the nature of an interface.

Interfaces works same as abstract classes, in the interface they declare a consistent set of methods that classes must implement. In the case of interface parent-child relationship does not exist. A class can access the properties of the interface with the help of **implements keyword**.

Note: At the same time you can extend class also.

Suppose an interface has two methods, and another class that implements the interface must have given the body.

Syntax

```
<?php
Interface NameOfInterface
{
    Name of method with the number of arguments
    .....
    .....
}
?>
```

Example #1 [Interface]

```
<?php
Interface MyFirstInterface
{
    Public function FirstMethod( $name);
    Public function SecondMethod( $name, $roll);
}
?>
```

Some Point about Interface

1. You cannot contain properties, only the methods declaration allowed in case of interface.
2. All method must be **public** in case of interface.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

You can then make a class that implements an interface using the **implements** keyword.

Example #2 [Class with Interface]

```
<?php
Class FirstClass implements MyFirstInterface
{
    Public function FirstMethod ( $name){
        //Method implementation here}
    Public function SecondMethod ( $name, $roll) {
        //Method implementation here }
}
?>
```

Example #3 [Two Interface Implementation]

You can implement more than one interfaces with the help of comma.

```
<?php
Class FirstClass implements MyFirstInterface, MySecondInterface
{
    // all Methods implementation from MyFirstInterface
    // and MySecondInterface.
}

Interface MyInterface
{ publicfunctionmyMethod( $name);}
Interface MySecondInterface
{publicfunctionmySecondMethod( $name);}

Class MyFirstClass implements MyInterface,MySecondInterface
{
    Public function myMethod( $name1){
        echo"$name1";}
    Public function mySecondMethod( $name1){
        echo"$name1"; }
}
$firstObj= new MyFirstClass;
$firstObj->myMethod("John"); $firstObj->mySecondMethod("Brown");
?>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Example #3 [Multiple Interface]

You can also extends mutiple interface using **extends** keyword in PHP.

```
<?php
Interface first
    {Public function set();}
Interface second extends first
    {publicfunctiondisplay();}
Class Computer implements second
{
    Public function set(){
        echo "First interface method<br>";}
    Public function display(){
        echo "Second interface method";}
}
$obj=new Computer(); $obj->set(); $obj->display();
?>
```

Implements

To implement an interface, the **implements** operator is used. All methods in the interface must be implemented within a class; failure to do so will result in a fatal error.

Classes may implement more than one interface if desired by separating each interface with a comma.

One Marks Question Answer

No	Question	Answer
1	What is Object interfaces?	Object interfaces allow you to create code which specifies which methods a class must implement, without having to define how these methods are handled.
2	All method must be _____ in case of interface.	Public

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Q.15 Explain Traits.

Marks –3

In PHP trait is the method of code re-usability.

PHP is single inheritance programming language.

The structure of PHP traits is almost matched with a PHP class but here we group the re-usable methods.

You can use that trait in those class needs to have the feature of the traits function.

if you have a group of the method which is same in the different class then you can club it into PHP Traits.

You are free to use access modifier also in your PHP traits.

Example #1 [trait]

```
<?php
    trait college
    {
        public function message(){
            echo "CCSIT-Junagadh";
        }
    }
    class test
    {
        use college;
    }
    $obj=new test();$obj->message();
?>
```

Example #2 [multiple traits in a single PHP class]

```
<?php
trait CollegeInfo
{
    function Address(){
        echo "<b>College : </b><br>CCSIT<br>Green City<br>Junagadh"; }
    }
trait CourseInfo
{
    function Course(){
        echo "<br><b>Courses Offered : </b><br>B.Sc.(IT)<br>B.C.A.<br>M.Sc.(IT)";}
    }
class Test
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
{  
    use CollegeInfo, CourseInfo;  
}  
$obj=new Test;$obj->Address();$obj->Course();  
?>
```

One Marks Question Answer

No	Question	Answer
1	Explain Traits	In PHP trait is the method of code re-usability.
2	You can use that trait in those _____ of the traits function	class needs to have the feature

Q.16 Explain Anonymous Classes.

Marks - 2

Anonymous classes are the classes which are defined without any name.

We can define an anonymous class using keyword '**new class**'.

This class can replace a full class definition.

One Marks Question Answer

No	Question	Answer
1	We can define an anonymous class using _____ keyword	new class.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Q.17 Explain Overloading.

Marks –3

Overloading in PHP provides the means to dynamically create members and methods via set of "magical" methods

1. Invoked with interacting with members or methods that have not been declared or are not visible in the current scope
2. All of the magic methods must be declared as public
3. None of the magic functions can be called with arguments, passed by reference

All overloading methods are invoked when accessing variable or method that is not declared or is inaccessible

- `__set($name, $value)` – when writing
- `__get ($name)` –when reading
- `__isset ($name)` – when calling `isset()` function
- `__unset ($name)` – when calling `unset()` function
- `__call ($name, $arguments)` - when calling a method
- `__callStatic ($name, $arguments)` – when calling a method in a static context

Must always be declared as static

PHP "overloading" is a lot different from most languages "overloading"

Usually it means the ability to declare two methods with different sets of parameters but same names

overriding

When a class inherits another, it can declare methods that override parent class methods

Method names are the same Parameters may differ

```
<?php
class ParentClass
{
    var $r,$area;
    const PI=3.14;
    public function funArea(){
        $this->r=10;
        $this->area=self::PI*$this->r*$this->r;
        echo "Area of Circle : ".$this->area;}
}
class child extends ParentClass
{
    var $l,$b;
    public function funArea(){
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```

echo parent::funarea();
echo '<br>';
$this->l=10;
$this->b=20;
echo "Area of Rectangle : ".$this->l*$this->b;}
}
$objChild=new child;
$objChild->funArea();
?>
    
```

No	Question	Answer
1	Explain Overloading	Overloading in PHP provides the means to dynamically create members and methods via set of "magical" methods
2	Explain Overloading Method	__set , __get , __isset , __unset , __call , __callStatic

Q.18 Explain Object Iteration.

Marks –3

PHP provides a way for objects to be defined so it is possible to iterate through a list of items, with, for example a foreach statement. By default, all visible properties will be used for the iteration.

Example #1 [Simple Iteration]

```

<?php
class Main
{
    var $a="Public";
    protected $b="Protected";
    private $c="Private";
}
$obj=new Main;
foreach($obj as $key=>$value)
    echo '<br>'.$key."=>".$value;
?>
    
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Example #3[Iteration Array]

```
<?php
class Main
{
    var $fruit=array('Apple','Banana','Grapes','Guava','Mango');
}
$obj=new Main;
foreach($obj->fruit as $val)
echo '<br>'.$val;
echo '<br>';
print_r($obj);
echo '<br>';
print_r($obj->fruit);
?>
```

One Marks Question Answer

No	Question	Answer
1	Explain Object Iteration	PHP provides a way for objects to be defined so it is possible to iterate through a list of items, with, for example a <u>foreach</u> statement. By default, all <u>visible</u> properties will be used for the iteration.

Q.19 Explain Magic Methods.

Marks -5

The "magic" methods are ones with special names, starting with two underscores(__), which denote methods which will be triggered in response to particular PHP events.

Magic methods in PHP are **some predefined function by PHP compiler which executes on some event**. Magic methods starts with prefix __.

__construct	__destruct	__get	__set	__call
__toString	__clone	__isset	__wakeup	__unset
__invoke	__sleep	__autoload		

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

1. __construct

The constructor is a magic method that gets called when the object is instantiated. It is usually the first thing in the class declaration but it does not need to be, it a method like any other and can be declared anywhere in the class. Constructors also inherit like any other method.

2. __destruct()

This magic method runs when your object is destroyed. By default, all objects are destroyed when the script is finished running.

Example [__construct and __destruct Magic Method]

```
<?php
//__constructor & __destruct magic method example
class MagicMethods
{
    function __construct()    {
        echo 'Constructor is called';    }
    function __destruct() {
        echo '<br>Destructor is called';}
}
$objMagic=new MagicMethods;
?>
```

3. __get()

The __get() method is called whenever you attempt to read a non-existing or private property of an object.

It accepts **1 parameter** which is the *name of the property* you tried accessing.

4. __set()

The __set() method is called whenever you attempt to write to a non-existing or private property of an object.

It accepts **2 parameters** which is the *name and the value of the property* you attempted creating or updating.

Example [__get and __Set Magic Method]

```
<?php //__get & __set magic method example
class magicMethod
{
    function __get($name)    {
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
        echo $name;}
function __set($name,$code){
    echo "College Name <u>$name</u> & Code of College <u>$code</u>";
}
$objMagic=new magicMethod();
$objMagic->CCSIT; //call __get magic method
echo '<br>';
$objMagic->CCSIT=15035; //call __set magic method
?>
```

5. __call()

This magic method runs when you try using a method that doesn't exist.

It accepts 2 parameters which is the name of the method and the parameters you tried passing in.

The parameters are converted into 1 array.

Example [__call Magic Method]

```
<?php //__call Magic Method example
class magicMethod
{
    function __call($name,$value){
        echo "Method Name : $name <br>";
        echo print_r($value); }
}
$objMagic=new magicMethod;
$objMagic->getData('CCSIT','Junagadh');
?>
```

6. __toString()

```
public string __toString ( void )
```

This magic method runs when you try echoing out the object itself.

If you define this method, you must return a string and not echo anything.

Example [__toString Magic Method]

```
<?php //__toString Magic Method example
class magicMethod
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
{
    function __toString() {
        return "Joomla! CMS";
    }
    $objMagic=new magicMethod;
    echo $objMagic;
?>
```

7. __clone()

This magic method runs when you try cloning an instance of the class.

To clone an instance, you use the keyword clone.

By cloning an instance, you're creating an object with the same properties and methods of the cloned object.

They become 2 separate instances after that.

This is useful when you want to save an instance at a certain state just in case.

8. __isset

__isset magic methods executes when function isset() is applied on property which is not available or not defined. It takes name of the parameter as an argument.

9. __unset

__unset magic method triggers when unset() method is applied on the property which is either not defined or not accessible. It takes name of the parameter as an argument.

Example [__isset & __unset Magic Method]

```
<?php //__isset & __unset Magic Method
class test
{
    function __isset($name){
        echo "__isset is called for $name";
    }
    function __unset($name){
        echo "<br>__unset is called for $name";
    }
}
$a = new test();
isset($a->x);
unset($a->c);
?>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

10. __invoke()

mixed __invoke ([\$...])

- ♣ This magic method runs when you try running an object as a method.

```
<?php
class test
{
    function __construct()    {
        echo "Object Oriented Programming";
    }
    function __invoke(){
        echo "<br>invoke magic method called";
    }
}
$obj=new test; echo $obj();
?>
```

If you try to do the same thing on an object without an __invoke() method, you'll see this error:

PHP Fatal error: Function name must be a string in filename.php on line X.

11. __sleep

__sleep methods trigger when you are going to serialize your class object.

public array __sleep (void)

```
<?php
class student
{
    private $name;
    function setName($name){
        $this->name=$name; }
    function __sleep(){
        return array("name");}
}
$obj=new student; $obj->setName("ABC"); $data=serialize($obj); echo $data;
?>
```

12. __wakeup

__wakeup executes when you are unserializing any class object.

void __wakeup (void)

```
<?php
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
class student
{
    private $name;
    function setName($name){
        $this->name=$name;}
    function __sleep(){
        return array("name");}
}
$obj=new student; $obj->setName("ABC"); $data=serialize($obj); echo $data;
var_dump(unserialize($data));
?>
```

13. **__autoload** → as per autoloading class.

One Marks Question Answer

No	Question	Answer
1	Explain Magic methods type ?	__construct __destruct __get __set __call
		__toString __clone __isset __wakeup __unset
		__invoke __sleep __autoload

Q.20 Explain Final Keyword.

Marks -2

PHP 5 introduces the final keyword, which prevents child classes from overriding a method by prefixing the definition with **final**.

If the class itself is being defined final then it cannot be extended.

Example #1 [Final methods example]

```
<?php //final method example (cannot overridden)
class Computer
{
    final function test(){
        echo "Samsung";}
}
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
class LapTop extends Computer
{
    function test() {
        echo "Dell";
    }
}
$objLaptop=new LapTop; $objLaptop->test();
?>
```

Fatal error: Cannot override final method Computer::test()

Example #2 [Final class example]

```
<?php //final class example (cannot be extended (inherited))
final class Computer
{
    function test() {
        echo "Samsung";
    }
}
class LapTop extends Computer
{
    function test() {
        echo "Dell";
    }
}
$objLaptop=new LapTop; $objLaptop->test();
?>
```

Fatal error: Class LapTop may not inherit from final class (Computer)

Note: Properties (variables) cannot be declared final, only classes and methods may be declared as final.

One Marks Question Answer

No	Question	Answer
1.	Explain Final Keyword.	PHP 5 introduces the final keyword, which prevents child classes from overriding a method by prefixing the definition with final .

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Q.21 Explain Object Cloning.

Marks -3

If you will directly copy objects in PHP, then it will copy by reference, not by value. Means if you will change main object data then copied object will be affected.

Also if you will change value of the copied object then main object value will be changed.

Syntax

```
$y=$x;
```

If you do this, \$x and \$y both print the same object.

If any changes you make to the properties of object \$y will automatically be made to object \$x.

Example #1 [Copying Object]

```
<?php
class Clone1
{
public $a;
private $b;
function __construct($x, $y){
    $this->a = $x;
    $this->b = $y;}
}
$a= new Clone1("Java", "PHP");
$b= $a; //Copy of the object
print_r($a); echo "<br> ";
$a->a = " Operating System ";
print_r($a); echo"<br> ";
print_r($b);
?>
```

In the above example, \$a is the object of Clone1 class and copy the reference of an object by using the assignment operator.

You can see the changes of the program after assigning the value of \$a to the \$b.

Example #2 [Copying Object]

```
<?php
class Clone1
{
public $name;
function __construct($z){
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
$this->name = $z;}  
function display(){  
    echo $this->name;}  
}  
$a= new Clone1("Audi");  
echo "Before modification:<br>";  
$a->display(); $b=$a;  
$a->name="BMW";  
echo"<br>after modification";  
echo"<br>Using object a<br>"; $a->display();  
echo"<br>Using object b<br>"; $b->display();  
?>
```

Clone Keyword

If you want to make a copy of an object or you want to create objects by value in PHP, you need to clone it with the clone keyword like this.

Cloning of an object is also known as shallow copy.

Example #3 [clone keyword]

```
Syntax    $copy_of_object = clone $object;  
<?php  
class Clone1  
{  
    public$a;  
    function __construct($z){  
        $this->a = $z;}  
}  
$obj= new Clone1("Java", "PHP");  
$obj1= $obj; //Copy of the object  
$obj2= clone $obj;  
print_r($obj);  
echo"<br> ";  
$obj->a = " Operating System ";  
print_r($obj);echo"<br> ";  
print_r($obj1);echo"<br> ";  
print_r($obj2)  
?>
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

In the above example, \$obj is the object of Clone1 class, \$obj is the copy of \$obj and \$obj2 is the clone of object \$obj that means if you change the property of the class using this object does not affect the \$obj2.

Object cloning with magic method __clone

Suppose you want to change value of your property a of the test class in case of cloning of object in PHP.

We can change behaviour of the clone object in PHP using magic method __clone.

Magic method clone executes when object cloning is performed.

As soon as PHP execute statement \$c = clone \$a, __clone method invoked.

Example #4 [__clone magic method]

Syntax : void __clone (void)

```
<?php
//__clone Magic Method example
class test
{
public $a;
private $b;
function __construct($a, $b) {
    $this->a = $a;
    $this->b = $b; }
function __clone() {
    $this->a = "c"; }
}

$a = new test("CCSIT" , "15035");
$b = $a; //Copy of the object
$c = clone $a; //clone of the object
$a->a = "no Data";
print_r($a); echo '<br>'; print_r($b); echo '<br>';
print_r($c); echo '<br>'; print_r($a);
?>
```

So here \$c->a will be "c". Because __clone method will be invoked and only for object \$c it will set value of \$this->a to 'c'.

One Marks Question Answer

No	Question	Answer
1	What is Object Cloning?	it will copy by reference, not by value. Means if you will change main object data then copied object will be affected.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Q.22 Explain Comparing Objects.

Marks -2

When using the comparison operator (==), object variables are compared in a simple manner, namely: Two object instances are equal if they have the same attributes and values (values are compared with ==), and are instances of the same class.

Example #1 [Example of Object Comparison]

```
<?php
class sum
{
    var $a,$b;
    function __construct($a,$b){
        $this->a=$a;
        $this->b=$b;}
}
$obj1=new sum(10,20);$obj2=new sum(10,0);
if($obj1==$obj2)
    echo "<br>Both Objects are Equal";
else
    echo "<br>Both Objects are not Equal";
?>
```

One Marks Question Answer

No	Question	Answer
1	Which Operator Use for Comparing Object?	(==)

Q.23 Explain Type Hinting.

Marks -3

PHP is not a strictly typed language means variable type declaration is not needed at all in PHP. Type declaration increases the readability of the code.

Type Hinting in PHP is a mechanism to allow developers to specify the type of function arguments and return. You can add type for function and method parameters which can be used as a Type Hint.

The definition of the Type Hinting specifies the parameter type when you call the function. If the argument type and the specified type does not match, PHP will generate a fatal error.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

In short, it allows you to control the execution how the type of data passed to the function. You can use type hinting to specify the expected data type of an argument in a function declaration. In PHP we can use type hinting for Object, Array and callable data type. You can also use interface in the type hinting. You can also override the method where you have used type hinting.
Note: Type Hinting in PHP is not supported with int and string data types.

Example

```
<?php
//Class where we are going to implement type hinting
class Typehinting_Test
{
//Implementation Type Hinting in PHP
//Forcing to pass argument as object of class Test1
public function type_hint_method(Test1 $parameter){
$parameter->test_method();
}
//Below is class
class Test1
{
public function __construct(){
//Do Nothing}
//Method to call in the type hinting class
public function test_method() {
echo 'Type Hinting in PHP works';}
}
$th_test = new Typehinting_Test();
//$th_test->type_hint_method(1221) //Will give fatal error: Catchable fatal error:
Argument 1 passed to Typehinting_Test::type_hint_method() must be an instance of
Test1, integer given
$t1 = new Test1();
$th_test->type_hint_method($t1); // Print Type Hinting in PHP works    ?>
```

Type Hinting also works with normal function outside of the class. You can force to pass only object of the class which implements the particular interface. Below is the example:

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

One Marks Question Answer

No	Question	Answer
1	Type Hinting in PHP is a _____ to allow developers	Mechanism
2	Which type _____ to specify the expected data type of an argument in a function declaration?	Hinting

Q.24 Explain Late Static Bindings.

