Experiment Number 1: To install and setup the HTML5 based Bootstrap framework and to deploy basic HTML elements using Bootstrap CSS.

What is Bootstrap?

Bootstrap is a widely used open-source front-end framework for developing responsive and visually appealing web applications. In this experiment, you will learn how to install and set up the HTML5-based Bootstrap framework and utilize it to deploy basic HTML elements styled with Bootstrap's CSS classes.

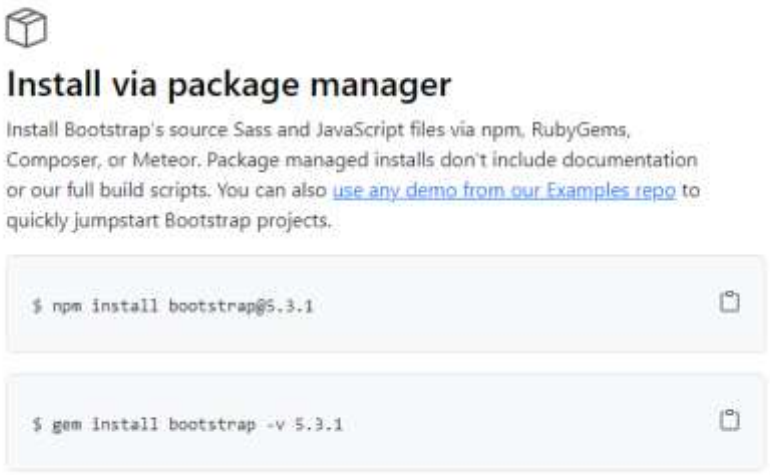
How to download Bootstrap 5:

1. Visit the Official Bootstrap Website:
   Open your web browser and go to the official Bootstrap website at https://getbootstrap.com.
2. Download Bootstrap:
   Once you are on the Bootstrap website, you can download the framework in two ways:

   a. Download Precompiled Bootstrap:

   -- Scroll down the Bootstrap website's homepage.
   -- In the "Download" section, you will find two main options:

   - **Download compiled CSS and JS:** This is the recommended option for most users. It includes precompiled CSS and JavaScript files that are ready to use in your projects.
   - **Download source files:** This option is for advanced users who want to customize Bootstrap's styles and components.



Install via package manager

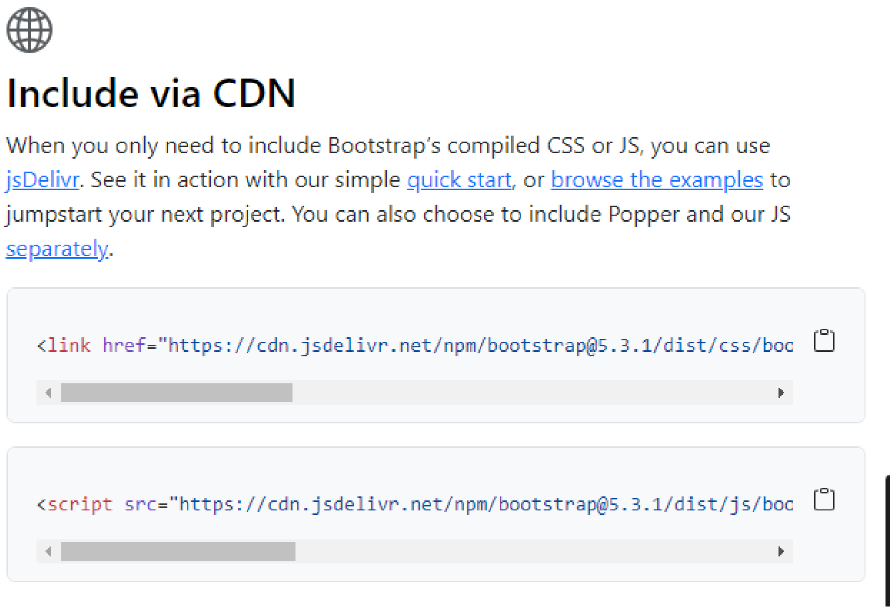
Install Bootstrap's source Sass and JavaScript files via npm, RubyGems, Composer, or Meteor. Package managed installs don't include documentation or our full build scripts. You can also use any demo from our Examples repo to quickly jumpstart Bootstrap projects.

```
$ npm install bootstrap@5.3.1
```

```
$ gem install bootstrap -v 5.3.1
```

b. Download Using CDN (Content Delivery Network):

- If you prefer not to download Bootstrap and instead want to use it via a CDN, you can find the CDN links in the "Quick Start" section of the Bootstrap website. These links allow you to include Bootstrap directly in your HTML files without downloading any files.



Include via CDN

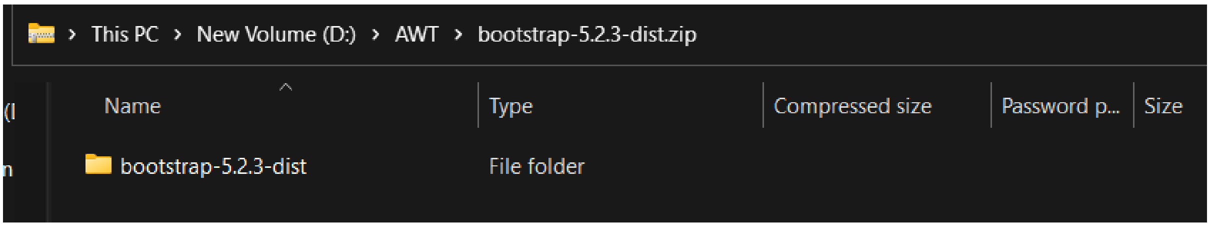
When you only need to include Bootstrap's compiled CSS or JS, you can use jsDelivr. See it in action with our simple quick start, or browse the examples to jumpstart your next project. You can also choose to include Popper and our JS separately.

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/css/boo
```

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/js/boo
```

3. Choose the Compiled Version:
   If you choose to download the compiled CSS and JS files, click the "Download" button next to the "Download compiled CSS and JS" option. This will start the download of a ZIP file containing Bootstrap.

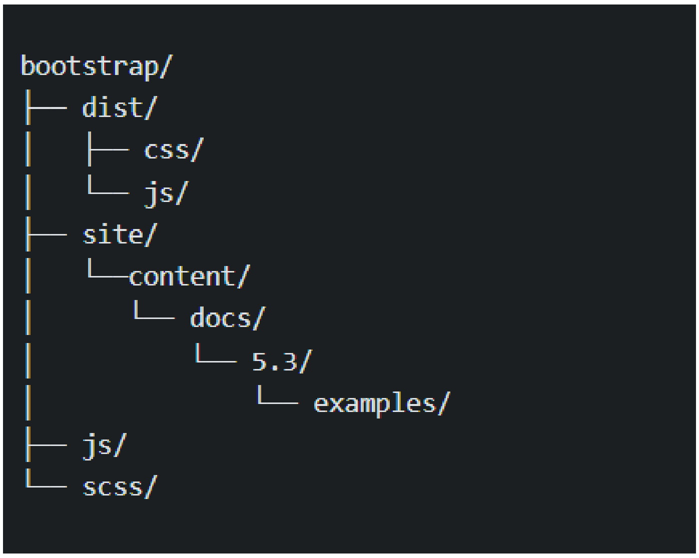
4. Extract the ZIP File:
   After the download is complete, locate the ZIP file on your computer and extract its contents to a folder of your choice.



This PC › New Volume (D:) › AWT › bootstrap-5.2.3-dist.zip

Name	Type	Compressed size	Password p...	Size
bootstrap-5.2.3-dist	File folder			

5. Access Bootstrap Files:
   Inside the extracted folder, you will find the necessary Bootstrap CSS and JavaScript files that you can include in your HTML documents.



```
bootstrap/
├── dist/
│   ├── css/
│   └── js/
├── site/
│   └──content/
│      └── docs/
│          └── 5.3/
│              └── examples/
├── js/
└── scss/
```

Bootstrap Versions

Bootstrap 5 (released 2021) is the newest version of Bootstrap (released 2013); with new components, faster stylesheet and more responsiveness.

Bootstrap 5 supports the latest, stable releases of all major browsers and platforms. However, Internet Explorer 11 and down is not supported.

The main differences between Bootstrap 5 and Bootstrap 3 & 4, is that Bootstrap 5 has switched to vanilla JavaScript instead of jQuery.

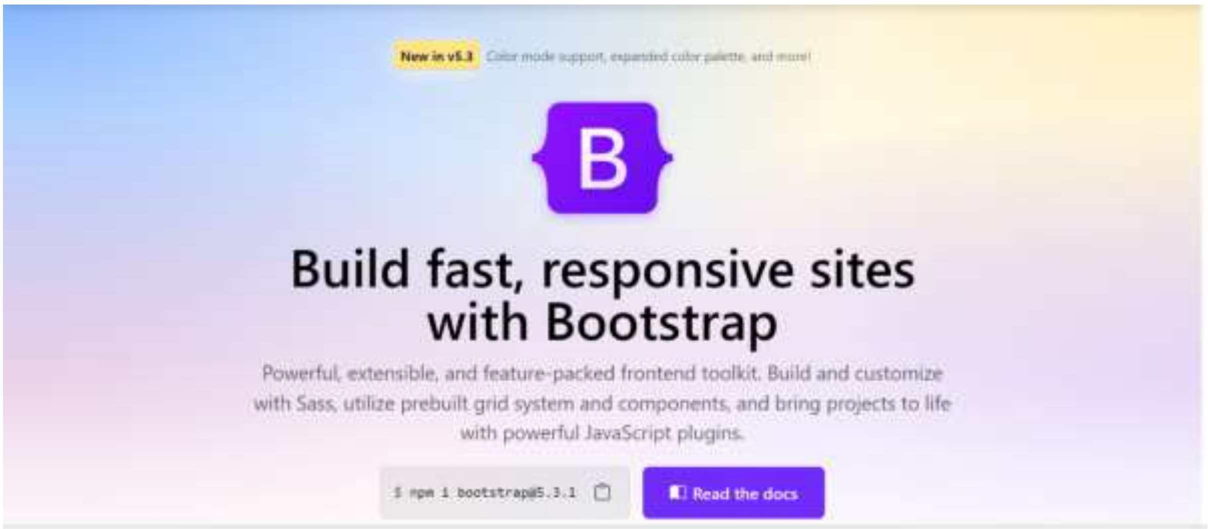
CDN or Content Delivery Network is a content distribution network of servers.

Bootstrap min V/S without min:

They have exactly the same function, but the . min. css version has been minified, meaning all whitespace has been removed to reduce file size and increase speed.

Bootstrap CDN	Bootstrap Download
Bootstrap CDN usage leads to faster loading time due to caching and geographically close servers of the CDN network.	The loading time will remain the same when we use Bootstrap Download.

CDN might be down sometimes or due to proxy settings, or security policies, the CDN link may not work.	Local copy has no dependency on CDN links and will always be up.
It reduces the amount of bandwidth used by your server.	Server bandwidth remains the same.



