

List of Programs

	Date	Page
1 Formatting Tags	1	
2 Images and Hyperlinks	2	
3 Tables	8	
4 Lists	12	
5 Forms	15	
6 CSS	18	
7 CSS Selectors	20	
8 Bootstrap CSS	22	
9 Calculator using JavaScript	25	
10 Reversing an Array	27	
11 TODO List	29	
12 Query Selector	31	
13 Animation	33	
14 Event Handler	35	
15 Library Database using Django	38	
16 Sessions	43	

Exercise 1

Formatting Tags

Problem Definition

Create a web page using basic HTML formatting tags.

Source Code

```
<html>
  <head>
    <title>FORMATTING TAGS</title>
  </head>
  <body bgcolor="skyblue" background="water.jpg">
    <center>
      <big>
        <strong>WATER</strong>
      </big>
      <small>
        (H<sub>2</sub>O)
      </small>
    </center>
    <b>Water</b> is an <i>inorganic compound</i> with the chemical formula H<sub>2</sub>O. It is a <u>transparent, tasteless, odorless, and nearly colorless</u> chemical substance, and it is the main constituent of Earth's <del> stratosphere</del> <ins> hydrosphere </ins> and the fluids of all known living organisms. It is vital for all known forms of life, despite not providing food energy, or organic micronutrients. The hydrogen atoms are attached to the oxygen atom at an angle of 104.45o. <mark>"Water"</mark> is also the name of the liquid state of H<sub>2</sub>O at standard temperature and pressure.
  </body>
</html>
```

Output

WATER (H_2O)
Water is an *inorganic compound* with the chemical formula H_2O . It is a *transparent, tasteless, odorless, and nearly colorless* chemical substance, and it is the main constituent of Earth's ~~stratosphere~~ hydrosphere and the fluids of all known living organisms. It is vital for all known forms of life, despite not providing food energy, or organic micronutrients. The hydrogen atoms are attached to the oxygen atom at an angle of 104.45°. "Water" is also the name of the liquid state of H_2O at standard temperature and pressure.



Exercise 2

Images and Hyperlinks

Problem Definition

Create a web page containing images and hyperlinks.

Source Code

about.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <title>Photo Gallery</title>
    <link rel="stylesheet" href="style1.css" />
  </head>
  <body>
    <div class="container">
      <div class="heading">
        <h2 class="heading-text">Explore Kasaragod</h2>
      </div>
      <ul class="image-gallery">
        <li>
          <a href="bekal.html">
            
            <div class="overlay">BEKAL FORT</div>
          </a>
        </li>
        <li>
          <a href="ranipuram.html">
            
            <div class="overlay">RANIPURAM</div>
          </a>
        </li>
        <li>
          <a href="malom.html">
            
            <div class="overlay">MALOM THE COORG OF KERALA</div>
          </a>
        </li>
        <li>
          <a href="kottancheri.html">
            
            <div class="overlay">KOTTANCHERI HILL</div>
          </a>
        </li>
      </ul>
    </div>
  </body>
</html>
```

bekal.html

EXERCISE 2. IMAGES AND HYPERLINKS

```
<html>
  <head>
    <title>Bekal Fort</title>
    <link rel="stylesheet" href="style1.css" />
    <style>
      #caption{
        background-image:url(bekalheader.jpg);
        color:brown ;
        height: 300px;
        width: 100%;
        background-repeat: no-repeat;
        background-position: center;
        align-items: center;
        text-align: center;
      }
    </style>
  </head>
  <body>
    <div id="caption">
      <h2>Bekal Fort-The Largest Fort in Kerala</h2>
    </div>
    <div class="content">
      <p class=intro> Bekal fort is the largest and one of the best preserved forts in Kerala.The 300 plus year old fort is a coastal fort lying 16 km south-east of Kasargod situated on the backdrop of Arabian seashore of Pallikkara village in the Kasargod district of North Kerala. The Bekal fort spreads over 40 acres (160,000 m2) of land. Due to its historic relevance, the fort is controlled by the Archeological Survey of India.</p>
      <h4 class=heading>THINGS TO SEE AT BEKAL FORT </h4>
      <ol class=intro>
        <li>Temple : <p>Anjaneya Temple at Nanganallur, Chennai is a Hindu temple dedicated to the god Hanuman. The idol was installed in 1989 and consecrated in 1995</p></li>
        <li>Malik Deena Mosque : <p>Malik Deena Mosque with a typical Malabar style of architecture was built by Malik Ibn Deenar. Considered to be one of the most well-kept and an attractive mosque of Kasaragod District. The place is a must-visit especially during the Uroos festival which is celebrated with great fervor.</p></li>
        <li>Underground Passages : <p>One can find many underground passages leading outside from various parts of the Fort. Two of them remain intact, one leading to the south of the fort and the other to the moat on the east.</p></li>
        <li>Sea Bastion : <p>The historic fort was endowed with impressive walls and ramparts which are hindered by massive bastions.</p></li>
      </ol>
    </div>
  </body>
</html>
```

malom.html

```
<html>
  <head>
    <title>Malom</title>
    <link rel="stylesheet" href="style1.css" />
    <style>
      #caption{
        background-image:url(malom.jpg);
        color:white;
        height: 300px;
        width: 100%;
        background-repeat: no-repeat;
        background-position: center;
        align-items: center;
        text-align: center;
      }
    </style>
  </head>
  <body>
```

EXERCISE 2. IMAGES AND HYPERLINKS

```
<div id="caption">
    <h2>Malom: Coorg of Kerala </h2>
</div>
<div class="content">
    <p class="intro"> Malom is a small town that lies lazily near the Western Ghats in Kasaragod district. Known as the 'Coorg' of Kerala, it is blessed with Nature's bounty. The word 'Malom' comes from the Malayalam word Malakalude lokam (the world of hills). Malom can be developed into a bio-eco tourist centre given the fact that it offers possibilities for trekking, waterfalls, and adventure safari to the nearby hills.</p>
    <p class="intro">Malom has a rich cultural heritage, which is quite distinct. A rare Theyyam of north Malabar, the 'Mukri Pokar', a Muslim Theyyam is seen at Malomkuloth. As it is situated near the Western Ghats and bordering Coorg, it is lush green with the same climatic conditions as that of Coorg. </p>
    <h4 class="heading">Things to do </h4>
    <ol class="intro">
        <li>Mountain biking</li>
        <li>Mountain trekking</li>
        <li>Eco trips</li>
        <li>Mountain village trails</li>
        <li>Marutham Thattu trekking</li>
    </ol>
</div>
</body>
</html>
```

kottancheri.html

