**TIMELINE**

**ABSTRACT:**

"Timeline" In this application, the user make their timeline what he/she learns in that time. It will replicate their work what he/she does in that period. In this timeline the user displayed their projects details which includes random number, custom range counter, bmi calculator and etc..

**FOLDER STRUCTURE OF WORD COUNTER APP:**

1. index.html 🡪 HTML file
2. style.css 🡪 CSS file

**STEP TO CREATE A FOLDER:**

STEP 1: Create a folder named “Random number generator App ” or whatever you want.

STEP 2: Create the HTML file “index.html” which represent as landing page of the our app.

STEP 3: Make a "Style.css" file to style the html files.

**INDEX.HTML:**

It is our app's base file. In this file, we write the html tags.

The basic HTML structure for creating a project is shown below by clicking Shift+1, it shows exclamatory(!) if you want to choose that, it will provide the base code.

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

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

  <title>Document</title>

</head>

<body>

</body>

</html>

The code starts with an h1 tag that contains the title "MY JOURNEY OF DEVELOPER". The timeline is contained within a div element with the class name "scrollbar".

The timeline itself is composed of several "container" elements, each with a "content" element inside. The "container" elements are positioned on the left or right of the timeline, depending on whether they occurred earlier or later in time. Each "container" element contains an image of a clock, which is followed by a "p" tag that displays the time spent on the project.

  <h1>MY JOURNEY OF DEVELOPER</h1>

    <div class="scrollbar">

      <div class="timeline">

        <div class="container left">

          <div class="content">

            <img src="clock.png" />

            <p id="time">10 Days</p>

            <p id="name">STATIC WEBSITE</p>

            <p id="para">

              Build your own personal portfolio website and a website to host

              conferences and events. Publish your website and share it with

              your friends, family and beyond.

            </p>

            <p id="html">HTML</p>

            <p id="css">CSS</p>

            <p id="bootstrap">BOOTSTRAP</p>

            <p id="javascript">JAVASCRIPT</p>

            <p id="react">REACT</p>

          </div>

        </div>

        <p id="year1">7 JANUARY 2023</p>

        <div class="container right">

          <div class="content">

            <img src="clock.png" />

            <p id="time">3 hours</p>

            <img id="img1" src="rm.png" />

            <p id="name">RANDOM NUMBER</p>

            <p id="para">

              Random Number Generator the user can generate a random number from

              the range of 0 to 100.

            </p>

            <p id="html">HTML</p>

            <p id="css">CSS</p>

            <p id="javascript">JAVASCRIPT</p>

            <br />

After the time spent on the project, there is an "img" tag that displays an image related to the project, followed by a "p" tag that displays the name of the project. Below the name of the project, there is a "p" tag that contains a brief description of the project, followed by several "p" tags that display the technologies used in the project.

      <a href="https://cheery-gumption-4750e6.netlify.app/">VISIT</a>

          </div>

        </div>

        <p id="year2">20 JANUARY 2023</p>

        <div class="container left">

          <div class="content">

            <img src="clock.png" />

            <p id="time">3 hours</p>

            <img id="img1" src="crc.png" />

            <p id="name">CUSTOM RANGE COUNTER</p>

            <p id="para">

              Custom Range Counter In this application the user can enter two

              values one is from value and another one is to value.

            </p>

            <p id="html">HTML</p>

            <p id="css">CSS</p>

            <p id="javascript">JAVASCRIPT</p>

            <br />

            <a href="https://verdant-meringue-eb6297.netlify.app">VISIT</a>

          </div>

        </div>

        <p id="year3">1 FEBRUARY 2023</p>

        <div class="container right">

          <div class="content">

            <img src="clock.png" />

            <p id="time">5 hours</p>

            <img id="img1" src="bmi.png" />

            <p id="name">BMI CALCULATOR</p>

            <p id="para">

              BMI calculator is used to calculate the body mass index of our

              body using our height and weight and it will show our risk factor.

