Commands:
In GitBash enter following commands
● Pwd→ present working directory (folder)
● Display the current location of CLI command line interface
● ls → List all files and folders available in current directory
● help → provides documentation on commands
● cd → change directory
eg:cd foldername
● cd subfoldername
● cd → navigate to home
● cd.. → navigate to parent folder
● cd - → going back to previous location
● cd ../.. → navigate to parent of parent folder
● mv → move directory
● ctrl + l → clear screen
● clear → clear screen
● mkdir foldername → make directory to create a new folder in current location
● rmdir foldername → removes empty folders
If the folder contains file then first remove file present in the folder and come back and remove folder when it is empty.
● rm -rf filename → to remove folder contains file
● touch → to create a new file . eg: touch test.txt
● echo python returns python
● echo → to enter data into file eg: echo test want to enter >> filename
● Eg: echo helloworld >> test.txt
● echo secondline>>test.txt
● cat test.txt → display data inside test file
● eg:hello world

secondline
If the file doesnt exist then create a file and copy data. If the file contain data it overrides
● cp file1 file2 : copy data from file1 to file2

Example:
cp hello.txt venkat.txt
Hello.txt contains “hai”
Venkat file doesnt exist when we execute it creates the file and copy data from hello.txt to venkat.txt
o/p:
Hello.txt:”hai”
venkat.txt:”hai”
If you do the same process then newly data override in venkat.txt
o/p:
Hello.txt:”hai”

venkat.txt:”hai”

● touch file1.txt file2.txt → creating multiple files at a time
● echo {1..110} → prints numbers from 1 to 110
● touch file{1..10}.txt → creating 10 files at a time
● echo {1..1000000} >>file.txt
● cat file.txt || more → to display all numbers between 1 to 1000000
● ls -l → memory used by files will display
● rm filename → to remove file
● rm file{6..10}.txt → to remove the files from 6 to 10
● rm *.txt → remove all files at a time

If the file doesnt exist then create a file and copy data. It adds data to the previous data

● cat file1>>file → create file1 and copy data of file1 into file2

Example:
cat hello.txt>>venkat.txt
Hello.txt contains “hai”
Venkat file doesnt exist when we execute it creates the file and copy data from hello.txt to venkat.txt
o/p:
Hello.txt:”hai”
venkat.txt:”hai”
If you do the same process then newly data adds to venkat.txt
o/p:
Hello.txt:”hai”
Venkat.txt:”hai”
“hai”

● Rename a file → mv oldfilename newfilename

● desktop→ git→ f1 (How to know whether f1 is empty or not?)
● In git folder enter ls f1 (it wont display nothing , it is empty)

Http:hypertext transfer protocol
https:hypertext transfer protocol secure

Web Designing with React

A web designer works on the appearance, layout, and, in some cases,content of a website. Appearance, for instance, relates to the colors, font,and images used.

Course Objective:

- HTML5
- CSS3
- CSS framework bootstrap
- JavaScript
- ES6 fundamentals
- JavaScript libraries React Js

Front End:
Front end developers build the visible parts of websites that users see and interact within their web browsers.

- *HTML*
- *CSS*
- *JavaScript*

frameworks like AngularJs, jQuery, and Node.js).

Backend:
Back end developers build the "under the hood" parts of websites that users don't interact with directly.

- PHP
- Ruby
- Python

WHAT SKILLS DO I NEED TO BECOME A FULL STACK DEVELOPER?

You'll typically see a mix of front and back end skills listed on full stack web developer job listings, including:

- HTML, CSS, JavaScript
- Ideally, one or more third-party library like ReactJS or Angular

- Programming languages and libraries like Ruby, PHP, Python
- Experience with databases
- Version control
- Knowledge of security concerns and best practices
- Ideally, some knowledge of web or visual design, plus user experience best practices

Download sublime:https://www.sublimetext.com/3

Download git: https://git-scm.com/download/win

Create acount on github

Version control system:

Version Control System (VCS) is a software that helps software developers to work together and maintain a complete history of their work.

Listed below are the functions of a VCS −

- *Allows developers to work simultaneously.*
- *Does not allow overwriting each other's changes.*
- *Maintains a history of every version.*

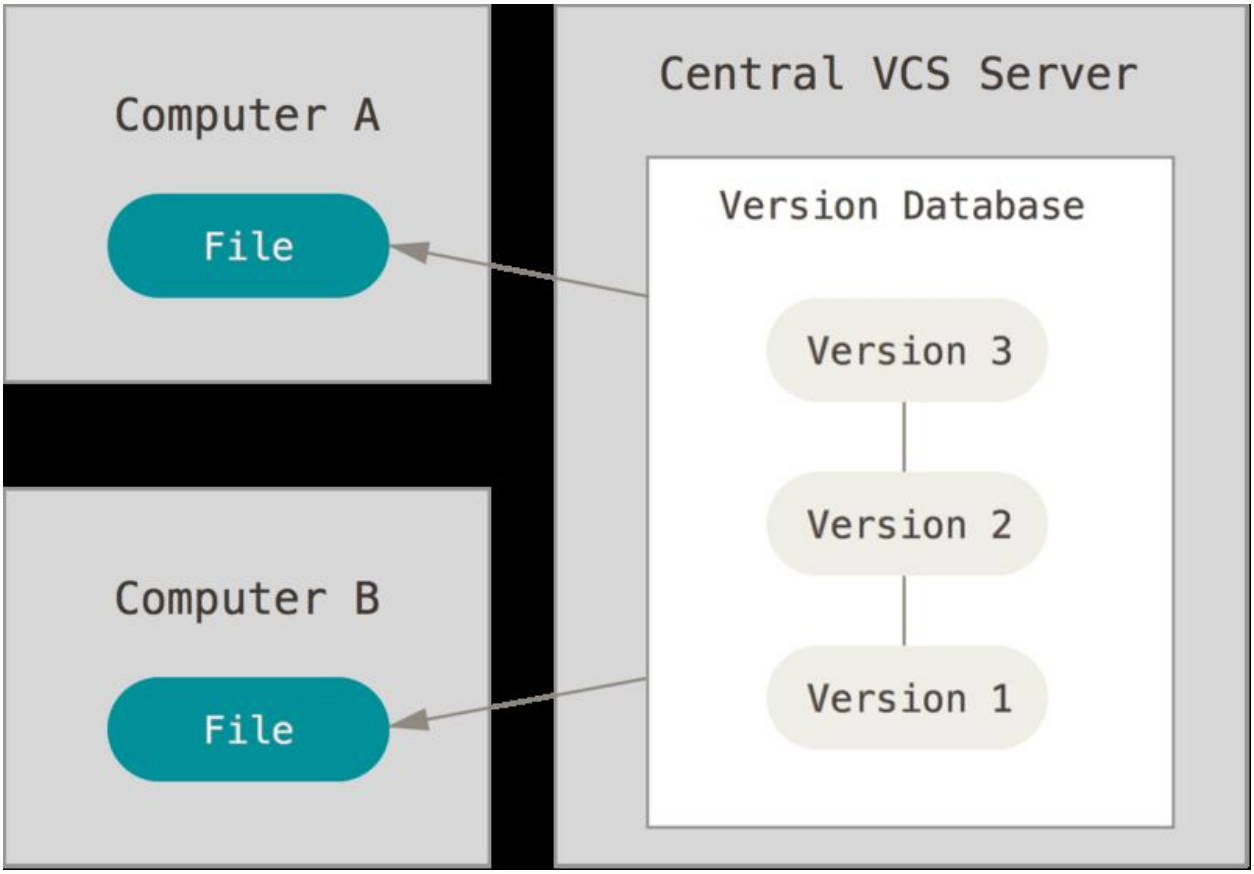
What is "version control", and why should you care? Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.