```
<html>
    <head>
        <title>Kottancheri</title>
        <link rel="stylesheet" href="style1.css" />
        <style>
            #caption{
                background-image:url(kottancheri.jpg);
                color:brown ;
                height: 300px;
                width: 100%;
                background-repeat: no-repeat;
                background-position: center;
                align-items: center;
                text-align: center;
            }
        </style>
    </head>
    <body>
        <div id="caption">
            <h2>Kottancherry Hills</h2>
        </div>
        <div class="content">
            <p class="intro"> Kottancherry Hills is a group of hills close to the Brahmagiri Hills of Kudagu in Karnataka and within the Western Ghats region in Kasaragod district. It is believed that the hills - Kottancherry mala, Kumbar mala, Pannyar mala, and Kanthanppara are in the shape of a fort, hence the name Kottancherry.
            To reach Kottancherry Hills, one has to travel eight kms from Konnakad in Kasargod. From Kumbar hills, one can enjoy the beauty of Kudagu Hills. From Malom Hills on Kerala-Karnataka border, it is 3 kms to Konnakadu. Kottancherry Hills trekking has all the ingredients of an adventure tourism destination. Walking 11 kms through the Kottancherry Forests will take you to Thalakaveri in Karnataka. Thalakaveri, which is the beginning of the Kaveri River, is best suited for river rafting and other adventurous expeditions.
        </p>
        <h4 class="heading">Things to do</h4>
        <ol class="intro">
            <li>Kottancherry Hills trekking</li>
            <li>Hill-village and social/tribal life</li>
            <li>Kottancherry-Thalakaveri adventure trekking</li>
        </ol>
    </div>
</body>
</html>
```

EXERCISE 2. IMAGES AND HYPERLINKS

```
<li>Mountain biking</li>
</ol>
</div>
</body>
</html>
```

ranipuram.html

```
<html>
<head>
    <title>Ranipuram Trek: Ooty of Kerala</title>
    <link rel="stylesheet" href="style1.css" />
    <style>
        #caption{
            background-image:url(ranipuram.jpg);
            color:white;
            height: 300px;
            width: 100%;
            background-repeat: no-repeat;
            background-position: center;
            align-items: center;
            text-align: center;
        }
    </style>
</head>
<body>
    <div id="caption">
        <h2>Ranipuram Trek: Ooty of Kerala </h2>
    </div>
    <div class="content">
        <p class=intro> In the Kasargod district of Kerala is Ranipuram, popularly known as the `Ooty of Kerala' and previously called Madathumala. It is an offbeat trekking destination, which is sparsely populated and provides panoramic views as you walk across misty, lush green grasslands. Due to the length of the trail, Ranipuram makes for a perfect one-day getaway if you reside in or around Mangalore or in the northern part of Kerala.
        <h4 class=heading>Where is Ranipuram </h4>
        <p class=intro>Where is Ranipuram
        The trail starts from the DTTC (District Tourism Promotion Council) guest house, located at the base of Ranipuram hills. One can reach there by getting a jeep from Panathady, 10km away from Ranipuram.</p>
        <ol class=intro>
            <li>Nearest railway station : <p> Kanhangad (45km from Ranipuram)</p></li>
            <li>Nearest airport : <p>Mangalore International Airport (113km away from Ranipuram)</p></li>
        </ol>
    </div>
</body>
</html>
```

style1.css

```
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}
p{
    padding:15px;
    text-align: justify;
    font-family: "arial", sans-serif;
}
body{
    font-family: "arial black", sans-serif;
    background-color: antiquewhite;
}
```

EXERCISE 2. IMAGES AND HYPERLINKS

```
.container{
  padding: 0px 10%;
}

.heading-text {
  margin-bottom: 2rem;
  font-size: 2rem;
  padding: 10px;
  text-align: center;
  background-color: #b3cde3;
}

ul {
  list-style: none;
}

.content {
  padding: 20px;
  margin: 10px 10%;
  font-size: large;
  font-family: 'Lucida Sans', sans-serif;
  background-color: antiquewhite;
}

.image-gallery {
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  gap: 20px;
}

.image-gallery > li {
  flex-basis: 300px;
  position: relative;
  cursor: pointer;
}

.image-gallery li img {
  object-fit: cover;
  max-width: 100%;
  height: 100%;
  vertical-align: middle;
  border-radius: 5px;
}

.overlay {
  position: absolute;
  width: 100%;
  height: 100%;
  background: rgba(121, 113, 113, 0.502);
  top: 0;
  left: 0;
  transform: scale(0);
  color: #fff;
  border-radius: 5px;
  display: flex;
  align-items: center;
  justify-content: center;
}

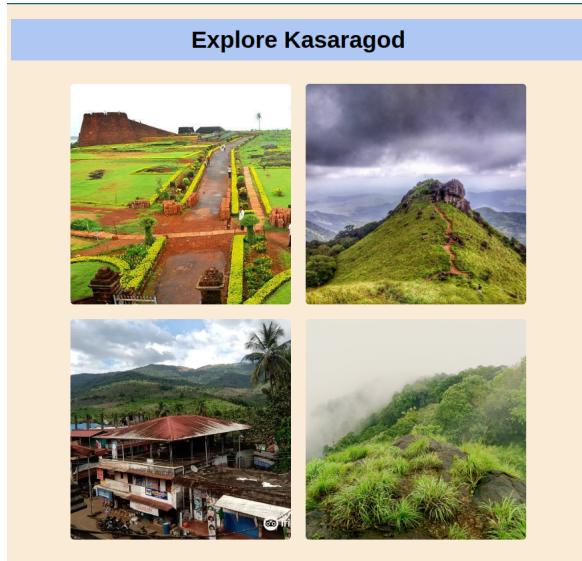
.image-gallery li:hover .overlay {
  transform: scale(1);
}

.heading {
  margin-top: 20px;
  text-align: center;
}

.intro {
  padding: 10px 20%;
}
```

EXERCISE 2. IMAGES AND HYPERLINKS

Output



Bekal fort is the largest and one of the best preserved forts in Kerala. The 300 plus year old fort is a coastal fort lying 16 km south-east of Kasaragod situated on the backdrop of Arabian seashore of Pallickara village in the Kasaragod district of North Kerala. The Bekal fort spreads over 40 acres (160,000 m²) of land. Due to its historic relevance, the fort is controlled by the Archeological Survey of India.

THINGS TO SEE AT BEKAL FORT

1. Temple :

Anjaneya Temple at Nanganallur, Chennai is a Hindu temple dedicated to the god Hanuman. The idol was installed in 1989 and consecrated in 1995

2. Malik Deena Mosque :

Malik Deena Mosque with a typical Malabar style of architecture was built by Malik ibn Deenar. Considered to be one of the most well-kept and an attractive mosque of Kasaragod District. The place is a must-visit especially during the Uroos festival which is celebrated with great fervor.

3. Underground Passages :

One can find many underground passages leading outside from various parts of the Fort. Two of them remain intact, one leading to the south of the fort and the other to the moat on the east.

4. Sea Bastion :

The historic fort was endowed with impressive walls and ramparts which are hindered by massive bastions.

Exercise 3

Tables

Problem Definition

Create a web page containing table.

Source Code

```
<!DOCTYPE html>
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Mark List</title>
    <style>
        table, tr, td, th{
            border: 2px solid black;
            text-align: center;
        }
        th{
            font-weight: bold;
        }
    </style>
</head>
<body>
    <h2 style="text-align: center;">KANNUR UNIVERSITY</h2>
    <h3 style="text-align: center;">(Examination Branch)</h3>
    <p style="padding: 3rem 1rem; width:59rem;">
        Programme : <b>Integrated M.Sc in Computer Science with Specialization in Artificial Intelligence and Machine Learning</b><br>
        Semester : <b>Fourth Semester, 2023</b><br>
        College : Nehru Arts and Science College, Kanhangad<br>
        Name : <b>ABCD</b><br>
        Reg No : <b>ZZ20CCSS23</b>
    </p>
    <center>
        <table style="width:59rem">
            <tr>
                <th rowspan="2">Course Code</th>
                <th rowspan="2" colspan="2">Courses Title</th>
                <th rowspan="2">Cr.</th>
                <th rowspan="2">Max. Marks</th>
                <th colspan="3">Marks Awarded</th>
                <th rowspan="2">G.P</th>
                <th rowspan="2">G</th>
                <th rowspan="2">C.P</th>
                <th rowspan="2">Result</th>
            </tr>
            <tr>
                <th>CE</th>
                <th>ESE</th>
                <th>Total</th>
            </tr>
            <tr>
                <th colspan="12" style="text-align: center; padding-left: 2rem;">CORE COURSES</th>
            </tr>
        </table>
    </center>
</body>
```

EXERCISE 3. TABLES

