

#### Bachelor of Technology

in

Computer Science and Engineering

Lab Manual

For

WEB TECHNOLOGIES LAB

18CS 5SE02L

*Global Campus*



Faculty of Engineering & Technology

45th km NH— 209, Jakkasandra Post, Kanakapura Rd, Bangalore

[www.set.jainuniversity.ac.in](http://www.set.jainuniversity.ac.in/) Fax STD Code:- 080 Fax:- 2757 71

CONTENTS

|  |  |  |
| --- | --- | --- |
| # | TITLE | PAGE NO. |
|  | Institute Vision and Mission | 3 |
| 2‘ | Department Vision and Mission | 3 |
| 3 | PEOs | 4 |
| 4 | Program Specific Outcomes (PSO) | 4 |

|  |  |  |
| --- | --- | --- |
|  | Program Outcomes (PO) | 4 |
| **6‘** | **Mapping of** PEOs **and** POs | 5 |
| 7’ | Course Outcome (CO) Statements and CO-PO Mapping | 6 |
| 8‘ | **List of** experiment **with CO Mapping** | 7 |
|  | List of Tools used **and** Ref **books** |  |
| 10. | Aim, Description, **Algorithms,** Program and Output of  each experiment |  |
| **11.** | **Rubrics for Evaluation** ( **CIA and** Semester End  Assessment) |  |



Faculty in-charge(s) **Head of the** Department

Institute **Vision** and Mission

## *Vision:*

### To be a leading technical institution that offers a transformative education to create leaders and innovators with ethical values to contribute for the sustainable development and economic growth of our nation.

*Missions*

the changing needs of the modern society.

To provide outcome-based education that transforms the students to understand and solve the societal, industrial problems through engineering and technology.

To collaborate with other leading technical institutions and organization to develop globally competitive technocrats and entrepreneurs.

**Department Vision and Mission**

***Vision:***

To emerge as a model center for education and research in the area of Computer Science and Engineering through Knowledge acquisition, dissemination and generation to meet societal demands.

***Mission:***

Ml: To impart the quality education in cutting edge technologies, teaching & learning ambience in Computer Science and Engineering.

M2: To establish a center of excellence in collaboration with industries, research laboratories and other agencies to meet the changing needs of society.

M3: To provide an environment conducive to develop innovation, team-spirit and Entrepreneurship. M4: To practice and promote high standards of professional ethics and transparency.



**Program Educational Objectives (PEOs)**

A few years after graduation, the Graduates of Computer Science and Engineering will be able

**PEO1:** To excel as professionals in the area of Computer Science and Engineering with an inclination towards continuous learning.

**PEO2:** To involve in interdisciplinary innovative and creative research work to solve societal needs and adopt themselves to rapidly evolving technologies.

**PEO3:** To develop entrepreneurial skills with leadership capabilities

**PEO4:** To exhibit professional ethics among the graduates to transform them as responsible citizens.

Program Specific Outcomes (PSO)

**PSO 1:** Design and develop network, web-based, cloud-based computational systems

PSO 2: Design efficient algorithms, understand software practices and implement code with optimization

**Program Outcomes**

**Engineering Graduate’s attributes**

|  |  |
| --- | --- |
| **Sl.No.** | **Program** Outcomes |
| 1. | **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex  engineering problems. |
| **2.** | **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first  principles of mathematics, natural sciences, and engineering sciences. |
| **3.** | **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural,  societal, and environmental considerations. |
| 4. | **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data,  and synthesis of the information to provide valid conclusions. |
| 5. | Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex  engineering activities with an understanding of the limitations. |
| 6. | The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent  responsibilities relevant to the professional engineering practice. |
| 7. | **Environment and sustainability:** Understand the impact of the professional  engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development. |
| 8. | **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities  and norms of the engineering practice. |
| 9. | **Individual** and **teamwork:** Function effectively as an individual, and as a member or  leader in diverse teams, and in multidisciplinary settings. |
| 10. | **Communication:** Communicate effectively on complex engineering activities with  the engineering community and with society at large, such as, being able to  comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions. |
| 11. | **Project management and finance:** Demonstrate knowledge and understanding of the |

|  |  |
| --- | --- |
|  |  |
|  |  |



**Course Outcome** Statements **and CO-PO Mapping**

### Course Outcome Statements

18CS5SE02L.

1

18CS5SE02L.

2

18CS5SE02L.

3

18CS5SE02L.

4

18CS5SE02L.

18CS5SE02L.

6

Understand the basic information about html tags, and attributes.

Develop skills in analyzing the usability of a web site.

Design the responsive web design with HTMLS and CSS3, and

Develop familiarity with the JavaScript language.

Become adept at implementing client-side interfaces through the use of the DOM, jQuery and AJAX.

Design the basics of creating XML documents, transforming XML documents, and validating XML documents.

CO — PO Mapping

***COIPO:* Mapping**

### (H/M/L indicates strength of correlation) 3-High, 2-Medium, I-Low

**Course Programme Outcome (POs)**

Outcome PO-1 PO-2 PO- PO-4 PO- PO- PO-7 PO- PO- PO- PO- PO- (COs) 3 5 6 8 9 10 11 12

CO-1 CO-2 CO-3 CO-4 CO-5 CO-6



Avg :

M M M

M M M

M M L

M H H

H H H

1. L L
2. M

L L

L L

H L

H L

M M M



**List of Experiment with CO Mapping**

|  |  |  |
| --- | --- | --- |
|  | Title of the Experiment | CO |
| 1. | Design a simple web page with head, body and footer, with heading tags, image tag | COl |
| 2. | Design a web site for book information, home page should contain books list, when particular book is clicked, information of the books should display in the next page. | CO1 |
| 3. | Design a page to display the product information such as name, brand, price and etc with table tag | CO1 |
| 4. | Design a web site for book information using frames, home page should contain two parts, left part should contain books list, and right part should contain book information. | CO2 |
| 5. | Design a web page to capture the user information such as name, gender, mobile number, mail id, city, state, and country using form elements. | CO2 |
| 6. | Design a web page with nice formatting like background image, text colors and border for text using external CSS. | CO3 |
| 7. | Design a web page to perform mathematical calculations such as addition, subtraction, multiplication, and division using form elements and Java Script. | CO4 |
| 8. | Design a web page to capture the user information such as name, gender, mobile number, mail id, city, state, and country using form elements and display them into other pages using Java Script. | CO4 |
| 9. | Design a web page to display timer in the left side of the web page using Java Script. | CO4 |

|  |  |  |
| --- | --- | --- |
|  | Design a web page to capture the student details such as student number, name, age, marks using Java Script Object. | CO4 |
| 11. | Design a web page to read data from an XML file and display the data in tabular format, take the data as employee information. | CO5 |
| 12. | Design a web site for online purchase using CSS, JS and XML, web site should contain the following web pages.   * Home page * Login page * Signup page * Product details page | CO6 |

**List of Tools used and Reference books**

**Tools / Software used**

# **Tools / Software Used**

visual **studio code**

### xampp server

2

**Licensed / Open source Open source**

Open Source



**Reference Text Books**

1. Dietel and Dietel, “Internet and World Wide Web - How to Program”, 5th Edition, PHI/Pearson Education, 2011
2. Herbert Schildt, “The complete Reference Java 2”, 8th Edition, TMH, 2011. 3. Phil Hanna, “The Complete Reference JSP”, 1st Edition, TMH, 2003.

**Overview**

### The steps below outline the processes for installing the XAMPP stack on a local machine. The XAMPP (pronounced Zamp) stack includes the following: Apache HTTP Server, MariaDB (essentially MySQL), Database Server, Per1, and the PHP Interpreter. The “X” in XAMPP is used to signify the cross-platform compatibility of the stack. The Apache HTTP Server and PHP are required to run phpMyAdmin which is a PHP application that is used for database administration tasks such as creating databases and tables, adding users, etc.

Alternative to XAMPP If you have experience with MySQL Workbench, you may prefer to install MySQL Server and MySQL Workbench via the MySQL Installer. MySQL Workbench performs the same functions as phpMyAdmin. However, unlike phpMyAdmin which is a web-based application, MySQL Workbench is a locally installed application and therefore does not require an HTTP Server (e.g. Apache) to run. Installing XAMPP Many of the steps listed have several alternatives (such as changing MySQL passwords via a command line) and students are welcomed and encouraged to explore alternatives. 1. Download XAMPP from the URL below and place the installer (.exe) in the location where you want to install XAMPP. Placing the installer (.exe) in the same location as the intended installation is not required but preferred. [**https://www.apachefriends.org/download.html**](http://www.apachefriends.org/download.html)

1. See the warning which recommends not installing to C:\Program Files (x86) which can be restricted by UAC (User Account Control). In the steps below XAMPP is installed to a USB flash drive for portability. However, I recommend your primary development environment be installed in a local directory such as C:\xampp. It is a best practice to install all development tools to the primary local drive. Since the installation steps are not complex or time consuming, reinstallation for practice can be performed. Before reinstallation, be sure to backup project and database work. Also, uninstall existing versions of XAMPP via Control Panel | Programs and Features to perform a clean removal.

XAMPP Weh Development Stack



Select the components you want to m5tallf oear the components You do not a'ant to wsaB. Clicx

Next when you are ready to conbnue.

Click on a cc•rrgonenr a get a detaled descrgbon

'J, NySQL

,6 FiIeZi1fa FTP Seri-er

Mercury Mail Ser 'w

'I, Tomcat

;4\* p p yAdmin

1. I“ake Sendmsl

< Back

Next >

Cancc



 DEEMED-TO-BE UNIVERSITY

### XAMPP Web Development Stack Page 4 of 17 4. Install to your local drive location. Again, the steps below show a flash drive installation but you should replace E: with C: where shown to reduce the likelihood of future configuration issues:



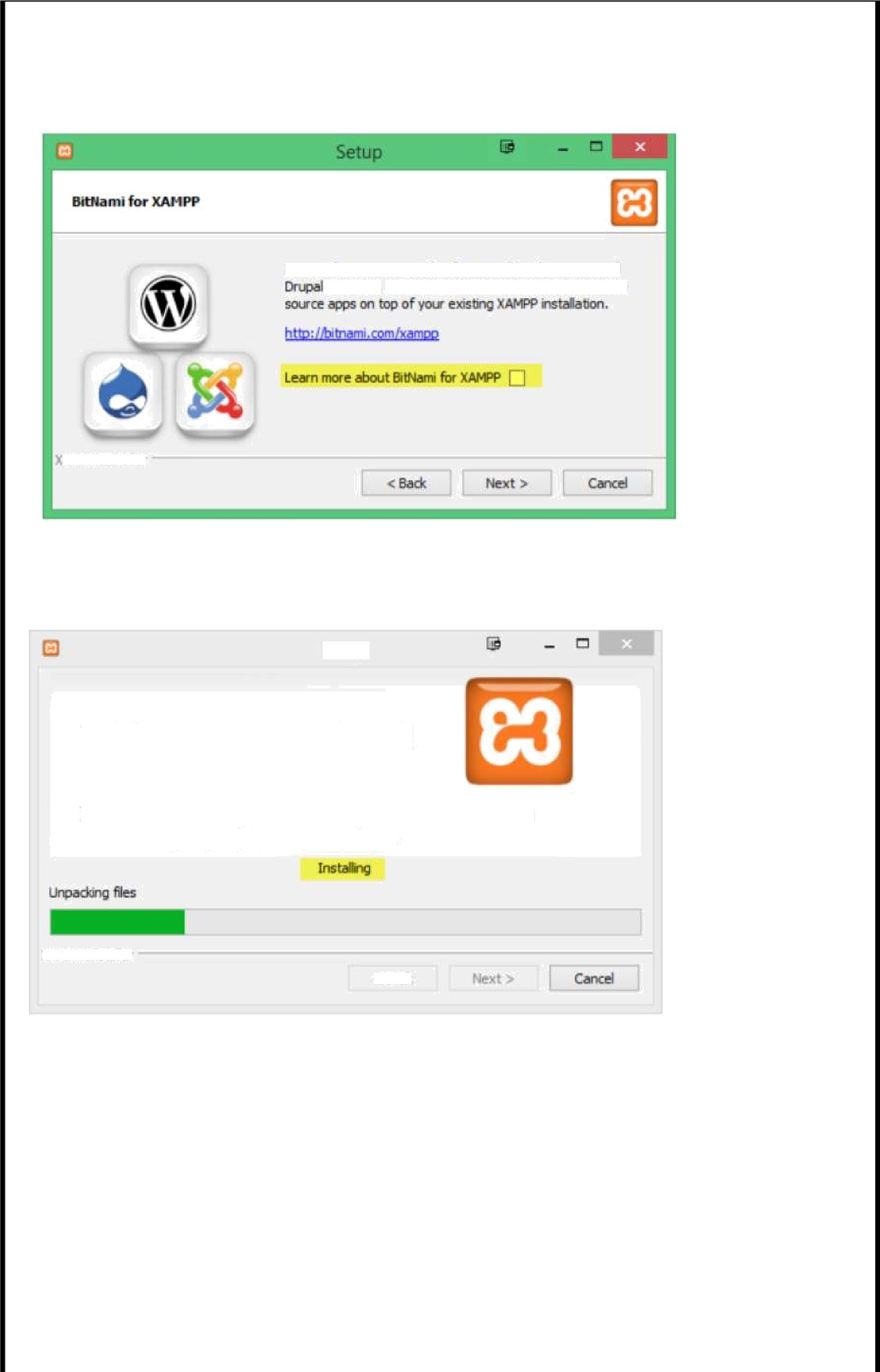


Ins a4ation folder 

Rease, onoose a folder to mstall XAMPP Select a fdder E: \xampg

NeXt :>

#### No need for Bitnami at this point:



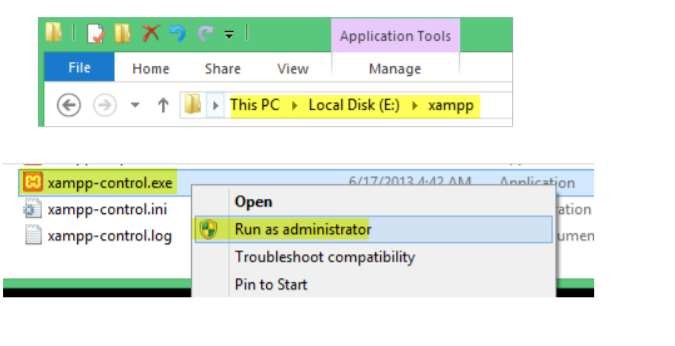
Setup

Welcome to XAMPP! '

# XANPP Is an easy to install Apache distribution contalnlng MySQL, PHP and Perl.

\.AlñPP Inc taIJef

#### 11

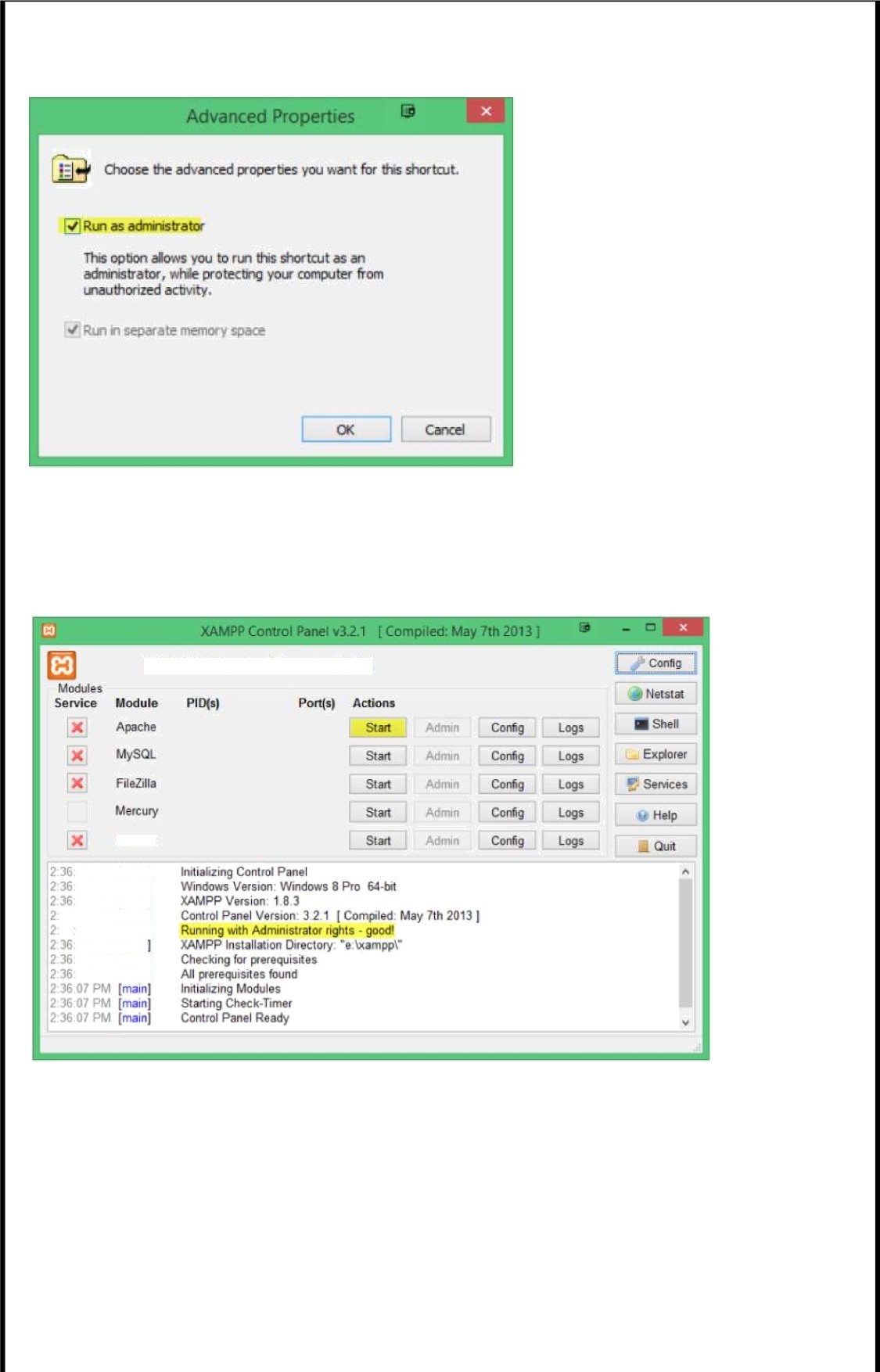


Check Start Control Panel and click Finish:

Navigate to the xampp folder (where you installed xampp). R-click xamppcontrol.exe and select Run as Adminstrator (stop — before you run from here, see the next step for a more lasting approach. If you started the Control Panel already, just close it.):

In the Advanced Properties dialog box, select Run as administrator | OK. Now double-click the shortcut to run the XAMPP Control Panel.

12



If Apache is not already running, select Start to start the Apache HTTP Server. Notice the other tools/servers included with XAMPP. Of those others listed, we will also use MySQL and we will be ready to start that shortly.

XAMPP Control Panel v3.2.1

Tomcat

[n”ainj

#### 13

Experiment -1



Design a simple web page with head, body and footer, with heading tags, image tag

<!D0CTYPE html>

<html>

##### ‹ head›

<style>

div.container { width: 73%; border: none; margin: auto;

header,

footer {

padding: 1em; color: white;

background-color: #007196; clear: left;

text-align: center;

nav {

float: rlght;

max-width: l60px; height: l34px; padding: lem; border: solid #333

nav u1 (

list-style-type: none; padding: 0;

nav u1 a (

text-decoration: none;

article {

padding: 1em;

</style>

<script type=“text/javascript“ src=”scripts.js”>

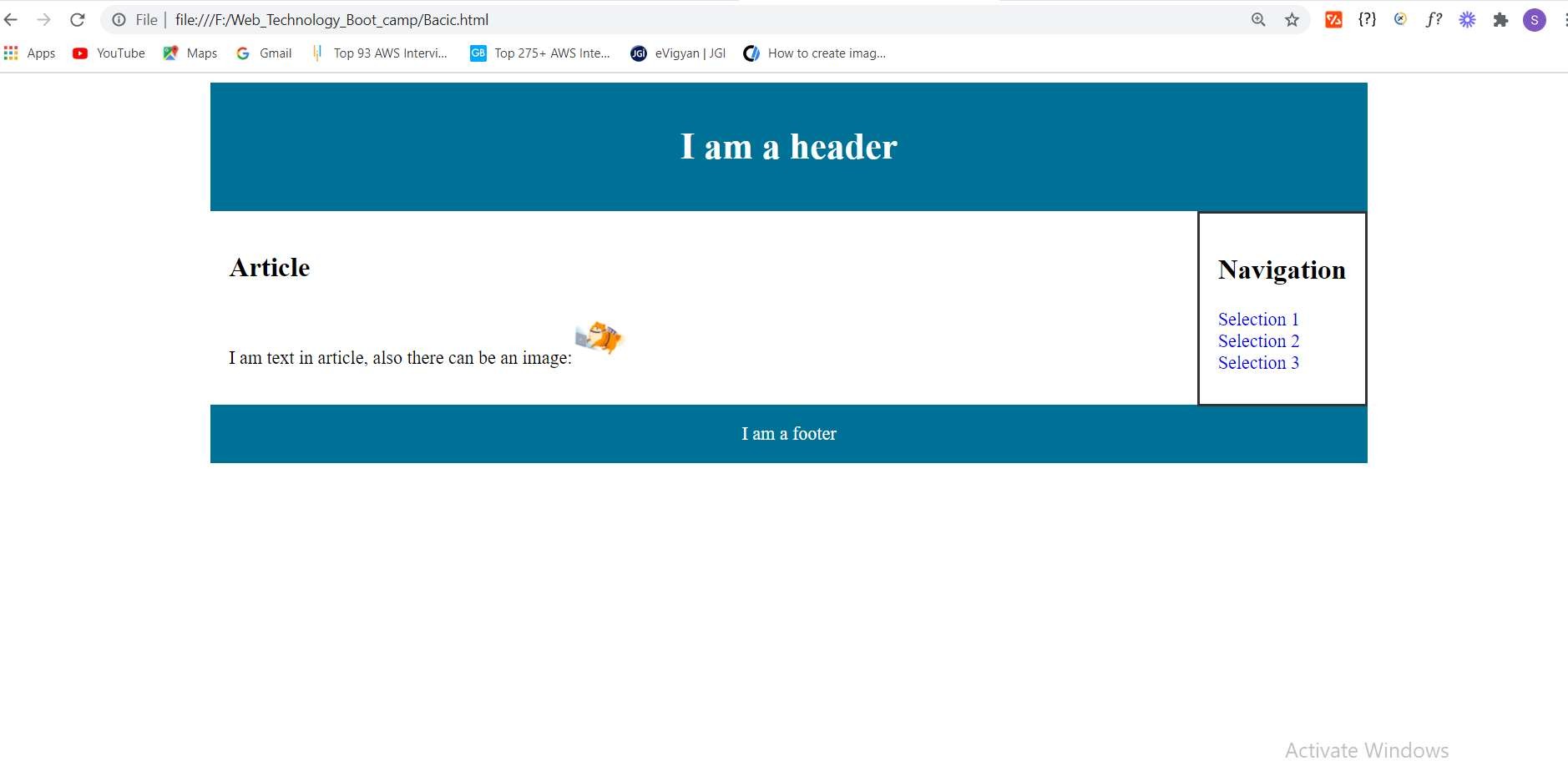
</script>

<link rel=“stylesheet” href="styles.css“>

</head>

##### ‹ body›

<div class="container”>



<header>

<h1>I am a header</h1>

</header>

<nav>

<h1>Navigation</h1>

<ul>

<li><a href=“#”>Selection 1</a></li>

<li><a href=“#”>Selection 2</a></li>

<li><a href=“#”>Selection 3</a></li>

</ul>

</nav>

<article>

<h1>Article</h1>

<p>I am text in article, also there can be an image:<img src="https://cdn.bitde gree.org/learn/pom-laptop.png” style=“height: 50px;“></p>

</article>

<footer>I am a footer</footer>

‹/d1v›

</body>

</html>

**OUTPUT:**

**Experiment -2**



1. Design a web site for book information, home page should contain books list, when particular book is clicked, information of the books should display in the next page.

<!D0CTYPE html>

<html lang=”en“>

<head>

<meta charset=“UTF-8“>

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

<title>Killing Floor</title>

</head>

<header>

<a href="Exp2.html“>Home(Jack Reacher's Book Collection)</a>

</header>

<body>

<h1 style=“color:steelblue;“>Killing Floor</h1>

<img [src=“https://www.freightlink.co.uk/sites/default/files/resize/u8261/killing-floor-](http://www.freightlink.co.uk/sites/default/files/resize/u8261/killing-floor-) 320x224.png” alt=“Image not Found! “>

<h2><u>About the Edition:</u></h2>

<ul>

<li><b>Originally published</b>: March 1997<br></li>

<li><b>Author:</b>Lee Child<br></li>

<li><b>Page count:</b> 522 (Paperback)</li>

<li><b>Genres:</b> Novel, Fiction, Mystery, Thriller</li>

<li><b>Publishers:</b> Bantam Press (UK), G. P. Putnam's Sons (US)</li>

</ul>

</p>

<details>

<h3>Description(Summary):</h3>

Killing Floor is the debut novel by Lee Child, first published in 1997 by Putnam. The book won the Anthony Award and Barry Award for best first novel.