Following are the types of VCS −

- *Centralized version control system (CVCS).*
- *Distributed/Decentralized version control system (DVCS).*

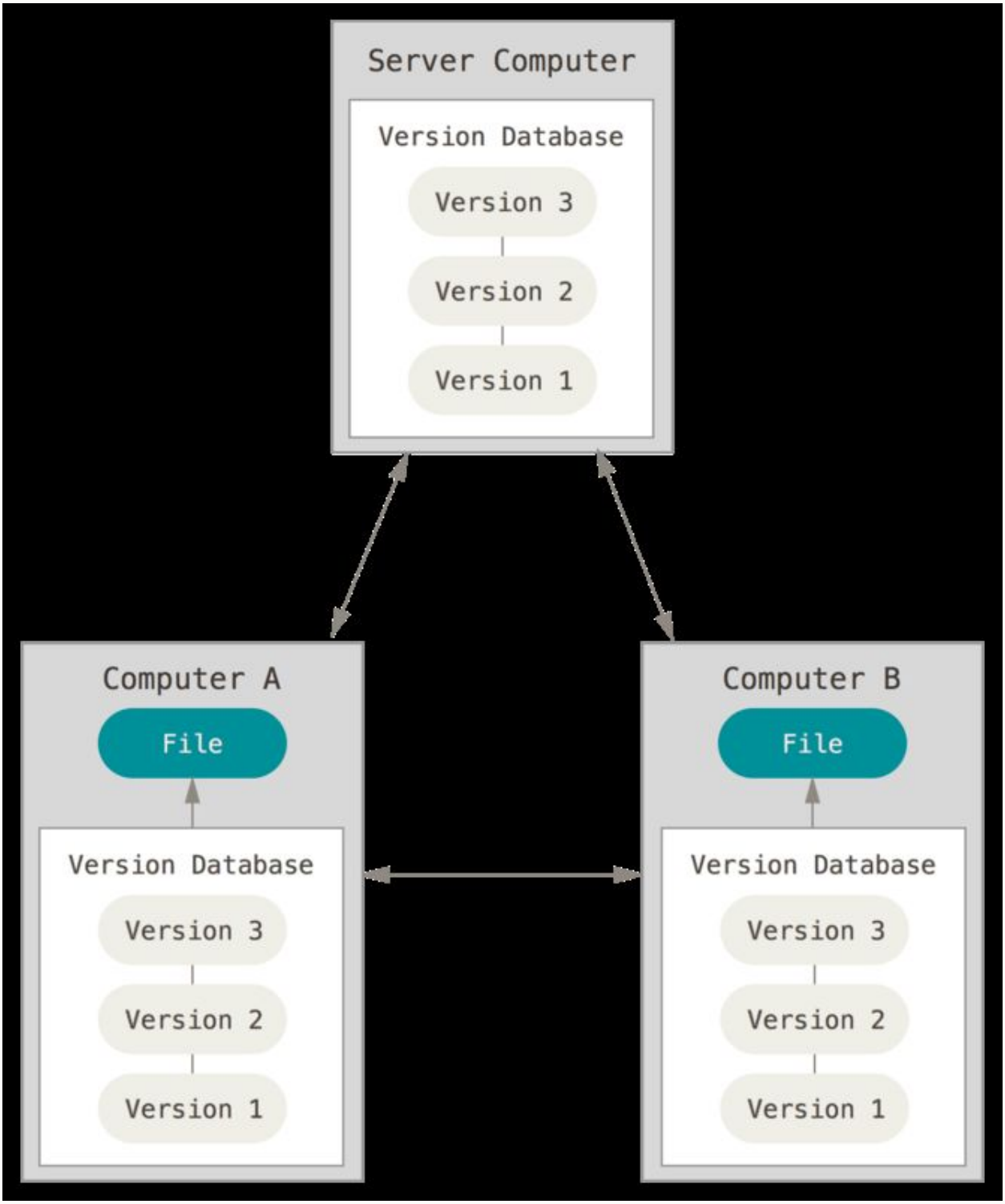
Centralized Version Control Systems

The next major issue that people encounter is that they need to collaborate with developers on other systems. To deal with this problem, Centralized Version Control Systems (CVCSs) were developed. These systems (such as CVS, Subversion, and Perforce) have a single server that contains all the versioned files, and a number of clients that check out files from that central place. For many years, this has been the standard for version control.



Distributed Version Control Systems

This is where Distributed Version Control Systems (DVCSs) step in. In a DVCS (such as Git, Mercurial, Bazaar or Darcs), clients don't just check out the latest snapshot of the files; rather, they fully mirror the repository, including its full history. Thus, if any server dies, and these systems were collaborating via that server, any of the client repositories can be copied back up to the server to restore it. Every clone is really a full backup of all the data.


Server Computer
Version Database
Version 3
Version 2
Version 1
Computer A
File
Version Database
Version 3
Version 2
Version 1
Computer B
File
Version Database
Version 3
Version 2
Version 1

git: *an open source, distributed version-control system*
GitHub: *a platform for hosting and collaborating on Git repositories*
commit: *a Git object, a snapshot of your entire repository compressed into a SHA*
branch: *a lightweight movable pointer to a commit*
clone: *a local version of a repository, including all commits and branches*

Advantages of Git

Free and open source

Fast and small

Implicit backup

Security

Easier branching

Git commands:

Git version

git config --global user.name "venkatasuryanarayana"
git config --global user.email "venkatasuryanarayana.n@apssdc.in"

git config user.name
git config user.email

cloning:

git clone paste the url

Staging

- takes a space-separated list of file names.

git add <file1><file2>...<filen>

example:
git add surya.txt hi.txt

All files:

- alternatively, the period `.` can be used in place of a list of files to tell Git to add the current directory (and all nested files)

git add .

git add --all (or) git add --a

- **I**f u need to remove(**unstage)**the staged files by using following command.

  (use git rm --cached <file>... to unstage)

- **A**fter adding all files u need to check the files either it added or not by using the following command.

  git status(if u get green color then it is added)
  (if u get red color then u need to add those files.)

Commit:

- **A**fter adding all files u need to commit the all the staged files by using following commands.

  git commit -m "your message"(# unique message for each commit.)

Push:

- After commiting completed you need to push the entair code into your Github Repository(global repository)by using following command.

  git push origin master(Give your credentials and get code on github)

Git:

- Create multiple files
- for example
  - First.txt
  - Second.txt
  - Third.txt
  - Fourth.txt

1. And write some data on it
2. Add the files in git(git add --all)
3. Commit the files(git commit -m "message")
4. Check the Git log(git log)

- And you will get the latest commits data

- Then type
- **git log --oneline**

then u only get ssh key and type what ssh key you want and which version you want to go
And type git log

- **Git checkout ssh key**

Type git checkout master then u will get head part
Git checkout HEAD~2

HTML:(machine readable text)
It is not a programming or scripting language
It is a markup language
Static - we cannot change
Dynamic - we can change-javascript

Tim berners lee--1991--html-1.0
Present version hyml5

Basic html tags

- Html
- Head
- Body
- Title

Block level elements:
Block level elements occupies the entire space of it's parent element

- Heading tags(H1,h2,....h6)
- Div
- Form
- hr
- P
- Semantic tags

Inline elements:
Inline elements only occupy space bounded by tags.