Marks -3

Late static binding in PHP can be used to reference the called class in context static inheritance.

In PHP, we use the self keyword to access static properties and methods of classes.

Self is resolved at compile time, which means that when class is inherited from another class, self will always relate to the class where it is mentioned not the class which is inheriting it.

Example #1 [using self keyword]

```
<?php
class Model{
    protected static $tableName = 'Model';
    public static function getTableName(){
        return self::$tableName;
    }
}
class User extends Model{
    protected static $tableName = 'User';
}
echo User::getTableName(); // Model, not User How it works.
?>
```

Example #2 [using static keyword]

```
<?php

class Model{
    protected static $tableName = 'Model';
    public static function getTableName(){
        return static::$tableName;
    }
}
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
}  
class User extends Model{  
    protected static $tableName = 'User';  
}  
echo User::getTableName(); // User Now we get the expected result.  
?>
```

Notice that the `static::` can only refer to static properties and static methods.

One Marks Question Answer

No	Question	Answer
1	Late static binding in PHP can be used to _____ class in context static inheritance	reference the called

Q.25 Explain Objects and References.

Marks -3

One of the key-points of PHP 5 OOP that is often mentioned is that "objects are passed by references by default". This is not completely true. This section rectifies that general thought using some examples.

A PHP reference is an alias, which allows two different variables to write to the same value. As of PHP 5, an object variable doesn't contain the object itself as value anymore. It only contains an object identifier which allows object accessors to find the actual object. When an object is sent by argument, returned or assigned to another variable, the different variables are not aliases: they hold a copy of the identifier, which points to the same object.

Example: References and Objects

```
<?php  
class A {  
    public $foo = 1;  
}  
$a = new A;  
$b = $a; // $a and $b are copies of the same identifier  
        // ($a) = ($b) = <id>  
$b->foo = 2;
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
echo $a->foo."\n";
$c = new A;
$d = &$c; // $c and $d are references
        // ($c,$d) = <id>
$d->foo = 2;
echo $c->foo."\n";
$e = new A;
function foo($obj) {
    // ($obj) = ($e) = <id>
    $obj->foo = 2;
}
foo($e);
echo $e->foo."\n";
?>
```

The above example will output:

2
2
2

One Marks Question Answer

No	Question	Answer
1	What is the key point of object and References?	PHP 5 OOP that is often mentioned is that "objects are passed by references by default"

Q.26 Explain Object Serialization.

Marks -3

serialize() returns a string containing a byte-stream representation of any value that can be stored in PHP.

Using serialize to save an object will save all variables in an object. The methods in an object will not be saved, only the name of the class.

That is, if you have an object of class A and serialize this, you'll get a string that refers to class A and contains all values of variables contained in it.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<?php
class sum
{
    var $a,$b;
    function __construct($a,$b){
        $this->a=$a;
        $this->b=$b;}
}
$obj1=new sum(10,20);
print_r($obj1); echo "<br>"; echo serialize($obj1);
?>
```

Object unserialization

- ♣ unserialize — Creates a PHP value from a stored representationmixed
unserialize (string \$str [, array \$options])

unserialize() takes a single serialized variable and converts it back into a PHP value.

One Marks Question Answer

No	Question	Answer
1	Which Function to save an object will save all variables in an object?	Serialize()

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Bootstrap Basics

• What is Bootstrap

Mark-1

Bootstrap is a powerful front-end framework for faster and easier web development. It includes HTML and CSS based design templates for common user interface components like Typography, Forms, Buttons, Tables, Navigations, Dropdowns, Alerts, Modals, Tabs, Accordion, Carousel and many other as well as optional JavaScript extensions.

Bootstrap also gives you ability to create responsive layout with much less efforts.

Advantages of Bootstrap

Mark-3

The biggest advantage of using Bootstrap is that it comes with free set of tools for creating flexible and responsive web layouts as well as common interface components.

Additionally, using the Bootstrap data APIs you can create advanced interface components like Scrollspy and Typeaheads without writing a single line of JavaScript.

Here are some more advantages, why one should opt for Bootstrap:

- **Save lots of time** — You can save lots of time and efforts using the Bootstrap predefined design templates and classes and concentrate on other development work.
- **Responsive features** — Using Bootstrap you can easily create responsive designs. Bootstrap responsive features make your web pages to appear more appropriately on different devices and screen resolutions without any change in markup.
- **Consistent design** — All Bootstrap components share the same design templates and styles through a central library, so that the designs and layouts of your web pages are consistent throughout your development.
- **Easy to use** — Bootstrap is very easy to use. Anybody with the basic working knowledge of HTML and CSS can start development with Bootstrap.
- **Compatible with browsers** — Bootstrap is created with modern browsers in mind and it is compatible with all modern browsers such as Mozilla Firefox, Google Chrome, Safari, Internet Explorer, and Opera.
- **Open Source** — And the best part is, it is completely free to download and use.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **File structure**

Mark-2

Precompiled Bootstrap

Once the compiled version Bootstrap is downloaded, extract the ZIP file, and you will see the following file/directory structure –

```
bootstrap/  
├── css/  
│   ├── bootstrap.css  
│   ├── bootstrap.min.css  
│   ├── bootstrap-theme.css  
│   └── bootstrap-theme.min.css  
├── js/  
│   ├── bootstrap.js  
│   └── bootstrap.min.js  
└── fonts/  
    ├── glyphicons-halflings-regular.eot  
    ├── glyphicons-halflings-regular.svg  
    ├── glyphicons-halflings-regular.ttf  
    └── glyphicons-halflings-regular.woff
```

As you can see, there are compiled CSS and JS (bootstrap.*), as well as compiled and minified CSS and JS (bootstrap.min.*). Fonts from Glyphicons are included, as it is the optional Bootstrap theme.

- **Bootstrap Source Code**

If you have downloaded the Bootstrap source code then the file structure would be as follows –

```
├── less/  
├── js/  
├── fonts/  
├── dist/  
│   ├── css/  
│   ├── js/  
│   └── fonts/  
├── docs-assets/  
├── examples/  
└── *.html
```

- The files under less/, js/, and fonts/ are the source code for Bootstrap CSS, JS, and icon fonts (respectively).
- The dist/ folder includes everything listed in the precompiled download section above.
- docs-assets/, examples/, and all *.html files are Bootstrap documentation.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **HTML Template**

A basic HTML template using Bootstrap would look like this –

```
<!DOCTYPE html>
<html>

<head>
  <title>Bootstrap 101 Template</title>
  <meta name = "viewport" content = "width = device-width, initial-scale = 1.0">

  <!-- Bootstrap -->
  <link href = "css/bootstrap.min.css" rel = "stylesheet">

  <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
  <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->

  <!--[if lt IE 9]>
  <script src = "https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
  <script src = "https://oss.maxcdn.com/libs/respond.js/1.3.0/respond.min.js"></script>
  <![endif]-->
</head>

<body>
  <h1>Hello, world!</h1>

  <!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
  <script src = "https://code.jquery.com/jquery.js"></script>

  <!-- Include all compiled plugins (below), or include individual files as needed -->
  <script src = "js/bootstrap.min.js"></script>

</body>
</html>
```

Here you can see the **jquery.js**, **bootstrap.min.js** and **bootstrap.min.css** files that are included to make a normal HTML file to the Bootstrapped Template. Just make sure to include jQuery library before you include Bootstrap library.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

• What is Bootstrap Grid System?

Mark-5

As put by the official documentation of Bootstrap for grid system –

Bootstrap includes a responsive, mobile first fluid grid system that appropriately scales up to 12 columns as the device or viewport size increases. It includes predefined classes for easy layout options, as well as powerful mixins for generating more semantic layouts.

Let us understand the above statement. Bootstrap 3 is mobile first in the sense that the code for Bootstrap now starts by targeting smaller screens like mobile devices, tablets, and then “expands” components and grids for larger screens such as laptops, desktops.

Mobile First Strategy

- **Content**
 - Determine what is most important.
- **Layout**
 - Design to smaller widths first.
 - Base CSS address mobile device first; media queries address for tablet, desktops.
- **Progressive Enhancement**
 - Add elements as screen size increases.

Working of Bootstrap Grid System

Grid systems are used for creating page layouts through a series of rows and columns that house your content. Here's how the Bootstrap grid system works –

- Rows must be placed within a **.container** class for proper alignment and padding.
- Use rows to create horizontal groups of columns.
- Content should be placed within the columns, and only columns may be the immediate children of rows.
- Predefined grid classes like **.row** and **.col-xs-4** are available for quickly making grid layouts. LESS mixins can also be used for more semantic layouts.
- Columns create gutters (gaps between column content) via padding. That padding is offset in rows for the first and the last column via negative margin on **.rows**.
- Grid columns are created by specifying the number of twelve available columns you wish to span. For example, three equal columns would use three **.col-xs-4**.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Creating Two Column Layouts

The following example will show you how to create two column layouts for small, medium and large devices like tables, laptops and desktops etc. However, on mobile phones, the columns will automatically become horizontal as default.

```
<div class="container">
  <!--Row with two equal columns-->
  <div class="row">
    <div class="col-sm-6"><!--Column left--></div>
    <div class="col-sm-6"><!--Column right--></div>
  </div>

  <!--Row with two columns divided in 1:2 ratio-->
  <div class="row">
    <div class="col-sm-4"><!--Column left--></div>
    <div class="col-sm-8"><!--Column right--></div>
  </div>

  <!--Row with two columns divided in 1:3 ratio-->
  <div class="row">
    <div class="col-sm-3"><!--Column left--></div>
    <div class="col-sm-9"><!--Column right--></div>
  </div>
</div>
```

Offsetting the Grid Columns

You can also move grid columns to the right for alignment purpose using the column offset classes like `.col-md-offset-*`, `.col-sm-offset-*`, etc.

These classes offset the columns by simply increasing its left margin by specified number of columns. For example, the class `.col-sm-offset-4` on the column `.col-sm-8` moves it to the right over four columns from its original position.

```
<div class="container">
  <div class="row">
    <div class="col-sm-4"></div>
    <div class="col-sm-8"></div>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
</div>
<div class="row">
  <div class="col-sm-8 col-sm-offset-4"><!--Column with 4 columns offset--></div>
</div>
</div>
```

- **Nesting of Grid Columns**

The Bootstrap grid columns are nestable, that means you can put rows and columns inside an existing column. However, the formula for placing the columns will be the same, i.e. the sum of column numbers should be equal to 12 or less within a single row.

```
<div class="container">
  <div class="row">
    <div class="col-xs-8"><!--Column left--></div>
    <div class="col-xs-4">
      <!--Column right with nested rows and columns-->
      <div class="row">
        <div class="col-xs-12"></div>
      </div>
    </div>
  </div>
  <div class="row">
    <div class="col-xs-12"></div>
  </div>
</div>
</div>
```

- **What is Responsive Web Design?**

Mark-2

Responsive web design makes your web page look good on all devices.

Responsive web design uses only HTML and CSS.

Responsive web design is not a program or a JavaScript.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



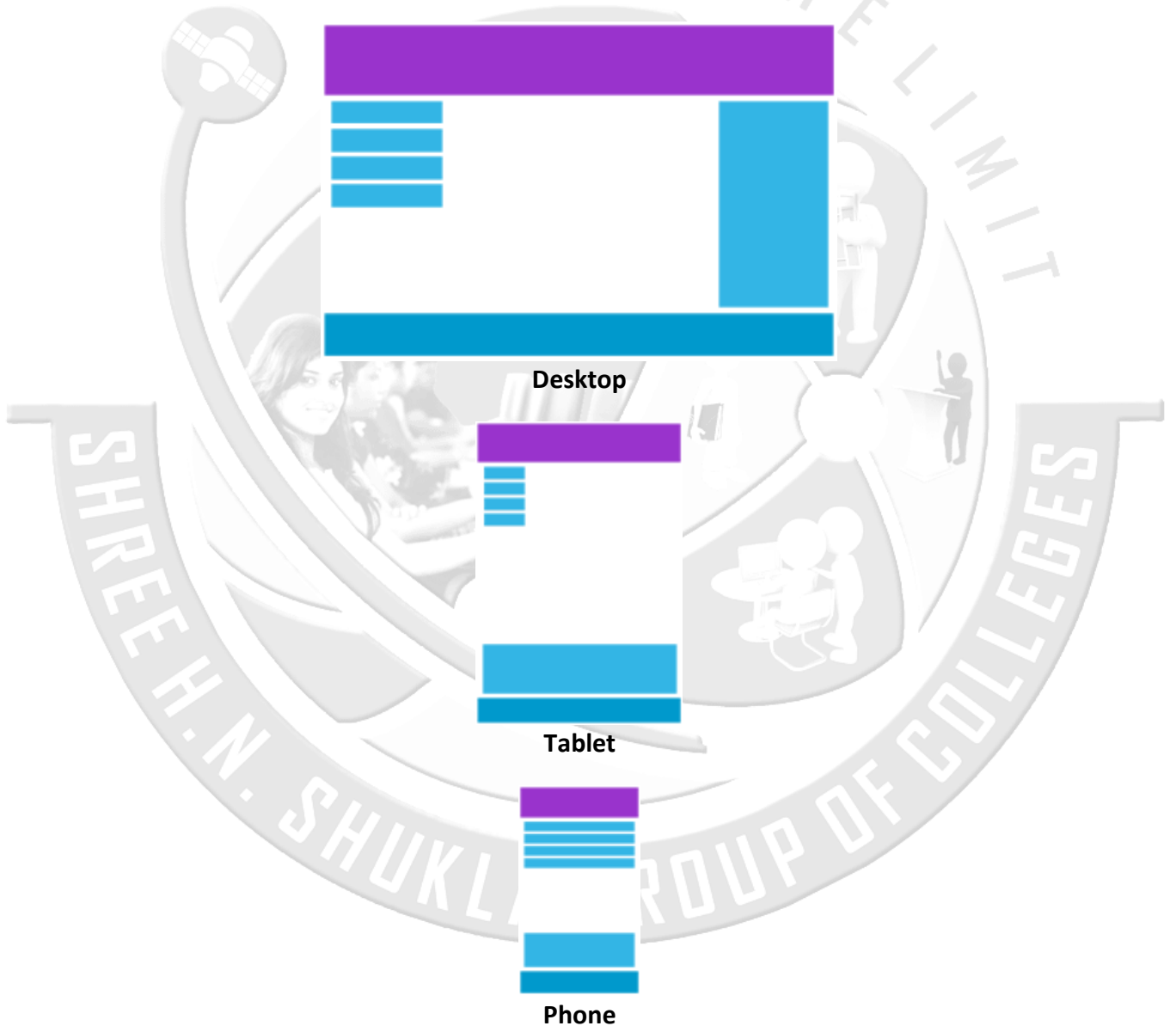
2 – Vaishalinagar
Nr. Amrapali Railway Crossing
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

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

Designing For The Best Experience For All Users

Web pages can be viewed using many different devices: desktops, tablets, and phones. Your web page should look good, and be easy to use, regardless of the device.

Web pages should not leave out information to fit smaller devices, but rather adapt its content to fit any device:



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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

It is called responsive web design when you use CSS and HTML to resize, hide, shrink, enlarge, or move the content to make it look good on any screen.

- **Typography**

Mark-5

Working with Headings

You can define all HTML headings, `<h1>` through `<h6>` — In the same way you define in simple HTML document. You can also utilize the heading classes `.h1` through `.h6` on other elements, if you want to apply the style on element's text same as headings.

- `<h1>h1. Bootstrap heading</h1>`
- `<h2>h2. Bootstrap heading</h2>`
- `<h3>h3. Bootstrap heading</h3>`
- `<h4>h4. Bootstrap heading</h4>`
- `<h5>h5. Bootstrap heading</h5>`
- `<h6>h6. Bootstrap heading</h6>`

The output of the above example will look something like this:

h1. Bootstrap heading

h2. Bootstrap heading

h3. Bootstrap heading

h4. Bootstrap heading

h5. Bootstrap heading

h6. Bootstrap heading

Creating Page Headers

You can make your `<h1>` heading appear differently than rest of the headings on a page using the page header component. You can also utilize `<small>` tag to mark header subtext.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<div class="page-header">
```

```
  <h1>Bootstrap <small>An intuitive front-end framework</small></h1>
```

```
</div>
```

The output of the above example will look something like this:

Bootstrap An intuitive front-end framework

- **Working with Paragraphs**

Bootstrap's global default font-size is **14px**, with a line-height of **1.428**. This is applied to the `<body>` and all paragraphs. In addition to that a bottom margin of half their line-height (10px by default) is applied to the all paragraphs i.e. the `<p>` elements.

You can also make a paragraph stand out by just adding the class `.lead`.

```
<p>This is how a normal paragraph looks like in Bootstrap.</p>
```

```
<p class="lead">This is how a paragraph stands out in Bootstrap.</p>
```

The HTML code in the above examples will produce the following result:

This is how a normal paragraph looks like in Twitter Bootstrap.

This is how a paragraph stands out in Twitter Bootstrap.

Bootstrap Text Formatting

You are free to use text formatting tags like ``, `<i>`, `<small>` to make your text bold, italic, small and so on, in the same way you do in simple HTML.

```
<p><b>This is bold text</b></p>
```

```
<p><big>This is big text</big></p>
```

```
<p><code>This is computer code</code></p>
```

```
<p><em>This is emphasized text</em></p>
```

```
<p><i>This is italic text</i></p>
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

<p><mark>This is highlighted text</mark></p>
<p><small>This is small text</small></p>

Text Transformation Classes

You can also transform the text to lowercase, uppercase or make them capitalize.

