Full Stack Development

V Semester Lab Manual – 2022-23

Experiment 1: Perform Local Git Operations

Step 1: Install git from the official website.

Step 2: Once installed, launch the git bash in Windows.

Step 3: Configure the name and email using the following commands.

```
git config --global user.name <USERNAME>

git config --global user.email <EMAIL>
```

Step 3: Create a folder with some files and launch a command prompt.

Step 4: Navigate to the folder.

Use the following commands to perform basic git operations:

Command 1# Initialize the repository.

```
git init
```

Command 2# View the status of the repository.

```
git status
```

Command 3# ADD files to the staging area.

```
git add <filename>
```

Command 4# Commit changes to the repository.

```
git commit -m "MESSAGE"
```

Command 5# View and create branches.

```
git branch
```

Command 6# View the commit history.

```
git log
```

Command 7# Switch between branches

```
git checkout <branch name>
```

Command 8# Merge two branches

```
git merge <branch name>
```

Command 9# git remote command to manage remote repositories

```
git remote add origin < github repository_url>
```

Command 10# To create a new branch and switch to it

```
git switch -c <New-Branch-Name>
```

Experiment 2: Basics of GitHub operations such as creating an account, create push and pull the repository between GitHub and Git local repository.

Step 1: Open GitHub official website

Step 2: Create an account by clicking on signing up on GitHub.

Step 3: Enter your email and click on continue.

Step 4: Create a password for your account and click continue

Step 5: Enter a username and click on continue

Step 6: Verify your account by solving captcha and click on create account.

Step 7: Verify your email and you can now sign in with your email

Step 8: Now click on profile button go to your repositories

Step 9: Click on New to create a new repository

Step 10: Provide a name for your repository and check the 'Add Readme file'.

Step 11: Click on create repository

Step 12: Go to Desktop and create a folder with some dummy files.

Step 13: Launch command prompt and navigate to the created folder.

Note: You must add your name and email in git bash and the ssh keys to your GitHub account before performing any git operations.

Step 14: Use the following command to initialize repository

```
git init
```

Step 15: Now commit all the files using following command

```
git commit -m "This is first commit"
```

Step 16: Now add the remote link of GitHub to the local git folder to connect the folder with the GitHub repository and push the repository.

Step 17: The files will appear on the GitHub.

Step 18: Click on Add new file > Create a new file > provide a name to your file and write some content.

Step 19: Add a commit message and click on commit

Step 20: To get the files on our local system we just added perform pull operation with following command.

```
git pull
```

Experiment 3: Create a repository name min-project-1. Push the same to GitHub

Step 1 : Open browser and login to your GitHub account

Step 2: Go to your repository and click on new to create a new one name 'mini-project1 by adding Readme file.

Step 3: Go to Desktop and create a folder with some dummy files.

Step 4: Launch command prompt and navigate to the created folder.

Step 5: Use the following command to initialize repository

```
git init
```

Step 6: Now commit all the files using following command

```
git commit -m "This is first commit"
```

Step 7: Now add the remote link of GitHub to the local git folder to connect the folder with the GitHub repository and push the repository.

Step 8: The files will appear on the GitHub.

Step 9: Now click on profile button go to your repositories

Step 10: Click on New to create a new repository

Step 11: Provide a name for your repository and check the 'Add Readme file'.

Step 12: Click on create repository

Experiment 4: Create an account on Amazon Web Services (AWS)