- Span
- Img
- Form controls(input,select,button,submit)
- navigation(a,href)
    - Mail to(email address)
    - Tel(phone number)

Semantic elements:
it clearly defines its content to both the browser and developer.

- By using semantic tags we will get more accessibility.

Semantic elements= elements with a meaning. Semantic elements have a simple and clear meaning for both, the browser and the developer.

- Header
- Section
- Article
- Aside
- Footer
- Nav
- Main
- audio,video,embed,track,source

css(cascading style sheet):
We can apply beautifications of the webpage

Selectors:

- Class selector(.)
- identifier(#)
- Tagnames(body)
- Universal selector(*)
- Descendant selector(parent child)
    - Parent h2{property:value;}

Kinds of css:

- Inline
- Internal
- External

Padding : 1% 0% 0% 1%;

```
<!DOCTYPE html>
<html>
<head>
	<title>My first webapplication</title>
	<meta name="viewport" content="width=device-width,initial-scale=1.0">
	<link rel="stylesheet" type="text/css" href="dummy.css">
	<style type="text/css">
		h2{
			color:green;
			background-color: skyblue;
		}
	</style>
</head>
<body>
	<h1 style="color:red;font-size: 80px">this is heading1</h1>
	<h2>This is my second heading</h2>
	<h3>This is my third heading</h3>
	<a href="https://www.google.com/">google</a>
	<a href="mailto:suryanarla54121@gmail.com">suryanarla54121</a>
	<a href="tel:999999999">9999999999</a>
	<div calss="parent">hello</div>
	<div id="child">haii</div>
	<ol type="1">
		<li>html</li>
		<li>css</li>
		<li>javascript</li>
		<li>bootstrap</li>
	</ol>
	<ol type="A">
		<li>html</li>
		<li>css</li>
		<li>javascript</li>
		<li>bootstrap</li>
	</ol>
	<ol type="a">
		<li>html</li>
		<li>css</li>
		<li>javascript</li>
		<li>bootstrap</li>
```

```
</ol>
<ol type="I">
        <li>html</li>
        <li>css</li>
        <li>javascript</li>
        <li>bootstrap</li>
</ol>
<ol type="i">
        <li>html</li>
        <li>css</li>
        <li>javascript</li>
        <li>bootstrap</li>
</ol>

<ul style="list-style-type:disc">
        <li>html</li>
        <li>css</li>
        <li>javascript</li>
        <li>bootstrap</li>
</ul>
<ul type="circle">
        <li>html</li>
        <li>css</li>
        <li>javascript</li>
        <li>bootstrap</li>
</ul>
<ul type="square">
        <li>html</li>
        <li>css</li>
        <li>javascript</li>
        <li>bootstrap</li>
</ul>
<ul type="none">
        <li>html</li>
        <li>css</li>
        <li>javascript</li>
        <li>bootstrap</li>
</ul>
<ul>
```

```
            <li>surya</li>
            <li>venkata</li>
            <ul>
                  <li>veera</li>
                  <li>vamsi</li>
            </ul>
            <li>sss</li>
            <li>uuu</li>
      </ul>
      <dl>
            <dt>tea</dt>
            <dd>coffee</dd>
            <dt>milk</dt>
            <dd>white</dd>
      </dl>

      <dl>
            <dt>HTML</dt>

      </dl>

      <table border="1">
            <caption>college data</caption>
            <colgroup style="background-color: ">
  <col span="2" style="background-color:red">
  <col style="background-color:yellow">
</colgroup>
<thead>
            <tr>
                  <th>sno</th>
                  <th>name</th>
                  <th>rollno</th>
            </tr>
            </thead>
            <tbody>
            <tr>
                  <td>1</td>
                  <td>surya</td>
                  <td>535</td>
```

```
        </tr>
        <tr>
            <td>2</td>
            <td>venkat</td>
            <td>536</td>
        </tr>
        </tbody>
        <tfoot>
        <tr>
            <td>3</td>
            <td>venkat</td>
            <td>537</td>
        </tr>
        </tfoot>
    </table>



</body>
</html>
```

viewport:

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

te:

The "width=device-width" part is used to set the width of the page to follow the screen-width of the device (vary according to the devices).

- The initial-scale=1 part is used to set the initial zoom level when the page is first loaded by the browser.

Flexboxes:
The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning

Forms:

The HTML <form> element provide a document section to take input from user. It provides various interactive controls for submitting information to web server such as text field, text area, password field, etc.

Note: The <form> element does not itself create a form but it is container to contain all required form elements, such as <input>, <label>, etc.

```
<form>
//Form elements
</form>
```

HTML <input> element

The HTML <input> element is fundamental form element. It is used to create form fields, to take input from user. We can apply different input filed to gather different information form user. Following is the example to show the simple text input.

```
<body>
  <form>
    Enter your name  <br>
   <input type="text" name="username">
  </form>
</body>
```

HTML TextField Control

The type="text" attribute of input tag creates textfield control also known as single line textfield control. The name attribute is optional, but it is required for the server side component such as JSP, ASP, PHP etc.

Note: If you will omit 'name' attribute then the text filed input will not be submitted to server.

HTML <textarea> tag in form

The <textarea> tag in HTML is used to insert multiple-line text in a form. The size of <textarea> can be specify either using "rows" or "cols" attribute or by CSS

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4.     <title>Form in HTML</title>
5. </head>
6. <body>
7.   <form>
8.         Enter your address:<br>
9.       <textarea rows="2" cols="20"></textarea>
10.  </form>
11. </body>
12. </html>
```

Label Tag in Form

It is considered better to have label in form. As it makes the code parser/browser/user friendly.

If you click on the label tag, it will focus on the text control. To do so, you need to have for attribute in label tag that must be same as id attribute of input tag.

NOTE: It is good to use <label> tag with form, although it is optional but if you will use it, then it will provide a focus when you tap or click on label tag. It is more worthy with touchscreens.

HTML Password Field Control

The password is not visible to the user in password field control

HTML 5 Email Field Control

The email field in new in HTML 5. It validates the text for correct email address. You must use @ and . in this field.

Radio Button Control

The radio button is used to select one option from multiple options. It is used for selection of gender, quiz questions etc.

If you use one name for all the radio buttons, only one radio button can be selected at a time.

Using radio buttons for multiple options, you can only choose a single option at a time.

Checkbox Control

The checkbox control is used to check multiple options from given checkboxes.

Note: These are similar to radio button except it can choose multiple options at a time and radio button can select one button at a time, and its displa

Submit button control

HTML **<input type="submit">** are used to add a submit button on web page. When user clicks on submit button, then form get submit to the server.

Syntax:

1. `<input type="submit" value="submit">`

The type = submit , specifying that it is a submit button

The value attribute can be anything which we write on button on web page.

The name attribute can be omit here.

HTML <fieldset> element:

The <fieldset> element in HTML is used to group the related information of a form. This element is used with <legend> element which provide caption for the grouped elements.