Build fast, responsive sites with Bootstrap

Powerful, extensible, and feature-packed frontend toolkit. Build and customize with Sass, utilize prebuilt grid system and components, and bring projects to life with powerful JavaScript plugins.

```
$ npm i bootstrap@5.3.1
```

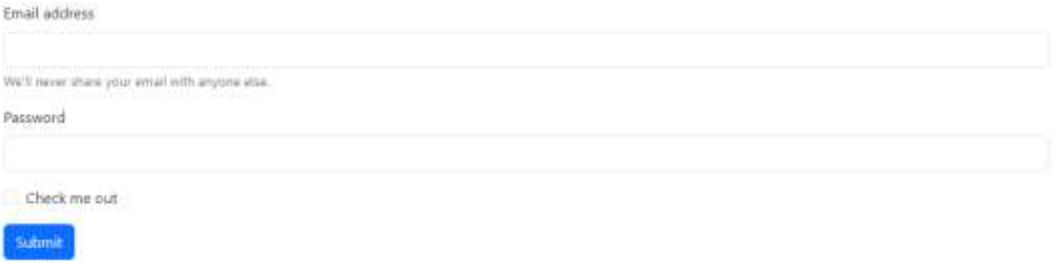
Basic HTML elements:

Form:

```
  <!DOCTYPE html>
<html lang="en">


<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-
4bw+/aepP/YC94hEpVNVgiZdgIC5+VKNBQNGCHeKRQN+PtmoHDEXuppvnDJzQIu9"
crossorigin="anonymous">
  <title>Form</title>
</head>


<body>
  <div class="container">
```

```
    <form>
          <div class="mb-3">
            <label for="exampleInputEmail1" class="form-label">Email address</label>
            <input type="email" class="form-control" id="exampleInputEmail1" aria-
describedby="emailHelp">
            <div id="emailHelp" class="form-text">We'll never share your email with anyone else.</div>
          </div>
          <div class="mb-3">
            <label for="exampleInputPassword1" class="form-label">Password</label>
            <input type="password" class="form-control" id="exampleInputPassword1">
          </div>
          <div class="mb-3 form-check">
            <input type="checkbox" class="form-check-input" id="exampleCheck1">
            <label class="form-check-label" for="exampleCheck1">Check me out</label>
          </div>
          <button type="submit" class="btn btn-primary">Submit</button>
        </form>
</div>
   <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-
HwwvtgBNo3bZJJLYd8oVXjrBZt8cqVSpeBNS5n7C8IVInixGAoxmnlMuBnhbgrkm"
    crossorigin="anonymous"></script>


 </body>


 </html>
```

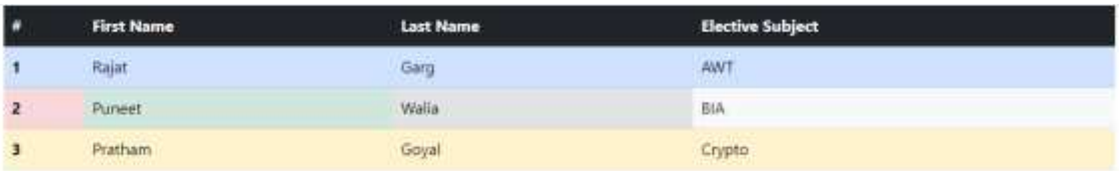
OUTPUT:



We'll never share your email with anyone else.

TABLE:

```
<!DOCTYPE html>
  <html lang="en">


  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link              href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/css/bootstrap.min.css"
rel="stylesheet"
      integrity="sha384-
4bw+/aepP/YC94hEpVNVgiZdgIC5+VKNBQNGCHeKRQN+PtmoHDEXuppvnDJzQIu9"
crossorigin="anonymous">
    <title>Form</title>
  </head>


  <body>
    <div class="container">
     <table class="table">
        <thead class="table-dark">
          <tr>
            <th scope="col">#</th>
            <th scope="col">First Name</th>
            <th scope="col">Last Name</th>
            <th scope="col">Elective Subject</th>
          </tr>
        </thead>
        <tbody>
          <tr class="table-primary">
            <th scope="row">1</th>
            <td>Rajat</td>
            <td>Garg</td>
            <td>AWT</td>
          </tr>
          <tr class="table-light">
            <th scope="row" class="table-danger">2</th>
            <td class="table-success">Puneet</td>
            <td class="table-secondary">Walia</td>
            <td>BIA</td>
          </tr>
          <tr class="table-warning">
            <th scope="row ">3</th>
            <td >Pratham</td>
            <td >Goyal</td>
            <td>Crypto</td>
          </tr>
        </tbody>
      </table>

    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/js/bootstrap.bundle.min.js"
      integrity="sha384-
HwwvtgBNo3bZJJLYd8oVXjrBZt8cqVSpeBNS5n7C8IVInixGAoxmnlMuBnhbgrkm"
      crossorigin="anonymous"></script>

  </body>

  </html>
```



#	First Name	Last Name	Elective Subject
1	Rajat	Garg	AWT
2	Puneet	Walia	BIA
3	Pratham	Goyal	Crypto

Experiment Number 2: To understand and deploy the multicolumn grid layout of Bootstrap.

Break Points:

Breakpoints are customizable widths that determine how your responsive layout behaves across device or viewport sizes in Bootstrap.

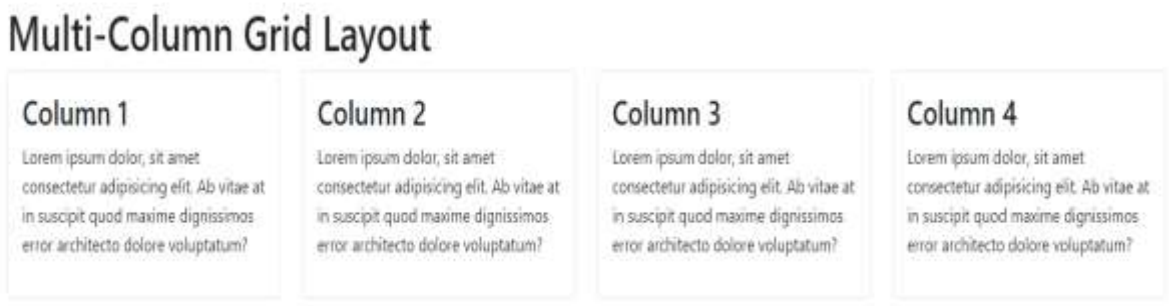
Breakpoint	Class infix	Dimensions
Extra small	*None*	<576px
Small	sm	≥576px
Medium	md	≥768px
Large	lg	≥992px
Extra large	xl	≥1200px
Extra extra large	xxl	≥1400px

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Multi-Column Grid Layout</title>
    <link rel="stylesheet" href="bootstrap-5.0.2-dist\css\bootstrap.css">
</head>
<body>
  <div class="container">

                <h1>Multi-Column Grid Layout</h1>
      <div class="row">
          <div class="col-lg-3 col-md-4 col-6">
              <div class="card">
                  <div class="card-body">
                      <h3>Column 1</h3>
                      <p>Lorem ipsum dolor, sit amet consectetur adipisicing
elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore
voluptatum?</p>
                  </div>
              </div>
          </div>
          <div class="col-lg-3 col-md-4 col-6">
              <div class="card">
                  <div class="card-body">
                      <h3>Column 2</h3>
```

```
                    <p>Lorem ipsum dolor, sit amet consectetur adipisicing
elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore
voluptatum?</p>
                </div>
            </div>
        </div>
        <div class="col-lg-3 col-md-4 col-6">
            <div class="card">
                <div class="card-body">
                    <h3>Column 3</h3>
                    <p>Lorem ipsum dolor, sit amet consectetur adipisicing
elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore
voluptatum?</p>
                </div>
            </div>
        </div>
        <div class="col-lg-3 col-md-4 col-6">
            <div class="card">
                <div class="card-body">
                    <h3>Column 4</h3>

                    <p>Lorem ipsum dolor, sit amet consectetur adipisicing
elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore
voluptatum?</p>
                </div>
            </div>
        </div>
    </div>
  </div>
</body>
</html>
```

Output

For large Screen size there are 4 columns



Multi-Column Grid Layout

Column 1

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

Column 2

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

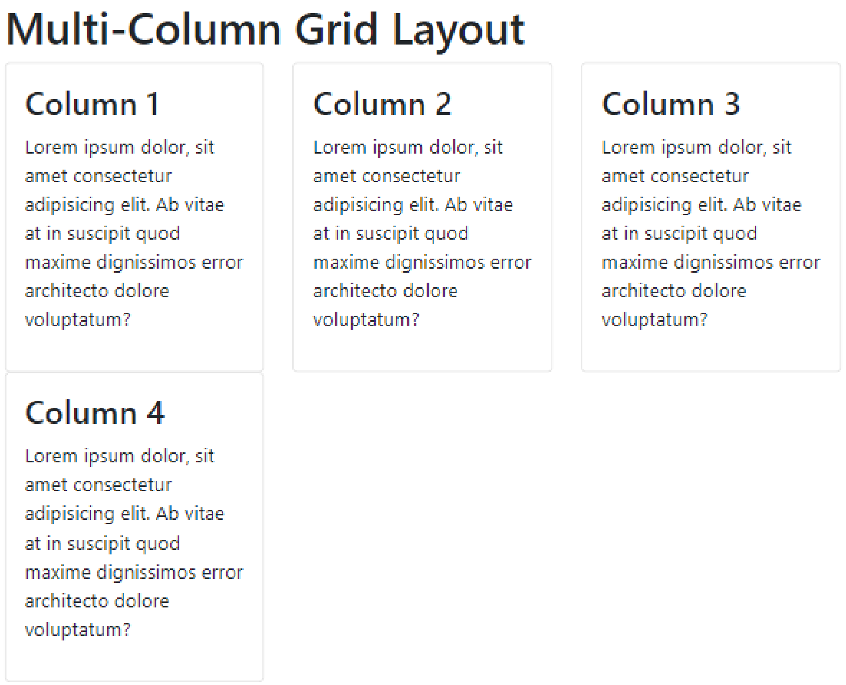
Column 3

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

Column 4

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

For medium Screen there are 3 columns



Multi-Column Grid Layout

Column 1

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

Column 2

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

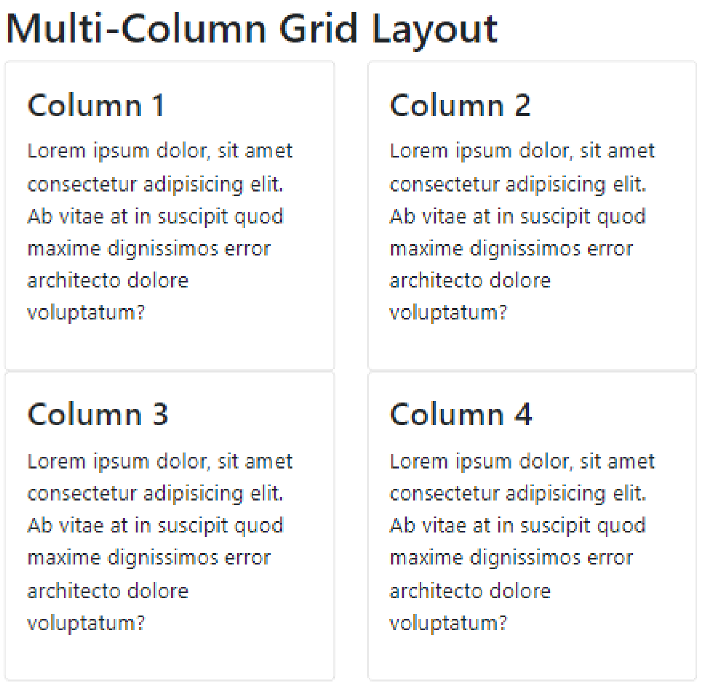
Column 3

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

Column 4

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

For Small Screen size there are 2 columns



Multi-Column Grid Layout

Column 1

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

Column 2

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

Column 3

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

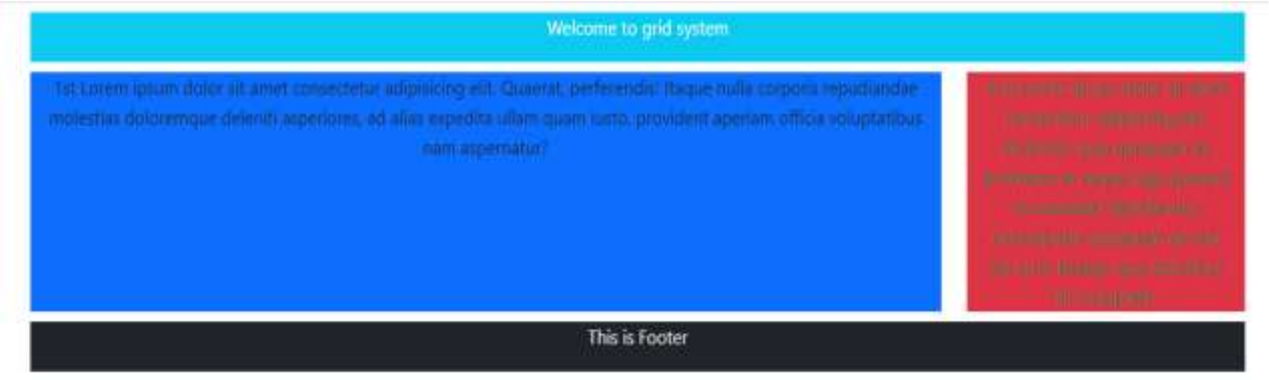
Column 4

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Ab vitae at in suscipit quod maxime dignissimos error architecto dolore voluptatum?

Difference In Container and Container-Fluid

Container: The .container class provides a responsive fixed width container.

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/css/bootstrap.min.css" rel="stylesheet"
        integrity="sha384-4bw+/aepP/YC94hEpVNVgiZdgIC5+VKNBQNGCHeKRQN+PtmoHDEXuppvnDJzQIu9" crossorigin="anonymous">

    <title>Bootstrap</title>
</head>

<body>
    <div class="container my-2 ">

        <div class="row bg-info"><p class="text-center text-white">Welcome to grid system</p></div>
        <div class="row">
            <div class="col col-6 col-lg-9 bg-primary text-center me-3 my-2 ">
                1st Lorem ipsum dolor sit amet consectetur adipisicing elit. Quaerat, perferendis! Itaque nulla corporis repudiandae molestias doloremque deleniti asperiores, ad alias expedita ullam quam iusto, provident aperiam officia voluptatibus nam aspernatur?
            </div>

            <div class="col  bg-danger text-center my-2 ms-2 text-success">

                2nd Lorem ipsum dolor sit amet consectetur adipisicing elit. Distinctio quia
quisquam ex architecto et eaque fuga aperiam recusandae! Repellendus consequatur
quisquam ea rem est error beatae, quo doloribus illo voluptate.


            </div>

        </div>

        <div class="row bg-dark"><p class="text-center text-light">This is
Footer</p></div>




    </div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/js/bootstrap.bundle.min.js"

        integrity="sha384-
HwwvtgBNo3bZJJLYd8oVXjrBZt8cqVSpeBNS5n7C8IVInixGAoxmnlMuBnhbgrkm"

        crossorigin="anonymous"></script>

</body>


</html>
```