```
<p class="text-lowercase">The quick brown fox jumps over the lazy dog.</p>  
<p class="text-uppercase">The quick brown fox jumps over the lazy dog.</p>  
<p class="text-capitalize">The quick brown fox jumps over the lazy dog.</p>
```

The output of the above example will look something like this:

the quick brown fox jumps over the lazy dog.

THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG.

The Quick Brown Fox Jumps Over The Lazy Dog.

Text Emphasis Classes

Colors are the powerful method of conveying important information in website design.

Bootstrap has handful of emphasis utility classes that can be used for this purpose such as showing success message in green color, warning or error message in red color, etc.

```
<p class="text-muted">Muted: This text is grayed out.</p>  
<p class="text-primary">Important: Please read the instructions carefully before proceeding.</p>  
<p class="text-success">Success: Your message has been sent successfully.</p>  
<p class="text-info">Note: You must agree with the terms and conditions to complete the sign up process.</p>  
<p class="text-warning">Warning: There was a problem with your network connection.</p>  
<p class="text-danger">Error: An error has been occurred while submitting your data.</p>
```

The output of the above example will look something like this:

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Muted: This text is grayed out.

Important: Please read the instructions carefully before proceeding.

Success: Your message has been sent successfully.

Note: You must agree with the terms and conditions to complete the sign up process.

Warning: There was a problem with your network connection.

Error: An error has been occurred while submitting your data.

Styling Blockquotes

You can also give pretty look to your blockquotes — Just define the blockquotes using the standard `<blockquote>` element and bootstrap's style sheet will do the rest.

```
<blockquote>
  <p>The world is a dangerous place to live; not because of the people who are evil, but because of
  the people who don't do anything about it.</p>
  <small>by <cite>Albert Einstein</cite></small>
</blockquote>
```

The output of the above example will look something like this:

The world is a dangerous place to live; not because of the people who are evil, but because of the people who don't do anything about it.

— by Albert Einstein

- **Bootstrap - Code**

Bootstrap allows you to display code with two different key ways –

- The first is the `<code>` tag. If you are going to be displaying code inline, you should use the `<code>` tag.
- Second is the `<pre>` tag. If the code needs to be displayed as a standalone block element or if it has multiple lines, then you should use the `<pre>` tag.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Make sure that when you use the `<pre>` and `<code>` tags, you use the unicode variants for the opening and closing tags – `<` and `>`;

Let us see an example below –

`<p><code><header></code>` is wrapped as an inline element.`</p>`

`<p>`To display code as a standalone block element use `<pre>` tag as:`</p>`

```
<pre>
  &lt;article&gt;
    &lt;h1&gt;Article Heading&lt;/h1&gt;
  &lt;/article&gt;
</pre>
```

`<header>` is wrapped as an inline element.

To display code as a standalone block element use `<pre>` tag as:

```
<article>
  <h1>Article Heading</h1>
</article>
```

- **Bootstrap – Tables**

Mark-5

Bootstrap provides a clean layout for building tables. Some of the table elements supported by Bootstrap are –

Tag	Description
<code><table></code>	Wrapping element for displaying data in a tabular format
<code><thead></code>	Container element for table header rows (<code><tr></code>) to label table columns.
<code><tbody></code>	Container element for table rows (<code><tr></code>) in the body of the table.
<code><tr></code>	Container element for a set of table cells (<code><td></code> or <code><th></code>) that appears on a single row.
<code><td></code>	Default table cell.
<code><th></code>	Special table cell for column (or row, depending on scope and placement) labels. Must be used within a <code><thead></code>
<code><caption></code>	Description or summary of what the table holds.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Basic Table**

If you want a nice, basic table style with just some light padding and horizontal dividers, add the base class of .table to any table as shown in the following example –

```
<table class = "table">
  <caption>Basic Table Layout</caption>

  <thead>
    <tr>
      <th>Name</th>
      <th>City</th>
    </tr>
  </thead>

  <tbody>
    <tr>
      <td>Tanmay</td>
      <td>Bangalore</td>
    </tr>

    <tr>
      <td>Sachin</td>
      <td>Mumbai</td>
    </tr>
  </tbody>
</table>
```

- **Optional Table Classes**

Along with the base table markup and the .table class, there are a few additional classes that you can use to style the markup. Following sections will give you a glimpse of all these classes.

- **Striped Table**

By adding the .table-striped class, you will get stripes on rows within the <tbody> as seen in the following example –

```
<table class = "table table-striped">
  <caption>Striped Table Layout</caption>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<thead>
<tr>
<th>Name</th>
<th>City</th>
<th>Pincode</th>
</tr>
</thead>
```

```
<tbody>
<tr>
<td>Tanmay</td>
<td>Bangalore</td>
<td>560001</td>
</tr>

<tr>
<td>Sachin</td>
<td>Mumbai</td>
<td>400003</td>
</tr>

<tr>
<td>Uma</td>
<td>Pune</td>
<td>411027</td>
</tr>
</tbody>
```

```
</table>
```

- **Bordered Table**

By adding the .table-bordered class, you will get borders surrounding every element and rounded corners around the entire table as seen in the following example –

```
<table class = "table table-bordered">
<caption>Bordered Table Layout</caption>

<thead>
<tr>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<th>Name</th>
<th>City</th>
<th>Pincode</th>
</tr>
</thead>
```

```
<tbody>
<tr>
<td>Tanmay</td>
<td>Bangalore</td>
<td>560001</td>
</tr>

<tr>
<td>Sachin</td>
<td>Mumbai</td>
<td>400003</td>
</tr>

<tr>
<td>Uma</td>
<td>Pune</td>
<td>411027</td>
</tr>
</tbody>
```

```
</table>
```

- **Hover Table**

By adding the `.table-hover` class, a light gray background will be added to rows while the cursor hovers over them, as seen in the following example –

```
<table class = "table table-hover">
<caption>Hover Table Layout</caption>

<thead>
<tr>
<th>Name</th>
<th>City</th>
<th>Pincode</th>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
</tr>
</thead>
```

```
<tbody>
<tr>
<td>Tanmay</td>
<td>Bangalore</td>
<td>560001</td>
</tr>
<tr>
<td>Sachin</td>
<td>Mumbai</td>
<td>400003</td>
</tr>
<tr>
<td>Uma</td>
<td>Pune</td>
<td>411027</td>
</tr>
</tbody>
```

```
</table>
```

- **Condensed Table**

By adding the `.table-condensed` class, row padding is cut in half to condense the table. as seen in the following example. This is useful if you want any denser information.

```
<table class = "table table-condensed">
<caption>Condensed Table Layout</caption>
<thead>
<tr>
<th>Name</th>
<th>City</th>
<th>Pincode</th>
</tr>
</thead>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<tbody>
<tr>
<td>Tanmay</td>
<td>Bangalore</td>
<td>560001</td>
</tr>

<tr>
<td>Sachin</td>
<td>Mumbai</td>
<td>400003</td>
</tr>

<tr>
<td>Uma</td>
<td>Pune</td>
<td>411027</td>
</tr>
</tbody>
```

</table>

- **Contextual classes**

The Contextual classes shown in following table will allow you to change the background color of your table rows or individual cells.

Class	Description
.active	Applies the hover color to a particular row or cell
.success	Indicates a successful or positive action
.warning	Indicates a warning that might need attention
.danger	Indicates a dangerous or potentially negative action

These classes can be applied to <tr>, <td> or <th>.

```
<table class = "table">
<caption>Contextual Table Layout</caption>

<thead>
<tr>
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<th>Product</th>
<th>Payment Date</th>
<th>Status</th>
</tr>
</thead>
```

```
<tbody>
<tr class = "active">
<td>Product1</td>
<td>23/11/2013</td>
<td>Pending</td>
</tr>
```

```
<tr class = "success">
<td>Product2</td>
<td>10/11/2013</td>
<td>Delivered</td>
</tr>
```

```
<tr class = "warning">
<td>Product3</td>
<td>20/10/2013</td>
<td>In Call to confirm</td>
</tr>
```

```
<tr class = "danger">
<td>Product4</td>
<td>20/10/2013</td>
<td>Declined</td>
</tr>
</tbody>
```

```
</table>
```

- **Responsive Tables**

By wrapping any .table in .table-responsive class, you will make the table scroll horizontally up to small devices (under 768px). When viewing on anything larger than 768px wide, you will not see any difference in these tables.

```
<div class = "table-responsive">
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<table class = "table">
```

```
<caption>Responsive Table Layout</caption>
```

```
<thead>
```

```
<tr>
```

```
<th>Product</th>
```

```
<th>Payment Date</th>
```

```
<th>Status</th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<tr>
```

```
<td>Product1</td>
```

```
<td>23/11/2013</td>
```

```
<td>Pending</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Product2</td>
```

```
<td>10/11/2013</td>
```

```
<td>Delivered</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Product3</td>
```

```
<td>20/10/2013</td>
```

```
<td>In Call to confirm</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Product4</td>
```

```
<td>20/10/2013</td>
```

```
<td>Declined</td>
```

```
</tr>
```

```
</tbody>
```

```
</table>
```

```
</div>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

• Bootstrap – Forms

Mark-3

Bootstrap makes it easy with the simple HTML markup and extended classes for different styles of forms. In this chapter we will study how to create forms with ease using Bootstrap.

Form Layout

Bootstrap provides you with following types of form layouts –

- Vertical (default) form
- In-line form
- Horizontal form
- **Vertical or Basic Form**

The basic form structure comes with Bootstrap; individual form controls automatically receive some global styling. To create a basic form do the following –

- Add a role form to the parent <form> element.
- Wrap labels and controls in a <div> with class .form-group. This is needed for optimum spacing.
- Add a class of .form-control to all textual <input>, <textarea>, and <select> elements.

```
<form role = "form">

<div class = "form-group">
  <label for = "name">Name</label>
  <input type = "text" class = "form-control" id = "name" placeholder = "Enter Name">
</div>

<div class = "form-group">
  <label for = "inputfile">File input</label>
  <input type = "file" id = "inputfile">
  <p class = "help-block">Example block-level help text here.</p>
</div>

<div class = "checkbox">
  <label><input type = "checkbox"> Check me out</label>
</div>

<button type = "submit" class = "btn btn-default">Submit</button>
</form>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

• Inline Form

To create a form where all of the elements are inline, left aligned and labels are alongside, add the class .form-inline to the <form> tag.

```
<form class = "form-inline" role = "form">

<div class = "form-group">
  <label class = "sr-only" for = "name">Name</label>
  <input type = "text" class = "form-control" id = "name" placeholder = "Enter Name">
</div>

<div class = "form-group">
  <label class = "sr-only" for = "inputfile">File input</label>
  <input type = "file" id = "inputfile">
</div>

<div class = "checkbox">
  <label><input type = "checkbox"> Check me out</label>
</div>

<button type = "submit" class = "btn btn-default">Submit</button>
</form>
```

• Horizontal Form

Horizontal forms stands apart from the others not only in the amount of markup, but also in the presentation of the form. To create a form that uses the horizontal layout, do the following –

- Add a class of .form-horizontal to the parent <form> element.
- Wrap labels and controls in a <div> with class .form-group.
- Add a class of .control-label to the labels.

```
<form class = "form-horizontal" role = "form">

<div class = "form-group">
  <label for = "firstname" class = "col-sm-2 control-label">First Name</label>

  <div class = "col-sm-10">
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<input type = "text" class = "form-control" id = "firstname" placeholder = "Enter First Name">
</div>
</div>

<div class = "form-group">
  <label for = "lastname" class = "col-sm-2 control-label">Last Name</label>

  <div class = "col-sm-10">
    <input type = "text" class = "form-control" id = "lastname" placeholder = "Enter Last Name">
  </div>
</div>

<div class = "form-group">
  <div class = "col-sm-offset-2 col-sm-10">
    <div class = "checkbox">
      <label><input type = "checkbox"> Remember me</label>
    </div>
  </div>
</div>

<div class = "form-group">
  <div class = "col-sm-offset-2 col-sm-10">
    <button type = "submit" class = "btn btn-default">Sign in</button>
  </div>
</div>

</form>
```

- **Supported Form Controls**

Bootstrap natively supports the most common form controls mainly input, textarea, checkbox, radio, and select.

Inputs

The most common form text field is the input field. This is where users will enter most of the essential form data. Bootstrap offers support for all native HTML5 input types: text, password, datetime, datetime-local, date, month, time, week, number, email, url, search, tel, and color. Proper type declaration is required to make Inputs fully styled.

```
<form role = "form">
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<div class = "form-group">  
  <label for = "name">Label</label>  
  <input type = "text" class = "form-control" placeholder = "Text input">  
</div>
```

```
</form>
```

- **Textarea**

The textarea is used when you need multiple lines of input. Change rows attribute as necessary (fewer rows = smaller box, more rows = bigger box).

```
<form role = "form">  
  
<div class = "form-group">  
  <label for = "name">Text Area</label>  
  <textarea class = "form-control" rows = "3"></textarea>  
</div>  
</form>
```

- **CheckBoxes and Radio Buttons**

Checkboxes and radio buttons are great when you want users to choose from a list of preset options.

- When building a form, use checkbox if you want the user to select any number of options from a list. Use radio if you want to limit the user to just one selection.
- Use `.checkbox-inline` or `.radio-inline` class to a series of checkboxes or radios for controls appear on the same line.

The following example demonstrates both (default and inline) types –

```
<label for = "name">Example of Default Checkbox and radio button </label>  
  
<div class = "checkbox">  
  <label>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<input type = "checkbox" value = "">Option 1
</label>
</div>

<div class = "checkbox">
<label>
<input type = "checkbox" value = "">Option 2
</label>
</div>

<div class = "radio">
<label>
<input type = "radio" name = "optionsRadios" id = "optionsRadios1" value = "option1" checked> Option
1
</label>
</div>

<div class = "radio">
<label>
<input type = "radio" name = "optionsRadios" id = "optionsRadios2" value = "option2">
Option 2 - selecting it will deselect option 1
</label>
</div>

<label for = "name">Example of Inline Checkbox and radio button </label>

<div>
<label class = "checkbox-inline">
<input type = "checkbox" id = "inlineCheckbox1" value = "option1"> Option 1
</label>

<label class = "checkbox-inline">
<input type = "checkbox" id = "inlineCheckbox2" value = "option2"> Option 2
</label>

<label class = "checkbox-inline">
<input type = "checkbox" id = "inlineCheckbox3" value = "option3"> Option 3
</label>

<label class = "checkbox-inline">
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<input type = "radio" name = "optionsRadiosinline" id = "optionsRadios3" value = "option1" checked>  
Option 1  
</label>
```

```
<label class = "checkbox-inline">  
<input type = "radio" name = "optionsRadiosinline" id = "optionsRadios4" value = "option2"> Option 2  
</label>  
</div>
```

- **Selects**

A select is used when you want to allow the user to pick from multiple options, but by default it only allows one.

- Use `<select>` for list options with which the user is familiar, such as states or numbers.
- Use `multiple = "multiple"` to allow the users to select more than one option.

The following example demonstrates both (select and multiple) types –

```
<form role = "form">  
  
<div class = "form-group">  
<label for = "name">Select list</label>  
  
<select class = "form-control">  
<option>1</option>  
<option>2</option>  
<option>3</option>  
<option>4</option>  
<option>5</option>  
</select>  
  
<label for = "name">Multiple Select list</label>  
  
<select multiple class = "form-control">  
<option>1</option>  
<option>2</option>  
<option>3</option>  
<option>4</option>
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<option>5</option>
</select>
```

```
</div>
```

```
</form>
```

- **Static Control**

Use the class `.form-control-static` on a `<p>`, when you need to place plain text next to a form label within a horizontal form.

```
<form class = "form-horizontal" role = "form">
  <div class = "form-group">
    <label class = "col-sm-2 control-label">Email</label>

    <div class = "col-sm-10">
      <p class = "form-control-static">email@example.com</p>
    </div>
  </div>

  <div class = "form-group">
    <label for = "inputPassword" class = "col-sm-2 control-label">Password</label>

    <div class = "col-sm-10">
      <input type = "password" class = "form-control" id = "inputPassword" placeholder = "Password">
    </div>
  </div>
</form>
```

- **Form Control States**

In addition to the `:focus` (i.e., a user clicks into the input or tabs onto it) state, Bootstrap offers styling for disabled inputs and classes for form validation.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Input Focus**

When an input receives `:focus`, the outline of the input is removed and a box-shadow is applied.

- **Disabled Inputs**

If you need to disable an input, simply adding the `disabled` attribute will not only disable it; it will also change the styling and the mouse cursor when the cursor hovers over the element.

- **Disabled Fieldsets**

Add the `disabled` attribute to a `<fieldset>` to disable all the controls within the `<fieldset>` at once.

- **Validation States**

Bootstrap includes validation styles for errors, warnings, and success messages. To use, simply add the appropriate class (`.has-warning`, `.has-error`, or `.has-success`) to the parent element.

The following example demonstrates all the form control states –

```
<form class = "form-horizontal" role = "form">
  <div class = "form-group">
    <label class = "col-sm-2 control-label">Focused</label>
    <div class = "col-sm-10">
      <input class = "form-control" id = "focusedInput" type = "text" value = "This is focused...">
    </div>
  </div>
  <div class = "form-group">
    <label for = "inputPassword" class = "col-sm-2 control-label">Disabled</label>
    <div class = "col-sm-10">
      <input class = "form-control" id = "disabledInput" type = "text" placeholder = "Disabled input here..."
disabled>
    </div>
  </div>
  <fieldset disabled>
    <div class = "form-group">
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<label for = "disabledTextInput" class = "col-sm-2 control-label">  
  Disabled input (Fieldset disabled)  
</label>
```

```
<div class = "col-sm-10">  
  <input type = "text" id = "disabledTextInput" class = "form-control" placeholder = "Disabled input">  
</div>  
</div>
```

```
<div class = "form-group">  
  <label for = "disabledSelect" class = "col-sm-2 control-label">  
    Disabled select menu (Fieldset disabled)  
  </label>
```

```
<div class = "col-sm-10">  
  <select id = "disabledSelect" class = "form-control">  
    <option>Disabled select</option>  
  </select>  
</div>
```

```
</div>  
</fieldset>
```

```
<div class = "form-group has-success">  
  <label class = "col-sm-2 control-label" for = "inputSuccess">  
    Input with success  
  </label>
```

```
<div class = "col-sm-10">  
  <input type = "text" class = "form-control" id = "inputSuccess">  
</div>  
</div>
```