```
<!DOCTYPE html>

<html>

<head>

        <title></title>


</head>

<body>

<form>

        <fieldset>

                <legend>Registration from</legend>


        Enter name:<input type="text" name="username"><br>


        First Name:<input type="text" name="firstname"><br>


        Last Name:<input type="text" name="lastname"><br>


        Enter your address:<textarea rows="2" cols="20"></textarea><br>


        <label for="rollno">rollno :</label>

        <input type="text" id="rollno" name="rollno"><br>


        <label for="collegename">collegename :</label>
```

```
    <input type="text" id="collegename" name="collegename"><br>

    <label for="password">Password: </label>
    <input type="password" id="password" name="password"><br>

    <label for="email">Email:</label>
    <input type="email" id="email" name="email"><br>

    <label for="gender">Gender:</label>
    <input type="radio" id="gender" name="gender" value="male">Male
    <input type="radio" id="gender" name="gender" value="female">female<br>

    Hobby:<br>
    <input type="checkbox" id="cricket" name="cricket" value="Cricket">
    <label for="cricket">cricket</label><br>
        <input type="checkbox" id="reading" name="reading books" value="reading books">
    <label for="reading">reading books</label><br>
        <input type="checkbox" id="cooking" name="cooking" value="cooking">
    <label for="cooking">cooking</label><br>


<label for="register"></label>
    <select name="country" id="register">

<option selected disabled>select your country</option>
    <option value="india">india</option>
    <option value="pakisthan">pakisthan</option>
```

```
        <option value="china">china</option></select>

        <input type="submit" value="submit">
        </fieldset>


</form>
</body>
</html>

<!DOCTYPE html>
<html>
<head>
        <title></title>

</head>
<body>
<form>
        <fieldset>
                <legend>Registration from</legend>
        <table>
        <tr><td>Enter name:</td>

        <td><input type="text" name="username"></td></tr>

        <tr><td>First Name:</td><td><input type="text" name="firstname"></td></tr>
```

```
<tr><td>Last Name:</td><td><input type="text" name="lastname"></td></tr>

<tr><td>Enter your address:</td><td><textarea rows="2" cols="20"></textarea></td></tr>

<tr><td><label for="rollno">rollno :</label></td>
<td><input type="text" id="rollno" name="rollno"></td></tr>

<tr><td><label for="collegename">collegename :</label></td>
<td></td><input type="text" id="collegename" name="collegename"></tr></tr>

<tr><td><label for="password">Password: </label></td>
<td><input type="password" id="password" name="password"></td></tr>

<tr><td><label for="email">Email:</label></td>
<td><input type="email" id="email" name="email"></td></tr>

<tr><td><label for="gender">Gender:</label></td>
<td><input type="radio" id="gender" name="gender" value="male">Male
<input type="radio" id="gender" name="gender" value="female">female</td></tr>

<tr><td>Hobby:<br>
<input type="checkbox" id="cricket" name="cricket" value="Cricket">
<label for="cricket">cricket</label><br>
        <input type="checkbox" id="reading" name="reading books" value="reading books">
<label for="reading">reading books</label><br>
        <input type="checkbox" id="cooking" name="cooking" value="cooking">
<label for="cooking">cooking</label></td></tr>
```

```
        <tr><label for="register"></label>
        <td>Selet your country</td>
        <td><select name="country" id="register">

   <option selected disabled>select your country</option>
        <option value="india">india</option>
        <option value="pakisthan">pakisthan</option>
        <option value="china">china</option></select></td></tr>

        </table>

<td colspan="2"><div align="center"><input type="submit" id="register_0" value="register"/> </div>
        </fieldset>


</form>
</body>
</html>
```

What is The Viewport?

The viewport is the user's visible area of a web page.

The viewport varies with the device, and will be smaller on a mobile phone than on a computer screen.

Before tablets and mobile phones, web pages were designed only for computer screens, and it was common for web pages to have a static design and a fixed size.

Then, when we started surfing the internet using tablets and mobile phones, fixed size web pages were too large to fit the viewport. To fix this, browsers on those devices scaled down the entire web page to fit the screen.

This was not perfect!! But a quick fix.

Setting The Viewport

HTML5 introduced a method to let web designers take control over the viewport, through the `<meta>` tag.

You should include the following `<meta>` viewport element in all your web pages:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

This gives the browser instructions on how to control the page's dimensions and scaling.

The `width=device-width` part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).

The `initial-scale=1.0` part sets the initial zoom level when the page is first loaded by the browser.

Here is an example of a web page *without* the viewport meta tag, and the same web page *with* the viewport meta tag:

Tip: If you are browsing this page with a phone or a tablet, you can click on the two links below to see the difference.

Svg :

SVG stands for Scalable Vector Graphics.

SVG defines vector-based graphics in XML format.

What is SVG?

- SVG stands for Scalable Vector Graphics
- SVG is used to define vector-based graphics for the Web
- SVG defines the graphics in XML format
- Every element and every attribute in SVG files can be animated
- SVG is a W3C recommendation
- SVG integrates with other W3C standards such as the DOM and XSL

SVG is a W3C Recommendation

Advantages of using SVG over other image formats (like JPEG and GIF) are:

- SVG images can be created and edited with any text editor
- SVG images can be searched, indexed, scripted, and compressed
- SVG images are scalable
- SVG images can be printed with high quality at any resolution
- SVG images are zoomable
- SVG graphics do NOT lose any quality if they are zoomed or resized
- SVG is an open standard
- SVG files are pure XML

```
<!DOCTYPE html>

<html>

<body>



<h1>My first SVG</h1>
```

```
<svg width="100" height="100">

  <circle cx="50" cy="50" r="40" stroke="green" stroke-width="4" fill="yellow"
/>

</svg>



</body>

</html>
```

Code explanation:

- An SVG image begins with an <svg> element
- The width and height attributes of the <svg> element define the width and height of the SVG image
- The <circle> element is used to draw a circle
- The cx and cy attributes define the x and y coordinates of the center of the circle. If cx and cy are not set, the circle's center is set to (0, 0)
- The r attribute defines the radius of the circle
- The stroke and stroke-width attributes control how the outline of a shape appears. We set the outline of the circle to a 4px green "border"
- The fill attribute refers to the color inside the circle. We set the fill color to yellow
- The closing </svg> tag closes the SVG image

Flexbox:

```
<!DOCTYPE html>

<html>

<head>

    <title></title>

    <meta name="viewport" content="width=device-width,initial-scale=1.0">

    <style type="text/css">
```

```
        .flex{
            display: flex;
            flex-direction: row;
            /*flex-direction: column;*/
            /*flex-direction: row-reverse;*/

/*flex-direction: column-reverse;*/
/*justify-content: flex-start;*/
/*justify-content: center;*/
justify-content: flex-end;
/*justify-content: space-around;*/
/*justify-content: space-between*/;


        }

        .flex div{
            padding: 1%;
            margin: 1%;
            text-align: center;
            width:100px;
            background-color: skyblue;
            font-size: 30px;
            border: 2px solid #ddd;
```

```
            }
        </style>

</head>
<body>
        <div class="flex">
                <div>A</div>
                <div>B</div>
                <div>C</div>
                <div>D</div>

        </div>

</body>
</html>
```

Flexbox2.html:

```
<!DOCTYPE html>
<html>
<head>
        <meta name="viewport" content="device-width,initial-scale=1.0">
        <title></title>
        <style type="text/css">
                .flex{
```

```
            display: flex;
            height: 800px;
            /*align-items: center;*/
        /*align-items: flex-start;*/
        /*align-items: flex-end;*/
        /*align-items: stretch;*/
        /*align-items: baseline;*/
        flex-wrap: wrap;
        /*align-content: space-between;*/
/*      align-content: space-around;*/
/*align-content: stretch;*/
/*align-content: center;*/
align-content: flex-start;
/*align-content: flex-end;*/
/*justify-content: center;
align-items: center;*/