```
<tr>
    <td>4B09ICSC</td>
    <td colspan="2">Computer Organization</td>
    <td>3</td>
    <td>50</td>
    <td>9</td>
    <td>24</td>
    <td>33</td>
    <td>6.6</td>
    <td>C</td>
    <td>19.8</td>
    <td>P</td>
</tr>
<tr>
    <td>4B10ICSC</td>
    <td colspan="2">Database Management System</td>
    <td>3</td>
    <td>50</td>
    <td>9</td>
    <td>34</td>
    <td>43</td>
    <td>8.6</td>
    <td>A</td>
    <td>25.8</td>
    <td>P</td>
</tr>
<tr>
    <td>4B11ICSC</td>
    <td colspan="2">Object Oriented Programming using Java</td>
    <td>3</td>
    <td>50</td>
    <td>7</td>
    <td>21</td>
    <td>28</td>
    <td>5.6</td>
    <td>D</td>
    <td>16.8</td>
    <td>P</td>
</tr>
<tr>
    <td>4B12ICSC</td>
    <td colspan="2">Lab 4: Object Oriented Programming using Java</td>
    <td>2</td>
    <td>25</td>
    <td>5</td>
    <td>19</td>
    <td>24</td>
    <td>9.6</td>
    <td>A+</td>
    <td>19.2</td>
    <td>P</td>
</tr>
<tr>
    <td>4B13ICSC</td>
    <td colspan="2">Lab 5: Database Management System</td>
    <td>2</td>
    <td>25</td>
    <td>5</td>
    <td>20</td>
    <td>25</td>
    <td>10.0</td>
    <td>A+</td>
    <td>20.0</td>
    <td>P</td>
</tr>
```

EXERCISE 3. TABLES

```
<tr>
    <th colspan="12" style="text-align:center; padding-left: 2rem;">
        COMPLEMENTARY ELECTIVE COURSE
    </th>
</tr>
<tr>
    <td>4C04STA-
    ICS</td>
    <td colspan="2">Statistical Inference</td>
    <td>3</td>
    <td>50</td>
    <td>10</td>
    <td>2</td>
    <td>12</td>
    <td>2.4</td>
    <td>F</td>
    <td>7.2</td>
    <td>F</td>
</tr>
<tr>
    <td>4C04MAT-
    ICS</td>
    <td colspan="2">Foundation Mathematics for Machine Learning II</td>
    <td>3</td>
    <td>50</td>
    <td>10</td>
    <td>7</td>
    <td>17</td>
    <td>3.4</td>
    <td>F</td>
    <td>10.2</td>
    <td>F</td>
</tr>
<tr>
    <td>Total</td>
    <td colspan="2"></td>
    <td>19</td>
    <td>300</td>
    <td colspan="2">****</td>
    <td>182</td>
    <td colspan="2">***</td>
    <td>-</td>
    <td> </td>
</tr>
<tr>
    <td colspan="2">Total Marks (%)</td>
    <td colspan="2">-</td>
    <td colspan="2">SGPA</td>
    <td colspan="2">-</td>
    <td colspan="2">Grade</td>
    <td colspan="2">-</td>
    <td colspan="2">Failed</td>
</tr>
</table>
</center>
</body>
</html>
```

EXERCISE 3. TABLES

Output

KANNUR UNIVERSITY

(Examination Branch)

Programme : **Integrated M.Sc in Computer Science with Specialization in Artificial Intelligence and Machine Learning**

Semester : **Fourth Semester, 2023**

College : Nehru Arts and Science College, Kanhangad

Name : **ABCD**

Reg No : **ZZ20CCSS23**

Course Code	Courses Title	Cr.	Max. Marks	Marks Awarded			G.P	G	C.P	Result
				CE	ESE	Total				
CORE COURSES										
4B09ICSC	Computer Organization	3	50	9	24	33	6.6	C	19.8	P
4B10ICSC	Database Management System	3	50	9	34	43	8.6	A	25.8	P
4B11ICSC	Object Oriented Programming using Java	3	50	7	21	28	5.6	D	16.8	P
4B12ICSC	Lab 4: Object Oriented Programming using Java	2	25	5	19	24	9.6	A+	19.2	P
4B13ICSC	Lab 5: Database Management System	2	25	5	20	25	10.0	A+	20.0	P
COMPLEMENTARY ELECTIVE COURSE										
4C04STA- ICS	Statistical Inference	3	50	10	2	12	2.4	F	7.2	F
4C04MAT- ICS	Foundation Mathematics for Machine Learning II	3	50	10	7	17	3.4	F	10.2	F
Total			19	300	***	182	***	-	-	
Total Marks(%)	-			SGPA	-		Grade	-		Failed

Exercise 4

Lists

Problem Definition

Create a web page containing all types of lists.

Source Code

```
<!DOCTYPE html>
<head>
    <title>Restaurant</title>
</head>
<style>
    body{
        font-family: "Poppins", sans-serif;
    }
    #desc
    {
        font-size:30px;
        font-weight: bold;
        text-align:center;
        font-family: 'Calibri', sans-serif;
        margin:20px auto 2px;
        width:80%;
        background-color: #f4c531;
        padding:10px 2px;
    }
    #recipe-desc
    {
        font-size: 16px;
        width: 80%;
        margin:auto;
        line-height: 1.5;
        background-color: antiquewhite;
        font-family: 'Lucida Sans', Arial, sans-serif;
    }
    .sub-heading{
        background-color: navy;
        padding: 10px;
        font-family: Arial, Helvetica, sans-serif;
        font-size: 1.5rem;
        font-weight: bold;
        color:white;
    }
    img{
        display: block;
        margin:auto;
        border:2px solid;
    }
</style>
<body>
    