It also is the first book to feature the character Jack Reacher. <br>

It is written in the first person. The novel has three prequels: The Enemy, Night S chool, The Affair.

</p>

</details>

</footer>

</body>

</html>

<!D0CTYPE html>



‹html lang="en“›

<head>

<meta charset=“UTF-8“>

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

<title>Jack Reacher's Book Collection</title>

‹style>

img

width:1000px; height:600px; float:right;

border: 2px solid black; border-radius:15px;

</style>

‹/head›

‹ body›

<h1>‹u>Jack Reacher's Book Collection:</u>‹/h1>

<img [src=“https://www.novelsuspects.com/wp-content/uploads/2019/07/The-10-Best-Jack-](http://www.novelsuspects.com/wp-content/uploads/2019/07/The-10-Best-Jack-) Reacher-Books-Accordlng-to-GOodreads-Featured-Image-Novel-

Suspects.png“ alt=“Jack Recher Book's Collection.”>

##### ‹li>

<a href="Exp2-1.html”>Killing Floor‹/a›

‹/li><br>

<li>

<a href=“Exp2-2.html“>Die Trying‹/a>

</li><br>

‹li>

<a href=“Exp2-3.html”>Tripwire‹/a>

‹/li><br>

<li>

‹a href=“Exp2-4.html”>The Visitor</a>

‹/li><br>

<li>

‹a href="Exp2-5.html“›Echo Burning‹/a>

</li><br>

<li>

<a href=“Exp2-6.html”>Without Fail</a>

‹/li><br>

<li>

<a href="Exp2-7.html”>Persuader</a>

</li>‹br>

<li>

<a href=“Exp2-8.html“>The Enemy</a>

</li>‹br>

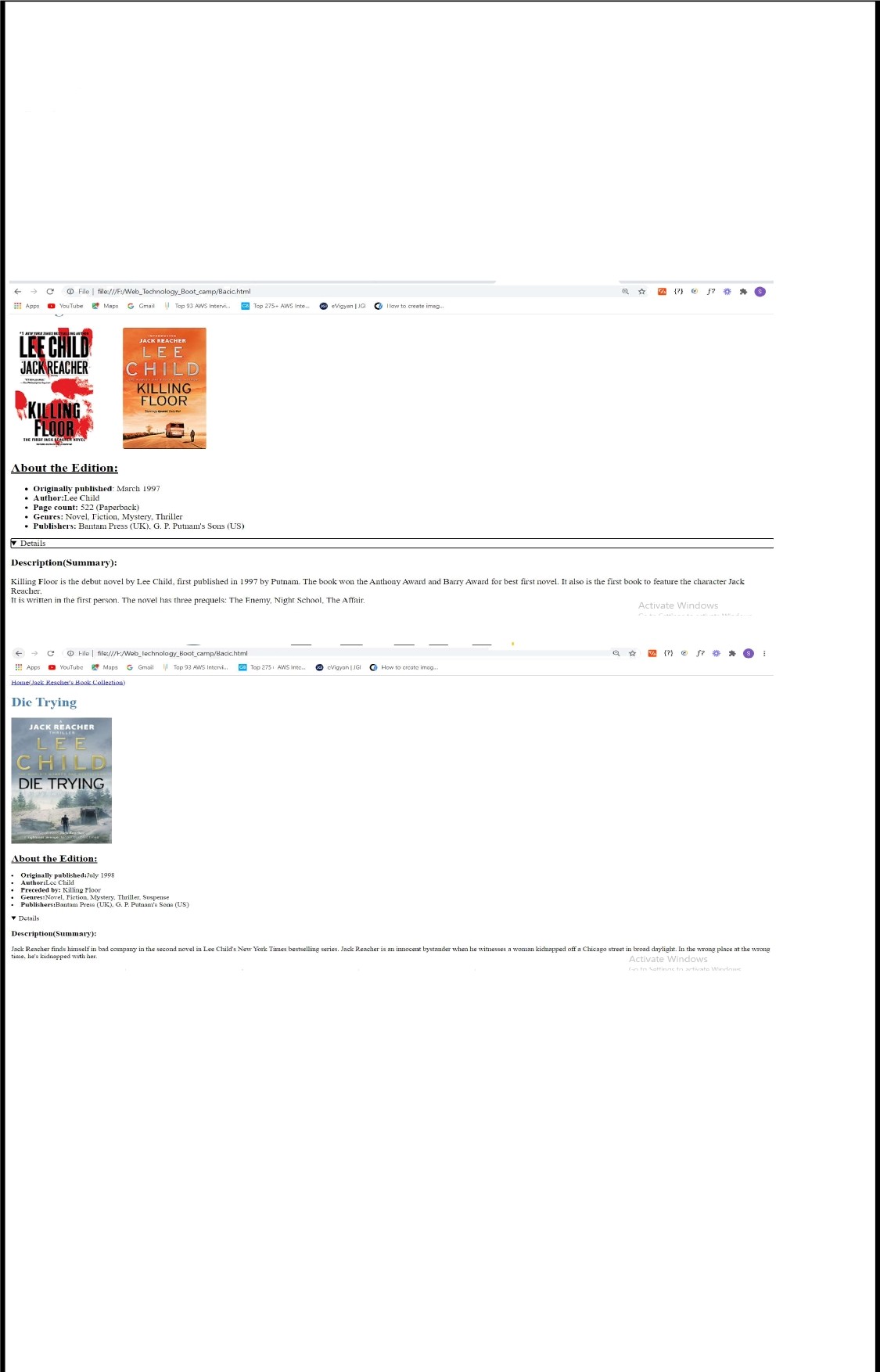
‹li>

<a href=“Exp2-9.html”>0ne Shot</a>

‹/li><br>

<li>

<a href=“Exp2-10.html“>The Hard Way‹/a>

</li>‹br>

</body>

</html>

OUTPUT:

18

DEEMED-TO-BE UNIVERSITY

FACULTY OF

ENGINEERING

JAIN

AND TECHNOLOGY



#### 19

Experiment -3



##### Design a page to display the product information such as name, brand, price and etc with table tag

<!D0CTYPE html>

<html>

<head>

<title>Exclusive Products</title>

<style>

table, th, td {

border: 1px solid black;

</style>

</head>

<body>

<h2>Exclusive Product Offers:</h2>

<table>

<th>Image</th>

<th>Product</th>

<th>Brand</th>

<th>Price</th>

<th>Description</th>

</tr>

<tr>

<td›‹img src="https://images-na.ssl-images- amazon.com/images/I/71yXShgxvpL.\_SL1500\_.jpg“ alt=”Image not found!” width=“200 px“ height= ”200px“></td>

<td>Mobile Phone</td>

<td>Redmi 8A Dual (Sea Blue, 2GB RAM, 32GB Storage) - Dual Cameras & 5,000 mAH

Battery</td>

‹td>Rs:7,499.00</td›

‹td>

‹li›13+2MP dual rear AI camera with PDAF | 8MP front camera</li>

‹li>(6.22-

inch) HD+ Dot notch display with 1520 x 720 pixels resolution and 19:9 aspect ratio | 2.5D curved glass‹/li>

‹li>Memory, Storage & SIM: 2GB | 32GB internal memory expandable up to 512GB with dedicated memory card slot.‹/li>

‹li›Dual SIM (nano+nano) dual-standby (4G+4G)‹/li>

<li>Android Pie v9.0 operating system with 1.95GHz Snapdragon 439 octa core

processor</li>

<li>5000mAH lithium-polymer battery</li>

‹/td›

</tr>

<tr>

‹td><img src="https://images-na.ssl-images- amazon.com/images/I/71ZSpNjEl0L.\_SL1500\_.jpg“ alt=”Image not found!” width=“200 px“ height= ”200px“></td›



‹td>Smart Watch</td>

<td>Mi Smart Band 4- India's No.1 Fitness Band, Up- to 20 Days Battery Life, Color AMOLED Full-Touch Screen‹/td>

<td>Rs:2,299.00</td>

‹td>

<li>Health and wellness tracking at its best with 24/7 automatic heart rati ng monitoring and alerts to warn you when the heart rate is high‹/li>

‹li>With a sturdy SATM waterproof built, you can now take your band for a s wim. It auto detects your swim style and captures 12 detailed data points for tracking. Con nectivity - Bluetooth 5.0‹/li>

‹li>Up to 20 days of long lasting battery life for uninterrupted performanc

e‹/li>

<li›Start your day right by understanding your sleep pattern with accurate

light and deep sleep monitoring‹/li>

##### ‹/td›

</tr>

<tr>

‹td><img src="https://images-na.ssl-images- amazon.com/images/I/61iE5piq4LL.\_SL1500\_.jpg“ alt=”Image not found!” width=“200 px“ height= ”200px“>‹/td›

‹td>Power Bank‹/td>

‹td›Redmi 20000mAh Li-Polymer Power Bank (Black)‹/td›

‹td>Rs:1,599.00</td›

##### ‹td›

‹li>20000mAh Li-Polymer Battery: Redmi Power Bank comes with high- density advanced Li-

polymer batteries that makes it more durable and optimizes charging efficiency‹/li›

<li>18W Fast Charging: The new Redmi Power Bank comes with 18W Fast Charging It supports 5.1V/2.4A, 9V/ 2A MAX and 12V/1.5A MAX charging outputs that ensures efficien

t and quick charging for your devices</li>

<li>Classic Dual Port: Dual input ports (Micro-USB/USB-

C) and Dual USB Output with smart charging -

Redmi Power Bank intelligently adjusts power output up to 18W to deliver fast and efficien t charging for each connected device‹/li›

<li>Easy to carry: The product has been designed keeping the Indian usage in mind and is made for carry it on-the-

go. The weight has been kept to a minimum for easy portability</li>

##### ‹/td›

</tr>

<tr>

<td><img src="https://images-na.ssl-images- amazon.com/images/I/91tltQezrxL.\_UL1500\_.jpg“ alt=”Image not found! width=“150px“ height= ”200px“></td>

<td>Analog Watch/td>

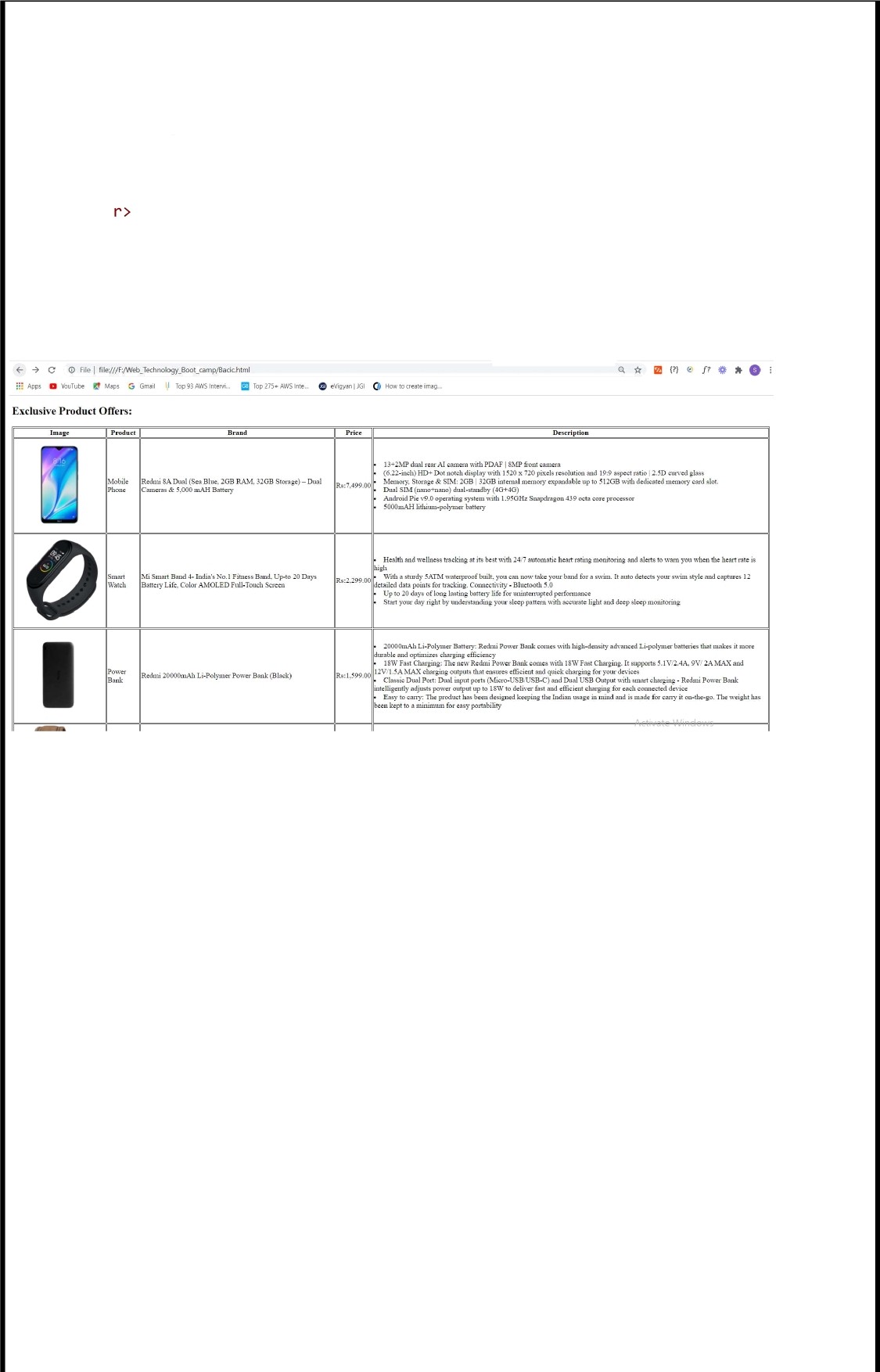
‹td>Fastrack Analog Grey Dial Men's Watch-NM38015PL03</td>