```
<div class = "form-group has-warning">  
  <label class = "col-sm-2 control-label" for = "inputWarning">  
    Input with warning  
  </label>
```

```
<div class = "col-sm-10">
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<input type = "text" class = "form-control" id = "inputWarning">
</div>
</div>

<div class = "form-group has-error">
<label class = "col-sm-2 control-label" for = "inputError">
  Input with error
</label>

<div class = "col-sm-10">
  <input type = "text" class = "form-control" id = "inputError">
</div>

</div>
</form>
```

• Bootstrap – Buttons

Mark-3

Anything that is given a class of **.btn** will inherit the default look of a gray button with rounded corners. However, Bootstrap provides some options to style buttons, which are summarized in the following table –

Class	Description
btn	Default/ Standard button.
btn-primary	Provides extra visual weight and identifies the primary action in a set of buttons.
btn-success	Indicates a successful or positive action.
btn-info	Contextual button for informational alert messages.
btn-warning	Indicates caution should be taken with this action.
btn-danger	Indicates a dangerous or potentially negative action.
btn-link	Deemphasize a button by making it look like a link while maintaining button behavior.

The following example demonstrates all the above button classes –

```
<!-- Standard button -->
<button type = "button" class = "btn btn-default">Default Button</button>
```

```
<!-- Provides extra visual weight and identifies the primary action in a set of buttons -->
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<button type = "button" class = "btn btn-primary">Primary Button</button>
```

```
<!-- Indicates a successful or positive action -->
```

```
<button type = "button" class = "btn btn-success">Success Button</button>
```

```
<!-- Contextual button for informational alert messages -->
```

```
<button type = "button" class = "btn btn-info">Info Button</button>
```

```
<!-- Indicates caution should be taken with this action -->
```

```
<button type = "button" class = "btn btn-warning">Warning Button</button>
```

```
<!-- Indicates a dangerous or potentially negative action -->
```

```
<button type = "button" class = "btn btn-danger">Danger Button</button>
```

```
<!-- Deemphasize a button by making it look like a link while maintaining button behavior -->
```

```
<button type = "button" class = "btn btn-link">Link Button</button>
```

- **Button Size**

The following table summarizes the classes used to get buttons of various sizes –

Class	Description
.btn-lg	This makes the button size large.
.btn-sm	This makes the button size small.
.btn-xs	This makes the button size extra small.
.btn-block	This creates block level buttons—those that span the full width of a parent.

The following example demonstrates this –

```
<p>  
<button type = "button" class = "btn btn-primary btn-lg">  
  Large Primary button  
</button>  
  
<button type = "button" class = "btn btn-default btn-lg">  
  Large button  
</button>  
</p>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

<p>

```
<button type = "button" class = "btn btn-primary">
```

Default size Primary button

```
</button>
```

```
<button type = "button" class = "btn btn-default">
```

Default size button

```
</button>
```

</p>

<p>

```
<button type = "button" class = "btn btn-primary btn-sm">
```

Small Primary button

```
</button>
```

```
<button type = "button" class = "btn btn-default btn-sm">
```

Small button

```
</button>
```

</p>

<p>

```
<button type = "button" class = "btn btn-primary btn-xs">
```

Extra small Primary button

```
</button>
```

```
<button type = "button" class = "btn btn-default btn-xs">
```

Extra small button

```
</button>
```

</p>

<p>

```
<button type = "button" class = "btn btn-primary btn-lg btn-block">
```

Block level Primary button

```
</button>
```

```
<button type = "button" class = "btn btn-default btn-lg btn-block">
```

Block level button

```
</button>
```

</p>

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Button State**

Bootstrap provides classes which allow you to change the state of buttons as active, disabled etc. each of which are discussed in the following sections.

- **Active State**

Buttons will appear pressed (with a darker background, darker border, and inset shadow) when active. The following table summarizes classes used to make button elements and anchor elements active –

Element	Class
Button element	Use .active class to show that it is activated.
Anchor element	Use .active class to <a> buttons to show that it is activated.

The following example demonstrates this –

```
<p>  
<button type = "button" class = "btn btn-default btn-lg ">  
  Default Button  
</button>  
  
<button type = "button" class = "btn btn-default btn-lg active">  
  Active Button  
</button>  
</p>  
  
<p>  
<button type = "button" class = "btn btn-primary btn-lg">  
  Primary button  
</button>  
  
<button type = "button" class = "btn btn-primary btn-lg active">  
  Active Primary button  
</button>  
</p>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Bootstrap - Images**

Bootstrap provides three classes that can be used to apply some simple styles to images –

- .img-rounded – adds border-radius:6px to give the image rounded corners.
- .img-circle – makes the entire image round by adding border-radius:500px.
- .img-thumbnail – adds a bit of padding and a gray border –

The following example demonstrates this –

```
<img src = "/bootstrap/images/download.png" class = "img-rounded">
```

```
<img src = "/bootstrap/images/download.png" class = "img-circle">
```

```
<img src = "/bootstrap/images/download.png" class = "img-thumbnail">
```

- **Bootstrap Icons**

Icons by Glyphicons

Bootstrap 3.3.7 includes more than 260 Glyphicons. As opposed to Bootstrap 2.x sprite version these icons are available in font format for better usability and scalability.

Since these Glyphicons are font based, so you can create icons of any color just through applying the CSS color property on the specific element.

How to Use Icons in Your Code

To use the Bootstrap icons you will require an `` tag along with a base class `glyphicon` and an individual icon class `glyphicon-*`. The general syntax for using Bootstrap icons is:

```
<span class="glyphicon glyphicon-class-name"></span>
```

Where `glyphicon-class-name` is the name of the particular icon class (e.g. `glyphicon-search`, `glyphicon-user`, `glyphicon-star` etc.) defined in `bootstrap.css`.

For example, to use search icon you can place the following code just about anywhere:

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- `<button type="submit" class="btn btn-default"> Search</button>`
- `<button type="submit" class="btn btn-primary"> Search</button>`

- **Bootstrap - Dropdowns**

Dropdown menus are toggleable, contextual menus for displaying links in a list format. This can be made interactive with the dropdown JavaScript plugin.

To use dropdown, just wrap the dropdown menu within the class `.dropdown`.The following example demonstrates a basic dropdown menu –

```
<div class = "dropdown">

  <button type = "button" class = "btn dropdown-toggle" id = "dropdownMenu1" data-toggle =
"dropdown">
  Topics
  <span class = "caret"></span>
</button>

<ul class = "dropdown-menu" role = "menu" aria-labelledby = "dropdownMenu1">
  <li role = "presentation">
    <a role = "menuitem" tabindex = "-1" href = "#">Java</a>
  </li>

  <li role = "presentation">
    <a role = "menuitem" tabindex = "-1" href = "#">Data Mining</a>
  </li>

  <li role = "presentation">
    <a role = "menuitem" tabindex = "-1" href = "#">
      Data Communication/Networking
    </a>
  </li>

  <li role = "presentation" class = "divider"></li>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<li role = "presentation">  
  <a role = "menuitem" tabindex = "-1" href = "#">Separated link</a>  
</li>  
</ul>
```

</div>

• Options

Alignment

Align the dropdown menu to right by adding the class **.pull-right** to **.dropdown-menu**. The following example demonstrates this –

```
<div class = "dropdown">  
  
  <button type = "button" class = "btn dropdown-toggle" id = "dropdownMenu1" data-toggle =  
"dropdown">  
    Topics  
    <span class = "caret"></span>  
  </button>  
  
  <ul class = "dropdown-menu pull-right" role = "menu" aria-labelledby = "dropdownMenu1">  
    <li role = "presentation">  
      <a role = "menuitem" tabindex = "-1" href = "#">Java</a>  
    </li>  
  
    <li role = "presentation">  
      <a role = "menuitem" tabindex = "-1" href = "#">Data Mining</a>  
    </li>  
  
    <li role = "presentation">  
      <a role = "menuitem" tabindex = "-1" href = "#">  
        Data Communication/Networking  
      </a>  
    </li>  
  
    <li role = "presentation" class = "divider"></li>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<li role = "presentation">  
  <a role = "menuitem" tabindex = "-1" href = "#">Separated link</a>  
</li>  
</ul>
```

</div>

- **Headers**

You can add a header to label sections of actions in any dropdown menu by using the class **dropdown-header**. The following example demonstrates this –

```
<div class = "dropdown">  
  
  <button type = "button" class = "btn dropdown-toggle" id = "dropdownMenu1" data-toggle =  
"dropdown">  
    Topics  
    <span class = "caret"></span>  
  </button>  
  
  <ul class = "dropdown-menu" role = "menu" aria-labelledby = "dropdownMenu1">  
    <li role = "presentation" class = "dropdown-header">Dropdown header</li>  
  
    <li role = "presentation" >  
      <a role = "menuitem" tabindex = "-1" href = "#">Java</a>  
    </li>  
  
    <li role = "presentation">  
      <a role = "menuitem" tabindex = "-1" href = "#">Data Mining</a>  
    </li>  
  
    <li role = "presentation">  
      <a role = "menuitem" tabindex = "-1" href = "#">  
        Data Communication/Networking  
      </a>  
    </li>  
  
    <li role = "presentation" class = "divider"></li>  
    <li role = "presentation" class = "dropdown-header">Dropdown header</li>  
  
    <li role = "presentation">
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<a role = "menuitem" tabindex = "-1" href = "#">Separated link</a>  
</li>  
</ul>
```

```
</div>
```

- **Bootstrap - Button Groups**

Button groups allow multiple buttons to be stacked together on a single line. This is useful when you want to place items like alignment buttons together. You can add on optional JavaScript radio and checkbox style behavior with Bootstrap Button Plugin.

Following table summarizes the important classes Bootstrap provides to use button groups –

Class	Description	Code Sample
.btn-group	This class is used for a basic button group. Wrap a series of buttons with class .btn in .btn-group .	<pre><div class = "btn-group"> <button type = "button" class = "btn btn-default">Button1</button> <button type = "button" class = "btn btn-default">Button2</button> </div></pre>
.btn-toolbar	This helps to combine sets of .btn-group into a .btn-toolbar for more complex components.	<pre><div class = "btn-toolbar" role = "toolbar"> <div class = "btn-group">...</div> <div class = "btn-group">...</div> </div></pre>
.btn-group-lg, .btn-group-sm, .btn-group-xs	These classes can be applied to button group instead of resizing each button.	<pre><div class = "btn-group btn-group- lg">...</div> <div class = "btn-group btn-group- sm">...</div> <div class = "btn-group btn-group- xs">...</div></pre>
.btn-group-vertical	This class make a set of buttons appear vertically stacked rather than horizontally.	<pre><div class = "btn-group-vertical"> ... </div></pre>

- **Basic Button Group**

The following example demonstrates the use of class **.btn-group** discussed in the above table –

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<div class = "btn-group">
```

```
<button type = "button" class = "btn btn-default">Button 1</button>
```

```
<button type = "button" class = "btn btn-default">Button 2</button>
```

```
<button type = "button" class = "btn btn-default">Button 3</button>
```

```
</div>
```

- **Button Toolbar**

The following example demonstrates the use of class **.btn-toolbar** discussed in the above table –

```
<div class = "btn-toolbar" role = "toolbar">
```

```
<div class = "btn-group">
```

```
<button type = "button" class = "btn btn-default">Button 1</button>
```

```
<button type = "button" class = "btn btn-default">Button 2</button>
```

```
<button type = "button" class = "btn btn-default">Button 3</button>
```

```
</div>
```

```
<div class = "btn-group">
```

```
<button type = "button" class = "btn btn-default">Button 4</button>
```

```
<button type = "button" class = "btn btn-default">Button 5</button>
```

```
<button type = "button" class = "btn btn-default">Button 6</button>
```

```
</div>
```

```
<div class = "btn-group">
```

```
<button type = "button" class = "btn btn-default">Button 7</button>
```

```
<button type = "button" class = "btn btn-default">Button 8</button>
```

```
<button type = "button" class = "btn btn-default">Button 9</button>
```

```
</div>
```

```
</div>
```

- **Button Size**

The following example demonstrates the use of class **.btn-group-*** discussed in the above table –

```
<div class = "btn-group btn-group-lg">
```

```
<button type = "button" class = "btn btn-default">Button 1</button>
```

```
<button type = "button" class = "btn btn-default">Button 2</button>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<button type = "button" class = "btn btn-default">Button 3</button>
</div>
```

```
<div class = "btn-group btn-group-sm">
  <button type = "button" class = "btn btn-default">Button 4</button>
  <button type = "button" class = "btn btn-default">Button 5</button>
  <button type = "button" class = "btn btn-default">Button 6</button>
</div>
```

```
<div class = "btn-group btn-group-xs">
  <button type = "button" class = "btn btn-default">Button 7</button>
  <button type = "button" class = "btn btn-default">Button 8</button>
  <button type = "button" class = "btn btn-default">Button 9</button>
</div>
```

- **Nesting**

You can nest button groups within another button group i.e, place a **.btn-group** within another **.btn-group**. This is done when you want dropdown menus mixed with a series of buttons.

```
<div class = "btn-group">
  <button type = "button" class = "btn btn-default">Button 1</button>
  <button type = "button" class = "btn btn-default">Button 2</button>
```

```
<div class = "btn-group">
  <button type = "button" class = "btn btn-default dropdown-toggle" data-toggle = "dropdown">
    Dropdown
    <span class = "caret"></span>
  </button>
```

```
<ul class = "dropdown-menu">
  <li><a href = "#">Dropdown link 1</a></li>
  <li><a href = "#">Dropdown link 2</a></li>
</ul>
```

```
</div>
```

```
</div>
```

- **Vertical Buttongroup**

The following example demonstrates the use of class **.btn-group-vertical** discussed in the above table –

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<div class = "btn-group-vertical">
```

```
<button type = "button" class = "btn btn-default">Button 1</button>
```

```
<button type = "button" class = "btn btn-default">Button 2</button>
```

```
<div class = "btn-group-vertical">
```

```
<button type = "button" class = "btn btn-default dropdown-toggle" data-toggle = "dropdown">
```

```
Dropdown
```

```
<span class = "caret"></span>
```

```
</button>
```

```
<ul class = "dropdown-menu">
```

```
<li><a href = "#">Dropdown link 1</a></li>
```

```
<li><a href = "#">Dropdown link 2</a></li>
```

```
</ul>
```

```
</div>
```

```
</div>
```

• Bootstrap - Navigation Elements

Bootstrap provides a few different options for styling navigation elements. All of them share the same markup and base class, **.nav**. Bootstrap also provides a helper class, to share markup and states. Swap modifier classes to switch between each style.

Tabular Navigation or Tabs

To create a tabbed navigation menu –

- Start with a basic unordered list with the base class of **.nav**
- Add class **.nav-tabs**.

The following example demonstrates this –

```
<p>Tabs Example</p>
```

```
<ul class = "nav nav-tabs">
```

```
<li class = "active"><a href = "#">Home</a></li>
```

```
<li><a href = "#">SVN</a></li>
```

```
<li><a href = "#">iOS</a></li>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<li><a href = "#">VB.Net</a></li>
<li><a href = "#">Java</a></li>
<li><a href = "#">PHP</a></li>
</ul>
```

• Pills Navigation

Basic Pills

To turn the tabs into pills, follow the same steps as above, use the class **.nav-pills** instead of **.nav-tabs**.

The following example demonstrates this –

```
<p>Pills Example</p>
```

```
<ul class = "nav nav-pills">
  <li class = "active"><a href = "#">Home</a></li>
  <li><a href = "#">SVN</a></li>
  <li><a href = "#">iOS</a></li>
  <li><a href = "#">VB.Net</a></li>
  <li><a href = "#">Java</a></li>
  <li><a href = "#">PHP</a></li>
</ul>
```

Vertical Pills

You can stack the pills vertically using the class **.nav-stacked** along with the classes – **.nav**, **.nav-pills**. The following example demonstrates this –

```
<p>Vertical Pills Example</p>
```

```
<ul class = "nav nav-pills nav-stacked">
  <li class = "active"><a href = "#">Home</a></li>
  <li><a href = "#">SVN</a></li>
  <li><a href = "#">iOS</a></li>
  <li><a href = "#">VB.Net</a></li>
  <li><a href = "#">Java</a></li>
  <li><a href = "#">PHP</a></li>
</ul>
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Justified Nav**

You can make tabs or pills of equal widths as of their parent at screens wider than 768px using class `.nav-justified` along with `.nav`, `.nav-tabs` or `.nav`, `.nav-pills` respectively. On smaller screens, the nav links are stacked.

The following example demonstrates this –

<p>Justified Nav Elements Example</p>

```
<ul class = "nav nav-pills nav-justified">
  <li class = "active"><a href = "#">Home</a></li>
  <li><a href = "#">SVN</a></li>
  <li><a href = "#">iOS</a></li>
  <li><a href = "#">VB.Net</a></li>
  <li><a href = "#">Java</a></li>
  <li><a href = "#">PHP</a></li>
</ul>
```

```
<br>
<br>
<br>
```

```
<ul class = "nav nav-tabs nav-justified">
  <li class = "active"><a href = "#">Home</a></li>
  <li><a href = "#">SVN</a></li>
  <li><a href = "#">iOS</a></li>
  <li><a href = "#">VB.Net</a></li>
  <li><a href = "#">Java</a></li>
  <li><a href = "#">PHP</a></li>
</ul>
```

- **Bootstrap - Navbar**

The navbar is one of the prominent features of Bootstrap sites. Navbars are responsive 'meta' components that serve as navigation headers for your application or site. Navbars collapse in mobile views and become horizontal as the available viewport width increases. At its core, the navbar includes styling for site names and basic navigation.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Default Navbar**

To create a default navbar –

- Add the classes **.navbar**, **.navbar-default** to the `<nav>` tag.
- Add **role = "navigation"** to the above element, to help with accessibility.
- Add a header class **.navbar-header** to the `<div>` element. Include an `<a>` element with class **navbar-brand**. This will give the text a slightly larger size.
- To add links to the navbar, simply add an unordered list with the classes of **.nav**, **.navbar-nav**.