        }

        .flex div{
            border: 2px solid #ddd;
            padding: 1%;
            margin: 1%;
```

```
                font-size: 120px;
                background-color: skyblue;
            }

        </style>

</head>
<body>
        <div class="flex">
        <div>A</div>
        <div>B</div>
        <div>C</div>
        <div>D</div>
        <div>E</div>
        <div>F</div>
        <div>G</div>
        <div>H</div>
        <div>I</div>
        <div>J</div>
        <div>k</div>
        <div>l</div>
        <div>m</div>
        <div>n</div>
        <div>o</div>
```

```
        <div>p</div>
        <div>q</div>
        <div>r</div>

</body>
</html>
```

Flexbox3.html

```
<!DOCTYPE html>
<html>
<head>
        <title></title>
        <meta name="viewport" content="width=device-width,initial-scale=1.0">
        <style type="text/css">
                .flex{
                        display: flex;
                        height: 600px;
                        /*align-items: baseline;*/
                        flex-wrap: wrap;
                        align-content: flex-start;
```

```
        }

        .flex div{
            padding: 1%;
            margin: 1%;
            text-align: center;
            width:100px;
            background-color: skyblue;
            font-size: 30px;
            border: 2px solid #ddd;
        }
    </style>

</head>
<body>
    <div class="flex">
        <div style="order: 3">1</div>
        <div style="order: 1">2</div>
        <div style="order: 2">3</div>
        <div style="flex-grow: 1">5</div>
        <div style="flex-shrink: 0">6</div>
        <div style="flex-basis: 400px">9</div>
        <div style="flex: 4 0 200px">7</div>
        <!-- <div>F</div>
```

```
        <div>G</div>
        <div>H</div>
        <div>I</div>
        <div>J</div>
        <div>K</div>
        <div>L</div> -->
    </div>


</body>
</html>
```

Media queries:

What is a Media Query?

Media query is a CSS technique introduced in CSS3.

Add a Breakpoint

Earlier in this tutorial we made a web page with rows and columns, and it was responsive, but it did not look good on a small screen.

Media queries can help with that. We can add a breakpoint where certain parts of the design will behave differently on each side of the breakpoint.

For mobiles:

```
/*
    ##Device = Most of the Smartphones Mobiles
    (Portrait)
    ##Screen = B/w 320px to 479px
    */
```

```
@media only screen and (min-device-width : 320px) and (max-device-width : 480px) {
/* Styles */
}
```

```
/*
    ##Device = Low Resolution Tablets, Mobiles
    (Landscape)
    ##Screen = B/w 481px to 767px
    */
```

```
@media only screen and (min-device-width : 481px) and (max-device-width : 767px) {
/* Styles */
}

  /*
        ##Device = Tablets, Ipads (portrait)
        ##Screen = B/w 768px to 1024px
       */
@media only screen and (min-device-width : 769px) and (max-device-width : 1024px) {
/* Styles */
}

  /*
        ##Device = Laptops, Desktops
        ##Screen = B/w 1025px to
       1280px
       */
@media only screen and (min-device-width : 1025px) and (max-device-width : 1280px) {
/* Styles */
}
  /
  *
```

##Device = Desktops

##Screen = 1281px to higher resolution desktops

*/

Bootstrap:

What is bootstrap:

- Bootstrap is the popular HTML, CSS and JavaScript framework for developing a responsive web applications.
    - Bootstrap was developed by Mark Otto and Jacob Thornton at Twitter
    - It is absolutely free to download and use.
    - It is a front-end framework used for easier and faster web development.
    - By using Bootstrap we can build Responsiveness web applications with the help of predefined classes

dvantages of using bootstrap:

ightweight and customizable:

- If we talk about lightweight and customizability then bootstrap 4 is one of the best. Because this is one of the lightweight frameworks. you can install this very easily and also loading speed is very fast.
- You can customize bootstrap 4 whatever you want. bootstrap fully supports customization.

ves lot of development time

- This will saves a lot of time. because it's very easy to develop. Bootstrap provides all the classes ready mate. There is a very large class library in the bootstrap. We can say bootstrap works on only classes.

pen Source:

- Yes, This is right! Bootstrap is a totally open source. If you want to use bootstrap then you don't have to pay. You just have to go on the bootstrap's official website and download directory from

there. Bootstrap also provides CDN for your use

Compatible with browsers:

- If you can use bootstrap 4 for your website, then you don't worry about the compatibility of browsers. because if you are installing bootstrap 4 on your website this will compatible with all the browsers. like Google Chrome, Firefox, Safari, Opera Mini, and many more browsers.

Great grid system:

- In this framework grid system is amazing. we can say one of the very useful for a responsive web design system that is the grid system. here we can see 1 to 12 pairs of the grid system and we can use it as per our need.

How to use bootstrap in the application:

- We can use bootstrap to websites in 2 ways
- 1.Offline(download directory)
- 2.Online(using CDN)

.Online CDN for Bootstrap:

CSS:

Have to follow some steps to install this CDN in websites.

```
step 1: Open your website's "index.html" page.
step 2: you will see <head> tag in the nested of **<html>** tag.
step 3: copy this CDN
```

- `step 4: Paste this CSS CDN in the top of </head> tag`

```
<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css"
integrity="sha384-9aIt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8WgZl5MYYxFfc+NcPb1dKGj7Sk"
crossorigin="anonymous">
```

Javascript & Jquery:

For pasting JavaScript in the body you have to follow bottom steps:

`Step 1: Scroll the bottom of your “index.html” file.`
`Step 2: There you will see the body tag is closing. like </body>.`

- `Step 3: Copy the bottom scripts and past on the top of the </body> tag.`

```
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj
" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"
integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo
" crossorigin="anonymous"></script>

<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"
integrity="sha384-OgVRvuATP1z7JjHLkuOU7Xw704+h835Lr+6QL9UvYjZE3Ipu6Tp75j7Bh/kR0JKI
" crossorigin="anonymous"></script>
```

Download Bootstrap for offline:

- First Download CDN's Directory and use by following steps

Step 1: Go to Website getbootstrap.com

Step 2: Select the version of bootstrap which you want to download.

step 3: After selection of versions move on the right side of the header and click on the download button.

step 4: Extract the bootstrap directory in local computer

step 5: It Have many files in the folder from those you have to open the dist folder

step 6: In the dist folder you will found two folders `1.CSS 2.JS`

step 7: In CSS Folder contains many files in that you have to select and copy bootstrap.min.css and

paste that file in the project directory.

Step 8: Same like as CSS files there also js files in JS folder from there we will copy the bootstrap.min.js file and place this in project directory

Step 9: Open the “Index.html” file.

Step 10: <link rel=”stylesheet” href=”Here your file directory path”> copy this line and past in the top of tag. This is for the “bootstrap.min.css” file.

Step 11: We have pasted CSS file now Past Javascript in the top of tag. for example <script src=”Here your file path”>. This is for the “bootstrap.min.js” file.

Example:

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>bootstrap</title>
    <link rel="stylesheet" href="/css/bootstrap.min.css">
  </head>
  <body>
    <h1>Welcome to bootstrap</h1>

    <script type="text/javascript" src="/js/bootstrap.min.js">
  </body>
</html>
```

Note*:

- If you are using Bootstrap directory online then this will actually access data from the bootstrap own directory. this will take time from retrieving data from there.
- And if you are using the Bootstrap 4.0 directory from the download. then this will access data from your own directory. So this will easy to access data for your website

Colors:

- Convey meaning through color with a handful of color utility classes. Includes support for styling

links with hover states, too.

```
<p class="text-primary">.text-primary</p>
<p class="text-secondary">.text-secondary</p>
<p class="text-success">.text-success</p>
<p class="text-danger">.text-danger</p>
<p class="text-warning">.text-warning</p>
<p class="text-info">.text-info</p>
<p class="text-light bg-dark">.text-light</p>
<p class="text-dark">.text-dark</p>
<p class="text-muted">.text-muted</p>
<p class="text-white bg-dark">.text-white</p>
```

- Contextual text classes also work well on anchors with the provided hover and focus states. Note that the .text-white and .text-muted class has no link styling.