Welcome to grid system

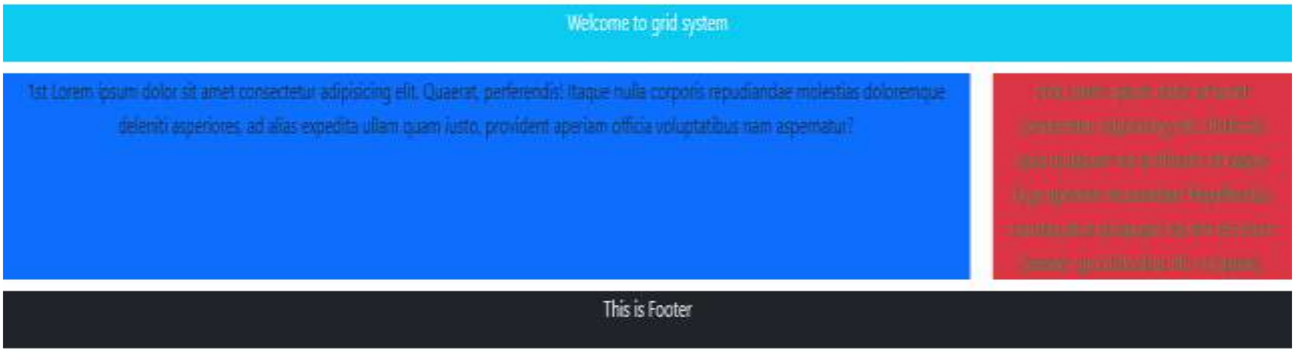
1st Lorem ipsum dolor sit amet consectetur adipisicing elit. Quaerat, perferendis! Itaque nulla corporis repudiandae molestias doloremque deleniti asperiores, ad alias expedita ullam quam iusto, provident aperiam officia voluptatibus nam aspernatur?

2nd Lorem ipsum dolor sit amet consectetur adipisicing elit. Distinctio quia quisquam ex architecto et eaque fuga aperiam recusandae! Repellendus consequatur quisquam ea rem est error beatae, quo doloribus illo voluptate.

This is Footer

Container-Fluid: The .container-fluid class provides a full width container, spanning the entire width of the viewport.

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/css/bootstrap.min.css" rel="stylesheet"
        integrity="sha384-4bw+/aepP/YC94hEpVNVgiZdgIC5+VKNBQNGCHeKRQN+PtmoHDEXuppvnDJzQIu9" crossorigin="anonymous">

    <title>Bootstrap</title>
</head>

<body>
    <div class="container-fluid my-2 ">

        <div class="row bg-info"><p class="text-center text-white">Welcome to grid system</p></div>
        <div class="row">
            <div class="col col-6 col-lg-9 bg-primary text-center me-3 my-2 ">
                1st Lorem ipsum dolor sit amet consectetur adipisicing elit. Quaerat, perferendis! Itaque nulla corporis repudiandae molestias doloremque deleniti asperiores, ad alias expedita ullam quam iusto, provident aperiam officia voluptatibus nam aspernatur?

            </div>
            <div class="col  bg-danger text-center my-2 ms-2 text-success">
        2nd Lorem ipsum dolor sit amet consectetur adipisicing elit. Distinctio quia
quisquam ex architecto et eaque fuga aperiam recusandae! Repellendus consequatur
quisquam ea rem est error beatae, quo doloribus illo voluptate.

      </div>

    </div>

    <div class="row bg-dark"><p class="text-center text-light">This is
Footer</p></div>



  </div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/js/bootstrap.bundle.min.js"

    integrity="sha384-
HwwvtgBNo3bZJJLYd8oVXjrBZt8cqVSpeBNS5n7C8IVInixGAoxmnlMuBnhbgrkm"

    crossorigin="anonymous"></script>



</body>

</html>
```



Welcome to grid system

1st Lorem ipsum dolor sit amet consectetur adipisicing elit. Quaerat, perferendis! Itaque nulla corporis repudiandae molestias doloremque deleniti asperiores, ad alias expedita ullam quam iusto, provident aperiam officia voluptatibus nam aspernatur?

2nd Lorem ipsum dolor sit amet consectetur adipisicing elit. Distinctio quia quisquam ex architecto et eaque fuga aperiam recusandae! Repellendus consequatur quisquam ea rem est error beatae, quo doloribus illo voluptate.

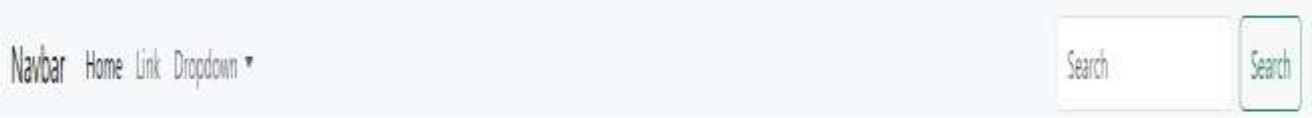
This is Footer

Experiment Number 3: To deploy different types of buttons, progress bars, modals and navigation bars using Bootstrap.

Navigation Bar: The navigation bar with Bootstrap is as follows:

```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Navbar</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Link</a>
        </li>
        <li class="nav-item dropdown">
          <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
            Dropdown
          </a>
          <ul class="dropdown-menu">
            <li><a class="dropdown-item" href="#">Action</a></li>
            <li><a class="dropdown-item" href="#">Another action</a></li>
```

```
         <li><hr class="dropdown-divider"></li>
         <li><a class="dropdown-item" href="#">Something else here</a></li>
        </ul>
       </li>

      </ul>
      <form class="d-flex" role="search">
       <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
       <button class="btn btn-outline-success" type="submit">Search</button>
      </form>
     </div>
    </div>
   </nav>
```

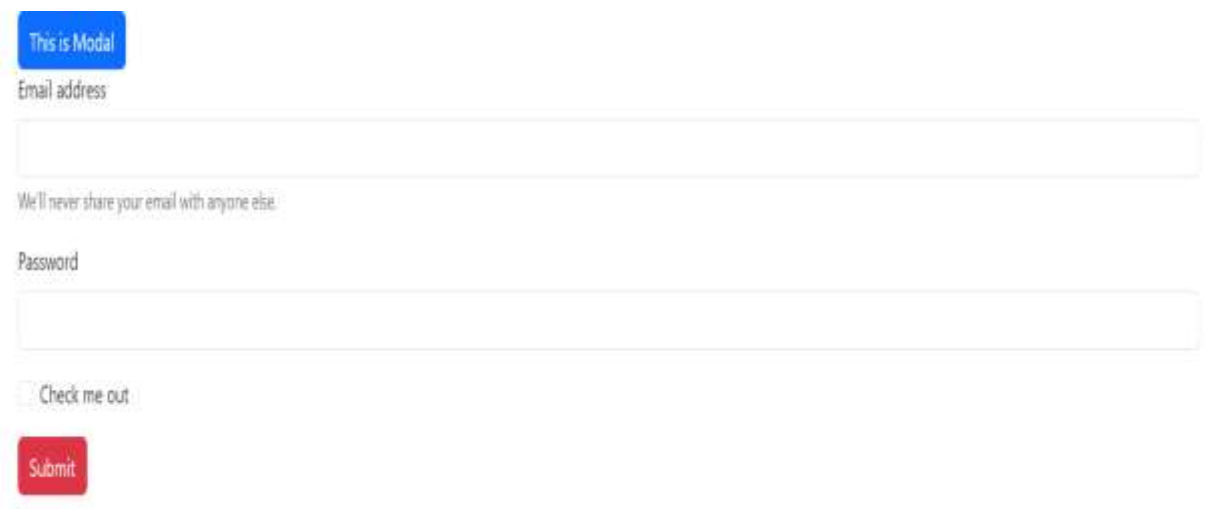


Modals and Buttons:

```
<div class="container">
    <!-- Button trigger modal -->
    <button type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#exampleModal">
      This is Modal
    </button>

    <!-- Modal -->
    <div class="modal fade" id="exampleModal" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
```