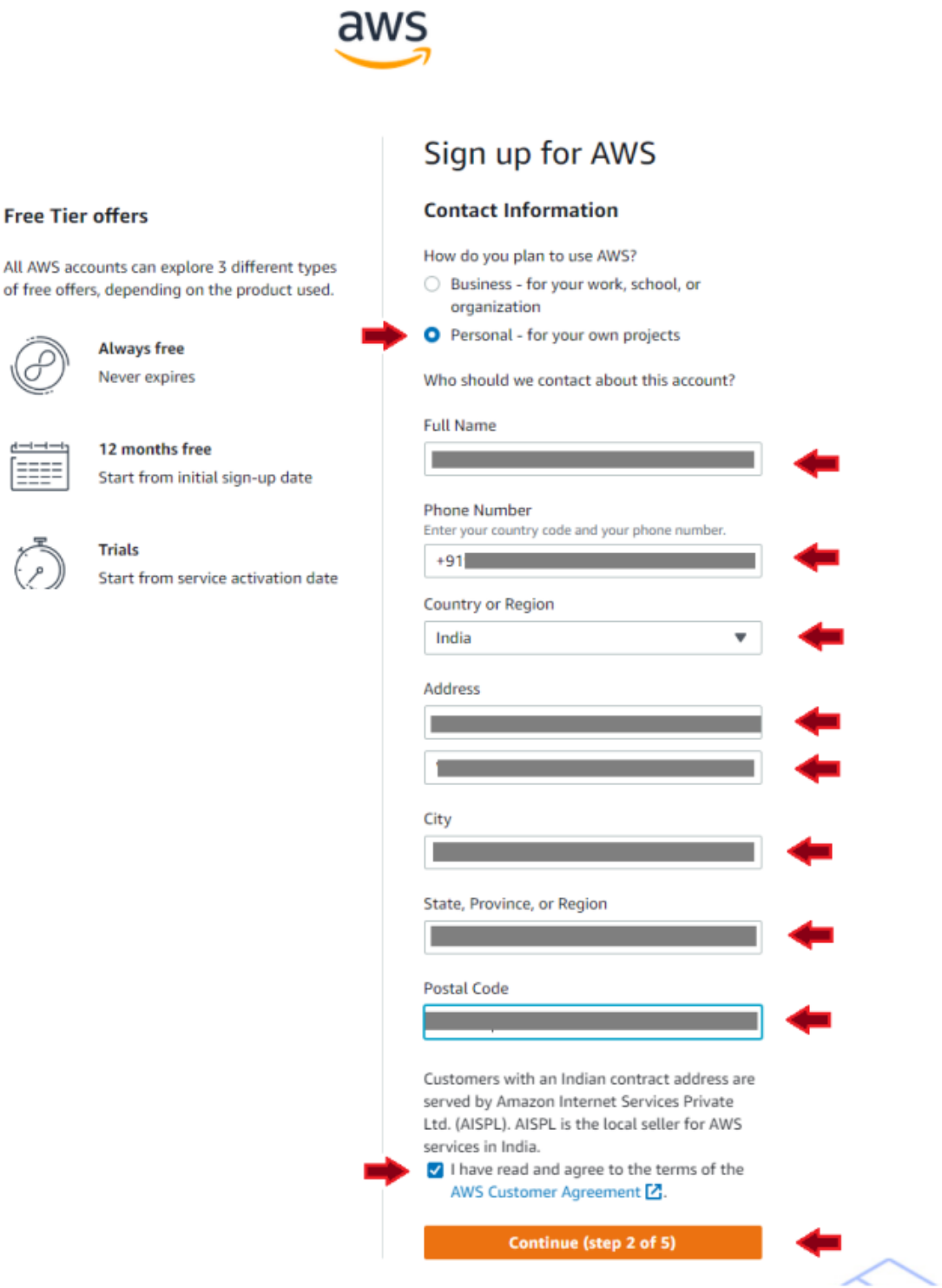
Step1: To create an account on AWS or go to AWS Console and click on SignIn button.

Step 2: After that click on create a new account.

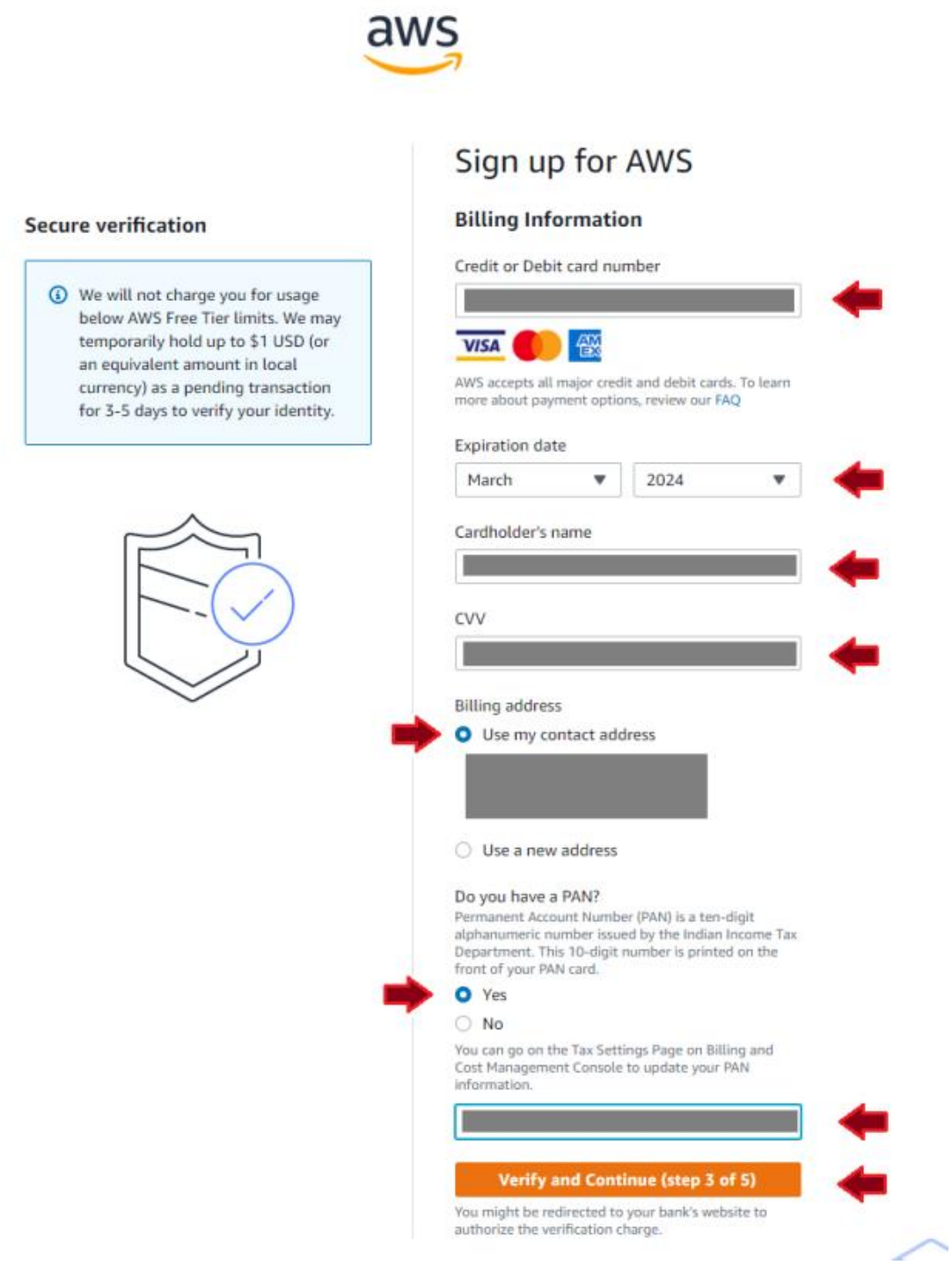
Step 3: Enter your email address and Account name. Verify your email address.