Links:

```
<p><a href="#" class="text-primary">Primary link</a></p>
<p><a href="#" class="text-secondary">Secondary link</a></p>
<p><a href="#" class="text-success">Success link</a></p>
<p><a href="#" class="text-danger">Danger link</a></p>
<p><a href="#" class="text-warning">Warning link</a></p>
<p><a href="#" class="text-info">Info link</a></p>
<p><a href="#" class="text-light bg-dark">Light link</a></p>
<p><a href="#" class="text-dark">Dark link</a></p>
<p><a href="#" class="text-muted">Muted link</a></p>
<p><a href="#" class="text-white bg-dark">White link</a></p>
```

Background color:

Similar to the contextual text color classes, easily set the background of an element to any contextual class. Anchor components will darken on hover, just like the text classes. Background utilities do not set color, so in some cases you'll want to use .text-* utilities.

m - sets margin

p - sets padding

- for classes that set margin, like this :

mt - for classes that set margin-top

mb - for classes that set margin-bottom

ml - for classes that set margin-left

mr - for classes that set margin-right

mx - for classes that set both margin-left and margin-right

my - for classes that set both margin-top and margin-bottom

0 - sets margin or padding to 0

1 - sets margin or padding to .25rem (4px if font-size is 16px)

2 - sets margin or padding to .5rem (8px if font-size is 16px)

3 - sets margin or padding to 1rem (16px if font-size is 16px)

4 - sets margin or padding to 1.5rem (24px if font-size is 16px)

5 - sets margin or padding to 3rem (48px if font-size is 16px)

auto - sets margin to auto

Example:

```
1. <div class="p-3 mb-2 bg-primary text-white">.bg-primary</div>
2. <div class="p-3 mb-2 bg-secondary text-white">.bg-secondary</div>
3. <div class="p-3 mb-2 bg-success text-white">.bg-success</div>
4. <div class="p-3 mb-2 bg-danger text-white">.bg-danger</div>
5. <div class="p-3 mb-2 bg-warning text-dark">.bg-warning</div>
6. <div class="p-3 mb-2 bg-info text-white">.bg-info</div>
7. <div class="p-3 mb-2 bg-light text-dark">.bg-light</div>
8. <div class="p-3 mb-2 bg-dark text-white">.bg-dark</div>
9. <div class="p-3 mb-2 bg-white text-dark">.bg-white</div>
```

Bootstrap Alerts:

- Bootstrap Alerts are used to provide an easy way to create predefined alert messages. Alert adds

a style to your messages to make it more appealing to the users.
- Provide contextual feedback messages for typical user actions with the handful of available and flexible alert messages.
- Alerts are available for any length of text, as well as an optional dismiss button. For proper styling, use one of the eight required contextual classes (e.g., .alert-success).

Example:

```
<div class="alert alert-primary" role="alert">
 This is a primary alert—check it out!
</div>

<div class="alert alert-secondary" role="alert">
 This is a secondary alert—check it out!
</div>

<div class="alert alert-success" role="alert">
 This is a success alert—check it out!
</div>

<div class="alert alert-danger" role="alert">
 This is a danger alert—check it out!
</div>

<div class="alert alert-warning" role="alert">
 This is a warning alert—check it out!
</div>
```

```
<div class="alert alert-info" role="alert">
 This is a info alert—check it out!
</div>


<div class="alert alert-light" role="alert">
 This is a light alert—check it out!
</div>


<div class="alert alert-dark" role="alert">
 This is a dark alert—check it out!
</div>
```

Dismissing:

Using the alert JavaScript plugin, it’s possible to dismiss any alert inline. Here’s how:

- Be sure you’ve loaded the alert plugin, or the compiled Bootstrap JavaScript.
- If you’re building our JavaScript from source, it requires util.js. The compiled version includes this.
- Add a dismiss button and the .alert-dismissible class, which adds extra padding to the right of the alert and positions the .close button.
- On the dismiss button, add the data-dismiss="alert" attribute, which triggers the JavaScript functionality. Be sure to use the element with it for proper behavior across all devices

Example:

```
<div class="alert alert-warning alert-dismissible fade show" role="alert">
  <strong>Holy guacamole!</strong> You should check in on some of those fields below.
  <button type="button" class="close" data-dismiss="alert" aria-label="Close">
    <span aria-hidden="true">&times;</span>
  </button>
```

```
</div>
```

Container:

- container is used to wrap the site contents
- There are two container classes.
    - .container
    - .container-fluid

Container:

- The .container class provides a responsive fixed width container

Syntax:

```
<div class="container">
      <!-- Content here -->
   </div>
```

.Container fluid

Use .container-fluid for a full width container, spanning the entire width of the viewport.

Example of container and container fluid:

```
<!DOCTYPE html>

<html lang="en" dir="ltr">

  <head>

    <meta charset="utf-8">
```

```
    <title></title>
    <link rel="stylesheet" href="/css/bootstrap.min.css">
  </head>
  <body>
<div style="border: 1px solid green;margin: 15px;">
    <div class="container">
        <div class="col-md-6">
         <h1>Bootstrap container</h1>
         <p>Normal Size</p>
         <p>
           <button type="button" class="btn btn-primary">container one</button>
           <button type="button" class="btn btn-danger">container two</button>
           <button type="button" class="btn btn-warning">container three</button>
         </p>
        </div>
    </div>
    <div class="container-fluid">
        <div class="col-md-6">
         <h1>Bootstrap fluid container</h1>
         <p>Normal Size</p>
         <p>
           <button type="button" class="btn btn-primary">container one</button>
```

```
          <button type="button" class="btn btn-danger">container demo</button>

          <button type="button" class="btn btn-warning">container demo</button>

        </p>

      </div>

  </div>

  </body>

</html>
```

Bootstrap Jumbotron:

- A Bootstrap jumbotron specifies a big box for getting extra attention to some special content or information. It is displayed as a grey box with rounded corners. It can also enlarge the font sizes of the text inside it.
- You can put any valid HTML or other Bootstrap elements/ classes inside a jumbotron.
- The class .jumbotron within the
  element is used to create a jumbotron.

umbotron Inside Container:

- The Inside container is used in jumbotron, if you want the jumbotron to not extend to the edge of the screen.
- Put the jumbotron inside the

```
<!DOCTYPE html>

<html lang="en">

<head>
```

```
  <title>Bootstrap Example</title>

  <meta charset="utf-8">

  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">

</head>

<body>



<div class="container">

  <div class="jumbotron">

    <h1>This is Jumbotron inside container!</h1>

    <p>Jumbotron specifies a big box for getting extra attention to some special
content or information.</p>

  </div>

  <p>This is some text.</p>

  <p>This is another text.</p>

</div>



  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>

  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>

</body>
```

```
</html>
```

Full-width Jumbotron:

- To get a jumbotron without rounded borders, you have to add the .jumbotron-fluid class and a .container or .container-fluid inside it.

```
<!DOCTYPE html>

<html lang="en">

<head>

 <title>Bootstrap Example</title>

 <meta charset="utf-8">

 <meta name="viewport" content="width=device-width, initial-scale=1">

 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/css/bootstrap.min.css">

 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>

 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.6/umd/popper.min.js"></script>

 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/js/bootstrap.min.js"></script>

</head>

<body>



<div class="jumbotron jumbotron-fluid">

 <div class="container">
```

```
  <h1>Full-width Jumbotron</h1>
 <p>Jumbotron specifies a big box for getting extra attention to some special content or information.</p>
</div>

<div class="container">
 <p>This is some text.</p>
 <p>This is another text.</p>
</div>

</body>
</html>
```

Bootstrap 4 Grid System:

Grid System nested of rows class. In the bootstrap grid system allows upto 12 columns across the page. we can use the grid as per our requirement but total of grids is must be 12.