‹td>Rs:3,050.00</td>

<td>

<li>24 Months Manufacturlng warranty and 6 months Battery warranty</li>

<li>Water Resistence Depth: 50 meters</li>

‹li›Recommended for customers looking for a small dial sized watches‹/li>

<li>Band Color: Brown, Band Material: Leather. The case thickness is 14 mm and it has a buckle lock mechanism</li>

‹/td›

</t

</table>

‹/body>

</html>

OUTPUT:

#### 22

**Experiment -4**

1. Design a web site for book information using frames, home page should contain two parts, left part should contain books list, and right part should contain book information.

<!D0CTYPE html>

<html>

<head>

<title>Programming Books</title>

</head>

<body>

<h1>Jack Reacher Books</h1>

</p>

<li>

HTML Page implemented

<a href="Exp4-1.html”

in Frames

target=“r-col">Killing Floor</a>

</li><br>

<li>

<a href="Exp4-2.html”

</li><br>

<li>

target=“r-col“>Die Trying</a>

<a href=“Exp4-3.html”

</li><br>

<li>

target=“r-col“>Tripwire</a>

<a href="Exp2-4.html”

</li><br>

<li>

target=“r-col“>The Visitor</a>

<a href=“Exp2-5.html”

</li><br>

<li>

target=“r-col“>Echo Burning</a>

<a href="Exp2-6.html”

</li><br>

<li>

target=“r-col“>Without Fail</a>

<a href="Exp2-7.html”

</li><br>

<li>

target=“r-col“>Persuader</a>

<a href="Exp2-8.html”

</li><br>

<li>

target=“r-col“>The Enemy</a>

<a href="Exp2-9.html”

</li><br>

<li>

target=“r-col“>0ne Shot</a>

<a href="Exp2-10.html“target=“r-col“ >The Hard Way</a>

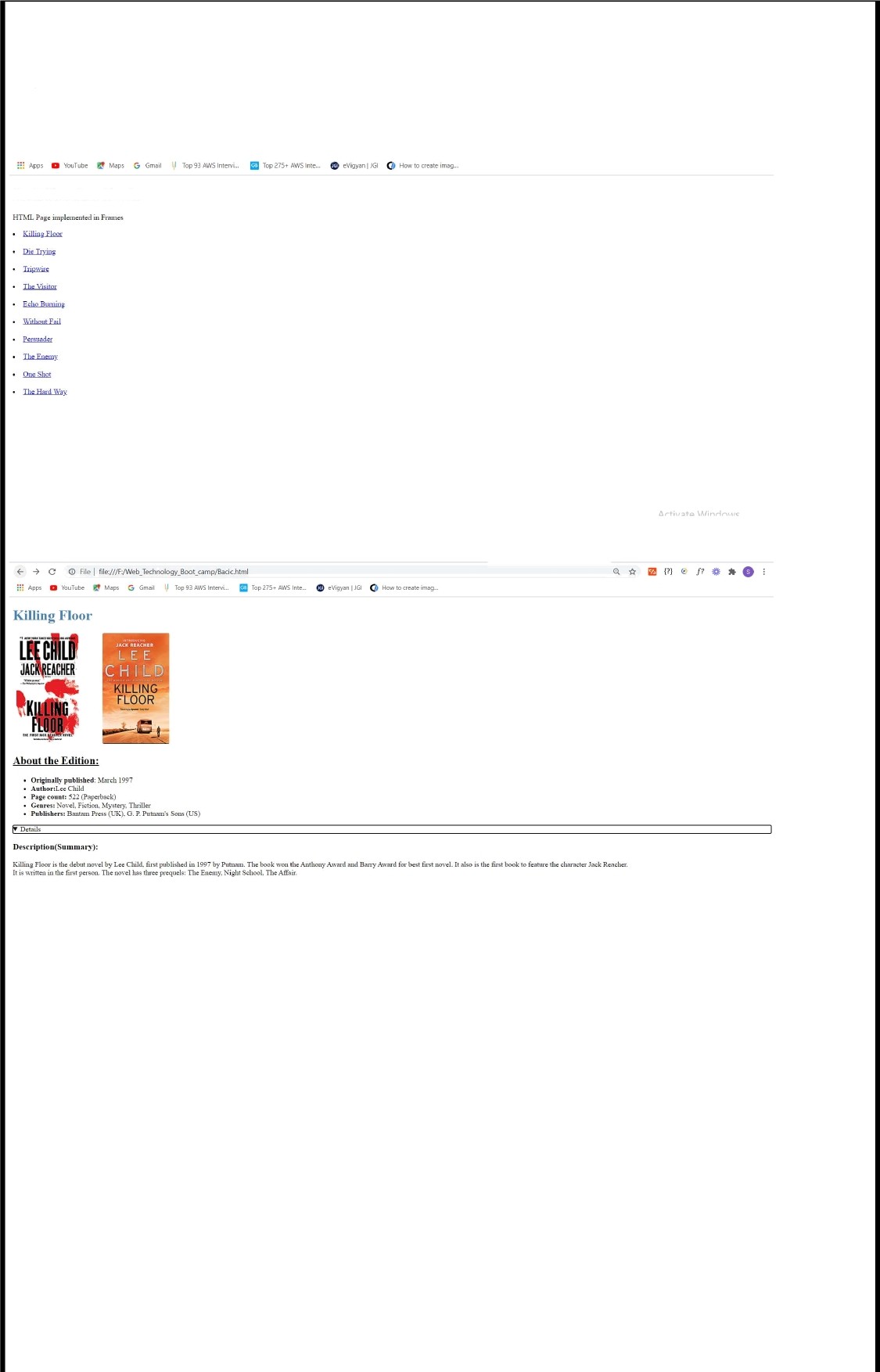


</li><br>

‹/body›

</html>

#### 23

Ref: [https://www.youtube.com/watch?v=q\_rmt9-Ikd8](http://www.youtube.com/watch?v=q_rmt9-Ikd8)

#### OUTPUT:

Jack Reacher Books

##### 24

**Experiment -5**

1. Design a web page to capture the user information such as name, gender, mobile number, mail id, city, state, and country using form elements.

<!D0CTYPE html>

<html lang=”en“>

<head>

<meta charset=“UTF-8“>

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

‹title>Document</title>

‹style>

details {

font-optical-sizing: 20;

letter-spacing: 10;

border-rlght-width: 10; background-color: pink; padding-block-end: 1000;

</style>

##### ‹/head ›

<body>

<center>

<h1>Students Bio-Data</h1>

</center>

<hr>

<form action=““>

<div>Name:<input type=”text“ name="myName“ class=“StudentData”> </div><br>

<div>USN:<input type=”text“ name="myUSN”> </div><br>



div>

<div>Mobile No:<input type="number” name=”mySem“></div><br>

<div>Year:<input type=”number“ name="Year”></div><br>

<div>Email:<input type="email“ name="myEmail”></div><br>

<div>Are you eligible?:<input type=“checkbox” name=”checkbox” value=“checkbox1“> </

<div> Gender: Male<input type=“radio“ name=”gender“> Female

<input type=“radio” name=“gender“></div><br>

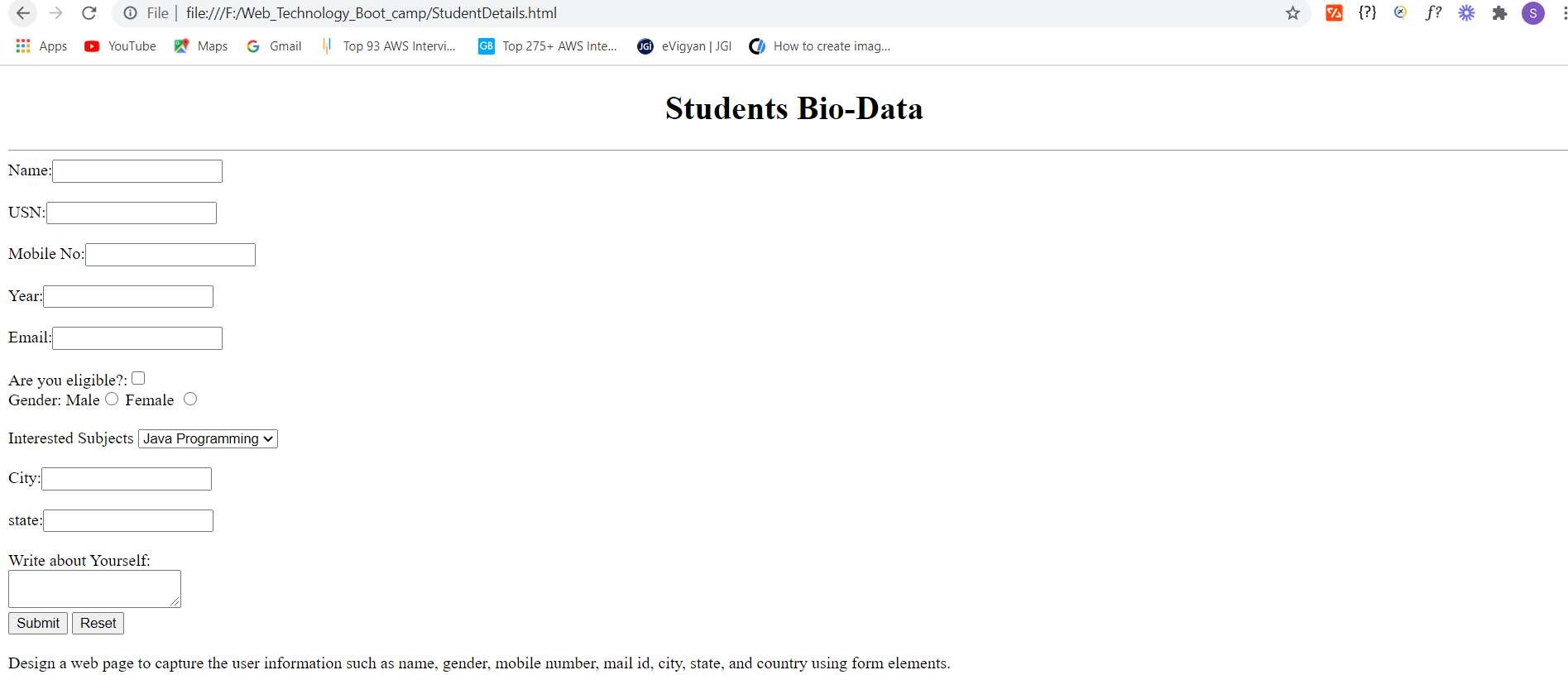
<div>

<label for=“Subject”>Interested Subjects</label>

<select name=“Subject” id=“sub">

<option value="WT“>Web Technology</option>

<option value="java“ selected>Java Programming</option>



<option value="cc“ >Cloud Compting</option>

</select>

</div><br>

<div>City:<input type=“text“ name="myUSN”> </div><br>

<div>state:<input type=“text" name=”myUSN“> </div><br>

<div>Write about Yourself:<br><textarea name=“mytext”></textarea></div>

<div><input type="submit” value=”Submit”>

<input type=“reset” name=“reset“></div><br>

</form>

</body>

</html> OUTPUT:

**Experiment -6**

### Design a web page with nice formatting like background image, text colors and border for text using external CSS.

<!D0CTYPE html>

<html lang="en">

+head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no"> otitle>Donate Blood</titles

<link rel="stylesheet" href="Style.css”>

<link href="https://fonts.googleapis.com/css2?family=Ubuntu&display=swap" rel="stylesheet">

</head>

<body>

+nav class=" navbar sticky">

<ul class="main-nav">

<li><a href=”Target.htm1"><b>Donate The B1ood</b></a></1i>

<li class="push"><a href="Target.html"><b>About Us</b></a></1i> Oli>ia href=”Target.htm1”><b>Signin</b></a></1i>

<1i><a href="Target.htm1"><b>Search /b></a></li>

<1i><a href="Target.htm1"><b>Home</b></a></lix

<li><a href="Target.html"><b>Donors</b></a></li>

<li><a href="Target.htm1"><b>Donate</b></a></li>

</nav>

<div class="wrapper">

<div class="box1">

<span><p>Search Donors<hr></p></span>

</div>

<div class="se1ect-wrapper">

<div class ="box2">

<select class="selectl" sty1e="width: 230px; height: 45px;">

<option va1ue="">Region</option>

<option va1ue="Bang">Bang1ore</option>

<option va1ue="che">Chennai</option>

<option va1ue="Mysore”>Mysore</option>

<option va1ue="Mumb">Mumbai</option>

<option value="Del">De1hi</option>

#### <option va1ue="pune">Pune</option>

<option va1ue="ko1">Ko1kata</option>

</se1ect>

</div>

<div c1ass="box3">

<select class="select2"style="width: 230px; height: 45px;”>

<option va1ue="">Blood Group</option>

<option va1ue="A+">A+</option>

<option va1ue="A-">A-</option>

<option va1ue="B+">B+</option>

<option va1ue="B-">B-</option>

<option value="AB+">AB+</option>

<option value="AB-">AB-</option>

<option va1ue="O+">O+</option> ooption value="O-">O-o/option>

</select>

</div>

</div>

<div c1ass="box4">

la href="Target.html"><button c1ass="search">Search+/button>

</div>

</div>

<div class= "footer">

<p><center><b>Donate Blood, Save Other’s Life &#128522;</body>

/htm1>

Style.css

/\* cyrillic-ext \*/ @font-face (

font-family: 'Ubuntu'; font-style: normal; font-weight: 400:

font-display: swap;

src: url(https://fonts.gstatic.com/s/ubuntu/v15/4iCs6KVjbNBYlgoKcg72j00.woff2) format('woff2');

unicode-range: U+0460-052F, U+1C80-1C88, U+20B4, U+2DE0-2DFF, U+A640-A69F, U+FE2E- FE2F;

/\* cyrillic \*/ @font-face (

font-family: 'Ubuntu'; font-style: normal; font-weight: 400:

font-display: swap;

src: url(https://fonts.gstatic.com/s/ubuntu/v15/4iCs6KVjbNBYlgoKew72j00.woff2) format('woff2');

unicode-range: U+0400-045F, U+0490-0491, U+04B0-04B1, U+2116;

/\* greek-ext ’/

@font-face (

font-family: 'Ubuntu'; font-style: normal; font-weight: 400;

font-display: swap;

src: url(https://fonts.gstatic.com/s/ubuntu/v15/4iCs6KVjbNBYlgoKcw72j00.woff2) format('woff2');

unicode-range: U+1F00-lFFF;

/\* greek \*/ @font-face (

font-family: 'Ubuntu'; font-style: normal; font-weight: 400;

font-display: swap;

src: url(https://fonts.gstatic.com/s/ubuntu/vl5/4iCs6KVjbNBYlgoKfA72j00.woff2) format('woff2');

unicode-range: U+0370-03FF;

/\* latin-ext \*/ @font-face (

font-family: 'Ubuntu'; font-style: normal; font-weight: 400;

font-display: swap;

src: url(https://fonts.gstatic.com/s/ubuntu/v15/4iCs6KVjbNBYlgoKcQ72j00.woff2) format('woff2');

unicode-range: U+0100-024F, U+0259, U+1E00-lEFF, U+2020, U+20A0-20AB, U+20AD-20CF, U+2113, U+2C60-2C7F, U+A720-A7FF;

/\* latin \*/ @font-face (

font-family: 'Ubuntu'; font-style: normal; font-weight: 400;

font-display: swap;

src: url(https://fonts.gstatic.com/s/ubuntu/v15/4iCs6KVjbNBYlgoKfw72.woff2) format('woff2');

unicode-range: U+0000-OOFF, U+0131, U+0152-0153, U+02BB-02BC, U+02C6, U+02DA, U+02DC, U+2000-206F, U+2074, U+20AC, U+2122, U+2191, U+2193, U+2212, U+2215, U+FEFF, U+FFFD;

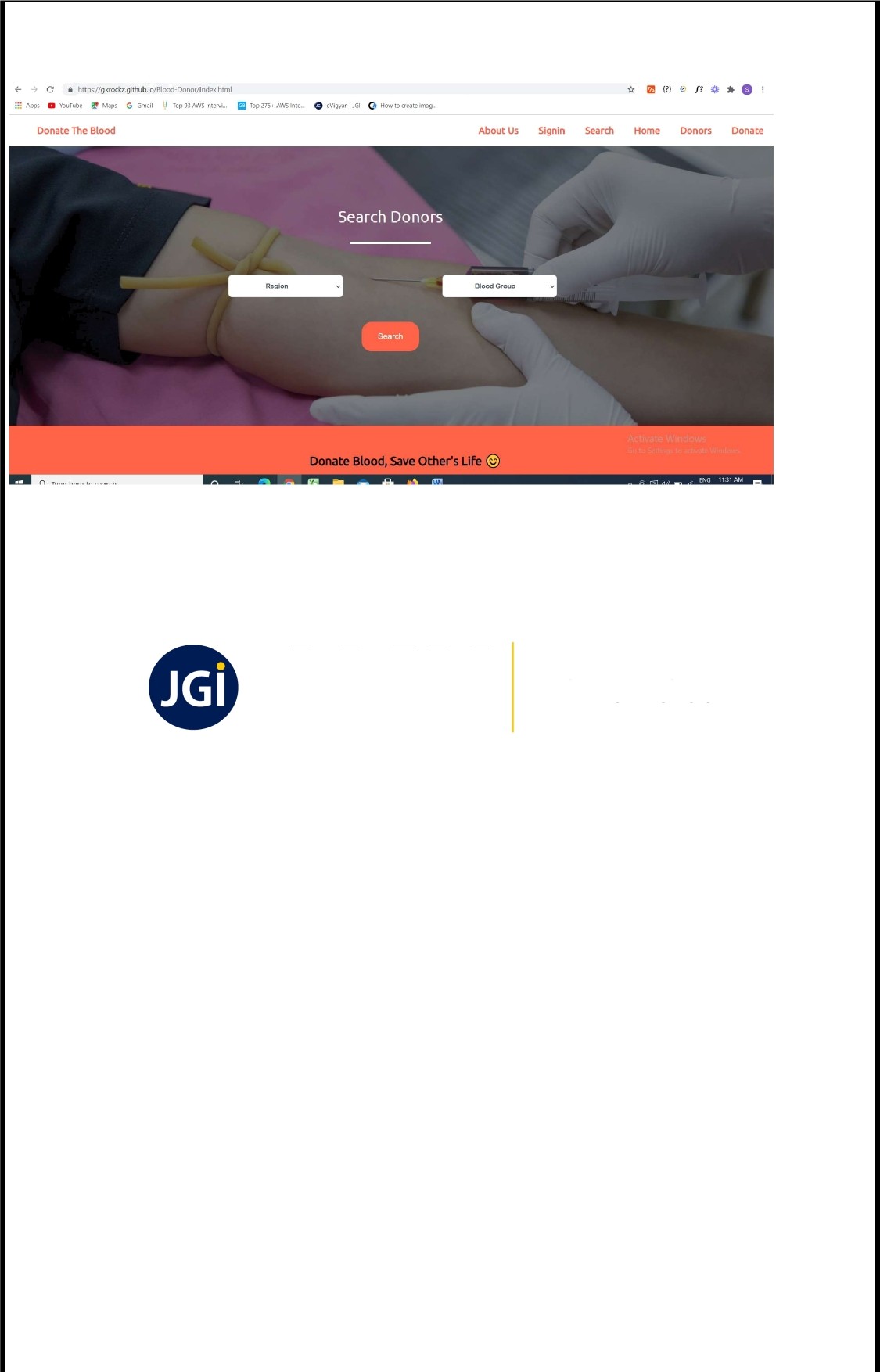
#### OUTPUT:

DEEMED-TO-BE UNIVERSITY

# FACULTY OF ENGINEERING

AND TECHNOLOGY

JAIN



30

**Experiment -7**

1. Design a web page to perform mathematical calculations such as addition, subtraction, multiplication, and division using form elements and Java Script.

***index.html***

<!D0CTYPE htm1>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-sca1e=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge"> blink rel="stylesheet" href="sty1e.css">

<title>Calculator</tit1e>

</head>

<body>

<div class="container">

<h1>Ca1culator </h1>

<div c1ass="ca1cu1ator">

<input type="text" name="screen" id="screen”>

<table>

<td><button>(</button></td>

<td><button>)</button></td>

<td><button>C</button></td>

<td><button>%</button></td>

<td><button>7</button></td>



<td><button>8</button></td>

<td><button>9</button></td>

<td><button>X</button></td>

<tr>

<td><button>4</button></td>

<td><button>5</button></td>

<td><button>6</button></td>

<td><button>-</button></td>

</tr>

<td><button>1</button></td>

<td><button>2</button></td>

<td><button>3</button></td>

<td><button>+</button></td>

</tr>

<td><button>0</button></td>

<td><button>.</button></td>

<td><button>/</button></td>

<td><button>=</button></td>

</tab1e>

</div>

</div>

</body>

<script src="index.js”></scripts

**style.css**

.container(

text-align: center; margin-top:23px

table(

margin: auto;

input{

border-radius: 21px; border: 5px solid #244624; font-size:34px;

height: 65px; width: 456px;

button{

border-radius: 20px; font-size: 40px; background: #978fa0; width: l02px;

height: 90px; margin: 6px;

.calculator(

border: 4px solid #13695 d; background-color: #ff99f7; padding: 23px;

border-radius: 53px; display: inline-block;

h1{

font-size: 28px;

font-family: 'Courier New', Courier, monospace;

**/nr/ex./s**

let screen = document.getE1ementById('screen'); buttons = document.querySe1ectorAll('button'); let screenValue = ”;

for (item of buttons) ( item.addEventListener('click', (e) => (

buttonText = e.target.innerText; console.log('Button text is ', buttonText); if (buttonText == 'X') (

buttonText = '\*'; screenValue += buttonText; screen.value = screenValue;