    <div id="desc">
        American Pancakes
    </div>
```

EXERCISE 4. LISTS

```
<div id="recipe-desc">
  <div class="sub-heading">
    Ingredients
  </div>
  <ul type="disc">
    <li>200g self-raising flour</li>
    <li>1.5 tsp baking powder</li>
    <li>1 tbsp golden caster sugar</li>
    <li>3 large eggs</li>
    <li>25g melted butter, plus extra for cooking</li>
    <li>200ml milk</li>
    <li>vegetable oil, for cooking</li>
  </ul>
  <div class="sub-heading">
    Method
  </div>
  <ol>
    <li>STEP 1</li>
    Mix 200g self-raising flour, 1.5 tsp baking powder, 1 tbsp golden caster sugar and a
    pinch of salt together in a large bowl.
    <li>STEP 2</li>
    Create a well in the centre with the back of your spoon then add 3 large eggs, 25g melted
    butter and 200ml milk.
    <li>STEP 3</li>
    Whisk together either with a balloon whisk or electric hand beaters until smooth then
    pour into a jug.
    <li>STEP 4</li>
    Heat a small knob of butter and 1 tsp of oil in a large, non-stick frying pan over a
    medium heat. When the butter looks frothy, pour in rounds of the batter, approximately 8cm
    wide. Make sure you don't put the pancakes too close together as they will spread during
    cooking. Cook the pancakes on one side for about 1-2 mins or until lots of tiny bubbles start
    to appear and pop on the surface. Flip the pancakes over and cook for a further minute on
    the other side. Repeat until all the batter is used up.
    <li> STEP 5</li>
    Serve your pancakes stacked up on a plate with a drizzle of maple syrup and any of your
    favourite toppings.
  </ol>
  <div class="sub-heading">
    To serve
  </div>
  <dl>
    <dt><b>maple syrup</b></dt>
    <dd> toppings of your choice, such as cooked bacon, chocolate chips, blueberries or
    peanut butter and jam</dd>
  </dl>
</div>
</body>
</html>
```

Output**American Pancakes****Ingredients**

- 200g self-raising flour
- 1.5 tsp baking powder
- 1 tbsp golden caster sugar
- 3 large eggs
- 25g melted butter, plus extra for cooking
- 200ml milk
- vegetable oil, for cooking

Method

1. STEP 1
Mix 200g self-raising flour, 1 ½ tsp baking powder, 1 tbsp golden caster sugar and a pinch of salt together in a large bowl.
2. STEP 2
Create a well in the centre with the back of your spoon then add 3 large eggs, 25g melted butter and 200ml milk.
3. STEP 3
Whisk together either with a balloon whisk or electric hand beaters until smooth then pour into a jug.
4. STEP 4
Heat a small knob of butter and 1 tsp of oil in a large, non-stick frying pan over a medium heat. When the butter looks frothy, pour in rounds of the batter, approximately 8cm wide. Make sure you don't put the pancakes too close together as they will spread during cooking. Cook the pancakes on one side for about 1-2 mins or until lots of tiny bubbles start to appear and pop on the surface. Flip the pancakes over and cook for a further minute on the other side. Repeat until all the batter is used up.
5. STEP 5
Serve your pancakes stacked up on a plate with a drizzle of maple syrup and any of your favourite toppings.

To serve**maple syrup**

toppings of your choice, such as cooked bacon, chocolate chips, blueberries or peanut butter and jam

Exercise 5

Forms

Problem Definition

Create a form with at least 5 form elements and appropriate validation.

Source Code

EXERCISE 5. FORMS

```
<th>Upload Marlist</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Class X</td>
<td><input type="text" name="boardx" required></td>
<td><input type="text" name="perx" required></td>
<td><input type="number" min="1900" max="2099" step="1" required/></td>
<td><input type="file" id="myFilex" name="filenamex" required /></td>
</tr>
<tr>
<td>2</td>
<td>Class XII</td>
<td><input type="text" name="boardxii" required /></td>
<td><input type="text" name="perxii" required></td>
<td><input type="number" min="1900" max="2099" step="1" required/></td>
<td><input type="file" id="myFilexii" name="filenamexii" /></td>
</tr>
</tbody>
</table>
</fieldset>
<div style="margin-top: 1rem;">
<input type="checkbox" value="ok" required/>
I have read and agreed to terms and conditions
<br />
<center><input type="submit" class="btn" value="Submit" /></center>
</div>
</form>
</div>
</div>
</body>
</html>
```

style.css

```
body{
background-color: antiquewhite;
width:80%;
margin:auto;
}
h1{
position: static;
text-align:center;
color: blue;
}
h2{
position: relative;
color: red;
}
.section1{
text-align:center;
}
li{
color:chocolate;
}
label{
width:200px;
display: inline-block;
}
```

Output

COLLEGE ADMISSION FORM

Enter your admission information below

Personal Information

First Name:	<input type="text"/>
Middle Name:	<input type="text"/>
Last Name:	<input type="text"/>
Date of Birth:	<input type="text"/> dd/mm/yyyy <input type="button" value="..."/>
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input type="radio"/> Other
Address:	<input type="text"/>
Email:	<input type="text"/>
Mobile number:	<input type="text"/>

Education Information

SI No.	Examination	Board	Percentage	Year of Passing	Upload Marklist
1	Class X	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="Choose file"/> No file chosen
2	Class XII	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="Choose file"/> No file chosen

I have read and agreed to terms and conditions

Exercise 6

CSS

Problem Definition

Create a web page using

- (a) In-line CSS
- (b) Internal CSS
- (c) External CSS

Note: use positioning css attributes

Source Code

```
<!DOCTYPE html>
<html>
  <head>
    <title>Nehru Arts and Science College Kanhangad</title>
    <style>
      p{
        background-color:skyblue;
        padding:10px;
        width:80%;
        margin:auto;
        line-height:20pt;
        font-size:14pt;
        text-align:justify;
        color:black;
      }
      .section1
      {
        text-align:center;
      }
      li
      {
        color:chocolate;
      }
    </style>
  </head>

  <body >
    <h1>Nehru Arts and Science College Kanhangad</h1>
    <div id="section1" style="text-align:center">
      
    </div>
    <h2>About Us</h2>
    <p>Early in <i>1964</i>, the enlightened public of <b>Kanhangad and Nileshwar</b> resolved to honour the memory of Jawaharlal Nehru, the architect of modern India by undertaking educational and cultural activities in this educationally backward area. In pursuit of this idea, a dedicated and learned team set up in 1966, an educational society known as "The Nehru Memorial Education Society" under the distinguished presidentship of the late Shri. C. K. Nair following which the society was duly registered. The establishment of Nehru Arts & Science College, Kanhangad in this backward coastal tract was the major task the society had set before themselves and had subsequently accomplished it.</p>
```

EXERCISE 6. CSS

```
<hr>
<h2>Course Offered</h2>
<ul>
  <li>B.A. Economics</li>
  <li>B.A. History</li>
  <li>B.Com.</li>
  <li>B.Sc. Mathematics</li>
  <li>B.Sc. Physics</li>
</ul>
</body>
</html>
```

style2.css

```
body{
  background-color: antiquewhite;
  width:60%;
  margin:auto;
}
h1{
  position: static;
  text-align:center;
  color: blue;
}
h2{
  position: relative;
  color: red;
}
```

Output

Nehru Arts and Science College Kanhangad



About Us

Early in 1964, the enlightened public of **Kanhangad and Nileswar** resolved to honour the memory of Jawaharlal Nehru, the architect of modern India by undertaking educational and cultural activities in this educationally backward area. In pursuit of this idea, a dedicated and learned team set up in 1966, an educational society known as "The Nehru Memorial Education Society" under the distinguished presidency of the late Shri. C. K. Nair following which the society was duly registered. The establishment of Nehru Arts & Science College, Kanhangad in this backward coastal tract was the major task the society had set before themselves and had subsequently accomplished it.

Course Offered

- B.A. Economics
- B.A. History
- B.Com.
- B.Sc. Mathematics
- B.Sc. Physics

Exercise 7

CSS Selectors

Problem Definition

Create a web page using various CSS selectors.

Source Code