Step 4: After the verification click on next and enter the password for and click on continue.

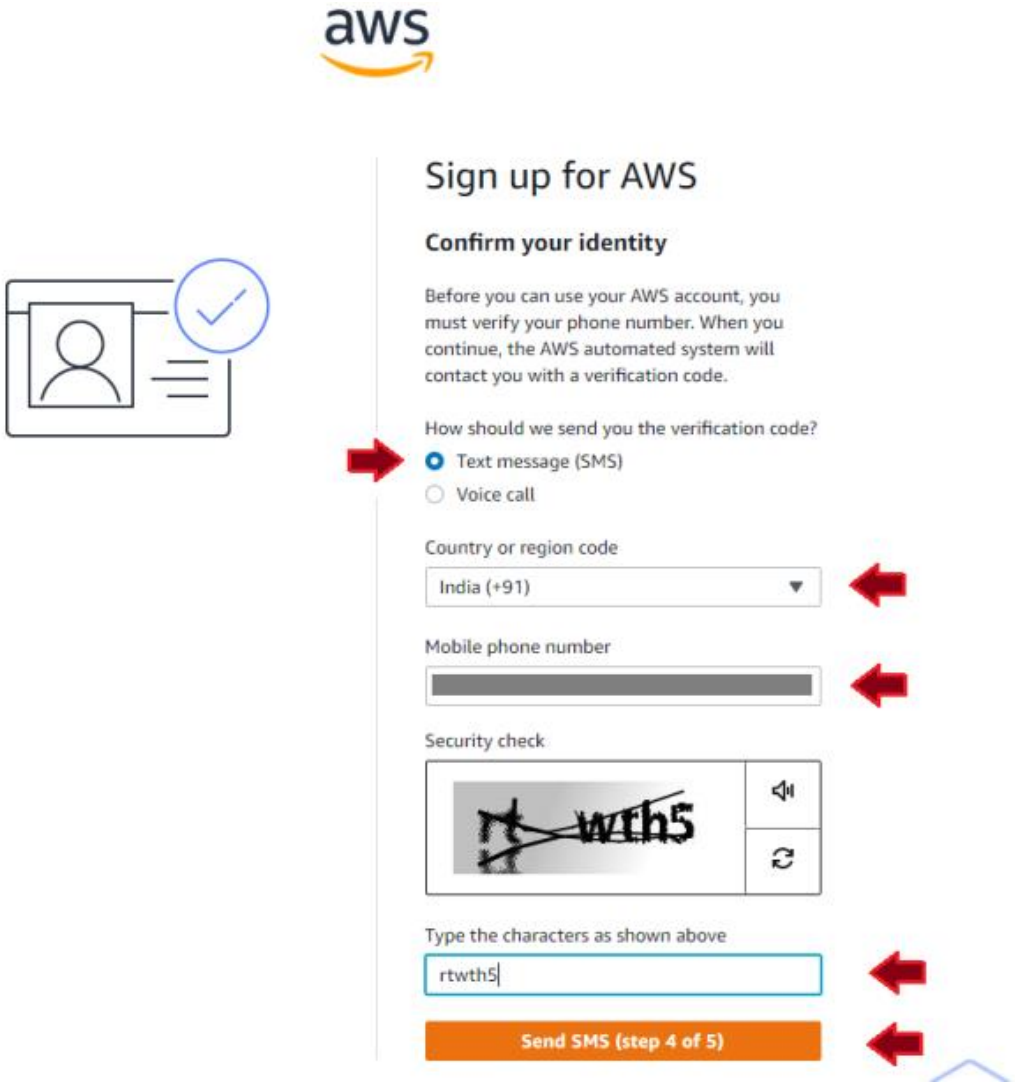
Step 5: In the next step we have to provide some information about us like user like full name mobile number, city etc. For your verification purpose once you are done click on continue


aws
Free Tier offers
All AWS accounts can explore 3 different types of free offers, depending on the product used.
Always free
Never expires
12 months free
Start from initial sign-up date
Trials
Start from service activation date
Sign up for AWS
Contact Information
How do you plan to use AWS?
Business - for your work, school, or organization
Personal - for your own projects
Who should we contact about this account?
Full Name
Phone Number
Enter your country code and your phone number.
+91
Country or Region
India
Address
City
State, Province, or Region
Postal Code
Customers with an Indian contract address are served by Amazon Internet Services Private Ltd. (AISPL). AISPL is the local seller for AWS services in India.
I have read and agree to the terms of the AWS Customer Agreement.
Continue (step 2 of 5)