The following example demonstrates this –

```
<nav class = "navbar navbar-default" role = "navigation">
  <div class = "navbar-header">
    <a class = "navbar-brand" href = "#">TutorialsPoint</a>
  </div>

  <div>
    <ul class = "nav navbar-nav">
      <li class = "active"><a href = "#">iOS</a></li>
      <li><a href = "#">SVN</a></li>
      <li class = "dropdown">
        <a href = "#" class = "dropdown-toggle" data-toggle = "dropdown">
          Java
          <b class = "caret"></b>
        </a>
        <ul class = "dropdown-menu">
          <li><a href = "#">jmeter</a></li>
          <li><a href = "#">EJB</a></li>
          <li><a href = "#">Jasper Report</a></li>
          <li class = "divider"></li>
          <li><a href = "#">Separated link</a></li>
          <li class = "divider"></li>
          <li><a href = "#">One more separated link</a></li>
        </ul>
      </li>
    </ul>
  </div>
</nav>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Responsive Navbar**

To add responsive features to the navbar, the content that you want to be collapsed needs to be wrapped in a <div> with classes **.collapse**, **.navbar-collapse**. The collapsing nature is tripped by a button that has the class of **.navbar-toggle** and then features two data- elements. The first, **data-toggle**, is used to tell the JavaScript what to do with the button, and the second, **data-target**, indicates which element to toggle. Then with a class **.icon-bar** create what we like to call the hamburger button. This will toggle the elements that are in the **.nav-collapse** <div>. For this feature to work, you need to include the Bootstrap Collapse Plugin.

The following example demonstrates this –

```
<nav class = "navbar navbar-default" role = "navigation">

<div class = "navbar-header">
  <button type = "button" class = "navbar-toggle"
    data-toggle = "collapse" data-target = "#example-navbar-collapse">
    <span class = "sr-only">Toggle navigation</span>
    <span class = "icon-bar"></span>
    <span class = "icon-bar"></span>
    <span class = "icon-bar"></span>
  </button>

  <a class = "navbar-brand" href = "#">TutorialsPoint</a>
</div>

<div class = "collapse navbar-collapse" id = "example-navbar-collapse">

<ul class = "nav navbar-nav">
  <li class = "active"><a href = "#">iOS</a></li>
  <li><a href = "#">SVN</a></li>

  <li class = "dropdown">
    <a href = "#" class = "dropdown-toggle" data-toggle = "dropdown">
      Java
      <b class = "caret"></b>
    </a>

    <ul class = "dropdown-menu">
      <li><a href = "#">jmeter</a></li>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<li><a href = "#">EJB</a></li>
<li><a href = "#">Jasper Report</a></li>
```

```
<li class = "divider"></li>
<li><a href = "#">Separated link</a></li>
```

```
<li class = "divider"></li>
<li><a href = "#">One more separated link</a></li>
</ul>
```

```
</li>
```

```
</ul>
```

```
</div>
```

```
</nav>
```

- **Bootstrap - Breadcrumb**

Breadcrumbs are a great way to show hierarchy-based information for a site. In the case of blogs, breadcrumbs can show the dates of publishing, categories, or tags. They indicate the current page's location within a navigational hierarchy.

A breadcrumb in Bootstrap is simply an unordered list with a class of **.breadcrumb**. The separator is automatically added by CSS (bootstrap.min.css) through the following class –

```
.breadcrumb > li + li:before {
  color: #CCCCCC;
  content: "/ ";
  padding: 0 5px;
}
```

The following example demonstrates breadcrumbs –

```
<ol class = "breadcrumb">
  <li><a href = "#">Home</a></li>
  <li><a href = "#">2013</a></li>
  <li class = "active">November</li>
</ol>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Bootstrap - Pagination**

Pagination, an unordered list is handled by Bootstrap like a lot of other interface elements.

Pagination

The following table lists the classes that Bootstrap provides to handle pagination.

Class	Description	Sample code
.pagination	Add this class to get the pagination on your page.	<pre><ul class = "pagination"> &laquo; 1 </pre>
.disabled, .active	You can customize links by using .disabled for unclickable links and .active to indicate the current page.	<pre><ul class = "pagination"> <li class = "disabled">&laquo; <li class = "active">1(current) </pre>
.pagination-lg, .pagination-sm	Use these classes to get different size items.	<pre><ul class = "pagination pagination-lg">... <ul class = "pagination">... <ul class = "pagination pagination-sm">...</pre>

- **Default Pagination**

The following example demonstrates the use of class .pagination discussed in the above table –

```
<ul class = "pagination">
  <li><a href = "#">&laquo;</a></li>
  <li><a href = "#">1</a></li>
  <li><a href = "#">2</a></li>
  <li><a href = "#">3</a></li>
  <li><a href = "#">4</a></li>
  <li><a href = "#">5</a></li>
  <li><a href = "#">&raquo;</a></li>
</ul>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- States

The following example demonstrates the use of class **.disabled**, **.active** discussed in the above table –

```
<ul class = "pagination">
  <li><a href = "#">&laquo;</a></li>
  <li class = "active"><a href = "#">1</a></li>
  <li class = "disabled"><a href = "#">2</a></li>
  <li><a href = "#">3</a></li>
  <li><a href = "#">4</a></li>
  <li><a href = "#">5</a></li>
  <li><a href = "#">&raquo;</a></li>
</ul>
```

- Sizing

The following example demonstrates the use of classes for sizing, **.pagination-*** discussed in the above table –

```
<ul class = "pagination pagination-lg">
  <li><a href = "#">&laquo;</a></li>
  <li><a href = "#">1</a></li>
  <li><a href = "#">2</a></li>
  <li><a href = "#">3</a></li>
  <li><a href = "#">4</a></li>
  <li><a href = "#">5</a></li>
  <li><a href = "#">&raquo;</a></li>
</ul>
```



```
<ul class = "pagination">
  <li><a href = "#">&laquo;</a></li>
  <li><a href = "#">1</a></li>
  <li><a href = "#">2</a></li>
  <li><a href = "#">3</a></li>
  <li><a href = "#">4</a></li>
  <li><a href = "#">5</a></li>
  <li><a href = "#">&raquo;</a></li>
</ul>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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


```
<ul class = "pagination pagination-sm">
```

```
<li><a href = "#">&laquo;</a></li>
```

```
<li><a href = "#">1</a></li>
```

```
<li><a href = "#">2</a></li>
```

```
<li><a href = "#">3</a></li>
```

```
<li><a href = "#">4</a></li>
```

```
<li><a href = "#">5</a></li>
```

```
<li><a href = "#">&raquo;</a></li>
```

```
</ul>
```

- **Bootstrap - Labels**

This chapter covers Bootstrap labels. Labels are great for offering counts, tips, or other markup for pages. Use class **.label** to display labels as shown in the following example –

```
<h1>Example Heading <span class = "label label-default">Label</span></h1>
```

```
<h2>Example Heading <span class = "label label-default">Label</span></h2>
```

```
<h3>Example Heading <span class = "label label-default">Label</span></h3>
```

```
<h4>Example Heading <span class = "label label-default">Label</span></h4>
```

You can modify the appearance of the labels using the modifier classes such as, **label-default**, **label-primary**, **label-success**, **label-info**, **label-warning**, **label-danger** as shown in the following example –

```
<span class = "label label-default">Default Label</span>
```

```
<span class = "label label-primary">Primary Label</span>
```

```
<span class = "label label-success">Success Label</span>
```

```
<span class = "label label-info">Info Label</span>
```

```
<span class = "label label-warning">Warning Label</span>
```

```
<span class = "label label-danger">Danger Label</span>
```

- **Bootstrap - Badges**

Badges are similar to labels; the primary difference is that the corners are more rounded.

Badges are mainly used to highlight new or unread items. To use badges just add **** to links, Bootstrap navs, and more.

The following example demonstrates this –

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<a href = "#">Mailbox <span class = "badge">50</span></a>
```

When there are no new or unread items, badges will simply collapse via CSS's **:empty** selector, provided no content exists within.

- **Active Nav States**

You can place badges in active states of pill and list navigations. You can achieve this by placing **** to active links, as demonstrated in the following example -

```
<h4>Example for Active State in Pill </h4>
```

```
<ul class = "nav nav-pills">  
  <li class = "active"><a href = "#">Home <span class = "badge">42</span></a></li>  
  <li><a href = "#">Profile</a></li>  
  <li><a href = "#">Messages <span class = "badge">3</span></a></li>  
</ul>
```

```
<br>
```

```
<h4>Example for Active State in navigations</h4>
```

```
<ul class = "nav nav-pills nav-stacked" style = "max-width: 260px;">  
  <li class = "active">  
    <a href = "#">  
      <span class = "badge pull-right">42</span>  
      Home  
    </a>  
  </li>  
  
  <li><a href = "#">Profile</a></li>  
  
  <li>  
    <a href = "#">  
      <span class = "badge pull-right">3</span>  
      Messages  
    </a>  
  </li>  
</ul>
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Bootstrap - Thumbnails**

A lot of sites need a way to lay out images, videos, text, etc, in a grid, and Bootstrap has an easy way to do this with thumbnails. To create thumbnails using Bootstrap –

- Add an `<a>` tag with the class of **.thumbnail** around an image.
- This adds four pixels of padding and a gray border.
- On hover, an animated glow outlines the image.

The following example demonstrates a default thumbnail –

```
<div class = "row">

<div class = "col-sm-6 col-md-3">
  <a href = "#" class = "thumbnail">
    <img src = "/bootstrap/images/kittens.jpg" alt = "Generic placeholder thumbnail">
  </a>
</div>

<div class = "col-sm-6 col-md-3">
  <a href = "#" class = "thumbnail">
    <img src = "/bootstrap/images/kittens.jpg" alt = "Generic placeholder thumbnail">
  </a>
</div>

<div class = "col-sm-6 col-md-3">
  <a href = "#" class = "thumbnail">
    <img src = "/bootstrap/images/kittens.jpg" alt = "Generic placeholder thumbnail">
  </a>
</div>

<div class = "col-sm-6 col-md-3">
  <a href = "#" class = "thumbnail">
    <img src = "/bootstrap/images/kittens.jpg" alt = "Generic placeholder thumbnail">
  </a>
</div>

</div>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

• Bootstrap - Alerts

Alerts provide a way to style messages to the user. They provide contextual feedback messages for typical user actions.

You can add an optional close icon to alert. For inline dismissal use the Alerts jQuery plugin.

You can add a basic alert by creating a wrapper `<div>` and adding a class of `.alert` and one of the four contextual classes (e.g., `.alert-success`, `.alert-info`, `.alert-warning`, `.alert-danger`). The following example demonstrates this –

```
<div class = "alert alert-success">Success! Well done its submitted.</div>
<div class = "alert alert-info">Info! take this info.</div>
<div class = "alert alert-warning">Warning ! Dont submit this.</div>
<div class = "alert alert-danger">Error ! Change few things.</div>
```

• Dismissal Alerts

To build a dismissal alert –

- Add a basic alert by creating a wrapper `<div>` and adding a class of `.alert` and one of the four contextual classes (e.g., `.alert-success`, `.alert-info`, `.alert-warning`, `.alert-danger`)
- Also add optional `.alert-dismissible` to the above `<div>` class.
- Add a close button.

The following example demonstrates this –

```
<div class = "alert alert-success alert-dismissible">
  <button type = "button" class = "close" data-dismiss = "alert" aria-hidden = "true">
    &times;
  </button>
```

```
  Success! Well done its submitted.
</div>
```

```
<div class = "alert alert-info alert-dismissible">
  <button type = "button" class = "close" data-dismiss = "alert" aria-hidden = "true">
    &times;
  </button>
```

```
  Info! take this info.
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

</div>

```
<div class = "alert alert-warning alert-dismissible">
```

```
<button type = "button" class = "close" data-dismiss = "alert" aria-hidden = "true">
```

```
&times;
```

```
</button>
```

Warning ! Dont submit this.

</div>

```
<div class = "alert alert-danger alert-dismissible">
```

```
<button type = "button" class = "close" data-dismiss = "alert" aria-hidden = "true">
```

```
&times;
```

```
</button>
```

Error ! Change few things.

</div>

Links in Alerts

To get links in alerts –

- Add a basic alert by creating a wrapper <div> and adding a class of **.alert** and one of the four contextual classes (e.g., **.alert-success**, **.alert-info**, **.alert-warning**, **.alert-danger**)
- Use the **.alert-link** utility class to quickly provide matching colored links within any alert.

```
<div class = "alert alert-success">
```

```
<a href = "#" class = "alert-link">Success! Well done its submitted.</a>
```

```
</div>
```

```
<div class = "alert alert-info">
```

```
<a href = "#" class = "alert-link">Info! take this info.</a>
```

```
</div>
```

```
<div class = "alert alert-warning">
```

```
<a href = "#" class = "alert-link">Warning ! Dont submit this.</a>
```

```
</div>
```

```
<div class = "alert alert-danger">
```

```
<a href = "#" class = "alert-link">Error ! Change few things.</a>
```

```
</div>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Bootstrap - Progress Bars**

The purpose of progress bars is to show that assets are loading, in progress, or that there is action taking place regarding elements on the page.

Default Progress Bar

To create a basic progress bar –

- Add a <div> with a class of **.progress**.
- Next, inside the above <div>, add an empty <div> with a class of **.progress-bar**.
- Add a style attribute with the width expressed as a percentage. Say for example, style = "60%"; indicates that the progress bar was at 60%.

Let us see an example below –

```
<div class = "progress">
  <div class = "progress-bar" role = "progressbar" aria-valuenow = "60"
    aria-valuemin = "0" aria-valuemax = "100" style = "width: 40%;">

    <span class = "sr-only">40% Complete</span>
  </div>
</div>
```

- **Alternate Progress Bar**

To create a progress bar with different styles –

- Add a <div> with a class of **.progress**.
- Next, inside the above <div>, add an empty <div> with a class of **.progress-bar** and class **progress-bar-*** where * could be **success, info, warning, danger**.
- Add a style attribute with the width expressed as a percentage. Say for example, style = "60%"; indicates that the progress bar was at 60%.

Let us see an example below –

```
<div class = "progress">
  <div class = "progress-bar progress-bar-success" role = "progressbar"
    aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style = "width: 90%;">

    <span class = "sr-only">90% Complete (Sucess)</span>
  </div>
</div>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

</div>
</div>

```
<div class = "progress">  
  <div class = "progress-bar progress-bar-info" role = "progressbar"  
    aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style = "width: 30%;">  
  
    <span class = "sr-only">30% Complete (info)</span>  
  </div>  
</div>
```

```
<div class = "progress">  
  <div class = "progress-bar progress-bar-warning" role = "progressbar"  
    aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style = "width: 20%;">  
  
    <span class = "sr-only">20%Complete (warning)</span>  
  </div>  
</div>
```

```
<div class = "progress">  
  <div class = "progress-bar progress-bar-danger" role = "progressbar"  
    aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style = "width: 10%;">  
  
    <span class = "sr-only">10% Complete (danger)</span>  
  </div>  
</div>
```

- **Striped Progress Bar**

To create a striped progress bar –

- Add a <div> with a class of **.progress** and **.progress-striped**.
- Next, inside the above <div>, add an empty <div> with a class of **.progress-bar** and class **progress-bar-*** where * could be **success**, **info**, **warning**, **danger**.
- Add a style attribute with the width expressed as a percentage. Say for example, style = "60%"; indicates that the progress bar was at 60%.

Let us see an example below –

```
<div class = "progress progress-striped">  
  <div class = "progress-bar progress-bar-success" role = "progressbar"
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style = "width: 90%;">
```

```
<span class = "sr-only">90% Complete (Sucess)</span>
```

```
</div>
```

```
</div>
```

```
<div class = "progress progress-striped">
```

```
<div class = "progress-bar progress-bar-info" role = "progressbar"
```

```
aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style = "width: 30%;">
```

```
<span class = "sr-only">30% Complete (info)</span>
```

```
</div>
```

```
</div>
```

```
<div class = "progress progress-striped">
```

```
<div class = "progress-bar progress-bar-warning" role = "progressbar"
```

```
aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style="width: 20%;">
```

```
<span class = "sr-only">20%Complete (warning)</span>
```

```
</div>
```

```
</div>
```

```
<div class = "progress progress-striped">
```

```
<div class = "progress-bar progress-bar-danger" role = "progressbar"
```

```
aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style = "width: 10%;">
```

```
<span class = "sr-only">10% Complete (danger)</span>
```

```
</div>
```

```
</div>
```

- **Animated Progress Bar**

To create an animated progress bar –

- Add a <div> with a class of **.progress** and **.progress-striped**. Also add class **.active** to **.progress-striped**.
- Next, inside the above <div>, add an empty <div> with a class of **.progress-bar**.
- Add a style attribute with the width expressed as a percentage. Say for example, style = "60%"; indicates that the progress bar was at 60%.

This will animate the stripes right to left.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Let us see an example below –

```
<div class = "progress progress-striped active">
  <div class = "progress-bar progress-bar-success" role = "progressbar"
    aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style = "width: 40%;">

    <span class = "sr-only">40% Complete</span>
  </div>
</div>
```

- **Stacked Progress Bar**

You can even stack multiple progress bars. Place the multiple progress bars into the same **.progress** to stack them as seen in the following example –

```
<div class = "progress">

  <div class = "progress-bar progress-bar-success" role = "progressbar"
    aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style = "width: 40%;">

    <span class = "sr-only">40% Complete</span>
  </div>

  <div class = "progress-bar progress-bar-info" role = "progressbar"
    aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style = "width: 30%;">

    <span class = "sr-only">30% Complete (info)</span>
  </div>

  <div class = "progress-bar progress-bar-warning" role = "progressbar"
    aria-valuenow = "60" aria-valuemin = "0" aria-valuemax = "100" style = "width: 20%;">

    <span class = "sr-only">20%Complete (warning)</span>
  </div>

</div>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Bootstrap - Wells**

A well is a container in `<div>` that causes the content to appear sunken or an inset effect on the page. To create a well, simply wrap the content that you would like to appear in the well with a `<div>` containing the class of `.well`. The following example shows a default well –

```
<div class = "well">Hi, am in well !!</div>
```

Hi, am in well !!

Sizing

You can change the size of well using the optional classes such as, `well-lg` or `well-sm`. These classes are used in conjunction with `.well` class. These affect the padding, making the well larger or smaller depending on the class.