```
<!DOCTYPE html>
<head>
    <title>Random color generator</title>
    <style>
        body{
            background-color: #000000;
            color: #ffffff;
            font-family: sans-serif;
            text-align: center;
        }
        .color-generator {
            margin-top: 100px;
        }
        #color {
            font-size: 50px;
            font-weight: bold;
        }
        button {
            background-color: #ffffff;
            color: #000000;
            font-size: 20px;
            padding: 10px 20px;
            border: none;
            border-radius: 5px;
            margin-top: 20px;
            transition: 0.5s ease-in-out;
            cursor: pointer;
        }
        button:hover {
            transform: scale(1.1)
        }
        button:active {
            transform: scale(.9)
        }
    </style>
</head>
<body>
    <h1>Random color generator</h1>
    <div class="color-generator">
        <p id="color"></p>
        <button onclick="generateColor()">Generate New Color</button>
    </div>
    <script>
        function generateColor() {
            let color = '#';
            let digits = '0123456789ABCDEF';
            for (let i = 0; i < 6; i++) {
```

EXERCISE 7. CSS SELECTORS

```
// generate a random number between 0 and 15
let randomDigit = Math.floor(Math.random() * 16);
// append the random number to the color string
color += digits[randomDigit];
}
// set the text value and background color to the random color
document.getElementById('color').innerHTML = color;
document.body.style.backgroundColor = color;
}
// call the function when the page loads
randomColor();
</script>
</body>
</html>
```

Output



Exercise 8

Bootstrap CSS

Problem Definition

Create a web page using bootstrap css framework

Source Code

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="css/bootstrap.min.css">
    <title>Simple Bootstrap Web Page</title>
</head>
<body>
    <a class="navbar-brand" href="#"></a><br>
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
        <ul class="navbar-nav">
            <li class="nav-item active">
                <a class="nav-link" href="#" onclick="showContent('home')">Home </a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="#" onclick="showContent('About')">About</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="#" onclick="showContent('Departments')">Departments</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="#" onclick="showContent('Contact')">Contact</a>
            </li>
        </ul>
    </nav>

    <div id="home" class="container mt-4">
        <div class="jumbotron">
            <h1 class="display-4">Welcome to Nehru Arts and Science College</h1>
        </div>
        <div class="card">
            <div class="card-body">
                <h5 class="card-title">History</h5>
                <p class="card-text">Early in 1964, the enlightened public of Kanhagad and Nileswar resolved to honour the memory of Jawaharlal Nehru, the architect of modern India by undertaking educational and cultural activities in this educationally backward area. In pursuit of this idea, a dedicated and learned team set up in 1966, an educational society known as "The Nehru Memorial Education Society" under the distinguished presidentship of the late Shri. C. K. Nair following which the society was duly registered. The establishment of Nehru Arts & Science College, Kanhagad in this backward coastal tract was the major task the society had set before themselves and had subsequently accomplished it.</p>
            </div>
        </div>
    </div>
    <div id="About" class="container mt-4" style="display: none;">
        <div class="card">
            <div class="card-body">
                <h5 class="card-title">Vision</h5>
```

EXERCISE 8. BOOTSTRAP CSS

```

<p class="card-text">Education for all and for all-round development</p>
<h5 class="card-title">Mission</h5>
<p class="card-text">
    <ul>
        <li>To keep the facilities for higher education open to all irrespective of caste, religion, sex and economic status and thereby strengthen the knowledge base of the society.
            <li>To help develop the spirit of patriotism and secularism among the students.</li>
    >
        <li>To preserve the ethnic values and cultural heritage of this locality with special accent on its linguistic forms and folklore. </li>
            <li>To conduct programmes and projects highlighting the challenges and trials faced by the society, to make the students alert to the threats of communalism, social fragmentation, exploitation and erosion of values.</li>
            <li>To stimulate interaction of the student community and the society through programmes of specific social benefit.</li>
    </ul>
</p>
</div>
</div>
<div id="Departments" class="container mt-4" style="display: none;">
<div class="card">
    <div class="card-body">
        <h5 class="card-title">Departments</h5>
        <p>
            <ul>
                <li>Botany</li>
                <li>Chemistry</li>
                <li>Commerce</li>
                <li>Computer Science</li>
                <li>Economics</li>
                <li>English</li>
                <li>Botany</li>
            </ul>
        </p>
    </div>
</div>
</div>
<div id="Contact" class="container mt-4" style="display: none;">
<div class="card">
    <div class="card-body">
        <h5 class="card-title">Contact Us</h5>
        <p> P.O. Padnekkat<br>
        Kasaragod (Dist.)<br>
        Kerala - 671314<br>
        India<br>
        +91-467 2280335<br>
        nascollegekanhangad@gmail.com</p>
    </div>
</div>
</div>
<script >
    // JavaScript function to show the selected content and hide others
    function showContent(sectionId) {
        // Hide all content sections
        document.querySelectorAll(".container").forEach(function(section) {
            section.style.display = "none";
        });

        // Show the selected content section
        document.getElementById(sectionId).style.display = "block";
    }
</script>
</body>
</html>

```

Output

The screenshot shows a website for 'NEHRU ARTS AND SCIENCE COLLEGE KANHANGAD'. The header includes the college's logo, name in English and Malayalam, NAAC re-accreditation information, and a portrait of Mahatma Gandhi. A navigation bar at the top has links for Home, About, Departments, and Contact. Below the header, a box contains the 'Vision' and 'Mission' statements. The 'Vision' is 'Education for all and for all-round development'. The 'Mission' includes points such as keeping facilities open to all, developing patriotism and secularism, preserving local heritage, conducting programmes to highlight challenges, and stimulating interaction between students and society.

नेहरु आर्ट्स एंड साइंस कॉलेज कांजंगाड
NEHRU ARTS AND SCIENCE COLLEGE KANHANGAD
ഐഎസ്റ്റ് ആർട്ട്സ് ആൻഡ് സയൻസ് കോളേജ്
NAAC RE-ACCREDITED WITH 'A' GRADE
Government Aided College - Affiliated to Kannur University

Home About Departments Contact

Vision
Education for all and for all-round development

Mission

- To keep the facilities for higher education open to all irrespective of caste, religion, sex and economic status and thereby strengthen the knowledge base of the society.
- To help develop the spirit of patriotism and secularism among the students.
- To preserve the ethnic values and cultural heritage of this locality with special accent on its linguistic forms and folklore.
- To conduct programmes and projects highlighting the challenges and trials faced by the society, to make the students alert to the threats of communalism, social fragmentation, exploitation and erosion of values.
- To stimulate interaction of the student community and the society through programmes of specific social benefit.

Exercise 9

Calculator using JavaScript

Problem Definition

Write a JavaScript code using functions to perform arithmetic operations on two numbers.