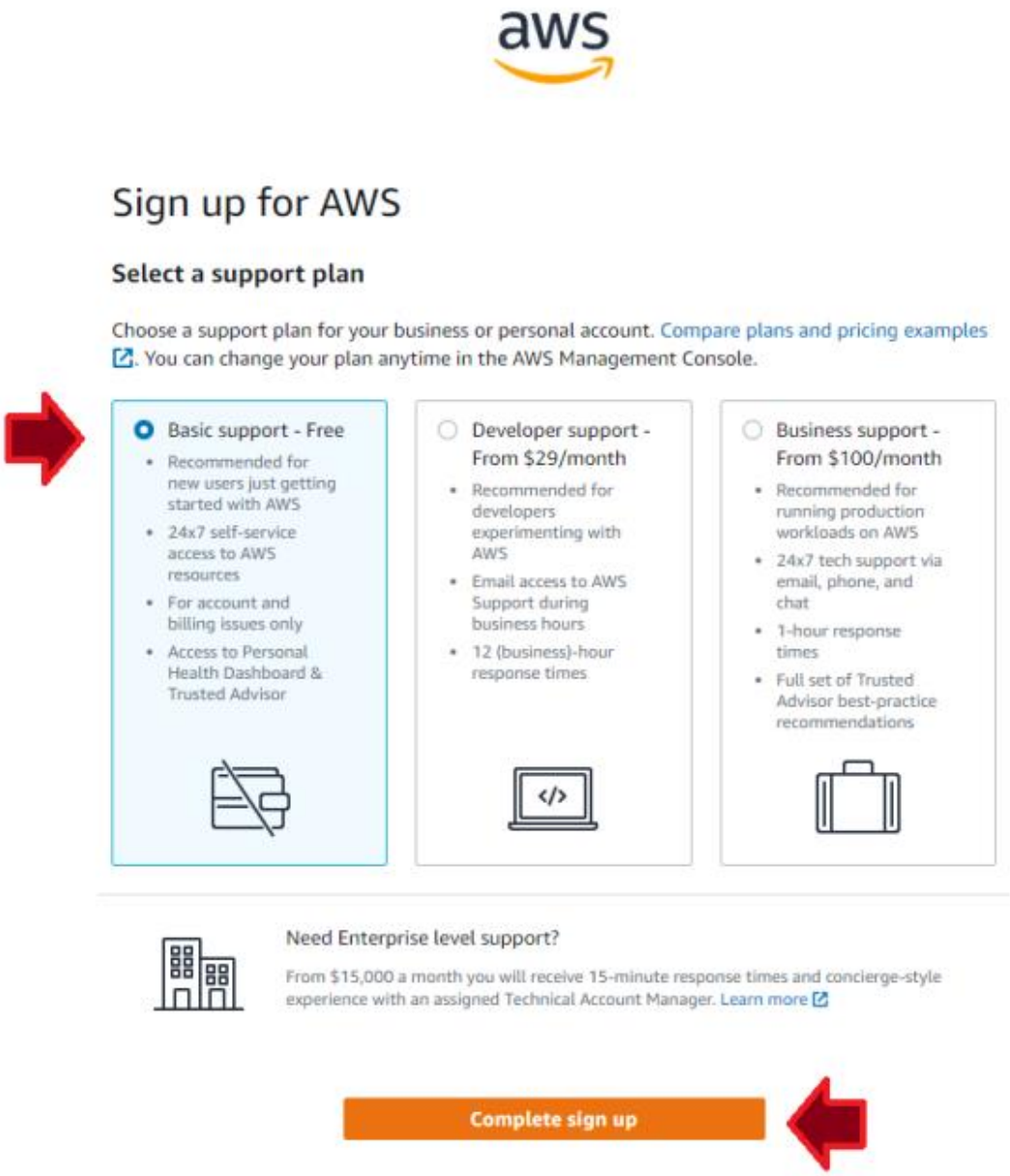
Step 6: In the next step we have to provide our card details for the payment and it is mandatory.



Step 7: After that you have to confirm your identity either via text or voice call.



Step 8: After confirming your identity, please go with the free tier plan and click on signup button.



Experiment 5: Build a basic web application on Amazon Web Services (AWS)

We must have an active AWS account to work on Amazon Web Services and deploy our web application there.

Step 1: Login to your AWS Console either by using IAM User/Password or Root Email/Password.

Step 2: Go to Services > Compute > EC2 Virtual Server in the cloud.

Step 3: Click on Launch Instances.