```
<div class = "well well-lg">Hi, am in large well !!</div>
```

```
<div class = "well well-sm">Hi, am in small well !!</div>
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

UNIT -2

Introduction to Laravel, Installation & Configuration

- What is Laravel ?

Laravel is a free, open-source PHP web framework, created by Taylor Otwell and intended for the development of web applications following the model–view–controller (MVC) architectural pattern. Some of the features of Laravel are a modular packaging system with a dedicated dependency manager, different ways for accessing relational databases, utilities that aid in application deployment and maintenance.

- Features of Laravel

Let's find features of PHP Laravel Framework that make it popular among developers and businesses alike:

1. Template Engine

Laravel framework is highly acknowledged for its **inbuilt lightweight templates** that help you **create amazing layouts using dynamic content seeding**. In addition to this, it has multiple widgets incorporating CSS and JS code with solid structures. The templates of laravel frameworks are innovatively designed to create a simple layout with distinctive sections.

2. Artisan

Laravel offers a built-in tool for command line known as Artisan which **allows performing the majority of those tedious and repetitive programming tasks** that many developers avoid performing manually. These artisans can also be utilized to create the database structure, a skeleton code, and build their migration that becomes pretty easy-to-manage the database system. In addition, it can be interlaced to generate the basic MVC files right away through the command-line and manage those assets and their respective configurations. Artisan even helps developers to create their own commands and do convenient things with it.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

3. Eloquent ORM (object-relational mapping)

Laravel framework offers the Eloquent ORM that includes a simple PHP Active Record implementation. It lets the web app developers issue database queries with PHP syntax rather than writing SQL code. Every table in the database possesses a corresponding Model through which the developer interacts with said table. **An ORM is relatively faster than all other PHP frameworks.**

4. Libraries & Modular:

Laravel is also popular due to its Object Oriented libraries as well as many other pre-installed libraries. These pre-installed libraries are not found in any other PHP frameworks. One of the preinstalled libraries is an Authentication library which is easy-to-implement and has many latest features, such as checking active users, Bcrypt hashing, password reset, CSRF (Cross-site Request Forgery) protection, and encryption. Furthermore, this framework is divided into individual modules that adopt modern PHP principles allowing developers to **build responsive, modular, and handy web apps.**

5. MVC Architecture Support:

The MVC pattern of laravel ensures clarity of logic and presentation. This architecture support helps in **improving the performance, allowing better documentation, and has multiple built-in functions.**

6. Migration system for databases

Laravel migration system **helps to expand the structure of the database of the web application without re-creating every time developers make a change.** Owing to this feature, the risk of losing data is very less. It not only offers the facility to change the database structure but it helps to do this by using PHP code instead of SQL. In addition to this, Laravel Schema Builder helps to create database tables and inserts indices or columns promptly.

7. Unit-Testing

Many web app developers like Laravel due to the way **it facilitates unit testing.** This framework of PHP web development is capable of running many tests to ensure that new changes done by programmers do

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

not unexpectedly break anything in the web application. Generally, laravel is considered to have stable releases in the industry as it is careful of the known failures. In addition to this, it makes it easy to write unit-tests for its own code.

8. Security

Application security is one of the most important concerns in web application development. While developing an application, every programmer has to use some effective ways to make it secure. **Laravel takes care of the web application security within its own framework.** It uses hashed and salted password that means the password would never save as the plain text in a database. It also uses “Bcrypt Hashing Algorithm” in order generate an encrypted representation of a password. Additionally, this PHP web development framework uses prepared SQL statements that make injection attacks unimaginable.

- **Model-View-Controller (MVC)**

let’s get a high-level overview of how Laravel applications work. You might have noticed that the standard Laravel application structure has an application directory called app/ with three subdirectories: models/, views/, and controllers/. This is a hint that Laravel follows the **model-view-controller (MVC)** architectural pattern, which enforces a separation between “business logic” from the input and presentation logic associated with a graphical user interface (GUI). In the case of Laravel web applications, the business logic typically consists of data models for things like users, blog posts, and the GUI is just a web page in a web browser. The MVC design pattern is very popular in the web development space.

There are three components to the MVC pattern:

- The **model** - The domain that your software is built around. Models are based on real-world items such as a person, bank account, or product. If you were building a blog, your models might be post and comment. Models are typically permanent and will be stored outside the application, often in a database. A model is more than just data; it enforces all the business rules that apply to that data. For example, if a discount shouldn’t be applied to orders of less than \$10, the model will enforce the constraint. This makes sense; by putting the implementation of these business rules in the model, we make sure that nothing else in the application can make our data invalid. The model acts as both a gatekeeper and a data store.
- The **view** - The visual representation of a model, given some context. It’s usually the resulting markup that the framework renders to the browser, such as the HTML representing the blog post. The view layer is responsible for generating a user interface, normally based on data in the model. For example, an online store will have a list of products to be displayed on a catalog screen. This list will be accessible via the model, but it will be a view that accesses the list from the model and formats it for the end user. Although the view may present the user with various ways of inputting data, the view itself never handles incoming data. The view’s work is done once the data is displayed.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



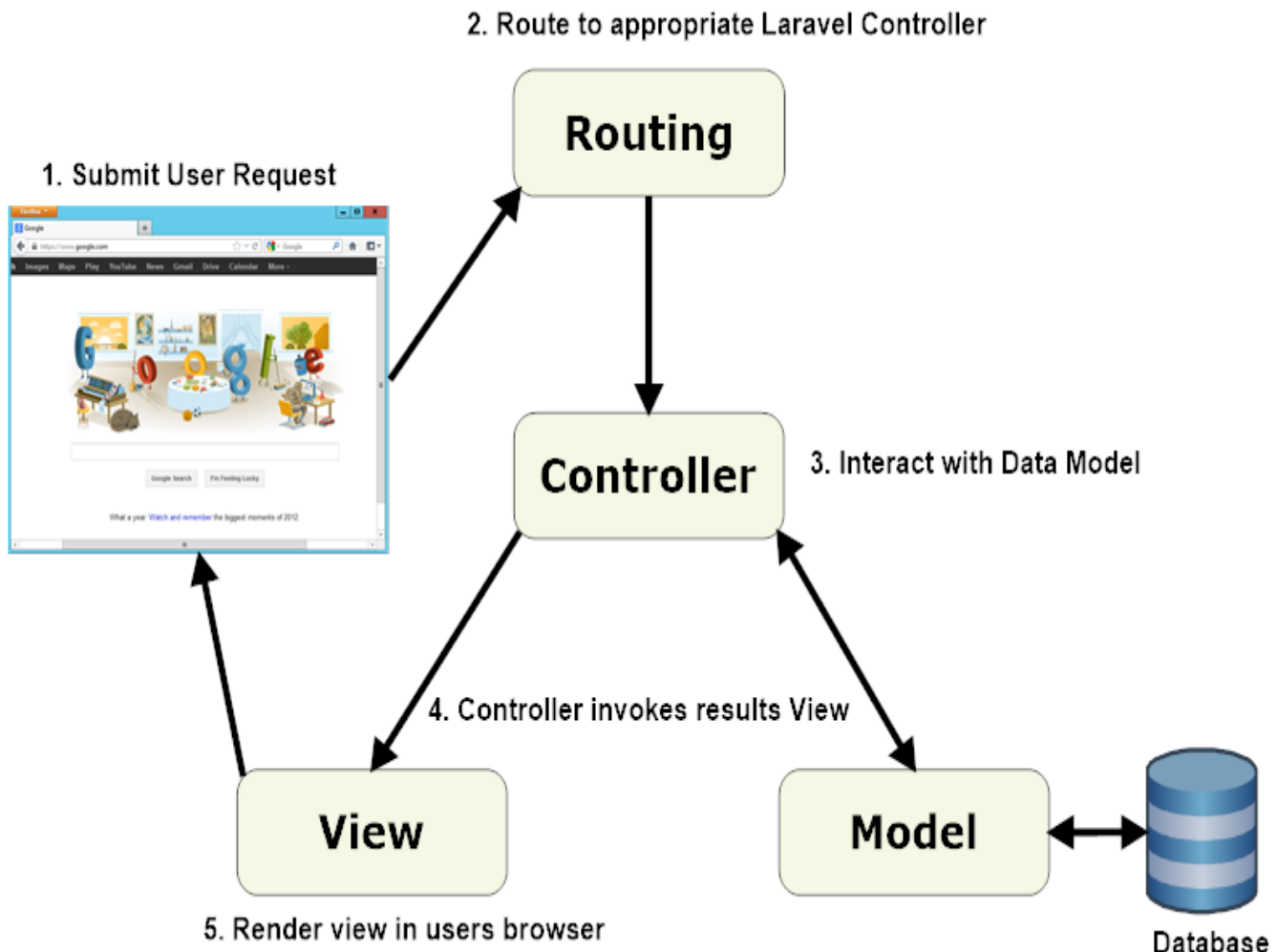
2 – Vaishalinagar
Nr. Amrapali Railway Crossing
Raiya Road
Rajkot – 360001
Ph.No–(0281)2440478,2472590

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

- The **controller** - The coordinator that provides the link between the view and the model. The controller is responsible for processing input, acting upon the model, and deciding on what action should be performed, such as rendering a view or redirecting to another page. Continuing the blog example, the controller might look up the most recent comments for a post (the model) and pass them to the view for rendering.

Components of Laravel

A typical Laravel application consists of the above mentioned MVC components, as you can see below:



When interacting with a Laravel application, a browser sends a request, which is received by a web server and passed on to the Laravel routing engine. The Laravel router receives the request and redirects to the appropriate controller class method based on the routing URL pattern.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

The controller class then takes over. In some cases, the controller will immediately render a view, which is a template that gets converted to HTML and sent back to the browser. More commonly for dynamic sites, the controller interacts with a model, which is a PHP object that represents an element of the application (such as a user, blog post) and is in charge of communicating with the database. After invoking the model, the controller then renders the final view (HTML, CSS, and images) and returns the complete web page to the user's browser.

Laravel promotes the concept that models, views, and controllers should be kept quite separate by storing the code for each of these elements as separate files in separate directories. This is where the Laravel directory structure comes into play.

Design patterns such as MVC are created to make a developer's life easier. This is where Laravel scores over plain PHP which doesn't follow any sort of paradigm. If this discussion seems a bit abstract right now, worry not! After you start working with Laravel, you won't even realize that you are working in a design pattern. It all becomes natural to you after a while.

- **Laravel Application Structure**

When we first open our project, we can see some application structure – folders and files that are already created,

app	16-04-2016 21:20	File folder	
bootstrap	16-04-2016 21:20	File folder	
config	16-04-2016 21:20	File folder	
database	16-04-2016 21:20	File folder	
public	16-04-2016 21:20	File folder	
resources	16-04-2016 21:20	File folder	
storage	16-04-2016 21:20	File folder	
tests	16-04-2016 21:20	File folder	
.env.example	16-04-2016 21:20	EXAMPLE File	1 KB
.gitattributes	16-04-2016 21:20	Text Document	1 KB
.gitignore	16-04-2016 21:20	Text Document	1 KB
artisan	16-04-2016 21:20	File	2 KB
composer.json	16-04-2016 21:20	JSON File	2 KB
composer.lock	16-04-2016 21:20	LOCK File	106 KB
gulpfile.js	16-04-2016 21:20	JavaScript File	1 KB
package.json	16-04-2016 21:20	JSON File	1 KB
phpunit.xml	16-04-2016 21:20	XML File	2 KB
readme.md	16-04-2016 21:20	MD File	2 KB
server.php	16-04-2016 21:20	PHP File	1 KB

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Laravel Application Structure**

app folder, is the main part of the application , all the major part of the code we develop resides in this app folder. It contains Controller files, Model files, Routing file, Request files, handler files and many more, the meat of the application lives here in the app directory.

Console	16-04-2016 21:20	File folder	
Events	16-04-2016 21:20	File folder	
Exceptions	16-04-2016 21:20	File folder	
Http	16-04-2016 21:20	File folder	
Jobs	16-04-2016 21:20	File folder	
Listeners	16-04-2016 21:20	File folder	
Policies	16-04-2016 21:20	File folder	
Providers	16-04-2016 21:20	File folder	
User.php	16-04-2016 21:20	PHP File	1 KB

- **App directory structure**

bootstrap folder, contains few files which are required to when starting the application. It contains autoload files , cache files etc.

config folder, as the name suggests it contains all configuration files for app configuration like whether to allow debugging or not ., database configuration like the database name , host ., mail configuration file, file system configurations.

database folder, contains all files and folders to create and delete schemas and tables.

public folder, contains the front controller, index file, which loads first when the application starts and some other files like .htaccess and robots.txt files.

resources folder, contains all raw assets like css and javacripts , all the view files resides in this directory.

storage folder, contains all the auto generated compiled view(blade template) files, some files about cache and sessions.

test folder, as name suggests it contains all the automated tests.

vendor folder, contains all composer dependencies . If vendor folder doesn't exist , we need to run **composer install** or **composer update** commands in the root directory of the project.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

artisan file, which is a command line interface, many of the commands we use while developing the application are provided by artisan command. `php artisan list` shows all the available commands with artisan

```
F:\laravel\sampleApplication>php artisan list
Laravel Framework version 5.2.29

Usage:
  command [options] [arguments]

Options:
  -h, --help            Display this help message
  -q, --quiet           Do not output any message
  -U, --version         Display this application version
  --ansi               Force ANSI output
  --no-ansi            Disable ANSI output
  -n, --no-interaction Do not ask any interactive question
  --env=ENV            The environment the command should run under.
  -vvvvvv, --verbose  Increase the verbosity of messages: 1 for normal output,
  2 for more verbose output and 3 for debug

Available commands:
  clear-compiled  Remove the compiled class file
  down            Put the application into maintenance mode
  env            Display the current framework environment
  help           Displays help for a command
  list           Lists commands
  migrate        Run the database migrations
  optimize       Optimize the framework for better performance
  serve          Serve the application on the PHP development server
  tinker         Interact with your application
  up            Bring the application out of maintenance mode

  app
  app:name       Set the application namespace
  auth
  auth:clear-resets Flush expired password reset tokens
  cache
  cache:clear    Flush the application cache
  cache:table    Create a migration for the cache database table
  config
  config:cache   Create a cache file for faster configuration loading
  config:clear   Remove the configuration cache file
  db
  db:seed        Seed the database with records
  event
  event:generate Generate the missing events and listeners based on registration
  key
  key:generate   Set the application key
  make
  make:auth      Scaffold basic login and registration views and routes
  make:console   Create a new Artisan command
  make:controller Create a new controller class
  make:event     Create a new event class
  make:job       Create a new job class
  make:listener  Create a new event listener class
  make:middleware Create a new middleware class
  make:migration Create a new migration file
  make:model     Create a new Eloquent model class
  make:policy    Create a new policy class
  make:provider  Create a new service provider class
  make:request   Create a new form request class
  make:seeder   Create a new seeder class
  make:test      Create a new test class
  migrate
  migrate:install Create the migration repository
  migrate:refresh Reset and re-run all migrations
  migrate:reset   Rollback all database migrations
  migrate:rollback Rollback the last database migration
  migrate:status Show the status of each migration
  queue
  queue:failed    List all of the failed queue jobs
  queue:failed-table Create a migration for the failed queue jobs database table
  queue:flush     Flush all of the failed queue jobs
  queue:forget    Delete a failed queue job
  queue:listen    Listen to a given queue
  queue:restart   Restart queue worker daemons after their current job
  queue:retry     Retry a failed queue job
  queue:table     Create a migration for the queue jobs database table
  queue:work      Process the next job on a queue
  route
  route:cache     Create a route cache file for faster route registration
  route:clear     Remove the route cache file
  route:list     List all registered routes
  schedule
  schedule:run   Run the scheduled commands
  session
  session:table  Create a migration for the session database table
  vendor
  vendor:publish Publish any publishable assets from vendor packages
  view
  view:clear     Clear all compiled view files
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **artisan command**

.env file, which is created when we create an application, it contains information about database settings, application key, port settings.

- **Server Requirements for installing Laravel**

The Laravel framework has a few system requirements. Of course, all of these requirements are satisfied by the Laravel Homestead virtual machine, so it's highly recommended that you use Homestead as your local Laravel development environment.

However, if you are not using Homestead, you will need to make sure your server meets the following requirements:

- PHP >= 5.6.4
- OpenSSL PHP Extension
- PDO PHP Extension
- Mbstring PHP Extension
- Tokenizer PHP Extension
- XML PHP Extension

- **Installing Laravel**

Laravel utilizes Composer to manage its dependencies. So, before using Laravel, make sure you have Composer installed on your machine.

Via Laravel Installer

First, download the Laravel installer using Composer:

```
composer global require "laravel/installer"
```

Make sure to place the \$HOME/.composer/vendor/bin directory (or the equivalent directory for your OS) in your \$PATH so the laravel executable can be located by your system.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Once installed, the laravel new command will create a fresh Laravel installation in the directory you specify. For instance, laravel new blog will create a directory named blog containing a fresh Laravel installation with all of Laravel's dependencies already installed:

laravel new blog

- **Via Composer Create-Project**

Alternatively, you may also install Laravel by issuing the Composer create-project command in your terminal:

```
composer create-project --prefer-dist laravel/laravel blog
```

- **Local Development Server**

If you have PHP installed locally and you would like to use PHP's built-in development server to serve your application, you may use the serve Artisan command. This command will start a development server at `http://localhost:8000`:

```
php artisan serve
```

Of course, more robust local development options are available via Homestead and Valet.

- **Configuration**

Public Directory

After installing Laravel, you should configure your web server's document / web root to be the public directory. The `index.php` in this directory serves as the front controller for all HTTP requests entering your application.

Configuration Files

All of the configuration files for the Laravel framework are stored in the `config` directory. Each option is documented, so feel free to look through the files and get familiar with the options available to you.

Directory Permissions

After installing Laravel, you may need to configure some permissions. Directories within the `storage` and the `bootstrap/cache` directories should be writable by your web server or Laravel will not run. If you are using the Homestead virtual machine, these permissions should already be set.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Application Key

The next thing you should do after installing Laravel is set your application key to a random string. If you installed Laravel via Composer or the Laravel installer, this key has already been set for you by the php artisan key:generate command.

Typically, this string should be 32 characters long. The key can be set in the .env environment file. If you have not renamed the .env.example file to .env, you should do that now. **If the application key is not set, your user sessions and other encrypted data will not be secure!**

- **Additional Configuration**

Laravel needs almost no other configuration out of the box. You are free to get started developing! However, you may wish to review the config/app.php file and its documentation. It contains several options such as timezone and locale that you may wish to change according to your application.

You may also want to configure a few additional components of Laravel, such as:

- Cache
- Database
- Session

- **An Introduction To Composer**

There are two separate elements to dependency management with Composer. The first is Composer itself which is a command line tool for grabbing and managing dependencies. The second is Packagist - the main composer repository. This is where the packages you may want to use are stored.

When using Composer the heart of your dependency management will be a single JSON file named composer.json. The nuts and bolts look something like this:

```
{
  "name": "danielpataki/my_project",
  "description": "My New Project",
  "authors": [
    {
      "name": "Daniel Pataki",
      "email": "mail@mymail.com"
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
}  
],  
"require": {  
    "monolog/monolog": "1.12.0"  
}  
}
```

- **Installing Composer**

Composer is available on any system you'd like. On Windows you should use the Composer Setup file which can be found on the Getting Started page. On Linux based systems, including OSX, you can use the following command to install it locally:

```
curl -sS https://getcomposer.org/installer | php
```

Run the above command from your project directory. This will give you a composer.phar file which you can use in that directory. I prefer to install the command globally so I can use it in any folder I like. To make this happen, issue the following command as well:

```
mv composer.phar /usr/local/bin/composer
```

There are two things which could go wrong. If you need admin privileges here you will need to run this command with sudo.