Source Code

```
<!DOCTYPE html>
<head>
<style>
body {
    background-color: #f1f1f1;
    font-family: Arial, sans-serif;
}
#heading {
    text-align: center;
    padding: auto;
    background-color: skyblue;
    color: white;
}
#calculator {
    background-color: #fff;
    width: 300px;
    margin: auto;
    padding: 20px;
    border-radius: 10px;
    box-shadow: 2px 2px 10px #888;
}
</style>
<script type="text/javascript">
function addition() {
    a = Number(document.my_cal.first.value);
    b = Number(document.my_cal.second.value);
    c = a + b;
    document.my_cal.total.value = c;
}
function multiply() {
    a = Number(document.my_cal.first.value);
    b = Number(document.my_cal.second.value);
    c = a * b;
    document.my_cal.total.value = c;
}

function subtraction() {
    a = Number(document.my_cal.first.value);
    b = Number(document.my_cal.second.value);
    c = a - b;
    document.my_cal.total.value = c;
}

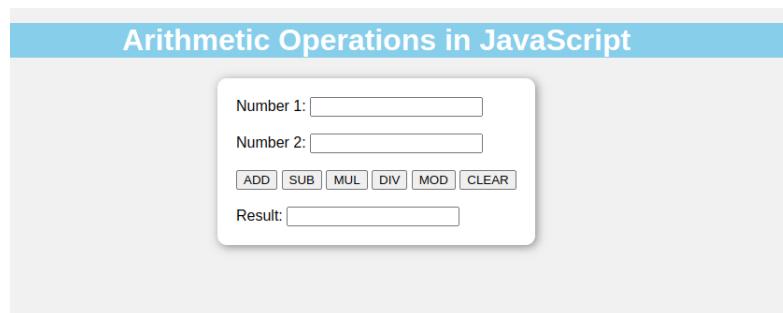
function division() {
    a = Number(document.my_cal.first.value);
    b = Number(document.my_cal.second.value);
    c = a / b;
    document.my_cal.total.value = c;
}
</script>
```

EXERCISE 9. CALCULATOR USING JAVASCRIPT

```
document.my_cal.total.value = c;
}

function modulus() {
    a = Number(document.my_cal.first.value);
    b = Number(document.my_cal.second.value);
    c = a % b;
    document.my_cal.total.value = c;
}
function clearAll() {
    my_cal.first.value=' ';
    my_cal.second.value=' ';
    my_cal.total.value=' ';
}
</script>
</head>
<body>
<div id="heading">
    <h1> Arithmetic Operations in JavaScript </h1>
</div>
<div id="calculator">
    <form name="my_cal">
        Number 1: <input type="text" name="first">
        <br>
        <br>
        Number 2: <input type="text" name="second">
        <br><br>
        <input type="button" value="ADD" onclick="addition()">
        <input type="button" value="SUB" onclick="subtraction();">
        <input type="button" value="MUL" onclick="multiply();">
        <input type="button" value="DIV" onclick="division();">
        <input type="button" value="MOD" onclick="modulus();">
        <input type="button" value="CLEAR" onclick="clearAll();">
        <br><br>
        Result: <input type="text" name="total">
    </form>
</div>
</body>
</html>
```

Output



Exercise 10

Reversing an Array

Problem Definition

Write a JavaScript code to sort and reverse array elements.

Source Code

```
<!DOCTYPE html>
<html>
  <head><title>Array Operations</title></head>
  <body>
    <form>
      <fieldset>
        <legend> Add elements to array </legend>
        <input type="text" id="text1"></input>
        <input type="button" id="button1" value="Add" onclick="add_element_to_array();"></input>
      </fieldset>
      <input type="button" id="button2" value="Display" onclick="display_array();"></input>
      <button type="button" onclick="sort()">Sort</button>
      <button type="button" onclick="rev()">Reverse</button>
      <button type="button" onclick="desc()">Descending Order</button>
      <div id="Result"></div>
      <div id="names"></div>
      <script>
        var x = 0;
        var array = Array();
        function add_element_to_array() {
          array[x] = document.getElementById("text1").value;
          alert("Element: " + array[x] + " Added at index " + x);
          x++;
          document.getElementById("text1").value = "";
        }
        function display_array() {
          document.getElementById("Result").innerHTML = "actual array=[ "+array+" ]";
        }
        function rev() {
          display_array()
          document.getElementById("names").innerHTML ="Reversed Array: "+array.reverse();
        }
        function sort() {
          display_array();
          array.sort();
          document.getElementById("names").innerHTML ="Ascending order Sorted Array:"+array;
        }
        function desc() {
          display_array();
          array.sort().reverse();
          document.getElementById("names").innerHTML ="Descending order Sorted Array:"+array;
        }
      </script>
    </form>
  </body>
</html>
```

EXERCISE 10. REVERSING AN ARRAY

Output

Add elements to array -

actual array=[9,5,1,-2]
Ascending order Sorted Array:-2,1,5,9

Exercise 11

TODO List

Problem Definition

Write a javascript code to create a TODO list.

Source Code

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Simple To Do List</title>
    <style>
      body{
        background: #e5e8d8;
      }
      #newtask{
        position: relative;
        padding: 30px 20%;
      }
      #newtask input{
        width: 75%;
        height: 45px;
        font-family: 'Poppins', sans-serif;
        font-size: 15px;
        border: 2px solid #d1d3d4;
        color: #111111;
        align-self: center;
        position: relative;
        border-radius: 5px;
      }
      #newtask button{
        position: relative;
        width: 10%;
        height: 45px;
        border-radius: 5px;
        font-size: 16px;
        background-color: #0d75ec;
        color: #ffffff;
        cursor: pointer;
      }
      #tasks{
        background-color: #e5e8d8;
        padding: 30px 20px;
        border-radius: 10px;
        position: relative;
      }
      .task{
        background-color: #c5e1e6;
        height: 50px;
        padding: 5px 10px;
        display: flex;
        border-radius: 5px;
        justify-content: space-between;
        border: 1px solid #939697;
      }
    </style>
  </head>
  <body>
    <div id="newtask">
      <input type="text" placeholder="New Task" />
      <button>Add</button>
    </div>
    <div id="tasks">
      <div class="task">
        <span>Task 1</span>
        <span>Edit</span>
        <span>Delete</span>
      </div>
      <div class="task">
        <span>Task 2</span>
        <span>Edit</span>
        <span>Delete</span>
      </div>
    </div>
  </body>
</html>
```

EXERCISE 11. TODO LIST

```
cursor: pointer;
font-family: 'Poppins', sans-serif;
font-size: 15px;
}
.delete {
background-color: #0a2ea4;
color: #ffffff;
width: 50px;
border-radius: 5px;
cursor: pointer;
}
</style>
</head>
<body>
<center><b><h3> TODO LIST</h3></b></center>
<div id="newtask">
<input type="text" placeholder="Task to be done.." >
<button id="push">Add</button>
</div>
<div id="tasks"></div>

<script type="text/javascript">
document.querySelector('#push').onclick = function(){
if(document.querySelector('#newtask input').value.length == 0){
alert("Please Enter a Task")
}
else{
document.querySelector('#tasks').innerHTML += `
<div class="task">
<span id="taskname">
${document.querySelector('#newtask input').value}
</span>
<button class="delete">
Delete
</button>
</div>
`;
var current_tasks = document.querySelectorAll(".delete");
for(var i=0; i<current_tasks.length; i++){
current_tasks[i].onclick = function(){
this.parentNode.remove();
}
}
}
}
</script>
</body>
</html>
```

Output



Exercise 12

Query Selector

Problem Definition

Write a javascript program using queryselector.

Source Code

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>QuerySelector Examples</title>
    <style>
      .circle{
        width: 40vmin;
        height: 40vmin;
        border: 5px solid rebeccapurple;
        border-radius: 50%;
      }
      body{
        height: 100vh;
        display: flex;
        flex-direction: column;
        justify-content: center;
        align-items: center;
      }
      .fancy{
        background: rebeccapurple;
      }
      button{
        padding: .5rem 1rem;
        margin: 1rem;
        border-radius: 1rem;
        background: #ddd;
        font-size: 1.5rem;
        font-family: MS Sans Serif, Geneva, sans-serif;
        font-variant: small-caps;
        cursor: pointer;
      }

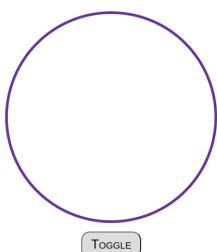
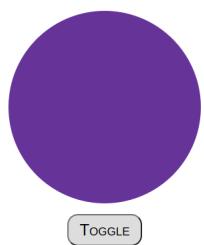
      button:active{
        background: #bbb;
      }

      body{
        margin: 0;
      }
    </style>
  </head>
  <body>
    <div class="circle"></div>
    <button>Toggle</button>
    <script>
      const circle = document.querySelector('.circle');
```

EXERCISE 12. QUERY SELECTOR

```
const button = document.querySelector('button');
button.addEventListener('click', function(){
  circle.classList.toggle('fancy');
});
</script>
</body>
</html>
```

Output



Exercise 13

Animation

Problem Definition

Write a javascript program to implement animations.