```
      <div class="modal-dialog">
        <div class="modal-content">
          <div class="modal-header">
            <h1 class="modal-title fs-5" id="exampleModalLabel">Modal title</h1>
            <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
          </div>
          <div class="modal-body">
            ...
          </div>
          <div class="modal-footer">
            <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
            <button type="button" class="btn btn-primary">Save changes</button>
          </div>
        </div>
      </div>
    </div>


    <form>
     <div class="mb-3">
      <label for="exampleInputEmail1" class="form-label">Email address</label>
      <input type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp">
      <div id="emailHelp" class="form-text">We'll never share your email with anyone else.</div>
     </div>
```

```
  <div class="mb-3">
    <label for="exampleInputPassword1" class="form-label">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword1">
  </div>
  <div class="mb-3 form-check">
    <input type="checkbox" class="form-check-input" id="exampleCheck1">
    <label class="form-check-label" for="exampleCheck1">Check me out</label>
  </div>
  <button type="submit" class="btn btn-danger">Submit</button>
</form>
```



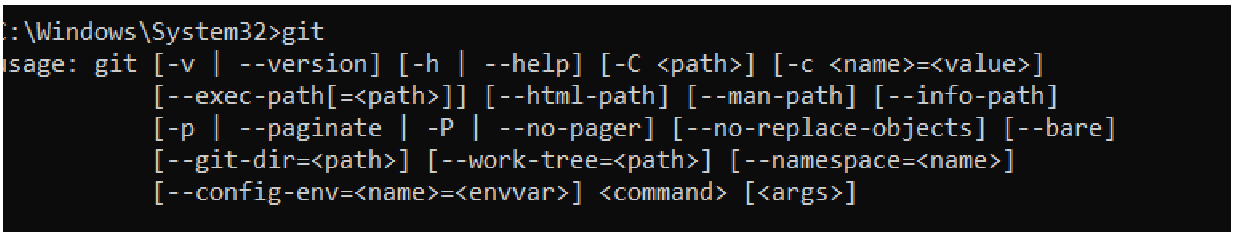
We'll never share your email with anyone else.

Experiment Number 4: To create and setup the Git repository on Bitbucket or github using SSH.

Step 1: Install Git

Download and install Git from the official website: https://git-scm.com/download/win

After writing command ***git*** this is shown on cmd.



```
:\Windows\System32>git
sage: git [-v | --version] [-h | --help] [-C <path>] [-c <name>=<value>]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           [--config-env=<name>=<envvar>] <command> [<args>]
```

Step 2: Configure Git

Set your Git username and email by running these commands in your terminal:

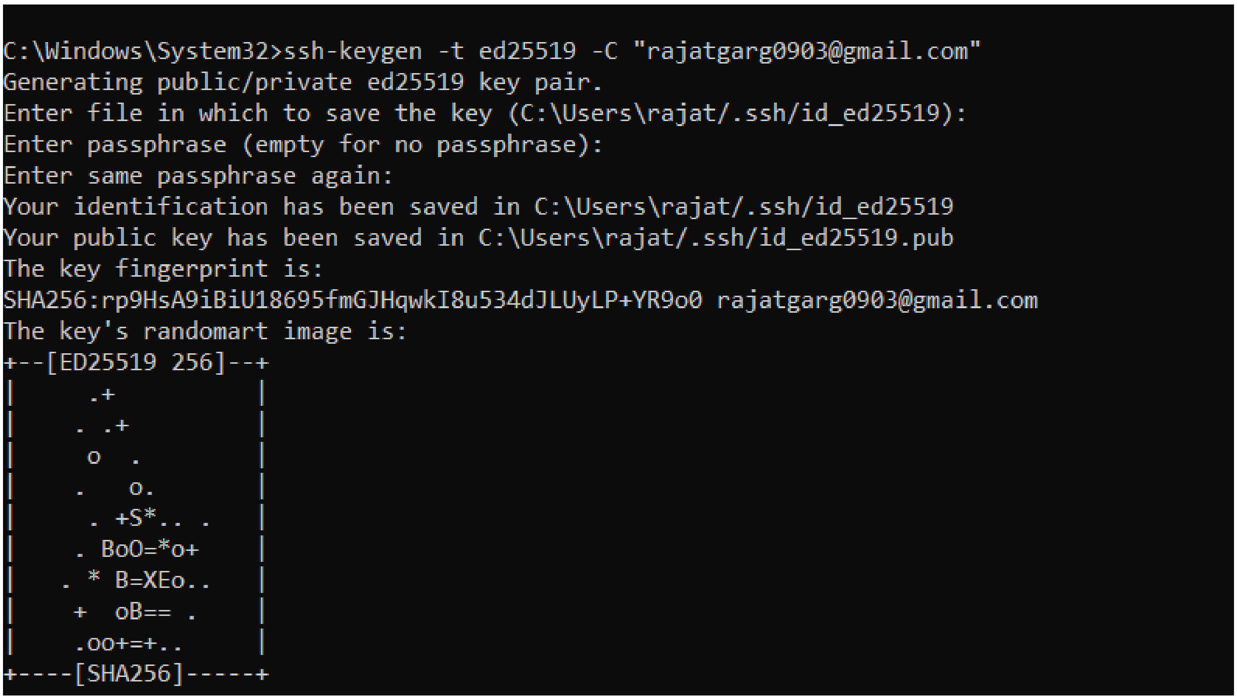
git config --global user.name "rajat-28"

git config --global user.email “rajatgarg0903@gmail.com”

Step 3: Generate SSH Key

Generate an SSH key to securely connect to GitHub:

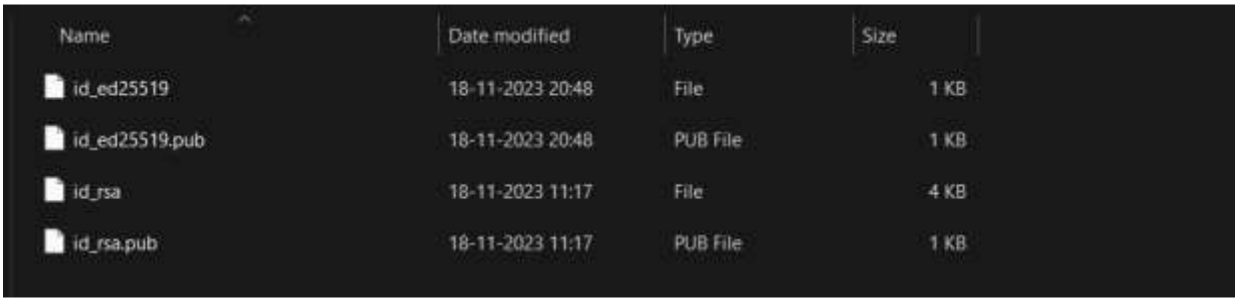
ssh-keygen -t ed25519 -C “rajatgarg0903@gmail.com”