How Works the bootstrap Grid System:

The grid system is the bundle of classes. There are 12 columns and a 1-row total bundle of 13 classes. There is no limitation for the use of columns. But you have to mind one thing, The thing is the total of your grids must be 12. This is the biggest reason for the system is flexible with all the screen sizes. This grid system supports a max value of 12 columns. Anything after the 12th column will be shifted to a new line.

This will re-arrange as per the device's screen size. You can use the grid in the .row class. Row class is must be required for grids. because this will help to arrange the columns of the grid. and the row class should be nested of .container class. You can use rows in the .container class.

Grid Classes: There are four classes in Bootstrap Grid System:

- xs(for phones)
- sm(for tablets)
- md(for laptops & smaller desktops)
- lg(for larger desktops)
- xl(projectors)

Bootstrap 4 Grid Classes: There are 5 classes in Bootstrap grid system.

- .col-(extra small devices)
- .col-sm- (small devices)
- .col-md- (medium devices)
- .col-lg- (large devices)

- .col-xl- (xlarge devices)

Size of grids in different screens:

Screen Size	Columns work
Less then 575px	col-*
Between 575px to 768px	col-sm-*
Between 768px to 992px	col-md-*
Between 992px to 1200px	col-lg-*
More then 1200px	col-xl-*

Container:

- .container, which sets a max-width at each responsive breakpoint
- .container-fluid, which is width: 100% at all breakpoints

```
<!DOCTYPE html>
<html>
<head>
	<title></title>
	<link rel="stylesheet" type="text/css" href="bootstrap.min.css">
	<meta name="viewport" content="width=device-width,initial-scale=1.0">
	<style type="text/css">
		.customise-row{
			border: 1px dashed red;

		}
		.customise-col{
			border: 1px dotted red;

		}
	</style>
</head>
<body>
	<div class="container" style="border:2px solid black;">

	<!-- row1 -->
		<div class="row customise-row">
			<!-- better to use even numbers of grid -->
			<div class="col-xs-2 col-sm-4 col-md-6 col-lg-4 customise-col">
				row1 - col1
			</div>

			<div class="col customise-col">
				row1 - col2
			</div>
    </div>

      <!-- row2 -->
		<div class="row customise-row">
			<div class="col customise-col">
				row2 - col1
			</div>
     </div>
</div>

</body>
</html>
```

Card header:

```
<!DOCTYPE html>
<html>
```

```
<head>
	<title></title>
	<link rel="stylesheet" type="text/css" href="bootstrap.min.css">
	<meta name="viewport" content="width=device-width,initial-scale=1.0">
</head>
<body>
<div class="row" style="justify-content: center">
	<div class="col-md-4 col-sm-8 col-lg-4">
		<div class="card-header bg-primary text-dark">card header</div>
		<div class="card-body">card body</div>
		<div class="card-footer">card footer</div>
	</div>
	<div class="col-md-4 col-sm-8 col-lg-4">
		<div class="card-header">card header</div>
		<div class="card-body">card body</div>
		<div class="card-footer">card footer</div>

  </div>

</div>
</body>
</html>
```

Tables:

```
<!DOCTYPE html>
<html>
<head>
	<title></title>
	<link rel="stylesheet" type="text/css" href="bootstrap.min.css">
	  <script src="jquery-3.5.1.min.js"></script>
	<script type="text/javascript" src="bootstrap.min.js"></script>
	<meta name="viewport" content="width=device-width,initial-scale=1.0">
</head>
<body>

	<!-- table-responsive-->
	<div class="table-responsive">
		<!-- table-dark -->
	<!-- table-striped= alternate rows different colors -->
	<!-- table-bordered= table border -->
	<!-- table-hover=mouse hover -->
		<table class="table table-striped">
		<tr>
			<!-- thead-dark -->
			<thead class="thead-dark">
			<th>sno</th>
			<th>Name</th>
			<th>rollno</th>
```

```
                <th>collegename</th>
            </tr>
            </thead>
            <tbody>

            <tr class="table-primary">
                        <td class="bg-danger">1</td>
                <td>surya</td>
                <td>535</td>
                <td>sai aditya engineering college</td>

            </tr>
            <tr class="bg-primary">
                <td class="bg-success">2</td>
                <td>venkata</td>
                <td>540</td>
                <td>aditya engineering college</td>
            </tr>
            <tr class="table-success">
                <td>3</td>
                <td>Narayana</td>
                <td>547</td>
                <td>sri aditya engineering college</td>
            </tr>
            </tbody>
        </table>
        </div>


</body>
</html>
```

Cards:

```
<!DOCTYPE html>
<html>
<head>
        <title></title>
        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css"
integrity="sha384-9alt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8WgZl5MYYxFfc+NcPb1dKGj7Sk"
crossorigin="anonymous">
        <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"
integrity="sha384-Q6E9RHvblyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo"
crossorigin="anonymous"></script>
```

```
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"
integrity="sha384-OgVRvuATP1z7JjHLkuOU7Xw704+h835Lr+6QL9UvYjZE3Ipu6Tp75j7Bh/kR0JKI"
crossorigin="anonymous"></script>
        <meta name="viewport" content="width=device-width,initial-scale=1.0">
</head>
<body>
<div class="row" style="justify-content: center">
        <div class="col-md-4 col-sm-8 col-lg-4">
                <div class="card-header bg-primary text-dark">card header</div>
                <div class="card-body">card body</div>
                <div class="card-footer">card footer</div>
        </div>
        <div class="col-md-4 col-sm-8 col-lg-4">
                <div class="card-header">card header</div>
                <div class="card-body">card body</div>
                <div class="card-footer">card footer</div>



        </div>


</div>
</body>
</html>
```

Profile html:

```
<!DOCTYPE html>

<html lang="en">

<head>

  <title>Bootstrap Example</title>

  <meta charset="utf-8">

  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">

  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
```

```
  <script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>

  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"></script>

 <style type="text/css">

   .hai{

    display: flex;

    flex-direction: row;

    justify-content: space-around;

    margin: 1%;

    padding:1%;

    }



    @media only screen and (min-width: 320px) and (max-width: 480px){

      .hai{

    display: flex;

    flex-direction: column;

    justify-content: space-around;

    margin: 1%;

    padding:1%;

    }

    }
```

```
 </style>
</head>
<body>

<div class="container hai">
  <div class="card" style="width:400px">

    <div class="card-body" style="text-align: center;font-size: 25px;">
      <img class="card-img-bottom" src="third1.jpg" alt="Card image"
style="width:100%;height:50%">
      <h4 class="card-title">Surya</h4>
      <p class="card-text">Software Trainer</p>
      <a href="#" class="btn btn-primary stretched-link">See Profile</a>
    </div>
  </div>
  <br>

  <div class="card" style="width:400px">
    <div class="card-body" style="text-align: center;font-size: 25px;">
      <img class="card-img-bottom" src="third1.jpg" alt="Card image"
style="width:100%;height:50%">
      <h4 class="card-title">Surya</h4>
```

```
      <p class="card-text">Software Trainer</p>

      <a href="#" class="btn btn-primary stretched-link">See Profile</a>

    </div>

  </div>

</div>



</body>

</html>
```