Step 4: An Instance wizard will appear. Provide the Name of EC2, in OS select Windows and for Image choose Microsoft Windows Server 2022. Select Instance type as t2.micro (free tier eligible). Create a key pair and save it in your local machine.

In Network Setting apart from default settings checkmark the both boxes saying 'Allow HTTP/s traffic from Internet'. Click on Launch Instance.

Step 5: Click on View All Instances.

Step 6: Wait for few minutes. Select the instance you just created and click on connect button on top right of window > RDP Client > Download the remote desktop file.

Click on Get Password and import the PEM file that you downloaded in step 4 >

Decrypt Password

Step 7: Open the remote desktop file you just downloaded and copy the decrypted Password. Allow the firewall and a login wizard will appear prompting for username and password. Provide the username (Administrator) and password (Copied password). Click on OK.

Your Created Virtual Machine will boot and will be displayed.

Experiment 6: To create a simple web application on AWS. Open anEC2 instance

Step 1: Click on Start > Server Manager and open it.

Step 2: Once Server Manager is opened click on "Add roles and features" under configure this local server menu.

Step 3: Now wizard will be opened for installing web server to our EC2 Instance.

Step 4: Click Next to display the Select server roles page

Step 5: Select the Web Server (IIS) option. When a dialog opens, click Add Features

Step 6: Click Next to display the Select features page. Do not change default settings in the Features scroll box.

Step 7: Click Next two times to display the Select role services page. Click on Install and wait for few minutes and click on close.

Then you test the IIS installation by opening your web browser and enter IPv4 public IP of EC2 in your browser and press enter and you will see the default screen for IIS.

Creating webpage and hosting it on EC2 Instance

Step 8: Now after installing IIS, open File Explorer and navigate to C:/inetpub/wwwroot. The www root folder is new in ASP.NET 5.0. All of the static files in your project go into this folder.

Step 9: Create a index.html file with some data in HTML tag and save in the www root folder.

Step 10: Then in the new tab and enter ipv4 public IP address of your EC2 machine in your browser and press enter and you will see the HTML page you have created.

Experiment 7: Create a HTML Page using CSS.

Step 1: Open VS Code and open a folder in VS Code.

Step 2: Create an index.html file. And write the following code.

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Index</title>
   <style>
        @importurl('https://fonts.googleapis.com/css?family=Raleway&display=
swap');
       body {
          font-family: 'Raleway', sans-serif;
          margin: 0;
          padding: 0;
          box-sizing: border-box;
          text-decoration: none;
       }
      nav {
          border: 2px  solid black;
          padding: 1.2rem;
          text-align: center;
          background-color: black;
       }
```

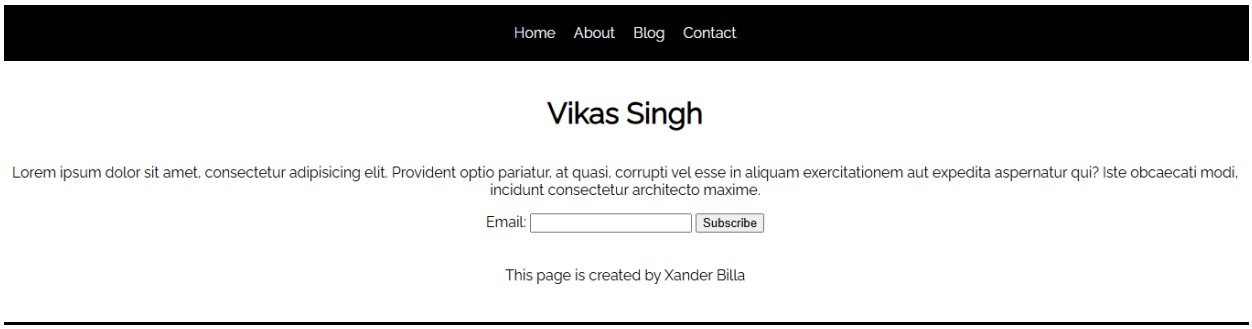
```
nav ul {
    list-style: none;
    padding: 0;
    margin: 0;
}

nav ul li {
    display: inline;
    margin: 0 0.5rem;
}

nav a {
    color: white;
    text-decoration: none;
}

nav a:hover {
    color: yellowgreen;
}
form {
    margin-top: 1rem;
    text-align: center;
}

.content {
    text-align: center;
```

```
        margin-bottom: 2rem;
      }

      h1 {
        padding: 1rem 0;
      }

      .footer {
        padding: 0 5rem;
        text-align: center;
      }
      .footer p {
        padding: 0.3rem 0;
      }
    </style>
</head>
<body>
    <nav>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">About</a></li>
        <li><a href="#">Blog</a></li>
        <li><a href="#">Contact</a></li>
      </ul>
    </nav>
    <div class="content">
```