```
C:\Windows\System32>ssh-keygen -t ed25519 -C "rajatgarg0903@gmail.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (C:\Users\rajat/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\rajat/.ssh/id_ed25519
Your public key has been saved in C:\Users\rajat/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:rp9HsA9iBiU18695fmGJHqwkI8u534dJLUyLP+YR9o0 rajatgarg0903@gmail.com
The key's randomart image is:
+--[ED25519 256]--+
|     .+          |
|    . .+         |
|     o  .        |
|    .   o.       |
|     . +S*.. .   |
|    . BoO=*o+    |
|   . * B=XEo..   |
|    +  oB== .    |
|    .oo+=+..     |
+----[SHA256]-----+
```

This is the place where ***SSH Key is generated***

C:\Users\rajat\.ssh



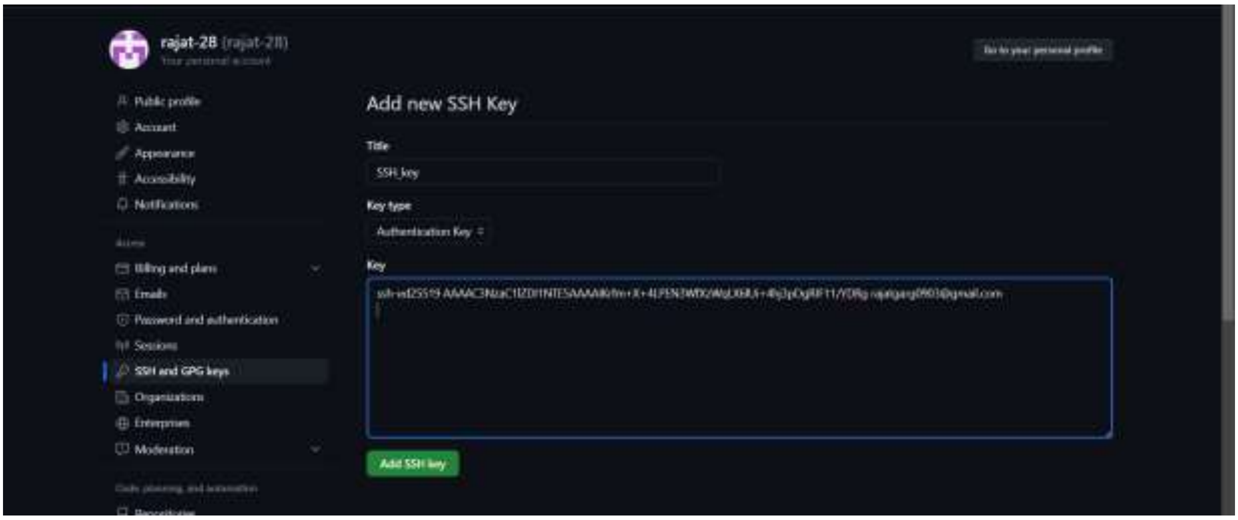
Name	Date modified	Type	Size
id_ed25519	18-11-2023 20:48	File	1 KB
id_ed25519.pub	18-11-2023 20:48	PUB File	1 KB
id_rsa	18-11-2023 11:17	File	4 KB
id_rsa.pub	18-11-2023 11:17	PUB File	1 KB

Step 5: Add SSH Key to GitHub

Copy your public SSH key to your clipboard:

cat ~/.ssh/id_ed25519.pub

Go to GitHub account settings, under "SSH and GPG keys," click "New SSH key," and paste public keythere.

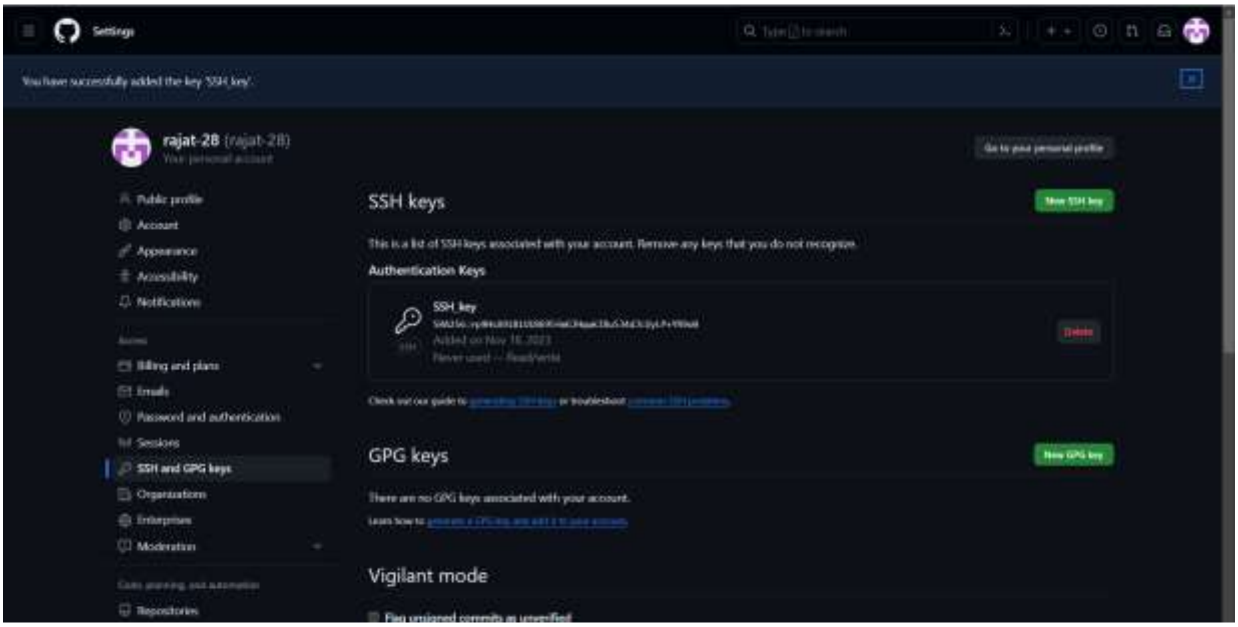


Add new SSH Key

Title: SSH_key

Key type: Authentication Key

Key: ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAA[illegible]@gmail.com



You have successfully added the key 'SSH key'.

SSH keys

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

Authentication Keys

SSH_key
SHA256:[illegible]
Added on Nov 16, 2023
Never used — Read/write

Check out our guide to generating SSH keys or troubleshoot common SSH problems.

GPG keys

There are no GPG keys associated with your account.

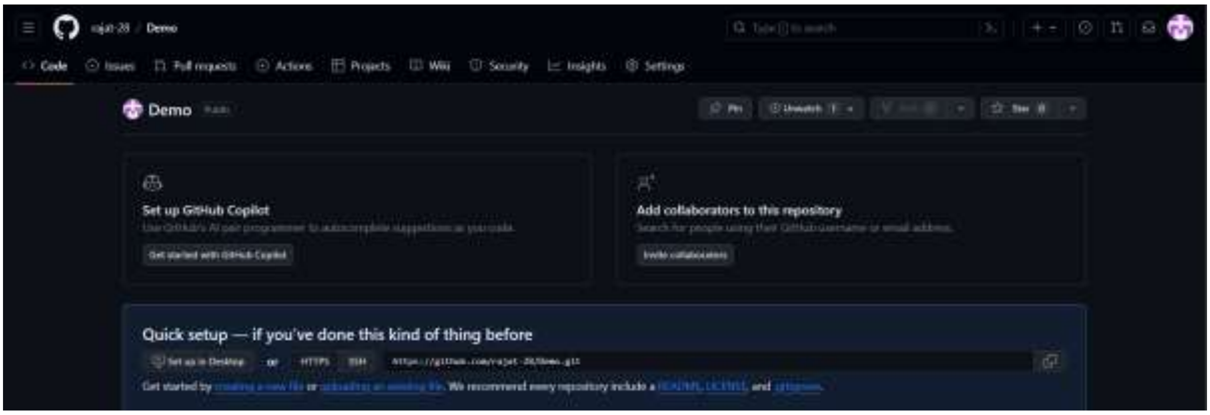
Learn how to generate a GPG key and add it to your account.

Vigilant mode

□ Flag unsigned commits as unverified

Step 6: Create a New Repository on GitHub

On GitHub, click the "+" icon in the top right corner and select "New repository." Fill in the repositoryname, description, and other options.



Set up GitHub Copilot

Use GitHub's AI pair programmer to autocomplete suggestions as you code.

Add collaborators to this repository

Search for people using their GitHub username or email address.

Quick setup — if you've done this kind of thing before

https://github.com/rajat-28/Demo.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

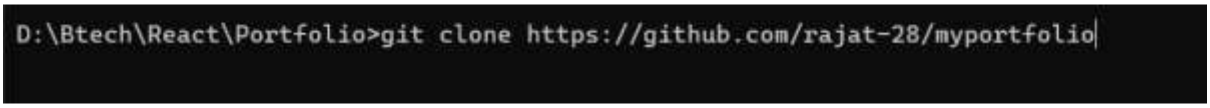
This is the SSH Key link of repository “Demo”.

Experiment Number 5: To perform push, c lone and patch operation to Git repository.

CLONE:

git clone <repository-url>

git clone https://github.com/rajat-28/myportfolio

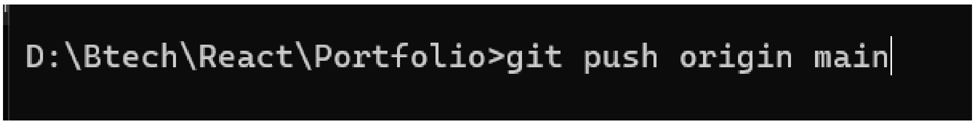


```
D:\Btech\React\Portfolio>git clone https://github.com/rajat-28/myportfolio
```

PUSH:

git push <remote-name> <branch-name>

git push origin main

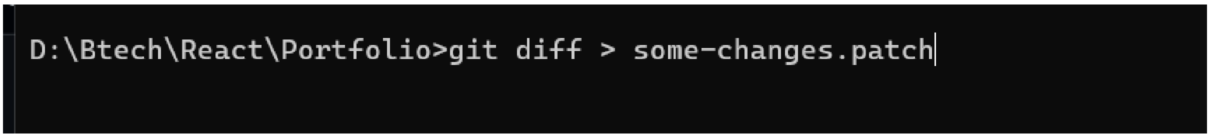


```
D:\Btech\React\Portfolio>git push origin main
```

PATCH:

1. Generate the patch:

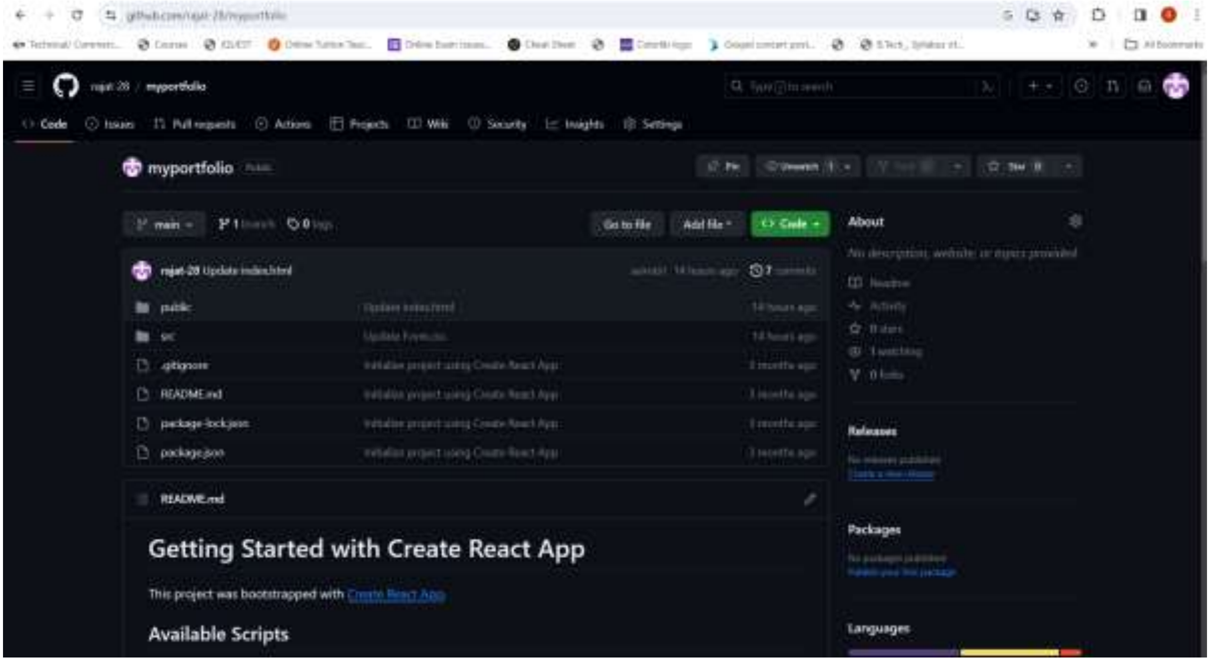
git diff > some-changes.patch