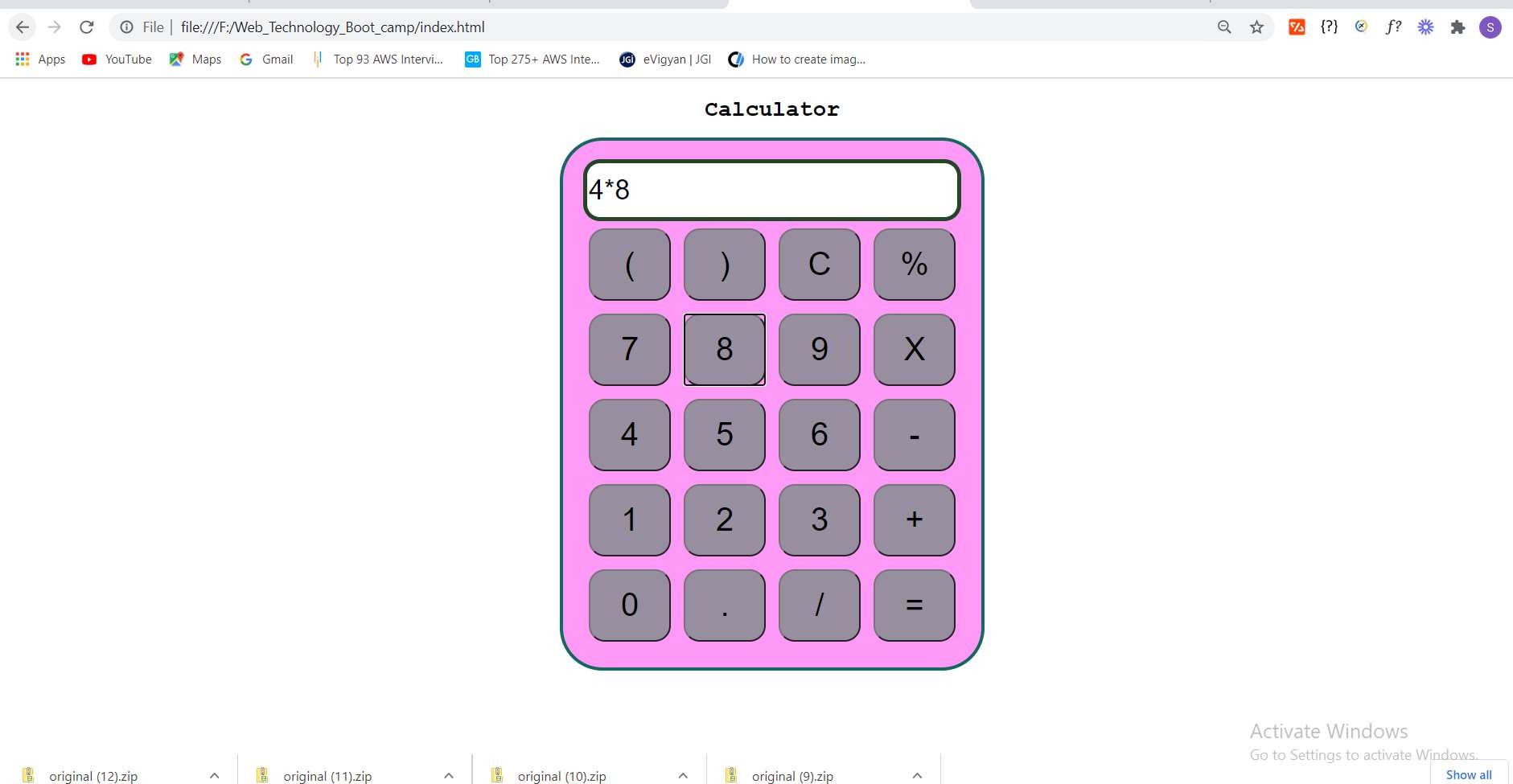
else if (buttonText == 'C') ( screenValue = ""; screen.value = screenValue;

else if (buttonText == '=') { screen.value = eval(screenValue);

else (

screenValue += buttonText; screen.value = screenValue;





**OUTPUT:**

**Experiment -8**

1. Design a web page to capture the user information such as name, gender, mobile number, mail id, city, state, and country using form elements and display them into other pages using Java Script.

<!doctype html>

<html lang=”en“>

<head>

<!-- Required meta tags -->

<meta charset=“utf-8“>

<meta name="viewport” content="width=device-width, initial-scale=1, shrink-to-fit=no”>

<!-- Bootstrap CSS -->

<link rel=“stylesheet” href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/boo tstrap.min.css“

integrity=“sha384- ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZwlT“ crossorigin=”anonymous”>

<title><center> User information</center> </title>

‹/head ›

<body>

<div id="success” class=“alert alert-success alert-dismissible fade“ role=”alert“>

<strong>Success!</strong> Your travel request has been successfully submitted

<button type=”button” class=“close“ data-dismiss=“alert“ aria-label=“Close“>

<span aria-hidden=”true“>x</span>

</button>

</div>

<div id="failure” class=“alert alert-danger alert-dismissible fade” role=”alert“>



<strong>Error!</strong> Your travel request has not been sent due to errors

<button type=“button” class=”close“ data-dismiss="alert“ aria-label="Close“>

<span aria-hidden="true“>x</span>

</button>

</div>

<div class="container”>

<h1>User Information</h1>

<form>

<div class=“form-group”>

<label for=“name”>Username</label>

<input type=“text“ class="form- control“ id="name“ placeholder="Enter your username">

‹small id=“namevalid“ class="form-text text-muted invalid-feedback”> Your username must be 2-

10 characters long and should not start with a number

</small>

</div>

<div class=“form-group”>

<label for=“email">Email address</label>

<input type=“text“ class="form- control“ id="email” placeholder="Enter your email”>

<small id=“emailvalid“ class="form-text text-muted invalid-feedback“> Your email must be a valid email

</small>

</div>

<div class=“form-group”>

<label for=“car”>State</label>

<select class=”form-control“ id=”car“>

<option>Karnataka</option>

<option>Pune</option>

<option>Hydrabad</option>

</select>

</div>

<div class="form-group“>

<label for=“car1”>State</label>

‹select class=”form-control“ id=”car“>

<option>India</option>

<option>UK</option>

<option>USA</option>

</select>

</div>

<div class=“form-group”>

<label for=“address“>Enter your address</label>

<textarea class="form-

control“ name=“address“ id=“address“ rows="3“></textarea>

</div>

<div class=“form-group”>

<label for=“address“>Enter your phone numbers/label>

‹input class=”form-control“ name=”phone“ id=”phone“ rows="3">



<small id=“emailvalid“ class="form-text text-muted invalid-feedback“> Your phone number must be 10 digit long

</small>

</div>

<div class=“form-group”>

<label for=“exampleFormControlTextarea1">Any other message...</label>

‹textarea class="form-

control“ id=”exampleFormControlTextareal” rows=“3”></textarea>

</div>

<button id=“submit“ class=“btn btn-primary“>Submit</button>

</form>

</div>

<!-- Optional JavaScript -->

<!-- jQuery first, then Popper.js, then Bootstrap JS -->

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js“

integrity=“sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo” crossorigin=”anonymous”></script>

<script src="https://cdnjs.cloudf1are.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js” integrity=“sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0Wl“ crossorigin=”anonymous”></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js“ integrity=“sha384-JjSmVgyd0p3pXBlrRibZUAYoIIy60rQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM” crossorigin=”anonymous”></script>

<script src=”js/project4.js“> console.log(”This is project 4”);

const name = document.getElementById('name'); const email = document.getElementById('email'); const phone = document.getElementById('phone'); let validEmail = false;

let validPhone = false; let validUser = false;

$('#failure').hide();

$('#success').hide();

// console.log(name, email, phone); name.addEventListener('blur', ()=>{ console.log("name is blurred");

// Validate name here

let regex = /’[a-zA-Z]([0-9a-zA-Z]){2,10}$/; let str = name.value;

console.log(regex, str); if(regex.test(str)){

console.log('Your name is valid'); name.classList.remove('is-invalid'); validUser = true;

else{

console.log('Your name is not valid'); name.classList.add('is-invalid');

validUser = false;



email.addEventListener('blur', ()=>{ console.log("email is blurred”);

// Validate email here

let regex = /’([\_\-\.0-9a-zA-Z]+)@([\_\-\.0-9a-zA-Z]+)\.([a-zA-Z]){2,7}$/; let str = email.value;

console.log(regex, str); if(regex.test(str)){

console.log('Your email is valid'); email.classList.remove('is-invalid'); validEmail = true;

else{

console.log('Your email is not valid'); email.classList.add('is-invalid'); validEmail = false;

phone.addEventListener('blur', ()=>{ console.log("phone is blurred”);

// Validate phone here

let regex = /’([0-9]){10}$/; let str = phone.value; console.log(regex, str); if(regex.test(str)){

console.log('Your phone is valid'); phone.classList.remove('is-invalid'); validPhone = true;

else{

console.log('Your phone is not valid'); phone.classList.add('is-invalid'); validPhone = false;

let submit = document.getElementById('submit'); submit.addEventListener('click', (e)=›{

e.preventDefault();

console.log('You clicked on submit'); console.log(validEmail, validUser, validPhone);

// Submit your form here

if(validEmail && validUser && validPhone){

let failure = document.getElementById('failure');

console.log('Phone, email and user are valid. Submitting the form'); let success = document.getElementById('success'); success.classList.add('show');



// failure.classList.remove('show');

// $('#failure').alert('close');

$('#failure').hide();

$('#success').show();

else{

console.log('One of Phone, email or user are not valid. Hence not submitting the fo rm. Please correct the errors and try again');

let failure = document.getElementById('failure'); failure.classList.add('show');

// success.classList.remove('show');

// $('#success').alert('hide');

$('#success').hide();

$('#failure').show();

‹/scr1pt›

</body>

</html>

OUTPUT:

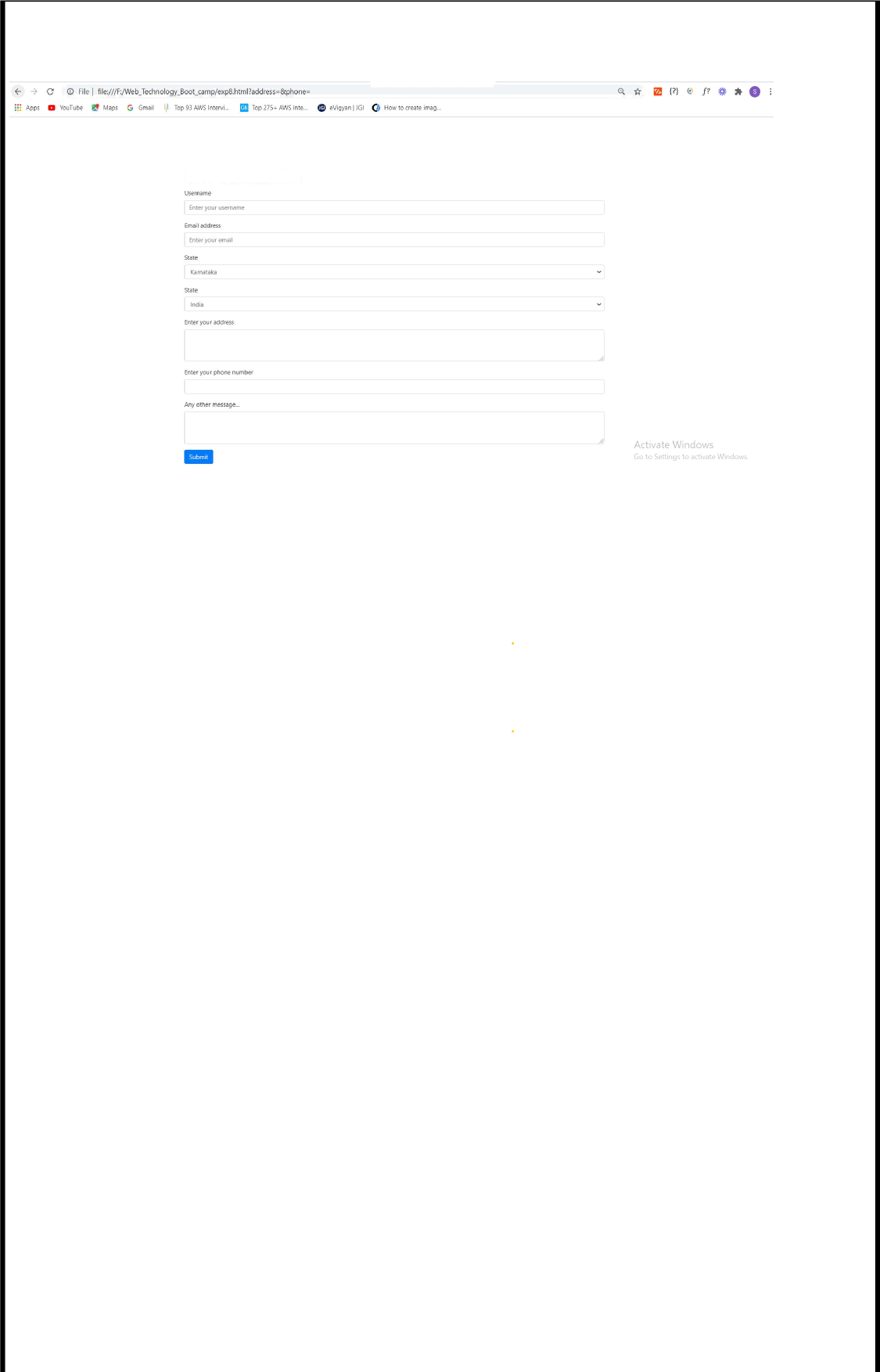
User Information

DEEMED-TO-BE UNIVERSITY

AI

# FACULTY OF ENGINEERING

AND TECHNOLOGY



##### 40

**Experiment -9**



1. Design a web page to display timer in the left side of the web page using Java Script.

<html>

‹ head›

‹link href=“css\timer.css“ type=“text/css“ rel="stylesheet” />

<style> #clock { margin: auto; padding: 0;

text-align: center; border: Spx solid blue; width: 35%;

line-height: 50px; font-size: larger;

background-color: white;