```
    <h1>Vikas Singh</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Provident optio pariatur, at quasi, corrupti vel esse in aliquam exercitationem aut expedita aspernatur qui? Iste obcaecati modi, incidunt consectetur architecto maxime.</p>
    <form action="">
      <label for="email">Email: </label>
      <input type="text" id="email" name="email">
      <button type="submit">Subscribe</button>
    </form>
  </div>
  <div class="footer">
    <p>This page is created by Xander Billa</p>
  </div>
</body>
</html>
```

Step 3: Click on Go Live present at the task bar of the VS Code in corner.

Output:

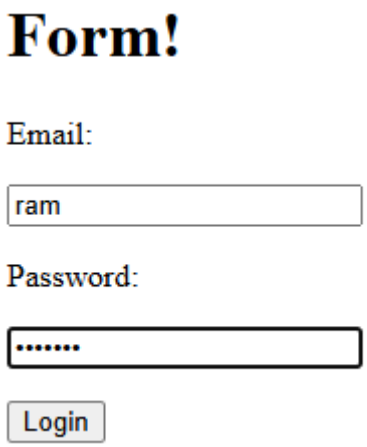


Experiment 8: Create a html form using html, css and javascript functionality to show success on form submit.

```
<!DOCTYPE html>
<html>
<head>
<title>Document</title>
</head>
<body>
<h1>Form!</h1>
<form>
<p>
<label for="email">Email: </label>
</p>
<input type="email">
<p>
<label for="password">Password: </label>
</p>
<input type="password">
<p>
<input type="submit" id="submit" value="Login">
</p>
</form>
<script>
        const signup=document.getElementById('submit');
        signup.addEventListener('click', (e) => {
            e.preventDefault();
```

```
        const formValue=document.querySelector('form');
        formValue.style.display = 'none';
        document.querySelector('h1').innerText = "Success"
    });
</script>
</body>
</html>
```

Output



Experiment 9: Setting up an Environment and tools for frontend development such as installing VS Code and installing required extension

Step 1: Open browser and go to visual studio code official website and download the windows exe file.

Step 2: Open the downloads folder and execute the downloaded file.

Step 3: Accept the license agreement and click on Next.

Step 4: Select the installation directory and click on Next.

Step 5: Keep clicking on Next keeping the setting default. And finally click on Next.

Step 6: Once the installation finish launch VS Code and go extensions to download the following extension –**ESLint**

Experiment 10: Setting up development environment for Typescript by installing the typescript compiler and live server.

Step 1: Launch command prompt.

Step 2: To install TypeScript is through npm, the Node.js Package Manager. If you have npm installed, you can install TypeScript globally (g) on your computer by:

```
npm install -g typescript
```

Step 3: To test your install by checking the version.

```
tsc --version
```

Step 5: Click on extensions and search for Live Server.

Step 6: Click on Install to install the extension

Experiment 11: Create and run first program in Typescript.

Step 1: Create a new folder HelloWorld and launch VS Code.

Step 2: Create one file named helloworld.ts then Add the following TypeScript code in that file.

```
let message : string = ' Hello World' ;

console.log (message) ;
```

Step 3: To compile your TypeScript code, you can open the Integrated Terminal by pressing key (Ctrl+`) and type

```
tsc helloworld.ts
```

Step 4: If you have Node.js installed, you can run

```
node helloworld.js.
```

Experiment 12: Setting up a React development environment-installing Node.js, create and run a ReactJS app.

Step 1: Open browser and go to nodejs.org

Step 2: Go to downloads and download the MSI installer for windows of latest version

Step 3: Open the download folder and run the MSI file.

Step 4: Accept the license terms and agreement, click Next.

Step 5: Select the installation directory, click on Install.

Step 6: Once installed verify the installation of Node JS on command following command. prompt with

```
node --version

npm --version
```

Step 7: To create a react application hit the following command.

```
cd Desktop

npx create-react-app my-app
```

Step 8: To run the react application run the following command.

```
cd my-app

npm start
```

Experiment 13: Using state (useState) in React JS.

Find file App.js inside src folder and write the below code in App.js

```
import React, { useState } from "react";
function App() {
 const [name, setName] = useState("Vikas");
  return (
    <div>
      <h2>My Name is {name}</h2>
      Click here to
      <button onClick={() => setName("Xander")}>Update Name</button>
    </div>
  );
}
export default App;
```

Experiment 14: Using Component, props, fragment in React.

Find file App.js inside src folder and write below code in App.js

```
import { Fragment } from "react";
import React from 'react';
 function Student (props){
  return (
    <div>
      <h1>
        Name:{props.name}
      </h1>
      <p>
        Marks:{props.marks}
      </p>
    </div>
  )
}
function App(){
   return (
     <Fragment>
       <Student name={"abhi"} marks={90}/>
     </Fragment>
   )
}
 export default App;
```

Experiment 15: Using List and keys in React.

Find file App.js inside src folder and write below code in App.js

```
import React, { Fragment } from "react";
import Student from "./Student";
function App() {
 const student=[
  {
   id:1,
   name:"Vikash singh",
   age:24
  },
  {
   id:2,
   name:"Anchal Rai",
   age:23
  }
 ]

 return (
  <Fragment>
   {
    student.map(item=>
     <Student key={item.id} stName={item.name} stAge={item.age} />
    )
   }
  </Fragment>
```

```
  );
}
export default App;
```

create one file named Student.js inside src folder and write below code inside Student.js