Source Code

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Animation - moving a car</title>
    <style>
      #container {
        width: 2000px;
        height: 550px;
        border: 1px solid #000;
        position: relative;
        background: url('grass1.jpg') no-repeat;
        background-position: 0 bottom;
        background-size: contain;
      }
      #car {
        position: absolute;
        height : 1000 px;
        width: 500px;
        bottom: 100px;
        left: 2px;
        opacity:20;
      }
    </style>
  </head>
  <body>
    <div id="container">
      
    </div>

    <script>
      (function (selector) {
        let start_pos = 2; // start from 2 pixel left
        let end_pos = document.querySelector("#container").getBoundingClientRect().width - 100;
        let interval = setInterval(function () {
          if(selector.offsetLeft >= end_pos) {
            start_pos = 2;
          } else {
            start_pos++;
          }
          selector.style = `left: ${start_pos}px`;
        }, 10);
      })(document.querySelector("#car"));
    </script>
  </body>
</html>
```

Output



Exercise 14

Event Handler

Problem Definition

Write a javascript program to implement an event handler.

Source Code

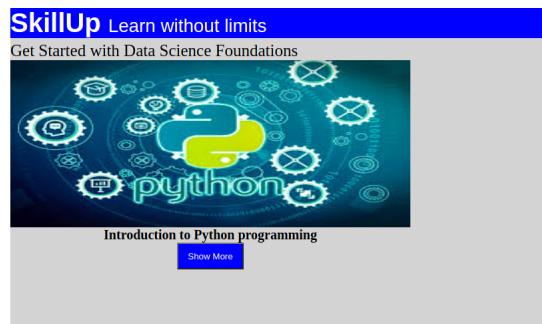
```
<html>
  <head>
    <title>course</title>
    <script>
      function less()
      {
        el=document.getElementById("task");
        while (el.firstChild)
          el.removeChild(el.firstChild);
      }
      function front(){
        document.querySelector('#back').innerHTML += `
        <div id="task">
          <span id="taskname">
            <b>Introduction to Python programming</b><br><hr>
            <dl>
              <dt><b>Course</b><br></dt>
              <dd><small>Gain insight into a topic and learn the fundamentals</small><br></dd>
              <hr>
              <dt><b>Intermediate level</b></dt>
              <dd><small>No previous experience necessary</small><br> </dd> <hr>
              <b>29 hours (approximately)</b><br><hr>
              <dt><b>Flexible schedule</b><br></dt>
              <dd>
                <small>Learn at your own pace</small><br></dd><br>
              </span>
              <button onclick="less () ">
                Show less
              </button>
            </div>
          `;
      }
    </script>
    <style>
      .scene{
        display: inline-block;
        width: 600px;
        height: 400px;
        perspective: 600px;
      }
      .card{
        position: relative;
        width: 100%;
        height: 100%;
        cursor: pointer;
      }
      .card__face{
```

EXERCISE 14. EVENT HANDLER

```
position: absolute;
width: 100%;
height: 100%;
color: black;
text-align: center;
font-weight: bold;
font-size: 20px;
}
.heading{
background-color: blue;
color: white;
font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
font-size: 40px;
}
.sub-heading{
position: relative;
font-size: 25px;
padding-top: 5px;
background-color: lightgrey
}
#front{
background-color: blue;
color: white;
height: 40px;
width: 100px;
}
</style>
</head>
<body>
<div class="heading">
<b>SkillUp</b>&nbsp;<small><small>Learn without limits</small></small>
</div>
<div class="sub-heading">
Get Started with Data Science Foundations<br>
<div class="scene">
<div class="card">
<div class="card_face">

<br>Introduction to Python programming<br>
<button id="front" onclick="front()">Show More</button>
</div>
</div>
<div id="back"></div>
</div>
</div>
</body>
</html>
```

Output



Introduction to Python programming

Course

Gain insight into a topic and learn the fundamentals

Intermediate level

No previous experience necessary

29 hours (approximately)

Flexible schedule

Learn at your own pace

[Show less]

Exercise 15

Library Database using Django

Problem Definition

Create a table with following schema books(title, author, shelfno). Create a django project which contains a form to accept the book's title, author and shelf no. It should have features to insert title, author and shelf no in the form to the books table. Insert 3 books' details using this. Create a url to display all the book details.

Procedure

1. Setup Django Project:

- Create a new Django project using following commands in VScode terminal

```
django-admin startproject library_project  
cd library_project
```

2. Create a Django App

- Create a Django app named 'book' using following command

```
python manage.py startapp books
```

- Add the books app to the `INSTALLED_APPS` list in the `library_project/settings.py` file.

```
INSTALLED_APPS = [# ...  
    'books',  
    # ...]
```

3. Define the Book Model

- In the `books/models.py` file, define the Book model with the specified schema.

```
from django.db import models  
  
class Book(models.Model):  
    title = models.CharField(max_length=100)  
    author = models.CharField(max_length=100)  
    shelf_no = models.IntegerField()  
  
    def __str__(self):  
        return self.title
```

4. Create a Form

- Create a form for adding book details in the `books/forms.py` file:

```
from django import forms  
from .models import Book  
  
class BookForm(forms.ModelForm):  
    class Meta:  
        model = Book  
        fields = ['title', 'author', 'shelf_no']
```

EXERCISE 15. LIBRARY DATABASE USING DJANGO

5. Create Views and Templates

- Create views to handle the form submission and to display all book details. Also, create templates for rendering the HTML in books/views.py.