```
D:\Btech\React\Portfolio>git diff > some-changes.patch
```

2. Apply the diff:

git apply /path/to/some-changes.patch



rajat-28 Update index.html · 14 hours ago · 7 commits

public	Update index.html	14 hours ago
src	Update Form.css	14 hours ago
.gitignore	Initialize project using Create React App	3 months ago
README.md	Initialize project using Create React App	3 months ago
package-lock.json	Initialize project using Create React App	3 months ago
package.json	Initialize project using Create React App	3 months ago

README.md

Getting Started with Create React App

This project was bootstrapped with Create React App.

Available Scripts

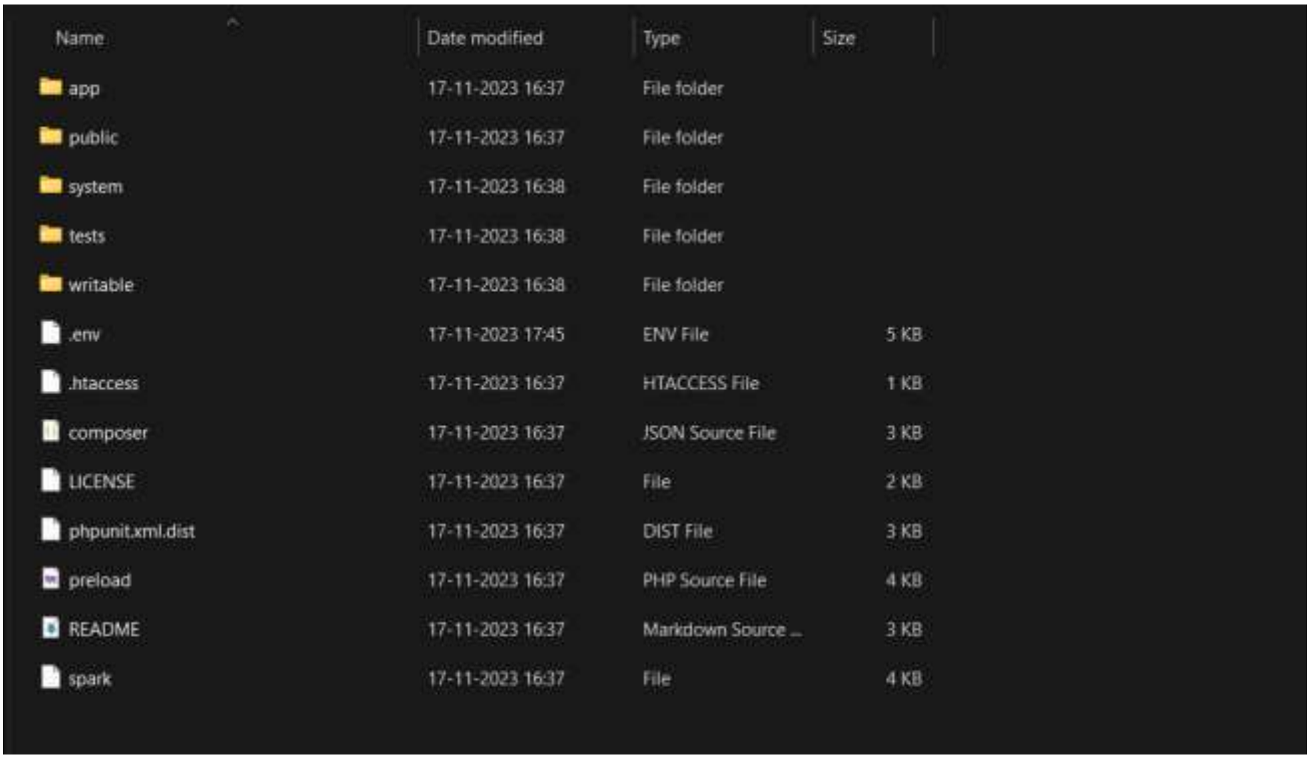
Experiment Number 6: To install and setup the CodeIgniter Framework and to understand its MVC architecture.

Installation with Composer:

In terminal go to the root folder of project and type the command

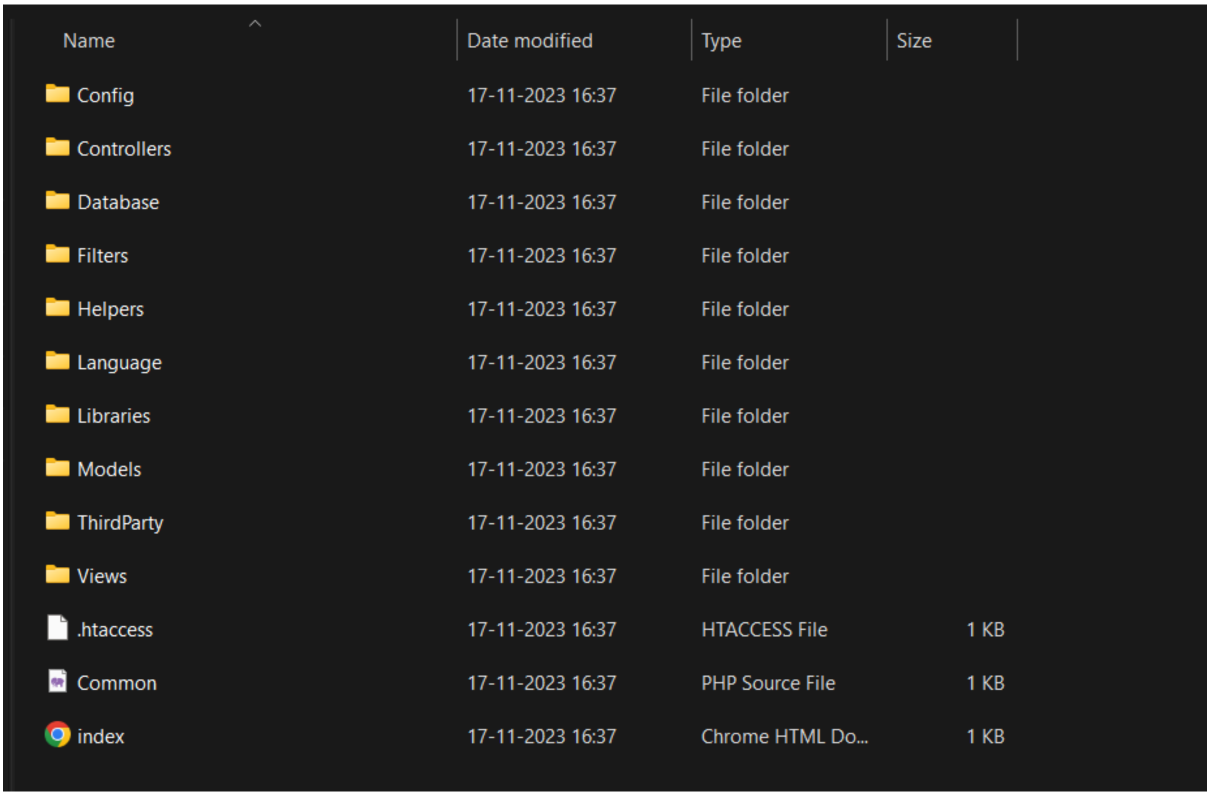
composer create-project codeigniter4/appstarter project-name

These files will be created:



Name	Date modified	Type	Size
app	17-11-2023 16:37	File folder	
public	17-11-2023 16:37	File folder	
system	17-11-2023 16:38	File folder	
tests	17-11-2023 16:38	File folder	
writable	17-11-2023 16:38	File folder	
.env	17-11-2023 17:45	ENV File	5 KB
.htaccess	17-11-2023 16:37	HTACCESS File	1 KB
composer	17-11-2023 16:37	JSON Source File	3 KB
LICENSE	17-11-2023 16:37	File	2 KB
phpunit.xml.dist	17-11-2023 16:37	DIST File	3 KB
preload	17-11-2023 16:37	PHP Source File	4 KB
README	17-11-2023 16:37	Markdown Source ...	3 KB
spark	17-11-2023 16:37	File	4 KB

As CodeIgniter is based on MVC architecture , In App folder there are different folders of Model, View and Controllers.



Name	Date modified	Type	Size
Config	17-11-2023 16:37	File folder	
Controllers	17-11-2023 16:37	File folder	
Database	17-11-2023 16:37	File folder	
Filters	17-11-2023 16:37	File folder	
Helpers	17-11-2023 16:37	File folder	
Language	17-11-2023 16:37	File folder	
Libraries	17-11-2023 16:37	File folder	
Models	17-11-2023 16:37	File folder	
ThirdParty	17-11-2023 16:37	File folder	
Views	17-11-2023 16:37	File folder	
.htaccess	17-11-2023 16:37	HTACCESS File	1 KB
Common	17-11-2023 16:37	PHP Source File	1 KB
index	17-11-2023 16:37	Chrome HTML Do...	1 KB

Experiment Number 7: To construct a simple login page web application to authenticate users using CodeIgniter Framework and also perform CURD operations.

CURD stands for Create, Update, Read and Delete.

- Create a new project using composer.
- Start the Xampp server
- Create a database named as demo

We will create a form for user details in which users can add their details and can update, delete and view those details.

1.Create: This will take input from user in a form.

2.Update: This will enable user to update the details.

3.Read: This will display the details of users in a table.

4.Delete: This will enable user to delete their details.

Firstly create views:

1.add_user.php:

```
<!DOCTYPE html>

<html>

<head>

  <title>Codeigniter 4 Add User With Validation Demo</title>

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

  <style>

    .container {

      max-width: 500px;

    }

    .error {

      display: block;
      padding-top: 5px;
      font-size: 14px;
      color: red;
    }
  </style>
</head>
<body>
  <div class="container mt-5">
    <form method="post" id="add_create" name="add_create"
    action="<?= site_url('/submit-form') ?>">
      <div class="form-group">
        <label>Name</label>
        <input type="text" name="name" class="form-control">
      </div>
      <div class="form-group">
        <label>Email</label>
        <input type="text" name="email" class="form-control">
      </div>
      <div class="form-group">
        <button type="submit" class="btn btn-primary btn-block">Update
Data</button>
      </div>
    </form>
  </div>
  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery-
validate/1.19.2/jquery.validate.js"></script>
```

```
  <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery-validate/1.19.2/additional-methods.min.js"></script>
  <script>
    if ($("#add_create").length > 0) {
      $("#add_create").validate({
        rules: {
          name: {
            required: true,
          },
          email: {
            required: true,
            maxlength: 60,
            email: true,
          },
        },
        messages: {
          name: {
            required: "Name is required.",
          },
          email: {
            required: "Email is required.",
            email: "It does not seem to be a valid email.",
            maxlength: "The email should be or equal to 60 chars.",
          },
        },
      })
    }
  </script>
```

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

2. edit_view.php:

```
<!DOCTYPE html>
<html>
<head>
  <title>Codeigniter 4 CRUD - Edit User Demo</title>
  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css"
>
  <style>
    .container {
      max-width: 500px;
    }
    .error {
      display: block;
      padding-top: 5px;
      font-size: 14px;
      color: red;
    }
  </style>
</head>
<body>
  <div class="container mt-5">
    <form method="post" id="update_user" name="update_user"
    action="<?= site_url('/update') ?>">
      <input type="hidden" name="id" id="id" value="<?php echo
$user_obj['id']; ?>">
```

```
      <div class="form-group">
        <label>Name</label>
        <input type="text" name="name" class="form-control" value="<?php echo
$user_obj['name']; ?>">
      </div>
      <div class="form-group">
        <label>Email</label>
        <input type="email" name="email" class="form-control" value="<?php
echo $user_obj['email']; ?>">
      </div>
      <div class="form-group">
        <button type="submit" class="btn btn-danger btn-block">Save
Data</button>
      </div>
    </form>
  </div>
  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery-
validate/1.19.2/jquery.validate.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery-
validate/1.19.2/additional-methods.min.js"></script>
  <script>
    if ($("#update_user").length > 0) {
      $("#update_user").validate({
        rules: {
          name: {
            required: true,
          },
          email: {
```

```
          required: true,
          maxlength: 60,
          email: true,
        },
      },
      messages: {
        name: {
          required: "Name is required.",
        },
        email: {
          required: "Email is required.",
          email: "It does not seem to be a valid email.",
          maxlength: "The email should be or equal to 60 chars.",
        },
      },
    })
  }
 </script>
</body>
</html>
```

3.user_view.php:

```
                                        <!doctype html>
<html lang="en">
 <head>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

```
  <title>Codeigniter 4 CRUD App Example - positronx.io</title>
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">
</head>
<body>
<div class="container mt-4">
    <div class="d-flex justify-content-end">
        <a href="<?php echo site_url('/user-form') ?>" class="btn btn-success mb-2">Add User</a>
                                                                        </div>
    <?php
     if(isset($_SESSION['msg'])){
        echo $_SESSION['msg'];
      }
     ?>
  <div class="mt-3">
     <table class="table table-bordered" id="users-list">
       <thead>
          <tr>
             <th>User Id</th>
             <th>Name</th>
             <th>Email</th>
             <th>Action</th>
          </tr>
       </thead>
       <tbody>
          <?php if($users): ?>
          <?php foreach($users as $user): ?>
          <tr>
            <td><?php echo $user['id']; ?></td>
            <td><?php echo $user['name']; ?></td>
            <td><?php echo $user['email']; ?></td>
            <td>
             <a href="<?php echo base_url('edit_view/'.$user['id']);?>" class="btn btn-primary btn-sm">Edit</a>
             <a href="<?php echo base_url('delete/'.$user['id']);?>" class="btn btn-danger btn-sm">Delete</a>
             <a href="<?php echo base_url('delete/'.$user['id']);?>" class="btn btn-danger btn-sm">Delete</a>
             </td>
          </tr>
         <?php endforeach; ?>
         <?php endif; ?>
       </tbody>
     </table>
  </div>
</div>


<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/1.10.21/css/jquery.dataTables.min.css">
<script type="text/javascript" src="https://cdn.datatables.net/1.10.21/js/jquery.dataTables.min.js"></script>
<script>
   $(document).ready( function () {
     $('#users-list').DataTable();
  } );
</script>
</body>
</html>
```

Models:

UserModel.php:

```
<?php
namespace App\Models;
use CodeIgniter\Model;
class UserModel extends Model
{
    protected $table = 'users';
    protected $primaryKey = 'id';

    protected $allowedFields = ['name', 'email'];
}
```

Controller:

UserCrud.php

```
<?php
namespace App\Controllers;
use App\Models\UserModel;
use CodeIgniter\Controller;
class UserCrud extends Controller
{
    // show users list
```

```
public function index(){
    $userModel = new UserModel();
    $data['users'] = $userModel->orderBy('id', 'DESC')->findAll();
    return view('user_view', $data);
}
// add user form
public function create(){
    return view('add_user');
}

// insert data
public function store() {
    $userModel = new UserModel();
    $data = [
        'name' => $this->request->getVar('name'),
        'email'  => $this->request->getVar('email'),
    ];
    $userModel->insert($data);
    return $this->response->redirect(site_url('/users-list'));
}
// show single user
public function singleUser($id = null){
    $userModel = new UserModel();
    $data['user_obj'] = $userModel->where('id', $id)->first();
    return view('edit_view', $data);
}
// update user data
```

```
    public function update(){
        $userModel = new UserModel();
        $id = $this->request->getVar('id');
        $data = [
            'name' => $this->request->getVar('name'),
            'email'  => $this->request->getVar('email'),
        ];
        $userModel->update($id, $data);
        return $this->response->redirect(site_url('/users-list'));
    }

    // delete user
    public function delete($id = null){
        $userModel = new UserModel();
        $data['user'] = $userModel->where('id', $id)->delete($id);
        return $this->response->redirect(site_url('/users-list'));
    }
}
```

Create the routes:

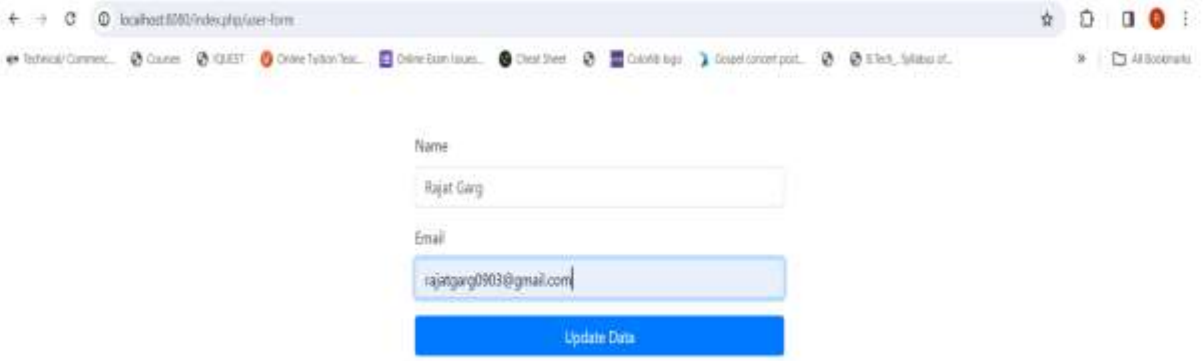
```
<?php

use CodeIgniter\Router\RouteCollection;

/**
 * @var RouteCollection $routes
 */
```

```
// CRUD RESTful Routes
$routes->get('/', 'UserCrud::index');
$routes->get('users-list', 'UserCrud::index');
$routes->get('user-form', 'UserCrud::create');
$routes->post('submit-form', 'UserCrud::store');
$routes->get('edit_view/(:num)', 'UserCrud::singleUser/$1');
$routes->post('update', 'UserCrud::update');
$routes->get('delete/(:num)', 'UserCrud::delete/$1');
```

In database.php add the database:

```
public array $default = [
    'DSN'         => '',
    'hostname'    => 'localhost',
    'username'    => 'root',
    'password'    => '',
    'database'    => 'demo',
    'DBDriver'    => 'MySQLi',
    'DBPrefix'    => '',
    'pConnect'    => false,
    'DBDebug'     => true,
    'charset'     => 'utf8',
    'DBCollat'    => 'utf8_general_ci',
    'swapPre'     => '',
    'encrypt'     => false,
    'compress'    => false,
    'strictOn'    => false,
    'failover'    => [],
    'port'         => 3306,
    'numberNative' => false,
];
```

Add User:

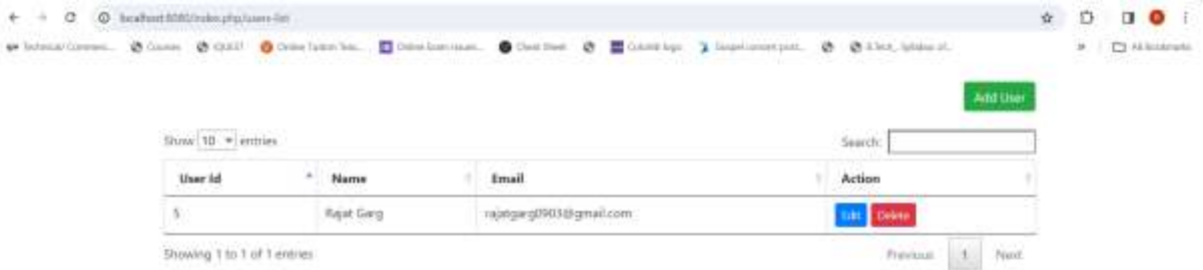


localhost:8080/index.php/user-form

Name: Rajat Garg

Email: rajatgarg0903@gmail.com

User View:



localhost:8080/index.php/users-list

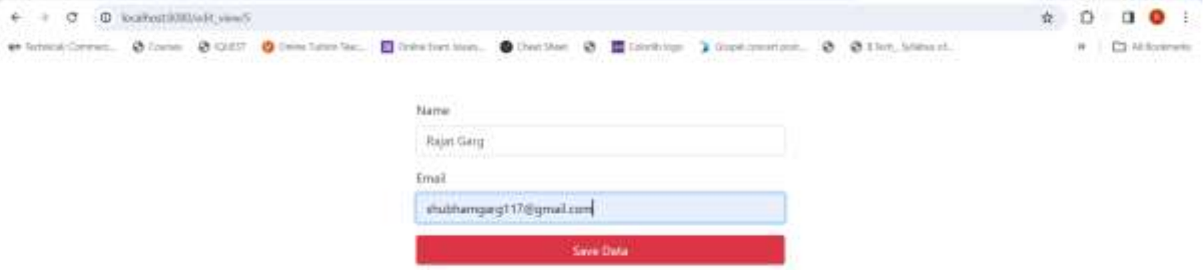
Show 10 entries

Search:

User Id	Name	Email	Action
5	Rajat Garg	rajatgarg0903@gmail.com	Edit Delete

Showing 1 to 1 of 1 entries

Update data:

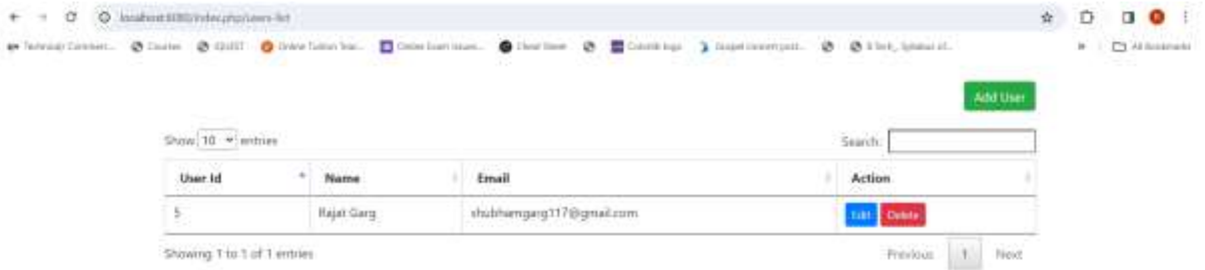


localhost:8080/edit_view/5

Name: Rajat Garg

Email: shubhamgarg117@gmail.com

Add new user:

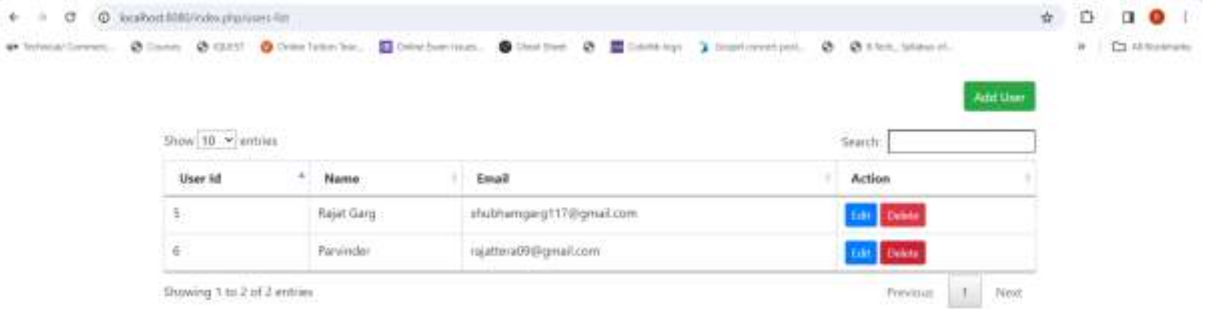


Show 10 entries

User Id	Name	Email	Action
5	Rajat Garg	shubhamgarg117@gmail.com	Edit Delete

Showing 1 to 1 of 1 entries

New user added:

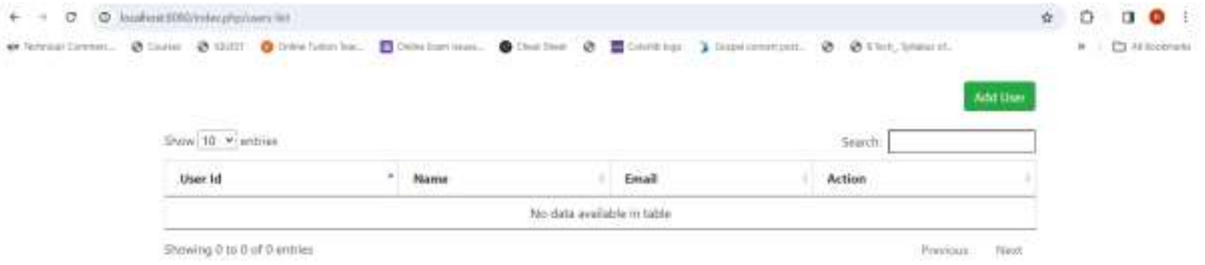


Show 10 entries

User Id	Name	Email	Action
5	Rajat Garg	shubhamgarg117@gmail.com	Edit Delete
6	Parvinder	rajattera09@gmail.com	Edit Delete

Showing 1 to 2 of 2 entries

Deleted the users:



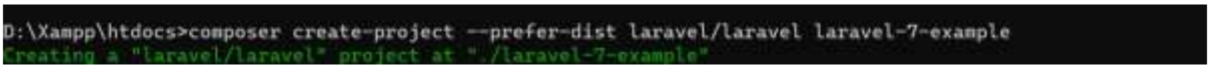
Show 10 entries

User Id	Name	Email	Action
No data available in table			

Showing 0 to 0 of 0 entries

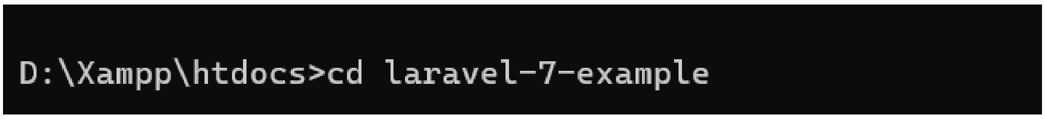
Experiment Number 8: To install and setup, configure the Laravel Framework.