.container {

background: #222;

width: 100%;

height: 100% ; display: flex;

justify-content: center; align-items: center;

input{

width: 45px; border: none; font-size: 40px;

font-weight: bold; color: black;

img{

width: 90%;

##### body(

background-color: #333;

</style>

<script src = ”js\timer.js“></script>

</head>

<body >

<div id=“clock“>

<h2>Click the image to start the timer (60 seconds)</h2>

<img src=“C:\Users\Rajesh\Desktop\1.jpg“ onclick=“countdown();“>

<div id="timeBox">

<h3>Time Left‹/h3>



<input id="min“ type=“text“><font size=”10“>:</font>

<input id="sec“ type=“text“>

<p id=“msg"></p>

</div>

</div>

‹script>

var mins = 1; var secs = mins \* 60; function countdown() {

setTimeout('Decrement()', 60);

function Decrement() {

minutes = document.getElementById(”min"); seconds = document.getElementById(”sec");

if (seconds < 59) { seconds.value = secs;

else {

minutes.value = getMinutes(); seconds.value = getseconds();

if (secs < 10) { minutes.style.color = ”red"; seconds.style.color = "red“;

document.getElementById("msg”).innerHTML "HURRY UP!!";

if (mins < 0) { alert('Timeout'); minutes.value = '00';

seconds.value = '00'; document.getElementById("msg”).innerHTML = “Time's up.“;

else {

secs--; setTimeout('Decrement()', 1000);

function getMinutes() {

mins = Math.floor(secs / 60); return mins;

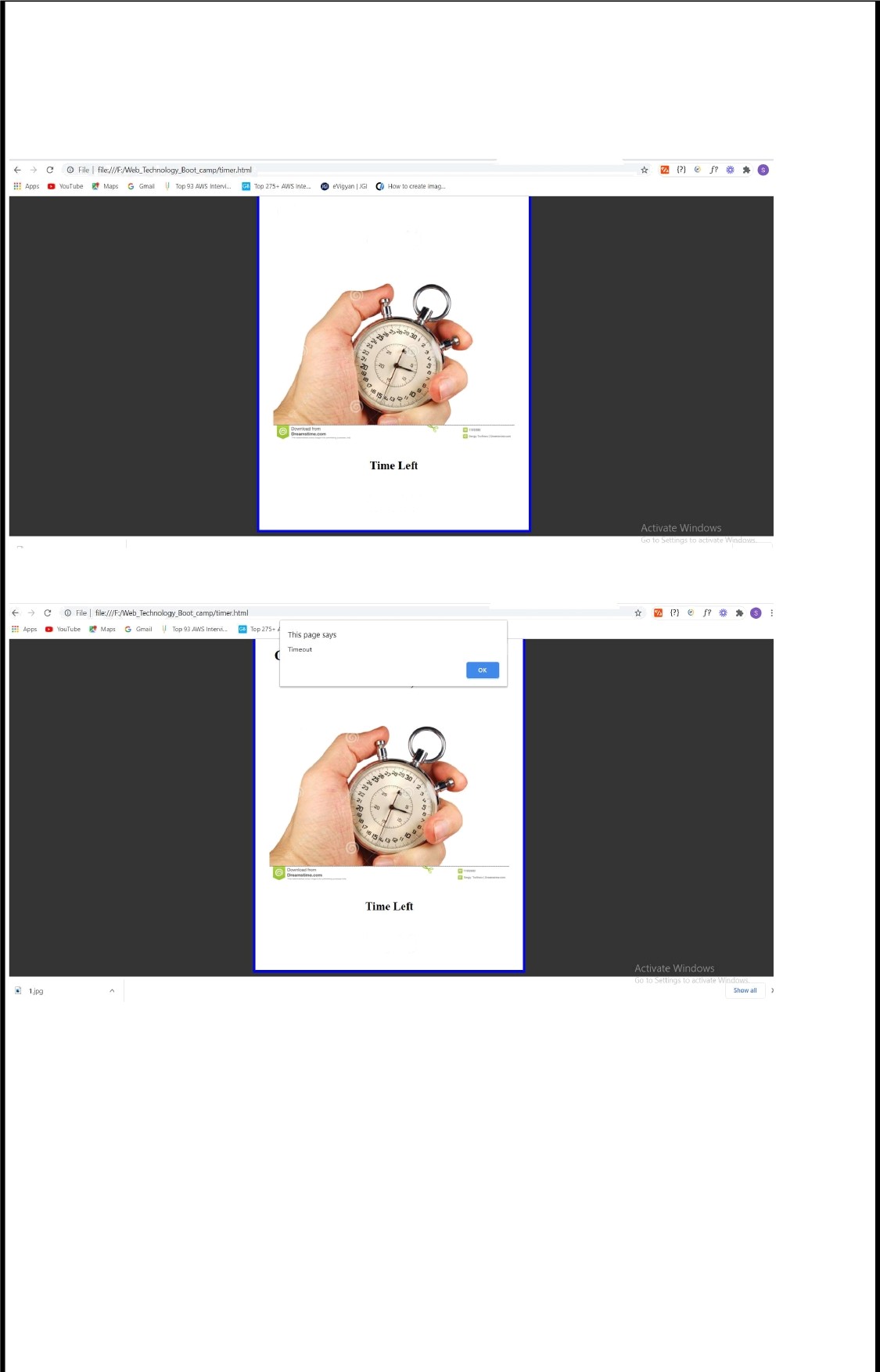
function getseconds() {

return secs - Math.round(mins \* 60);

</script>

</body>

</html>

OUTPUT:

Click the image to sturt the timer (60

seconds)

0 : 37

0 : 28

### 43

Experiment -10



1. Design a web page to capture the student details such as student number, name, age, marks using Java Script Object.

<!D0CTYPE html>

<html>

<head>

<meta charset=utf-8 />

<title>Compute the average marks and grade</title>

</head>

<body>

<script>

var students = [['David', 80], ['Vinoth', 77], ['Divya', 88], ['Ishitha', 95], ['Thomas', 6

]]I

var Avgmarks = 0;

for (var i=0; i < students.length; i++) { Avgmarks += students[i][1];

var avg = (Avgmarks/students.length);

console.log(”Average grade: ” + (Avgmarks)/students.length); if (avg < 60){

console.log(”Grade : F“);

else if (avg < 70) {

console . 1og( "Grade : D" ) ;

else if (avg < 80)

console.log("Grade : C“);

} else if (avg < 90) {

console.log(“Grade : B“);

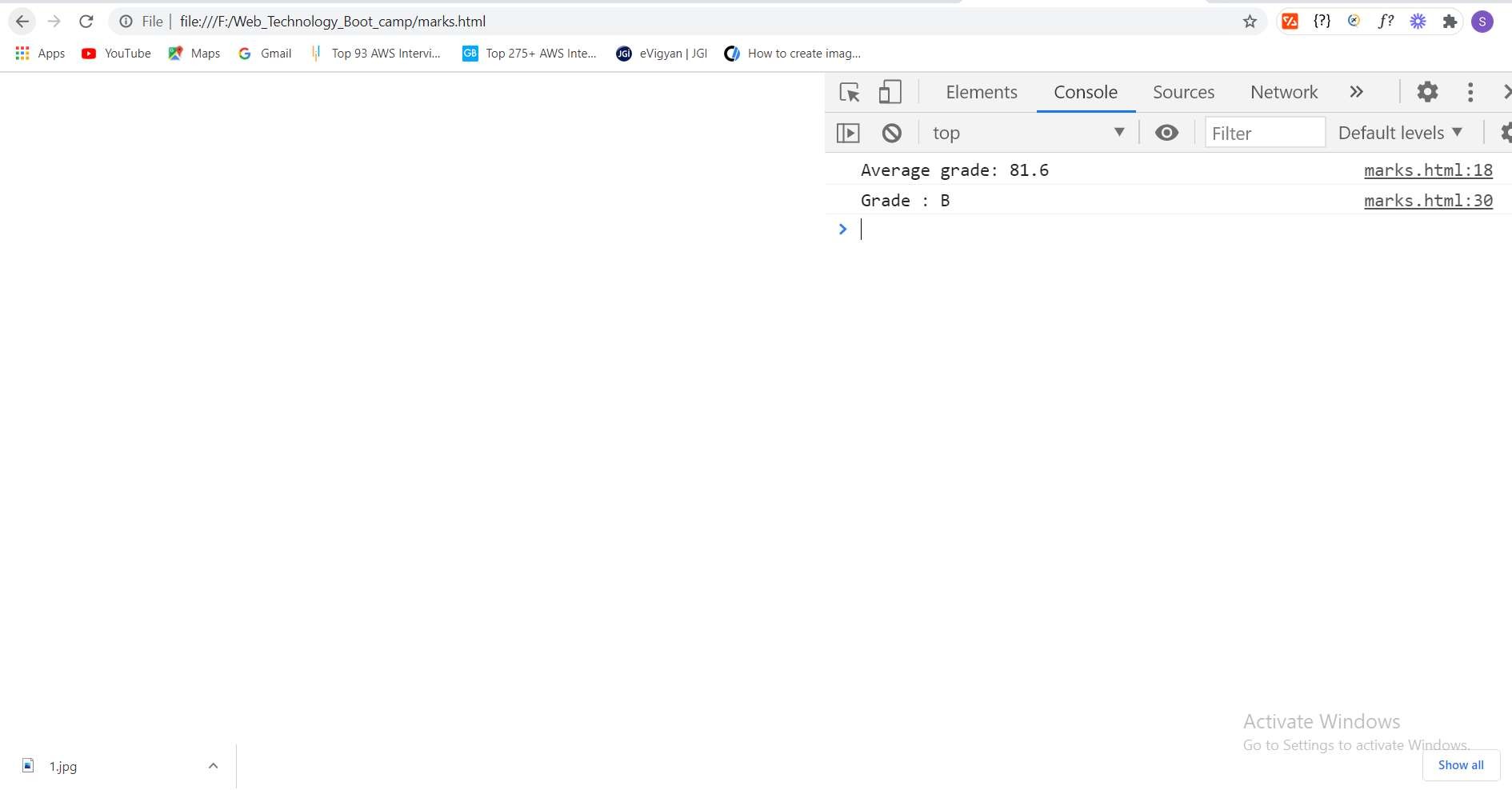
} else if (avg < 100) {

console.log("Grade : A“);

</script>

</body>

</html>



OUTPUT:

Experiment -11

1. Design a web page to read data from an XML file and display the data in tabular format, take the data as employee information.

User.xml

<?xml version=“1.0“ encoding=”UTF-8“?>

<employees>

<employee id=“501“>

<firstName>Sunil</firstName>

<lastName>Yadav</lastName>

<location>Hyd</location>

</employee>

<employee id=“502“>

<firstName>Trilok</firstName>

<lastName>Reddy‹/lastName>

<location>Chevella</location>

</employee>

<employee id=“503“>

<firstName>Mallikarjun</firstName>

<lastName>Tiger</lastName>

<location>Forest</location>

</employee>

<employee id=“504“>

<firstName>Neelima</firstName>

<lastName>Lakshmi</lastName>

<location>MP</location>

</employee>

<employee id=“505“>

<firstName>Abhi</firstName>

<lastName>Shambu</lastName>



<location>LD</location>

</employee>

<employee id=“506“>

<firstName>Nikitha</firstName>

<lastName>Reddy</lastName>

<location>UP</location>

</employee>

<employee id=“507“>

<firstName>Ashwini</firstName>

<lastName>Reddy</lastName>

<location>ADB</location>

</employee>

<employee id=“508“>

<firstName>Vinod</firstName>

<lastName>Kumar</lastName>

<location>MGD</location>

</employee>

<employee id=“509“>

<firstName>Raghu</firstName>

<lastName>Reddy</lastName>

<location>MBD</location>

</employee>

<employee id=“510“>

<firstName>Uma Shankar</firstName>

<lastName>Kommana</lastName>

<location>DSNR</location>

</employee>

</employees>

employeedetails .java

import org.w3c.dom.\*; import javax.xml.parsers.\*; import java.io.\*;

public class employeedetails {

public static void main(String[] args)throws Exception {

//Get Document Builder

DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance(); DocumentBuilder builder = factory.newDocumentBuilder();

//Build Document

Document document = builder.parse(new File(”User.xml”));

//Normalize the XML Structure; It's just too important !!