```
import React, { Fragment } from "react";
function Student(props){
   return(
      <Fragment>
         <h1>Student Info</h1>
         <h4> Name: { props.stName } | Age: { props.stAge } </h4>
      </Fragment>
   )
}
export default Student;
```

Experiment 16: Setting up environment for creating JAVA Application- Install JAVA, JAVA Editor (such as IntelliJ, Eclipse etc.), Install DBMS (MySQL, PostgreSQL or any other)

Install JDK JAVA

Step 1: Go to Java official website and go to downloads search for Java 17

Step 2: Download the MSI installer.

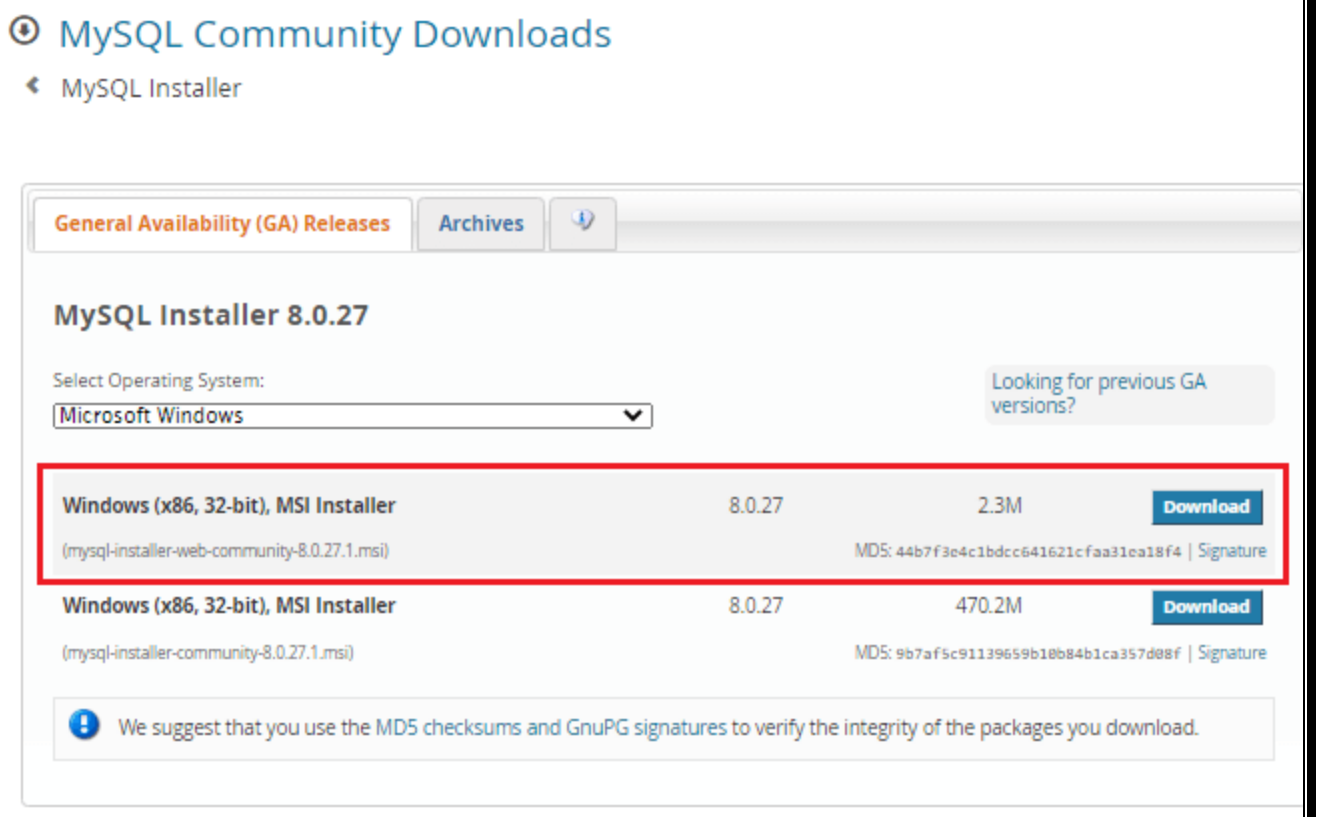
Step 3: Open the downloads folder and execute the downloaded JDK executable file.

Step 4: Select the installation directory and click on Install, once the JDK install click on close.

Download MySQL

The simplest and recommended method is to download MySQL Installer for Windows

from https://dev.mysql.com/downloads/installer/ and execute it.



Select mysql-installer-web-community-80.23.msi if you have good internet connection, otherwise choose mysql-installer-community-8.0.23.msi

Install MySQL

After downloading, unzip it, and double click the MSI installer.exe file.

Then follow the steps below.

Step1: "Choosing a Setup Type" screen: Choose "Full" setup type This installs all MySQL products and features. Then click the "Next" button to continue.

Step2: "Check Requirements" screen: The installer checks if your pc has the Requirements needed. If there is some failing requirements, click on each item to try to Resolve them by clicking on the Execute button that will install all requirements automatically, Click "Next".

Step 3: "Installation" screen: See what products that will be installed, Click "Execute" to download and install the Products. After finishing the installation, click "Next".

Step 4: "Product Configuration" screen: See what products that will be configured. Click the "MySQL Server 8.0.23" option to configure the MySQL Server: Click the "Next" button. Choose the "Standalone MySQL Server/Classic MySQL Replication" option and click on the "Next" button.

In page "Type and Networking" set Config Type to "Development Computer" and "Connectivity" to "TCP/IP" and "Port" to "3006". Then, click the "Next" button.

Step 5: "Authentication Method" screen: Choose "Use Strong Password Encryption for Authentication". Click "Next".

Step 6: "Accounts and Roles" screen: Set a password for the root account. Click "Next".

Step 7: "Windows Service" screen: Here, you configure the Windows Service to start the server. Keep the default setup, then click "Next".

Step 8: "Apply Configuration" screen: Click the "Execute" button to apply the Server configuration. After finishing, click the "Finish" button.

Step 9: "Product Configuration" screen: See that the Product Configuration is completed. Keep the default setting and click on the "Next" and "Finish" button to complete the MySQL package installation.

Step 10: in the next screen, you can choose to configure the Router. Click on "Next", Finish" and then click the "Next" button.

Step 11: "Connect To Server" screen: Type in the root password (from step 6). Click the "Check" button to check if the connection is successful or not. Click on the "Next" button.

Step 12: Apply Configuration" screen: Select the options and click the "Execute" button. After finishing, click the "Finish" button.

Step 13: "Installation Complete" screen: The installation is complete. Click the "Finish" button.

Install Eclipse

Step 1: Go to eclipse.org and download the exe file for windows

Step 2: Open the downloads folder and execute the downloaded eclipse executable file.

Step 3: Select the installation directory and click on Install, once the eclipse install click on Launch.

Experiment 17: Install MongoDB and perform CRUD operation on Database and documents

Step 1: Go to MongoDB official website

Step 2: Click on Products > MongoDB Community Edition

Step 3: Download the MongoDB Community Server for your OS

Step 4: To download the MongoDB shell click on Products > Tools > shell

Step 5: Download the zip file and extract it

Step 6: Now execute the executable file of MongoDB server after the download is finished

Step 7: Accept the license terms and agreement and click on next.

Step 8: Select complete and click Next every time keeping everything as it is. Click on Install.

Step 9: Once the installation is complete click on finish.

Step 10: Now Copy the extracted file to C:/drive and create sub folder with name data/db in the same drive.

Step 11: Now copy the bin folder path of both shell and server and add those paths to environment variables.

Step 12: Launch the command prompt and start the server by executing the command 'mongod'

Step13: Open another command prompt and launch the MongoDB' shell using mongosh command

Step 14: CRUD Operation in MongoDB (Let's take an

Step 15: Create Database