Nav:

```
<!DOCTYPE html>

<html>

<head>

      <title></title>

      <meta name="viewport" content="width=device-width,initial-scale=1.0">

      <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">

  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

  <script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>

  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"></script>

</head>
```

```
<body>

	<!-- justify-content-end -->

	<!-- justify-content-center -->

	<!-- flex-column -->

	<!-- nav nav-tabs,nav-link active-->

	<!-- nav nav-pills,nav-link active -->

	<!-- nav nav-pills nav-justified,nav-link active -->




	<ul class="nav nav-tabs nav-justified">

		<li class="nav-item">

			<a class="nav-link active" href="#">Home</a>

		</li>

		<li class="nav-item">

			<a class="nav-link active" href="#">Login</a>

		</li>

		<li class="nav-item">

			<a class="nav-link" href="#">Register</a>

		</li>

		<li class="nav-item">

			<a class="nav-link disabled" href="#">Disabled</a>

		</li>
```

```
        </ul>


</body>

</html>
```

Nav2 :

```
<!DOCTYPE html>

<html>

<head>

        <title></title>

        <meta name="viewport" content="width=device-width,initial-scale=1.0">

        <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">

 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>

 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"></script>

</head>

<body>

        <!-- justify-content-end -->
```

```
        <!-- justify-content-center -->

        <!-- flex-column -->

        <!-- nav nav-tabs,nav-link active-->

        <!-- nav nav-pills,nav-link active -->

        <!-- nav nav-pills nav-justified,nav-link active -->



<!-- nav nav-pills -->

        <ul class="nav nav-pills">

                <li class="nav-item">

                        <!-- data-toggle="tab" -->

                        <a class="nav-link active" data-toggle="pill" href="#home">Home</a>

                </li>

                <li class="nav-item">

                        <!-- data-toggle="tab" -->

                        <a class="nav-link" data-toggle="pill" href="#login">Login</a>

                </li>

                <li class="nav-item">

                        <!-- data-toggle="tab" -->

                        <a class="nav-link" data-toggle="pill" href="#register">Register</a>
```

```
            </li>
        </ul>

        <!-- Tab panes -->
<div class="tab-content">
  <div class="tab-pane container active" id="home">This is home page</div>
  <div class="tab-pane container fade" id="login">This is login page</div>
  <div class="tab-pane container fade" id="register">This is register page</div>
</div>

</body>
</html>
```

Dropdown nav:

```
<!DOCTYPE html>
<html>
<head>
        <title></title>
        <meta name="viewport" content="width=device-width,initial-scale=1.0">
        <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">
```

```
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"></script>
</head>
<body>
    <ul class="nav nav-tabs">
        <li class="nav-item">
            <a class="nav-link active" href="#">Active</a>
        </li>
        <li class="nav-item dropdown">
            <a class="nav-link dropdown-toggle" data-toggle="dropdown" href="">Dropdown</a>
            <div class="dropdown-menu">
                <a class="dropdown-item" href="#">link 1</a>
                <a class="dropdown-item" href="#">link 2</a>
                <a class="dropdown-item" href="#">link 3</a>
            </div>

        </li>
```

```
        <li class="nav-item">
            <a href="nav-link" href="">Link</a>

        </li>
            <li class="nav-item">
            <a href="nav-link disabled" href="">Disabled</a>

        </li>
        </ul>

</body>
</html>
```

Navbar:

```
<!DOCTYPE html>
<html>
<head>
        <title></title>
        <meta name="viewport" content="width=device-width,initial-scale=1.0">
        <link rel="stylesheet"
```

```
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"></script>
</head>
<body>



    <nav class="navbar navbar-expand-sm bg-dark navbar-dark">
        <!-- <a class="navbar-brand" href="">
            <img src="bird.PNG" alt="logo" style="width: 40px">
        </a> -->
        <img src="bird.PNG" alt="logo" style="width: 40px">
    <ul class="navbar-nav">
        <li class="nav-item">
            <a class="nav-link active" href="#">Home</a>
        </li>
        <li class="nav-item">
            <a class="nav-link active" href="#">Login</a>
```

```
            </li>
            <li class="nav-item">
                <a class="nav-link" href="#">Register</a>
            </li>
            <li class="nav-item">
                <a class="nav-link disabled" href="#">Disabled</a>
            </li>

        </ul>
        </nav>



</body>
</html>
```

Navbar collapsing:

```
<!DOCTYPE html>
<html>
```

```
<head>
        <title></title>
        <meta name="viewport" content="width=device-width,initial-scale=1.0">
        <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"></script>
</head>
<body>
        <nav class="navbar navbar-expand-sm bg-dark navbar-dark">
                <a class="navbar-brand" href="">
                        Navbar
                </a>
                <button class="navbar-toggler" type="button" data-toggle="collapse"
data-target="#collapsibleNavbar">
                <span class="navbar-toggler-icon"></span>
                </button>

                <div class="collapse navbar-collapse" id="collapsibleNavbar">
```

```
<ul class="navbar-nav">
        <li class="nav-item">
                <a class="nav-link active" href="#">Home</a>
        </li>
        <li class="nav-item">
                <a class="nav-link active" href="#">Login</a>
        </li>
        <li class="nav-item">
                <a class="nav-link" href="#">Register</a>
        </li>
        <li class="nav-item">
                <a class="nav-link disabled" href="#">Disabled</a>
        </li>


</ul>
</div>
</nav>
```

```
</body>

</html>
```

Navbar dropdown:

```
<!DOCTYPE html>

<html>

<head>

        <title></title>

        <meta name="viewport" content="width=device-width, initial-scale=1">

 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">

 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>

 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"></script>

</head>

<body>

        <!-- justify-content-end -->

        <!-- justify-content-center -->

        <!-- flex-column -->

        <!-- nav nav-tabs,nav-link active-->
```

```
<!-- nav nav-pills,nav-link active -->

<!-- nav nav-pills nav-justified,nav-link active -->



<nav class="navbar navbar-expand-sm bg-dark navbar-dark">

<a href="navbar-brand">logo</a>




<ul class="navbar-nav">

        <li class="nav-item">

                <a class="nav-link active" href="#">Home</a>

        </li>

        <li class="nav-item">

                <a class="nav-link active" href="#">Login</a>

        </li>



        <li class="nav-item dropdown">

                <a class="nav-link dropdown-toggle" href="#" id="navbardrop"
data-toggle="dropdown">Dropdownlink</a>



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

```
                <a class="dropdown-item" href="">link 1</a>

                <a class="dropdown-item" href="">link 2</a>

                <a class="dropdown-item" href="">link 1</a>

                <a class="dropdown-item" href="">link 1</a>

            </div>

            </li>

            <li class="nav-item">

                <a class="nav-link" href="#">Register</a>

            </li>

            <li class="nav-item">

                <a class="nav-link disabled" href="#">Disabled</a>

            </li>



        </ul>

        </nav>



</body>

</html>
```