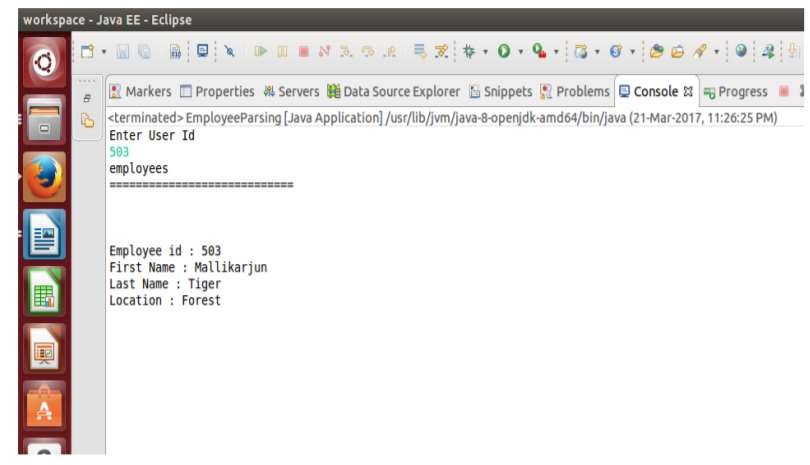
//document.getDocumentElement().normalize();

// TODO Auto-generated method stub

//Here comes the root node

Element root = document.getDocumentElement();

BufferedReader br=new BufferedReader(new InputStreamReader(System.in)); System.out.println(“Enter User Id");



String id=br.readLine();

//Get all employees

NodeList nList = document.getElementsByTagName("employee“); System.out.println(root.getNodeName()); System.out.println(“============================“);

for (int temp = 0; temp < nList.getLength(); temp++)

Node node = nList.item(temp); System.out.println(“”); //Just a separator if (node.getNodeType() == Node.ELEMENT\_N0DE)

//Print employee's details of given id Element eElement = (Element) node; if(eElement.getAttribute("id").equals(id))

System.out.println(”Employee id : ” + eElement.getAttribute(”id")); System.out.println(”First Name : ” + eElement.getElementsByTagName(”firstName”).item(0).getTextContent()); System.out.println(”Last Name : ” + eElement.getElementsByTagName(”lastName").item(0).getTextContent()); System.out.prlntln(”Location : " + eElement.getElementsByTagName(”location").item(0).getTextContent());

OUTPUT :

**Experiment -12**



### Design a web site for online purchase using CSS, JS and XML, web site should contain the following web pages.

#### Home page

* + Login page
  + Signup page
  + Product details page

**Exp12.html**

<!doctype html>

<html lang=”en“>

<head>

<!-- Required meta tags -->

<meta charset=“utf-8“>

<meta name="viewport” content="width=device-width, initial-scale=1, shrink-to-fit=no”>

<!-- Bootstrap CSS -->

<link rel=“stylesheet” href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/boo tstrap.min.css“

1ntegr1ty=" sha384-

9aIt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8WgZl5MYYxFfc+NcPb1dKGj7Sk“ crossorigin=”anonymous”>

<title>Product Details</title>

‹/head ›

‹ body›

‹nav class="navbar navbar-expand-lg navbar-dark bg-dark”›

<a class=“navbar-brand” href=“#”>Product Details</a>

<button class="navbar-toggler” type="button” data-toggle="collapse“ data- 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 mr-auto“>

<li class=“nav-item active“>

<a class=”nav-link“ href="#“>Home <span class=“sr- only“>(current)‹/span></a>

</li>

<li class=“nav-item“>

<a class=”nav-link“ href="#“>About</a>

</li>

<li class=“nav-item dropdown“>

<a class=”nav-link dropdown- toggle“ href=“#“ id=”navbarDropdown“ role="button“

</a>

data-toggle=“dropdown“ aria-haspopup="true“ aria-expanded="false”> Items

<div class=”dropdown-menu“ aria-labelledby="navbarDropdown”>



<a class=“dropdown-item” href="#“>Action</a>

‹a class=“dropdown-item” href="#“>Another action</a>

<div class="dropdown-divider“></div>

‹a class=“dropdown-item” href="#“>Something else here</a>

‹/div>

##### ‹/li>

‹li class=“nav-item“>

‹a class=”nav-link“ href="#“ tabindex=”-1" aria- disabled=“true“>Contact</a>

‹/li>

</ul>

<form class=“form-inline my-2 my-lg-0”>

‹input class=”form-control mr-sm-

2” type=“search“ placeholder=”Search“ aria-label=”Search“>

‹button class=”btn btn-outline-success my-2 my-sm- 0” type=“submit“>Search</button>

‹/form>

</div>

</nav>

<div class="container my-4”>

<h2 class=“text-center”>Product Details</h2>

<div class=“form-group”>

<label for=“title“>Title</label>

<input type=“text“ class=”form-control” id=”title“ aria- describedby="emailHelp“>

‹small id=“emailHelp“ class="form-text text- muted“>Add an item to the list</small>

</div>

<div class=“form-group”>

<label for=“description”>Description</label>

<textarea class="form-control” id=”description” rows=”3"></textarea>

</div>

<button id=“add“ class=“btn btn-primary“>Add to list</button>

<button id=“clear” class=“btn btn- primary“ onclick=“clearStorage()“>Clear list</button>

<div id="items” class=”my-4“>

<h2>Your Items</h2>

<table class=“table”>

‹thead>

<th scope=”col“>SNo</th>

<th scope=”col“>Item Title</th>



<th scope=”col“>Item Description</th>

<th scope=”col“>Actions</th>

</tr>

</thead>

<tbody id=“tableBody“>

<tr>

<th scope=”row“>1</th>

<td>Get some Coffee</td>

<td>You need coffee as you are a coder</td>

<td><button class=“btn btn-sm btn-primary">Delete</button></td>

</tr>

‹/tbody>

</table>

</div>

</div>

<!-- Optional JavaScript -->

<!-- jQuery flrst, then Popper.js, then Bootstrap JS --›

‹script src="https://code.jquery.com/jquery-3.5.1.slim.min.js“

integrity=“sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+0GpamoFVy38MVBnE+IbbVYUew+0rCXaRkfj“ crossorigin=”anonymous”></script>

‹script [src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js“](https://cdn.jsdelivr.net/npm/popper.js%401.16.0/dist/umd/popper.min.js) integrity=“sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9I0Yy5n3zV9zzTtmI3UksdQRVvoxMfooAo” crossorigin=”anonymous”>‹/script>

<script src=“https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js“ integrity=“sha384-0gVRvuATP1z7JjHLku0U7Xw704+h835Lr+6QL9UvYjZE3Ipu6Tp75j7Bh/kR0JKI“ crossorigin=”anonymous”></script>

<script›

function getAndUpdate(){ console.log("Updating List...");

tit = document.getElementById('title').value;

desc document.getElementById('description').value; if (localstorage.getItem('ltemsJson')==null){

itemJsonArray = []; itemJsonArray.push([tit, desc]);

localstorage.setItem('itemsJson', JSON.stringify(itemJsonArray))

else{

itemJsonArrayStr localstorage.getItem('itemsJson') itemJsonArray = JSON.parse(itemJsonArrayStr); itemJsonArray.push([tit, desc]); localstorage.setItem('itemsJson', JSON.stringify(itemJsonArray))

update();

function update(){

if (localstorage.getItem('itemsJson')==null){ itemJsonArray = [];

localstorage.setItem('itemsJson', JSON.stringify(itemJsonArray))

else{



itemJsonArrayStr localstorage.getItem('itemsJson') itemJsonArray = JSON.parse(itemJsonArrayStr);

// Populate the table

let tableBody = document.getElementById("tableBody“); let str = "“;

itemJsonArray.forEach((element, index) =› { str +=

<tr>

<th scope=”row”>${index + 1}</th>

<td>${element[0]}</td>

<td>${element[1]}</td>

<td><button class="btn btn-sm btn- primary" onclick=”deleted(${index})">Delete</button></td>

</tr> ;

tableBody.innerHTML = str;

add = document.getElementById(”add"); add.addEventListener(”click”, getAndUpdate); update();

function deleted(itemIndex){ console.log("Delete“, itemIndex);

itemJsonArrayStr localstorage.getItem('itemsJson') itemJsonArray = JSON.parse(itemJsonArrayStr);

// Delete itemIndex element from the array itemJsonArray.splice(itemIndex, 1); localstorage.setItem('itemsJson', JSON.stringify(itemJsonArray)); update();

function clearstorage(){

if (confirm("Do you areally want to clear?")){ console.log('Clearing the storage') localStorage.clear();

update()

</script>

</body>

</html>

**OUTPUT:**

Product Details

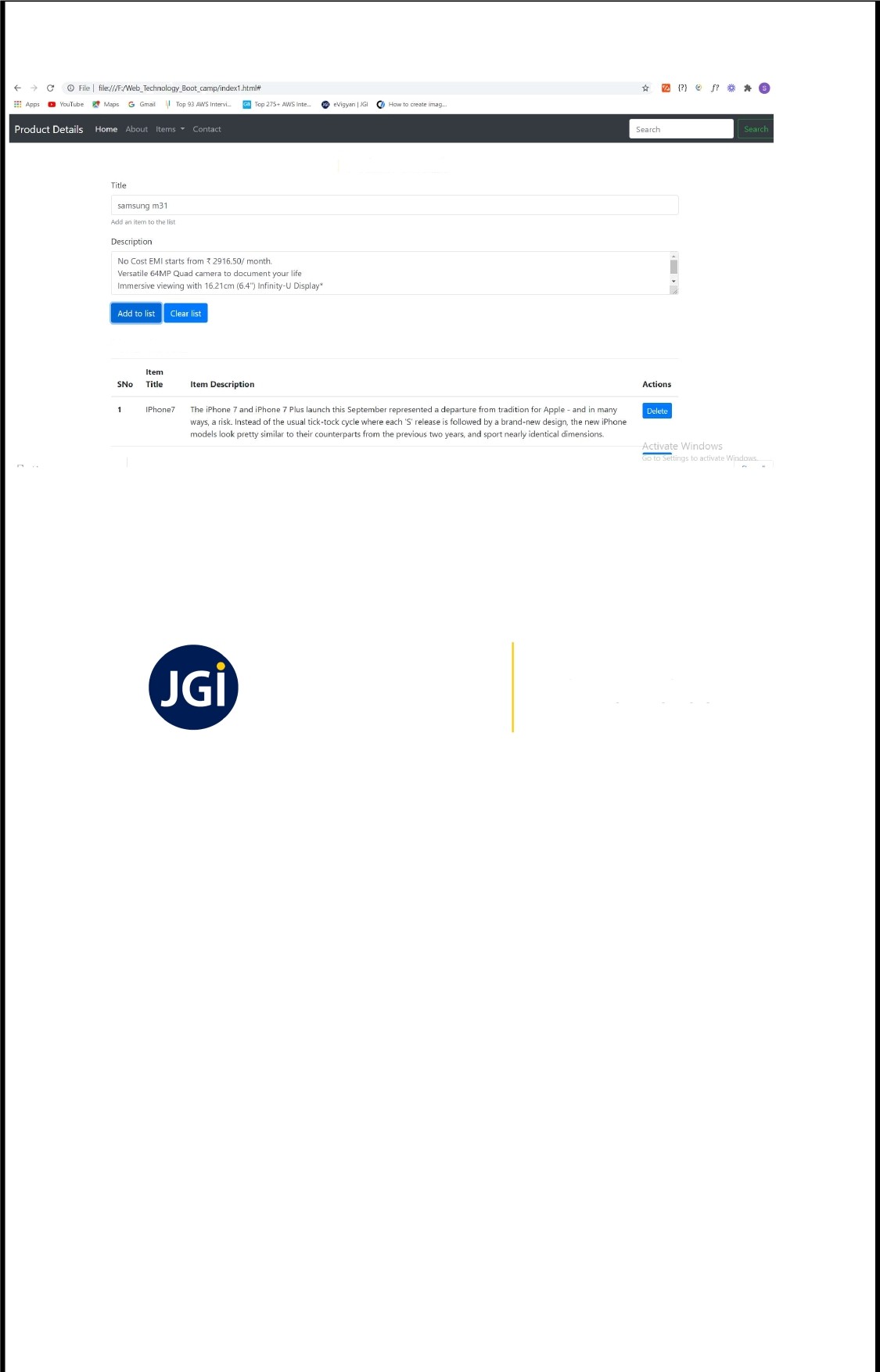
Your Items

DEEMED-TO-BE UNIVERSITY

# FACULTY OF ENGINEERING

AND TECHNOLOGY

JAIN



#### 52