```
mv composer.phar /usr/local/bin/composer
```

- **Finding and installing new packages**

Using the search on <http://packagist.org>, you can find packages to add common features, such as image manipulation or PDF generation to your application. Indicators of good packages beyond the number of downloads and the number of stars on GitHub are the quality of documentation, the test coverage, and the overall project activity. Before adding a new package, you can also browse the different versions of a package and its dependencies on Packagist:

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

laravel/framework - Packagist

packagist.org/packages/laravel/framework

Create a new account | Login

Packagist
The PHP package archivist.

Submit Package

Search packages...

laravel/framework framework laravel

The Laravel Framework.

Maintainer: [taylorotwell](#)

Canonical: <http://github.com/laravel/framework>

Source: <https://github.com/laravel/framework/tree/master>

Issues: <https://github.com/laravel/framework/issues>

Overall: 482816 installs
This month: 25539 installs
Today: 664 installs

dev-master / 4.1.x-dev reference: 12ae2aa 2013-11-08 23:01 UTC MIT

require: "laravel/framework": "4.1.*@dev"

Author

- Taylor Otwell <taylorotwell@gmail.com>

Requires

- ircmaxell/password-compat: 1.0.*
- classpreloader/classpreloader: 1.0.*
- redis/redis: 0.8.*
- php: >=5.3.0
- patchwork/utf8: 1.1.*
- nesbot/carbon: 1.*
- swiftmailer/swiftmailer: 5.0.*
- monolog/monolog: 1.6.*
- clintfw/bons: 1.0.*
- phpseclib/phpseclib: 0.3.*
- jeremiasma/superclosure: 1.0.*
- symfony/browser-kit: 2.4.*
- symfony/console: 2.4.*
- symfony/css-selector: 2.4.*
- symfony/debug: 2.4.*
- symfony/dom-crawler: 2.4.*
- symfony/finder: 2.4.*
- symfony/http-foundation: 2.4.*
- symfony/http-kernel: 2.4.*
- symfony/process: 2.4.*
- symfony/translation: 2.4.*
- stack/builder: 1.0.*
- filp/whoops: 1.0.9

Requires (Dev)

- phpunit/phpunit: 3.7.*
- mockery/mockery: 0.8.0
- aws/aws-sdk-php: 2.4.*
- iron-io/iron_mq: 1.4.*
- pda/pheanstalk: 2.1.*

Suggests

- doctrine/dbal: Allow renaming columns and dropping SQLite columns.

In development it is fine to use a dev-master branch, but in production, it is better to stick with a precise version number to avoid potential compatibility issues.

To install a package, open composer.json in a text editor and insert its name and the desired version in the require object:

```
"require": {  
    "laravel/laravel": "v4.1.*",  
    "intervention/image": "dev-master"  
}
```

Since it is a JSON file, you need to be careful not to leave any trailing commas on the last line. To check if there are any errors in your composer.json file, you can use the composer validate command. Then, simply run composer install and Composer will fetch the package and its...

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

UNIT -3

Artisan

- **Artisan Command Line Tool**

Artisan is the command-line interface included with Laravel. It provides a number of helpful commands that can assist you while you build your application. To view a list of all available Artisan commands, you may use the list command:

```
php artisan list
```

Every command also includes a "help" screen which displays and describes the command's available arguments and options. To view a help screen, simply precede the name of the command with help:

```
php artisan help migrate
```

- **Writing Commands**

In addition to the commands provided with Artisan, you may also build your own custom commands. Commands are typically stored in the

```
app/Console/Commands directory;
```

however, you are free to choose your own storage location as long as your commands can be loaded by Composer.

- **Generating Commands**

To create a new command, use the `make:command` Artisan command. This command will create a new command class in the `app/Console/Commands` directory. Don't worry if this directory does not exist in your application, since it will be created the first time you run the `make:command` Artisan command. The generated command will include the default set of properties and methods that are present on all commands:

```
php artisan make:command SendEmails
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Artisan Migration**

Introduction

Migrations are a type of version control for your database. They allow a team to modify the database schema and stay up to date on the current schema state. Migrations are typically paired with the Schema Builder to easily manage your application's schema.

- **Database Creation**

Laravel has made processing with database very easy. Laravel currently supports following databases:

MySQL , SQLite, SQL Server

The query to the database can be fired using raw SQL, the fluent query builder. To understand the all CRUD (Create, Read, Update, Delete) operations with Laravel, we will use simple student management system. Configure the database in config/database.php file and create the college database with structure in MySQL as shown in the following table.

Database: College

Table: student

Column Name	Column Datatype	Extra
Id	int(11)	Primary key Auto increment
Name	varchar(25)	

We will see how to add, delete, update and retrieve records from database using Laravel in student table.

- **Insert Records**

We can insert the record using the DB facade with insert method. The syntax of insert method is as shown in the following table.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- Syntax - `bool insert(string $query, array $bindings = array())`
- Parameters - `$query(string)` – query to execute in database

`$bindings(array)` – values to bind with queries

- Returns - `bool`
- Description - Run an insert statement against the database.

Example

Step 1: Execute the below command to create a controller called

`StudInsertController`

`php artisan make:controller StudInsertController --plain`

Step 2: After successful execution, you will receive the following

output : `Controller created successfully.`

Step 3: Write the following code to file

`app/Http/Controllers/StudInsertController.php`

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use DB;
use App\Http\Requests;
use App\Http\Controllers\Controller;
class StudInsertController extends Controller
```

```
{
    public function insertform(){
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
return view('stud_create');  
  
}  
  
public function insert(Request $request){  
    $name = $request->input('stud_name');  
    DB::insert('insert into student (name) values(?)',[$name]);  
    echo "Record inserted successfully.<br/>";  
    echo '<a href="/insert">Click Here</a> to go back.';  
}  
}
```

Step 4 – Create a view file called **resources/views/stud_create.php** and copy the following code in that file.

resources/views/stud_create.php

```
<html>  
<head>  
<title>Student Management | Add</title>  
</head>  
<body>  
<form action = "/create" method = "post">  
<input type = "hidden" name = "_token" value = "<?php echo csrf_token(); ?>">  
<table> <tr> <td>Name</td> <td>  
<input type='text' name='stud_name' /></td> </tr>
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

<tr>

<td colspan = '2'>

<input type = 'submit' value = "Add student"/>

</td>

</tr>

</table>

</form>

</body>

</html>

Step 5 – Add the following lines in

app/Http/routes.php

```
Route::get('insert','StudInsertController@insertform');
```

```
Route::post('create','StudInsertController@insert');
```

Step 6 – Visit the

following URL to insert record in database.

<http://localhost:8000/insert>

- **Retrieve Records**

After configuring the database, we can retrieve the records.

Example

Step 1 – Execute the below command to create a controller called

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

StudViewController.

php artisan make:controller StudViewController --plain

Step 2 – After

successful execution, you will receive the following output –

Controller created successfully.

Step 3 – Copy the following code to file

app/Http/Controllers/StudViewController.php

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use DB;
use App\Http\Requests;
use App\Http\Controllers\Controller;
class StudViewController extends Controller
{
    public function index()
    {
        $users = DB::select('select * from student');
        return view('stud_view',['users'=>$users]);
    }
}
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Step 4 – Create a view file called `resources/views/stud_view.blade.php` and copy the following code in that file.

`resources/views/ stud_view.blade.php`

```
<html>
<head>
<title>View Student Records</title>
</head>
<body>
<table border = 1>
<tr>
<td>ID</td>
<td>Name</td>
</tr>
@foreach ($users as $user)
<tr>
<td>{{ $user->id }} </td>
<td>{{ $user->name }} </td>
</tr> @endforeach
</table>
</body>
</html>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Step 5 – Add the following lines in `app/Http/routes.php`.

`app/Http/routes.php`

```
Route::get('view-records','StudViewController@index');
```

Step 6 – Visit the following URL to see records from database.

`http://localhost:8000/view-records`

- **Update Records**

Step 1 – Execute the below command to create a controller called

StudViewController.

```
php artisan make:controller StudUpdateController --plain
```

Step 2 – After successful execution, you will receive the following

output – Controller created successfully.

Step 3 – Copy the following code to file `app/Http/Controllers/ StudUpdateController.php`

`app/Http/Controllers/StudUpdateController.php`

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use DB; use App\Http\Requests;
use App\Http\Controllers\Controller;
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
class StudUpdateController extends Controller {  
  
    public function index(){  
  
        $users = DB::select('select * from student');  
  
        return view('stud_edit_view',['users'=>$users]);  
  
    }  
  
    public function show($id) {  
  
        $users = DB::select('select * from student where id = ?',[$id]);  
  
        return view('stud_update',['users'=>$users]);  
  
    }  
  
    public function edit(Request $request,$id) {  
  
        $name = $request->input('stud_name');  
  
        DB::update('update student set name = ? where id = ?',[$name,$id]);  
  
        echo "Record updated successfully.<br/>";  
  
        echo '<a href = "/edit-records">Click Here</a> to go back.';  
  
    }  
  
}
```

Step 4 – Create a view file called **resources/views/stud_edit_view.blade.php** and copy the following code in that file.

resources/views/stud_edit_view.blade.php

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<html>
<head>
<title>View Student Records</title>
</head>
<body>
<table border = "1">
<tr>
<td>ID</td>
<td>Name</td>
<td>Edit</td>
</tr>
@foreach ($users as $user)
<tr> <td> {{ $user->id }} </td>
<td> {{ $user->name }} </td>
<td> <a href = 'edit/{{ $user->id }}'>Edit</a></td></tr>
@endforeach
</table>
</body>
</html>
```

Step 5 – Create another view file called **resources/views/stud_update.php** and copy the following code in that file.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

resources/views/stud_update.php

```
<html>
<head>
<title>Student Management | Edit</title>
</head>
<body>
<form action = "/edit/<?php echo $users[0]->id; ?>" method = "post">
<input type = "hidden" name = "_token" value = "<?php echo csrf_token(); ?>">
<table>
<tr> <td>Name</td>
<td> <input type = 'text' name = 'stud_name' value = '<?php echo $users[0]->name; ?>' /> </td> </tr>
<tr> <td colspan = '2'> <input type = 'submit' value = "Update student" /> </td> </tr>
</table>
</form>
</body>
</html>
```

Step 6 – Add the following lines in **app/Http/routes.php**.

app/Http/routes.php.

```
Route::get('edit-records','StudUpdateController@index');
```

```
Route::get('edit/{id}','StudUpdateController@show');
```

```
Route::post('edit/{id}','StudUpdateController@edit');
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Step 7 – Visit the following URL to update records in database.

<http://localhost:8000/edit-records>

- **Delete Records**

Step 1 – Execute the below command to create a controller called

StudDeleteController.

```
php artisan make:controller StudDeleteController --plain
```

Step 2 – After successful execution, you will receive the following output –

Controller created successfully.

Step 3 – Copy the following code to file

```
app/Http/Controllers/StudDeleteController.php
```

```
app/Http/Controllers/StudDeleteController.php
```

```
<?php
```

```
namespace App\Http\Controllers;
```

```
use Illuminate\Http\Request;
```

```
use DB;
```

```
use App\Http\Requests;
```

```
use App\Http\Controllers\Controller;
```

```
class StudDeleteController extends Controller {
```

```
    public function index(){
```

```
        $users = DB::select('select * from student');
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
return view('stud_delete_view',[ 'users'=>$users]);
```

```
}
```

```
public function destroy($id) {
```

```
DB::delete('delete from student where id = ?',[$id]);
```

```
echo "Record deleted successfully.<br/>";
```

```
echo '<a href="/delete-records">Click Here</a> to go back.';
```

```
}
```

```
}
```

Step 4 – Create a view file called

resources/views/stud_delete_view.blade.php and copy the following

code in that file.

resources/views/stud_delete_view.blade.php

```
<html>
```

```
<head>
```

```
<title>View Student Records</title>
```

```
</head>
```

```
<body>
```

```
<table border = "1">
```

```
<tr>
```

```
<td>ID</td>
```

```
<td>Name</td>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<td>Edit</td>
```

```
</tr>
```

```
@foreach ($users as $user)
```

```
<tr> <td> {{ $user->id }} </td>
```

```
<td> {{ $user->name }} </td>
```

```
<td> <a href = 'delete/{{ $user->id }}>Delete</a></td> </tr>
```

```
@endforeach
```

```
</table>
```

```
</body>
```

```
</html>
```

Step 5 – Add the following lines in **app/Http/routes.php**.

app/Http/routes.php

```
Route::get('delete-records','StudDeleteController@index');
```

```
Route::get('delete/{id}','StudDeleteController@destroy');
```

- **Artisan Migration**

Laravel migrations provide mechanisms for creating and modifying database tables. You do not have to worry about the specific SQL syntax for the database engine that you are creating tables for. Migrations also allow you to roll back the most recent changes that you made to a database.

- **Migration Structure**

A migration class contains two methods: up and down. The up method is used to add new tables, columns, or indexes to your database, while the down method should simply reverse the operations performed by the up method.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Within both of these methods you may use the Laravel schema builder to expressively create and modify tables.

For example, this migration example creates a users table:

```
<?php
use Illuminate\Support\Facades\Schema;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Database\Migrations\Migration;
class CreateUsersTable extends Migration {
public function up() {
    Schema::create('users', function (Blueprint $table) {
        $table->increments('id');
        $table->string('name');
    });
}
public function down()
{
    Schema::drop('users');
}
}
```

- **Creating Migrations**

To create a migration, you may use the `make:migration` command on the Artisan CLI:

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
php artisan make:migration create_users_table --create=users
```

The migration will be placed in your database/migrations folder, and will contain a timestamp which allows the framework to determine the order of the migrations.

The --table and --create options may also be used to indicate the name of the table, and whether the migration will be creating a new table:

```
php artisan make:migration add_city_to_users_table
```

```
php artisan make:migration create_users_table --create=users
```

Add the following code in **AppServiceProvider.php**

```
root/app/Providers/AppServiceProvider.php
```

The code block

```
use Illuminate\Support\Facades\Schema;
```

```
function boot()  
{  
    Schema::defaultStringLength(191);  
}
```

• Database Seeding

Laravel includes a simple method of seeding your database with test data using seed classes. All seed classes are stored in the database/seeds directory.

Writing Seeders

To generate a seeder, execute the make:seeder Artisan command. All seeders generated by the framework will be placed in the database/seeds directory:

```
php artisan make:seeder UsersTableSeeder
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

A seeder class only contains one method by default: run. This method is called when the db:seed Artisan command is executed. Within the run method, you may insert data into your database however you wish.

As an example, let's modify the default DatabaseSeeder class and add a database insert statement to the run method:

```
public function run()
{
    DB::table('users')->insert([
        'name' => str_random(10),
        'email' => str_random(10).'@gmail.com',
        'password' => bcrypt('secret'),
    ]);
}
```

Now run this command on cmd : php artisan db:seed

Using Model Factories

For example, let's create 50 users and attach a relationship to each user:

```
public function run() {
    factory(App\User::class, 50)->create();
}
```

Now run this command on cmd : php artisan db:seed

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

UNIT -3

Routing in Laravel

- **Basic Routing**

Basic routing is meant to route your request to an appropriate controller. The routes of the application can be defined in app/Http/routes.php file. Here is the general route syntax for each of the possible request.

Example

```
app/Http/routes.php
```

```
<?php
```

```
Route::get('/', function () {  
    return view('welcome');  
});
```

- **Routing Parameters**

Often in the application, we intend to capture the parameters passed with the URL. To do this, we need to modify the code in routes.php file accordingly. There are two ways by which we can capture the parameters passed with the URL.

Required Parameters

Optional Parameters

Example

routes.php

```
<?php
```

```
// First Route method – Root URL will match this method
```

```
Route::get('/', function () {  
    return view('welcome');  
});
```

```
// Second Route method – Root URL with ID will match this method
```

```
Route::get('ID/{id}',function($id){  
    echo 'ID: '.$id;
```

If we execute the below URL, it will execute the 2nd method and the argument/parameter ID will be passed to the variable \$id.

http://localhost:8000/ID/5

```
// Third Route method – Root URL with or without name will match this method
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
Route::get('/user/{name?}',function($name = 'Virat Kohli'){        echo "Name: ".$name;
});
```

If we execute the below URL, it will execute the 3rd method and the optional argument/parameter name will be passed to the variable \$name. The default name will be used that we have passed in the function as 'Virat Kohli'

http://localhost:8000/user/Virat

- **Named Routing**

Named routes allow the convenient generation of URLs or redirects for specific routes. You may specify a name for a route by chaining the name method onto the route definition:

```
Route::get('user/profile', function () {
    //
})->name('profile');
```

You may also specify route names for controller actions:

```
Route::get('user/profile', 'UserController@showProfile')-> name('profile');
```

Once you have assigned a name to a given route, you may use the route's name when generating URLs or redirects via the global route function:

```
// Generating URLs...
$url = route('profile');
// Generating Redirects...
return redirect()->route('profile');
```

- **Middleware**

As the name suggest, Middleware acts as a middle man between request and response. It is a type of filtering mechanism. For example, Laravel includes a middleware that verifies whether user of the application is authenticated or not. If the user is authenticated, he will be redirected to the home page otherwise, he will be redirected to the login page.