```
from django.shortcuts import render, redirect
from .forms import BookForm
from .models import Book

def add_book(request):
    if request.method == 'POST':
        form = BookForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('book_list')
    else:
        form = BookForm()
    return render(request, 'books/add_book.html', {'form': form})

def book_list(request):
    books = Book.objects.all()
    return render(request, 'books/book_list.html', {'books': books})
```

6. Create HTML Templates

- Create three HTML templates for adding books (add_book.html), displaying book details (list_books.html) and displaying main page (index.html) inside a books/templates/ directory.

add_book.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Add Book</title>
  </head>
  <body>
    <div name="add" style="background-color:#007BFF">
      <h1>Add a Book</h1>
    </div>
    <form method="post">
      {% csrf_token %}
      {{ form.as_p }}
      <input type="submit" value="Submit">
    </form>
  </body>
</html>
```

list_books.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Book List</title>
  </head>
  <body>
    <div name="list" style="background-color:#007BFF">
      <h1>Book List</h1>
    </div>
    <table border="1">
      <tr><th>Title</th><th>Author:</th><th> Shelf No</th></tr>
      {% for book in books %}
        <tr><td> {{ book.title }}</td>
        <td> {{ book.author }}</td>
        <td> {{ book.shelfno }}</td>
      </tr>
      {% endfor %}
    </table>
  </body>
</html>
```

EXERCISE 15. LIBRARY DATABASE USING DJANGO

```
</tr>
{%
empty %
No books available
% endfor %
</table>
</body>
</html>
```

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Library Management System</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f0f0f0;
            margin: 0;
            padding: 0;
        }
        header {
            background-color: #007BFF;
            color: #fff;
            padding: 20px;
            text-align: center;
        }
        .container {
            max-width: 800px;
            margin: 20px auto;
            padding: 20px;
            background-color: #ffffff;
            box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.2);
        }
        h1 {
            color: #ffffff;
        }
        p {
            margin-bottom: 20px;
        }
        .buttons {
            text-align: center;
            margin-top: 20px;
        }
        .button {
            background-color: #007BFF;
            color: white;
            border: none;
            padding: 10px 20px;
            margin-right: 10px;
            cursor: pointer;
            text-decoration: none;
        }
        .button:hover {
            background-color: #0056b3;
        }
    </style>
</head>
<body>
    <header>
        <h1>Library Management System</h1>
    </header>
    <div class="container">
        <h2>Welcome to the Library</h2>
        <p>Manage books, borrowers, and more with our Library Management System.</p>
```

EXERCISE 15. LIBRARY DATABASE USING DJANGO

```
<div class="buttons">
    <a href={% url 'add_book' %} class="button">Add Books</a>
    <a href={% url 'list_books' %} class="button">View Books</a>
</div>
</body>
</html>
```

7. Configure URLs

- Configure the URLs for the books app by creating a `urls.py` file in the books directory:

```
from django.urls import path
from . import views

urlpatterns = [
    path('add_book/', views.add_book, name='add_book'),
    path('list_books/', views.list_books, name='list_books'),
    path('', views.index, name='index'),
]
```

- Include these URLs in the main `urls.py`:

```
from django.contrib import admin
from django.urls import path, include
urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('books.urls')),
]
```

8. Run migration and the Development Server

- Run the following command to create the database table for the Book model.

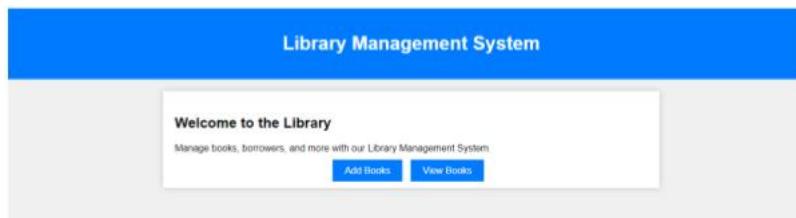
```
python manage.py makemigrations
python manage.py migrate
```

- Start the development server:

```
python manage.py runserver
```

Output

- Main page (`http://127.0.0.1:8000/`)



- Add Book page `http://127.0.0.1:8000/add_book/`

EXERCISE 15. LIBRARY DATABASE USING DJANGO

Add a Book

Title:

Author:

Shelfno:

- View Book (http://127.0.0.1:8000/list_books/)

Book List

Title	Author:	Shelf No
Hands-on MachineLearning with Keras and TensorFlow	Aurelien Geron	5
Deep Learning	Ian Goodfellow, Yoshua Bengio and Aaron Courville	11
Natural Language Processing with Transformers	Lewis Tunstall, Thomas Wolf	4

Exercise 16

Sessions

Problem Definition

Create a django project which uses sessions.

Procedure

1. Setup Django Project:

- Create a new Django project using following commands in VScode terminal

```
django-admin startproject session_demo  
cd session_demo
```

2. Create a Django App

- Create a Django app named ‘login’ using following command

```
python manage.py startapp login
```

- In session_demo/settings.py file add the login app to the INSTALLED_APPS and configure session settings

```
INSTALLED_APPS = [# ...  
'login',  
# ...]  
MIDDLEWARE = [  
# ...  
'django.contrib.sessions.middleware.SessionMiddleware',  
]  
SESSION_ENGINE = 'django.contrib.sessions.backends.db'  
SESSION_COOKIE_NAME = 'login_session'
```

3. Create a User Model

- Create a superuser to access the Django admin site for user management

```
python manage.py createsuperuser
```

4. Create Views

- Create views in app’s views.py for login, logout, and a simple dashboard.

```
from django.contrib.auth import login, logout, authenticate  
from django.shortcuts import render, redirect  
  
def user_login(request):  
    if request.method == 'POST':  
        username = request.POST.get('username')  
        password = request.POST.get('password')  
        user = authenticate(request, username=username, password=password)  
        if user is not None:  
            login(request, user)  
            return redirect('dashboard')
```

EXERCISE 16. SESSIONS

```
else:
    return render(request, 'login.html', {'error_message': 'Invalid username or password'})
return render(request, 'login.html')

def user_logout(request):
    logout(request)
    return redirect('login')

def dashboard(request):
    if request.user.is_authenticated:
        return render(request, 'dashboard.html')
    else:
        return redirect('login')
```

5. Create URLs

- Define URL patterns for views in app's urls.py:

```
from django.urls import path
from . import views

urlpatterns = [
    path('', views.user_login, name='login'),
    path('logout/', views.user_logout, name='logout'),
    path('dashboard/', views.dashboard, name='dashboard'),
]
```

- Include these URLs in the main urls.py:

```
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('login.urls')),
]
```

6. Create Templates

- Create HTML templates for login and dashboard pages in app's templates directory

login.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Login</title>
        <style>
            .login {
                max-width: 20%;
                margin: 20px auto;
                padding: 20px;
                align: center;
                background-color: #012fff;
            }
        </style>
    </head>
    <body>
        <div style="background-color: grey; text-align: center;">
            <h1>Login</h1>
        </div>
        {% if error_message %}
            <p>{{ error_message }}</p>
        
```

EXERCISE 16. SESSIONS

```
{% endif %}
<form method="post" action="{% url 'login' %}">
    {% csrf_token %}
    <div class="login">
        <label for="username">Username:</label>
        <input type="text" id="username" name="username"><br>
        <label for="password">Password:</label>
        <input type="password" id="password" name="password"><br>
        <input type="submit" value="Login">
    </div>
</form>
</body>
</html>
```

dashboard.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Dashboard</title>
    </head>
    <body>
        <h1>Welcome to the Dashboard</h1>
        <p><a href="{% url 'logout' %}">Logout</a></p>
    </body>
</html>
```

- Set the Login URL: In the project's settings.py, set the `LOGIN_URL` to the login view URL. This ensures that unauthenticated users are redirected to the login page:

```
LOGIN_URL = 'login:user_login'
```

- Migrate the Database

- Run migrations to create the necessary database tables

```
python manage.py makemigrations
python manage.py migrate
```

- Run the Development Server

```
python manage.py runserver
```

Output

- login.html



EXERCISE 16. SESSIONS

-
- dashboard.html



Welcome to the Dashboard

[Logout](#)