Installing Laravel via composer:

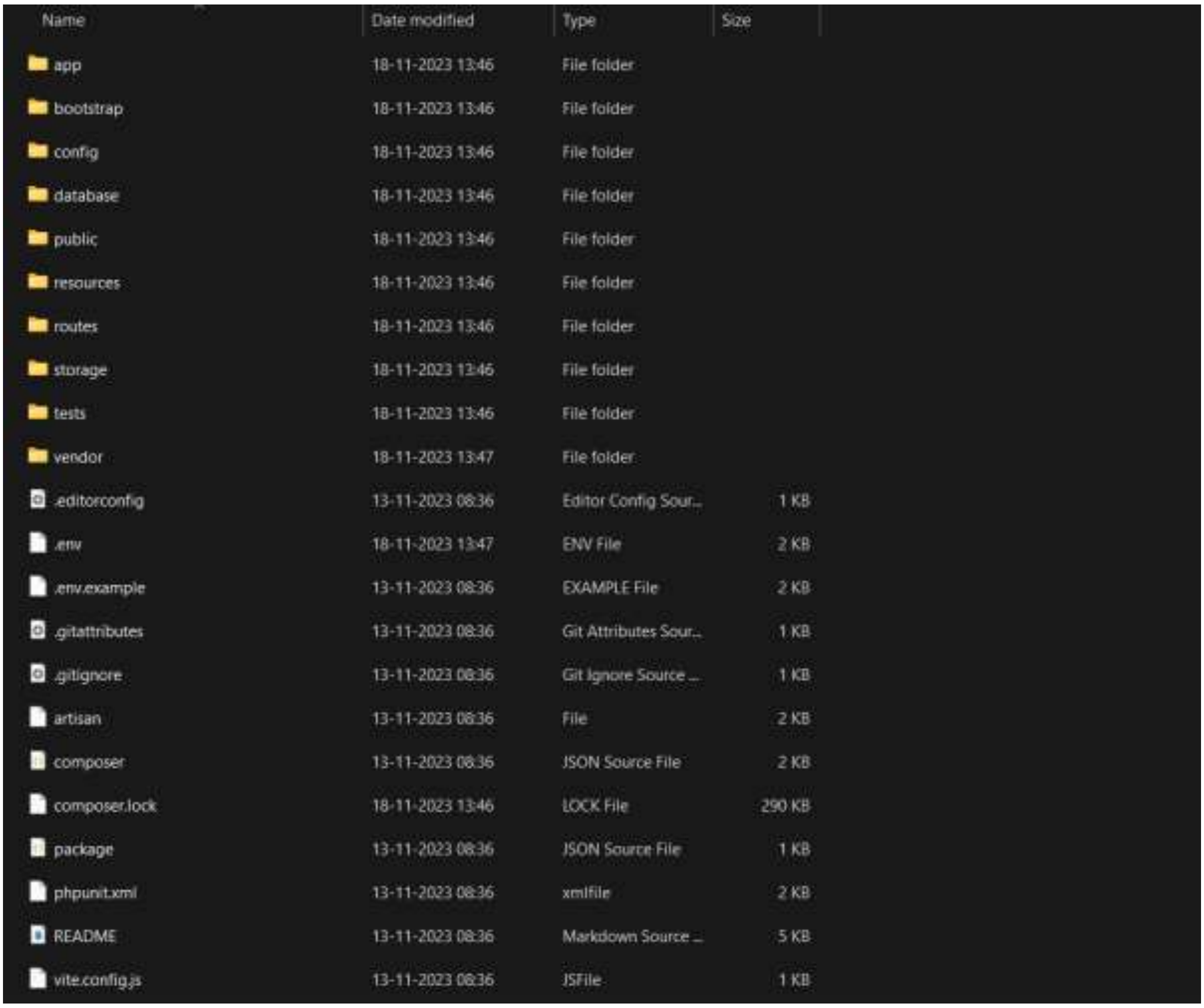


```
D:\Xampp\htdocs>composer create-project --prefer-dist laravel/laravel laravel-7-example
Creating a "laravel/laravel" project at "./laravel-7-example"
```

Then go to the specified project:

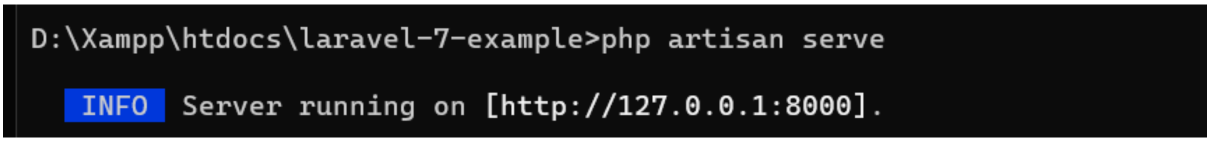


```
D:\Xampp\htdocs>cd laravel-7-example
```

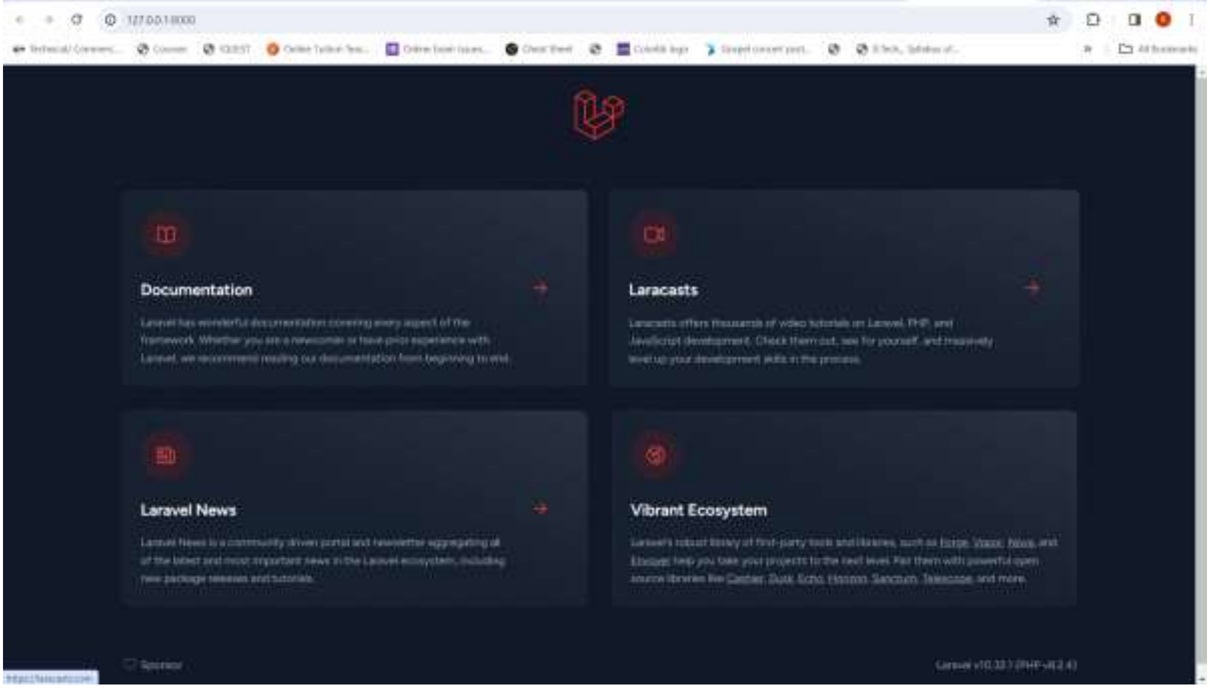


Name	Date modified	Type	Size
app	18-11-2023 13:46	File folder	
bootstrap	18-11-2023 13:46	File folder	
config	18-11-2023 13:46	File folder	
database	18-11-2023 13:46	File folder	
public	18-11-2023 13:46	File folder	
resources	18-11-2023 13:46	File folder	
routes	18-11-2023 13:46	File folder	
storage	18-11-2023 13:46	File folder	
tests	18-11-2023 13:46	File folder	
vendor	18-11-2023 13:47	File folder	
.editorconfig	13-11-2023 08:36	Editor Config Sour...	1 KB
.env	18-11-2023 13:47	ENV File	2 KB
.env.example	13-11-2023 08:36	EXAMPLE File	2 KB
.gitattributes	13-11-2023 08:36	Git Attributes Sour...	1 KB
.gitignore	13-11-2023 08:36	Git Ignore Source ...	1 KB
artisan	13-11-2023 08:36	File	2 KB
composer	13-11-2023 08:36	JSON Source File	2 KB
composer.lock	18-11-2023 13:46	LOCK File	290 KB
package	13-11-2023 08:36	JSON Source File	1 KB
phpunit.xml	13-11-2023 08:36	xmlfile	2 KB
README	13-11-2023 08:36	Markdown Source ...	5 KB
vite.config.js	13-11-2023 08:36	JSFile	1 KB

Start the development server:



```
D:\Xampp\htdocs\laravel-7-example>php artisan serve

  INFO  Server running on [http://127.0.0.1:8000].
```



127.0.0.1:8000

Documentation

Laravel has wonderful documentation covering every aspect of the framework. Whether you are a newcomer or have prior experience with Laravel, we recommend reading our documentation from beginning to end.

Laracasts

Laracasts offers thousands of video tutorials on Laravel, PHP, and JavaScript development. Check them out, see for yourself, and massively level up your development skills in the process.

Laravel News

Laravel News is a community driven portal and newsletter aggregating all of the latest and most important news in the Laravel ecosystem, including new package releases and tutorials.

Vibrant Ecosystem

Laravel's robust library of first-party tools and libraries, such as Forge, Vapor, Nova, and Envoyer help you take your projects to the next level. Pair them with powerful open source libraries like Cashier, Dusk, Echo, Horizon, Sanctum, Telescope, and more.

This is by default welcome page of Laravel

We can configure the files according to our requirements.

Experiment Number 9: To construct the any simple web application using Laravel Framework.

Firstly we will create the controller in http/Controllers/Controller.php

```
<?php

namespace App\Http\Controllers;

use Illuminate\Foundation\Auth\Access\AuthorizesRequests;
use Illuminate\Foundation\Validation\ValidatesRequests;
use Illuminate\Routing\Controller as BaseController;
use App\User;
use App\Http\Controllers\Controller;

class Controller extends BaseController
{
    public function display(){
        return view('welcome',['name'=>'Rajat']);
    }
}
```

Now we will create the routes in routes/web.php

```
<?php

use Illuminate\Support\Facades\Route;
use App\Http\Controllers\Controller;

Route::get('/', [Controller::class,'display']);
```

Now create the model Models/User.php

```
<?php

namespace App\Models;

// use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Illuminate\Notifications\Notifiable;
use Laravel\Sanctum\HasApiTokens;

class User extends Authenticatable
{
    use HasApiTokens, HasFactory, Notifiable;

    /**
     * The attributes that are mass assignable.
     *
     * @var array<int, string>
     */
    protected $fillable = [
        'name',
        'email',
        'password',
    ];

    /**
     * The attributes that should be hidden for serialization.
     *
     * @var array<int, string>
     */
    protected $hidden = [
        'password',
        'remember_token',
    ];

    /**
     * The attributes that should be cast.
     *
     * @var array<string, string>
     */
    protected $casts = [
        'email_verified_at' => 'datetime',
        'password' => 'hashed',
    ];
}
```

Now we will create views in Resources/view/welcome.blade.php

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <h1>Name is: <?php echo $name?></h1>
</body>
</html>
```