Middleware can be created by executing the following command –
php artisan make:middleware <middleware-name>

Replace the <middleware-name> with the name of your middleware. The middleware that you create can be seen at **app/Http/Middleware** directory.

```
Route::middleware(['first', 'second']->group(function () {
    Route::get('/', function () {
        // Uses first & second Middleware
    });
});
```

```
Route::get('user/profile', function () {
    // Uses first & second Middleware
});
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
});  
});
```

- **Route Groups**

Route groups allow you to share route attributes, such as middleware or namespaces, across a large number of routes without needing to define those attributes on each individual route. Shared attributes are specified in an array format as the first parameter to the `Route::group` method.

```
Route::group(['middleware' => 'auth'], function () {  
    Route::get('/', function () {  
        // Uses Auth Middleware  
    });  
    Route::get('user/profile', function () {  
        // Uses Auth Middleware  
    });  
});
```

- **Sub-Domain Routing**

Route groups may also be used to handle sub-domain routing. Sub-domains may be assigned route parameters just like route URIs, allowing you to capture a portion of the sub-domain for usage in your route or controller. The sub-domain may be specified by calling the `domain` method before defining the group:

```
Route::domain('{account}.myapp.com')->group(function () {  
    Route::get('user/{id}', function ($account, $id) {  
        //  
    });  
});
```

- **Route Prefixing**

The `prefix` method may be used to prefix each route in the group with a given URI. For example, you may want to prefix all route URIs within the group with `admin`:

```
Route::prefix('admin')->group(function () {  
    Route::get('users', function () {  
        // Matches The "/admin/users" URL  
    });  
});
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **Route Model Binding**

Laravel model binding provides a convenient way to inject class instances into your routes. For example, instead of injecting a user's ID, you can inject the entire User class instance that matches the given ID.

```
Route::get('users', function() {  
    return User::all();  
});  
Route::get('users/{id}', function($id) {  
    return User::find($id);  
});
```

- **Throwing 404 Errors**

Some exceptions describe HTTP error codes like 404, 500 etc. To generate such response anywhere in an application, you can use abort() method as follows.

```
abort(404)
```

Laravel makes it very easy for us to use the custom error pages for each separate error codes. For example, if you want to design custom page for error code 404, you can create a view at resources/views/errors/404.blade.php. Same way, if you want to design error page for error code 500, it should be stored at resources/views/errors/500.blade.php.

Example

Step 1: Add the following lines in app/Http/routes.php.

```
app/Http/routes.php  
Route::get('/error',function(){  
    abort(404);  
});
```

Step 2: Create a view file called resources/views/errors/404.blade.php and copy the following code in that file.

```
resources/views/errors/404.blade.php
```

```
<html>  
<head>  
    <title>404</title>  
</head>  
<body>  
    <div>404 Error</div>  
</body>  
</html>
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

UNIT -4

Blade Template

- **Template Inheritance**

Laravel is very powerful framework in PHP. Laravel provides a lot of library to create complex functionality in easy manner. Blade is one of them powerful **templating engine** with Laravel, which will generate theme layout in HTML format. All blade files uses `.blade.php` file extension and stored in `resources/views` directory.

Blade engines provides fast views without any overhead due to cached views until they are modified. Blade engine are using template inheritance and sections.

Laravel layout files structure

- **header.blade.php** : This file will contains html template header part.
- **sidebar.blade.php** : This file will contains html template sidebar part.
- **footer.blade.php** : This file will contains html template footer part.
- **master.blade.php** : This file will use to include above partial files and create master layout template which will render on each laravel request.

You want to include one Blade template within another.

You combined some common HTML code into a single template and want to pull in the template within another template.

Use the `@include` statement.

```
<html>
<body>
    @include('common-header')
    <h1>Page One</h1>
    <p>This is a demo page</p>
</body>
</html>
```

- **Master Layout**

Defining A Layout

Two of the primary benefits of using Blade are template inheritance and sections. To get started, let's take a look at a simple example. First, we will examine a "master" page layout. Since most web applications maintain the same general layout across various pages, it's convenient to define this layout as a single Blade view:

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<html>
<head>
  <title>App Name - @yield('title')</title>
</head>
<body>
  @section('sidebar')
    This is the master sidebar.
  @show
  <div class="container">
    @yield('content')
  </div>
</body>
</html>
```

- **Template Inheritance**

As you can see, this file contains typical HTML mark-up. However, take note of the @section and @yield directives.

The @section directive, as the name implies, defines a section of content.

The @yield directive is used to display the contents of a given section.

- **Extending the Master Layout**

Extending A Layout

When defining a child view, use the Blade @extends directive to specify which layout the child view should "inherit". Views which extend a Blade layout may inject content into the layout's sections using @section directives. Remember, as seen in the example above, the contents of these sections will be displayed in the layout using @yield:

```
@extends('layouts.app')
@section('title', 'Page Title')
@section('sidebar')
  @parent
  <p>This is appended to the master sidebar.</p>
@endsection
```

```
@section('content')
  <p>This is my body content.</p>
@endsection
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

In this example, the sidebar section is utilizing the @parent directive to append (rather than overwriting) content to the layout's sidebar. The @parent directive will be replaced by the content of the layout when the view is rendered.

Blade views may be returned from routes using the global view helper:

```
Route::get('blade', function ()  
{  
    return view('child');  
});
```

Example Coding

Controller file

```
class PostController extends Controller
```

```
{  
    public function about(){  
        $title = 'About Us';  
        $content = 'This is an about us page';  
        return view('aboutus',compact('title','content'));  
    }  
    public function contact(){  
        $title = 'Contact Us';  
        $content = 'This is an contact us page';  
        return view('contactus',compact('title','content'));  
    }  
}
```

Master file

```
<html>  
<head> <title>@yield('title')</title> </head>  
<body>  
<table>  
    <tr> <td> <h1> @yield('title') </h1> </td> </tr>  
    <tr> <td> @yield('content') </td> </tr>  
</table>  
</body>  
</html>
```

About us file

```
@extends('master')  
@section('title')
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
{{ $title }}  
@stop  
@section('content')  
{{ $content }}  
@stop
```

Contact us file

```
@extends('master')  
@section('title')  
{{ $title }}  
@stop  
@section('content')  
{{ $content }}  
@stop
```

Route file

```
Route::get('aboutus', 'PostController@about');  
Route::get('contactus', 'PostController@contact');
```

- **Displaying a Variable in a Blade Template**

- You want to echo the contents of a variable on a web page.
- Instead of using `<?php echo $variable; ?>`, Use the curly brace syntax to display the variable.

```
Route::get('greeting', function () {  
    return view('welcome', ['name' => 'Samantha']);  
});
```

You may display the contents of the name variable like so:

Hello, {{ \$name }}.

You may also echo the results of any PHP function. In fact, you can put any PHP code you wish inside of a Blade echo statement:

The current UNIX timestamp is **{{ time() }}**.

- **Control Structures**

Blade also provides convenient shortcuts for common PHP control structures, such as conditional statements and loops.

- **If Statements**

You may construct if statements using the `@if`, `@elseif`, `@else`, and `@endif` directives. These directives function identically to their PHP counterparts:

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
@if (count($records) === 1)
    I have one record!
@elseif (count($records) > 1)
    I have multiple records!
@else
    I don't have any records!
@endif
```

Example :

```
@if(Auth::check())
    @extends("template/index")
@else
    @extends("template/login")
@endif
```

• Switch Statements

Switch statements can be constructed using the `@switch`, `@case`, `@break`, `@default` and `@endswitch` directives:

```
@switch($i)
    @case(1)
        First case...
    @break
    @case(2)
        Second case...
    @break
    @default
        Default case...
@endswitch
```

• Loops

In addition to conditional statements, Blade provides simple directives for working with PHP's loop structures.

```
@for ($i = 0; $i < 10; $i++)
    The current value is {{ $i }}
@endfor
@foreach ($users as $user)
    <p>This is user {{ $user->id }}</p>
@endforeach
@forelse ($users as $user)
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
<li>{{ $user->name }}</li>
```

```
@empty
```

```
<p>No users</p>
```

```
@endforelse
```

```
@while (true)
```

```
<p>I'm looping forever.</p>
```

```
@endwhile
```

• Executing PHP functions in blade

There are many useful PHP functions which an developer might use from time to time in their application. To use PHP in a blade there are two methods:

- Traditional method
- Blade Template method
- **Traditional Method**

You can simply insert your PHP code within php tags and it will work. For example:

```
<?php echo date(); ?>
```

- **Blade Template Method**

above PHP code can be passed as following way:

```
@php
```

```
echo date('l jS \of F Y h:i:s A');
```

```
@endphp
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

UNIT -4

SQL Interaction

- **Introduction**

Laravel makes connecting to a database and managing data in it extremely simple and secure. Instead of requiring the developer to write complicated SQL queries it provides a convenient way of telling the application how you want to manipulate the data and Laravel would automatically translate those commands into SQL queries behind the scenes.

- **Running Queries**

Once you have configured your database connection, you may run queries using the DB class.

Running A Select Query

```
$results = DB::select('select * from users where id = ?', array(1));
```

The select method will always return an array of results.

Running An Insert Statement

```
DB::insert('insert into users (id, name) values (?, ?)', array(1, 'Dayle'));
```

Running An Update Statement

```
DB::update('update users set votes = 100 where name = ?', array('John'));
```

Running A Delete Statement

```
DB::delete('delete from users');
```

- **Database Transactions**

To run a set of operations within a database transaction, you may use the transaction method:

```
DB::transaction(function()  
{  
    DB::table('users')->update(array('votes' => 1));  
    DB::table('posts')->delete();  
});
```

Sometimes you may need to begin a transaction yourself:

```
DB::beginTransaction();
```

You can rollback a transaction via the rollback method:

```
DB::rollback();
```

Lastly, you can commit a transaction via the commit method:

```
DB::commit();
```


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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

UNIT -5

Eloquent ORM

- **Eloquent ORM Models**

A Laravel Model makes it very easy to store, read (retrieve), update and delete (CRUD) a resource in a Laravel application.

Each database table has a corresponding "Model" which is used to interact with that table. Models allow you to query for data in your tables, as well as insert new records into the table.

- **Naming conventions**

The law of the gods state that model names should be singular and table names should be plural. Laravel automatically pluralizes the model name to get the table name.

Eloquent Models provide the above as the default implementation. You can explicitly specify a table name if you want to. We will now create a model for the categories table. The name of the model will be category.

php artisan make:model Category

php artisan make:model Category creates a model named Category in /app/Category.php

Open the newly created model in /app/Category.php

You will get the following

```
<?php
namespace App;
use Illuminate\Database\Eloquent\Model;
class Category extends Model {
    //
}
```

use Illuminate\Database\Eloquent\Model; imports the Eloquent model class namespace
class Category extends Model defines a model Category that extends Model

- **Table name and primary key**

By default, the plural form of the model name is used as the table name and the primary key field name is assumed to be id. This section shows you how you can explicitly define both the table and primary key field names Add the following lines to Category model

```
protected $primaryKey = 'id';
protected $table = 'categories';
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

protected \$primaryKey = 'id'; defines the primary key field name
protected \$table = 'categories'; defines the name of the table

- **Record timestamps**

By default, Laravel assumes you have added the following fields to all of your database tables
created_at and **updated_at**

These fields are updated whenever you create a new record or update an existing record. If you do not want to use these timestamp fields in your database tables, you can set the following property to turn them off.

```
public $timestamps = false;  
HERE, public $timestamps = false;  
Eloquent ORM model not to consider created_at and updated_at fields.
```

- **Basic Operations**

Retrieving All Records

```
$users = User::all();
```

Retrieving A Record By Primary Key

```
$user = User::find(1); var_dump($user->name);
```

Querying Using Eloquent Models

```
$users = User::where('votes', '>', 100)->take(10)->get();  
foreach ($users as $user) {  
    var_dump($user->name);  
}
```

Insert, Update, Delete

To create a new record in the database from a model, simply create a new model instance and call the save method.

Saving A New Model

```
$user = new User; $user->name = 'John';  
$user->save();
```

You may also use the create method to save a new model in a single line.

```
$user = User::create(['name' => 'John']);
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

Updating A Retrieved Model

To update a model, you may retrieve it, change an attribute, and use the save method:

```
$user = User::find(1);  
$user->email = 'john@example.com';  
$user->save();
```

You may also run updates as queries against a set of models:

```
$affectedRows = User::where('votes', '>', 100)->update(['status' => 2]);
```

Deleting An Existing Model

To delete a model, simply call the delete method on the instance:

```
$user = User::find(1);  
$user->delete();
```

Deleting An Existing Model By Key

```
User::destroy(1);  
User::destroy([1, 2, 3]);  
User::destroy(1, 2, 3);
```

you may also run a delete query on a set of models:

```
$affectedRows = User::where('votes', '>', 100)->delete();
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

UNIT -5

Validation

- **Defining The Routes**

First, let's assume we have the following routes defined in our routes/web.php file:

```
Route::get('post/create', 'PostController@create');
```

```
Route::post('post', 'PostController@store');
```

The GET route will display a form for the user to create a new blog post, while the POST route will store the new blog post in the database.

- **Creating The Controller**

Next, let's take a look at a simple controller that handles these routes. We'll leave the store method empty for now:

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use App\Http\Controllers\Controller;
class PostController extends Controller
{
    public function create()
    {
        return view('post.create');
    }
    public function store(Request $request)
    {
        // Validate and store the blog post...
    }
}
```

- **Writing The Validation Logic**

Now fill in our store method with the logic to validate the new blog post. To do this, we will use the validate method provided by the Illuminate\Http\Request object.

```
public function store(Request $request)
{
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
$validatedData = $request->validate([  
    'title' => 'required|unique:posts|max:255',  
    'body' => 'required', ]); // The blog post is valid...  
}
```

We simply pass the desired validation rules into the validate method. Again, if the validation fails, the proper response will automatically be generated.

Stopping On First Validation Failure

Sometimes you may wish to stop running validation rules on an attribute after the first validation failure. To do so, assign the bail rule to the attribute:

```
$request->validate([  
    'title' => 'bail|required|unique:posts|max:255',  
    'body' => 'required',  
]);
```

In this example, if the unique rule on the title attribute fails, the max rule will not be checked. Rules will be validated in the order they are assigned.

- **Customizing The Error Messages**

You may customize the error messages used by the form request by overriding the messages method. This method should return an array of attribute / rule pairs and their corresponding error messages:

```
public function messages()  
{  
    return [  
        'title.required' => 'A title is required',  
        'body.required' => 'A message is required',  
    ];  
}
```

- **Manually Creating Validators**

If you do not want to use the validate method on the request, you may create a validator instance manually using the Validator.

```
<?php  
namespace App\Http\Controllers;  
use Validator;  
class PostController extends Controller  
{
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
public function store(Request $request)
{
    $validator = Validator::make($request->all(), [
        'title' => 'required|unique:posts|max:255',
        'body' => 'required',
    ]);
    if ($validator->fails())
    {
        return redirect('post/create')
            ->withErrors($validator)
            ->withInput();
    }
}
```

• Available Validation Rules in Laravel

- **accepted**
The field under validation must be *yes*, *on*, or *1*. This is useful for validating "Terms of Service" acceptance.
- **after:date**
The field under validation must be a value after a given date. The dates will be passed into the PHP `strtotime` function.
- **alpha**
The field under validation must be entirely alphabetic characters.
- **alpha_dash**
The field under validation may have alpha-numeric characters, as well as dashes and underscores.
- **alpha_num**
The field under validation must be entirely alpha-numeric characters.
- **array**
The field under validation must be of type array.
- **before:date**
The field under validation must be a value preceding the given date. The dates will be passed into the PHP `strtotime` function.
- **between:min,max**
The field under validation must have a size between the given *min* and *max*. Strings, numerics, and files are evaluated in the same fashion as the size rule.
- **boolean**
The field under validation must be able to be cast as a boolean. Accepted input are `true`, `false`, `1`, `0`, `"1"` and `"0"`.

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

- **date**
The field under validation must be a valid date according to the strtotime PHP function.
- **digits:value**
The field under validation must be *numeric value*.
- **digits_between:min,max**
The field under validation must have a length between the given *min* and *max*.
- **email**
The field under validation must be formatted as an e-mail address.
- **exists:table,column**
The field under validation must exist on a given database table.
- **image**
The file under validation must be an image (jpeg, png, bmp, or gif)
- **in**
The field under validation must be included in the given list of values.
- **integer**
The field under validation must have an integer value.
- **max:value**
The field under validation must be less than or equal to a maximum *value*. Strings, numerics, and files are evaluated in the same fashion as the size rule.
- **min:value**
The field under validation must have a minimum *value*. Strings, numerics, and files are evaluated in the same fashion as the size rule.
- **not_in**
The field under validation must not be included in the given list of values.
- **numeric**
The field under validation must have a numeric value.
- **regex:pattern**
The field under validation must match the given regular expression.
- **required**
The field under validation must be present in the input data.
- **string:value**
The field under validation must be a string type.

Example of Validation

```
$this->validate($request,[  
    'name' => 'required|min:5|max:35',  
    'email' => 'required|email|unique:users',  
    'mobilenos' => 'required|numeric',  
    'password' => 'required|min:3|max:20',
```

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

(AFFILIATED TO SAURASHTRA UNIVERSITY)



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

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

```
'confirm_password' => 'required|min:3|max:20|same:password',  
'details' => 'required'
```

```
],[  
  
'firstname.required' => 'The first name field is required.',  
'firstname.min' => 'The first name must be at least 5 characters.',  
'firstname.max' => 'The first name may not be greater than 35 characters.',  
'lastname.required' => 'The last name field is required.',  
'lastname.min' => 'The last name must be at least 5 characters.',  
'lastname.max' => 'The last name may not be greater than 35 characters.', ]];
```