```
use Student
```

Step 16: Create Collection

```
db.createCollection ( ' studentInfo ' )
```

Step 17: Show Collection

```
show collections
```

Step 18: Show Database

```
show dbs
```

Step 19: Insert a single Documents

```
db.studentInfo.insertOne ( { id : 101 , name : " Vikas Singh " , age : 24 , address : "Bangalore" } )
```

Step 20: Insert multiple documents

```
db.studentInfo.insertMany ( [ { id : 102 , name : "Mohan Yadav ", age:25, address : "Varanasi " }, { id : 103 , name : "Aanchal Rai ", age:26, address:”Mumbai” } ] )
```

Step 21 : Show documents in a collection

```
db.studentInfo.find()
```

Step 22: Show documents in a collection using filter

```
db.studentInfo.find ( { id : 102 })
```

Step 23: Update one document

```
db.studentInfo.updateOne ( {id:102 } , ( address : " Bangalore"} )
```

Step 25: Delete single document

```
db.studentInfo.deleteOne ( { address : "Bangalore" })
```

Step 26: Delete multiple documents

```
db.studentInfo.deleteMany( {})
```

Experiment 18: Create a docker image and run a container using docker image. Also do the same using docker file.

Step 1 : Go to play with docker official website

Step 2: Click on Login

Step 3: Click on start

Step 4: Click on create new instance at left side o fthe page

Step 5: To create a docker image run the following command

```
docker create <IMAGE NAME>
```

Step 6: Show the existing images of docker use the command - docker images

Step 7: To run a docker image in a container execute the following command -

```
docker run –it  < IMAGE>
```

Step 8: Exit the container you're in.

Step 9: Click on Editor to create a dockerfile

Step 10: Enter the following script and save the file

```
FROM ubuntu
MAINTAINER vikas
RUN apt -get update
CMD ["echo" , " Hello World"]
```

Step 11: Now rename the file to 'Dockerfile'

```
mv <filename> Dockerfile
```

Step 12: To create an image and run container using the docker file hit the following command

```
docker build .
```

Step 13: If you want to deploy a web application create a NGINX container To do so

follow the steps below to Create a NGINX image with -

Step 13.a: Run the NGINX container by binding the port number using -p flag

```
docker run -p 8085: 80 nginx
```

Step 13.b: Now click on open port on top of screen and enter the port number you banded

Step 13.c: The deployed application will be opened in a new tab