            </p>

            <p id="html">HTML</p>

            <p id="css">CSS</p>

            <p id="javascript">JAVASCRIPT</p>

            <br />

            <a href="https://prismatic-sorbet-225a74.netlify.app">VISIT</a>

          </div>

        </div>

        <p id="year4">25 FEBRUARY 2023</p>

        <div class="container left">

At the end of each "container" element, there is an anchor tag with a link to the deployed project.

There are also "p" tags with ids "year1", "year2", "year3", and "year4" that display the date each project was completed.

          <div class="content">

            <img src="clock.png" />

            <p id="time">7 hours</p>

            <img id="img1" src="am.png" />

            <p id="name">APPOINTMENT MANAGER</p>

            <p id="para">

Appointment manger In this application, the user can book their

              appointment and the developer can easily manage the appointments

            </p>

            <p id="html">HTML</p>

            <p id="css">CSS</p>

            <p id="javascript">JAVASCRIPT</p>

            <br />

            <a href="https://eloquent-dodol-378d26.netlify.app">VISIT</a>

          </div>

        </div>

      </div>

    </div>

The external CSS file is attached to our index.html file.

 <link rel="stylesheet" href="style.css">

**STYLE.CSS:**

In style.css, we did styling for our html page by id, selectors and attributes. We added background image for our page, gave border for our div tags using their id and assigning in a right manner.

This is a CSS code that creates a vertical timeline. The timeline is contained in a div with class "timeline". The timeline consists of several div containers with class "container" that contain the content and the circles that denote each event. The circles are created using the ::after pseudo-element. The left and right containers are differentiated using the classes "left" and "right".

\* {

*box-sizing*: border-box;

}

body {

*background-color*: #8c8cc8;

*font-family*: Helvetica, sans-serif;

}

/\* The actual timeline (the vertical ruler) \*/

.timeline {

*position*: relative;

*max-width*: 1200px;

*margin*: 0 auto;

}

/\* The actual timeline (the vertical ruler) \*/

.timeline::after {

*content*: "";

*position*: absolute;

*width*: 6px;

*background-color*: rgb(234, 114, 17);

*top*: 0;

*bottom*: 0;

*left*: 50%;

*margin-left*: -3px;

}

/\* Container around content \*/

.container {

*padding*: 10px 40px;

*position*: relative;

*background-color*: inherit;

*width*: 50%;

}

/\* The circles on the timeline \*/

.container::after {

*content*: "";

*position*: absolute;

*width*: 25px;

*height*: 25px;

*right*: -17px;

*background-color*: white;

*border*: 4px solid #ff9f55;

*top*: 15px;

*border-radius*: 50%;

*z-index*: 1;

}

/\* Place the container to the left \*/

.left {

*left*: 0;

}

/\* Place the container to the right \*/

.right {

*left*: 50%;

}

/\* Add arrows to the left container (pointing right) \*/

.left::before {

*content*: " ";

*height*: 0;

*position*: absolute;

*top*: 22px;

*width*: 0;

*z-index*: 1;

*right*: 30px;

*border*: medium solid white;

*border-width*: 10px 0 10px 10px;

*border-color*: transparent transparent transparent white;

}

/\* Add arrows to the right container (pointing left) \*/

.right::before {

*content*: " ";

*height*: 0;

*position*: absolute;

*top*: 22px;

*width*: 0;

*z-index*: 1;

*left*: 30px;

*border*: medium solid white;

*border-width*: 10px 10px 10px 0;

*border-color*: transparent white transparent transparent;

}

/\* Fix the circle for containers on the right side \*/

.right::after {

*left*: -16px;

}

/\* The actual content \*/

.content {

*padding*: 20px 30px;

*background-color*: white;

*position*: relative;

*border-radius*: 6px;

}

/\* Media queries - Responsive timeline on screens less than 600px wide \*/

@media screen and (*max-width*: 600px) {

  /\* Place the timelime to the left \*/

  .timeline::after {

*left*: 31px;

  }

  /\* Full-width containers \*/

  .container {

*width*: 100%;

*padding-left*: 70px;

*padding-right*: 25px;

  }

  /\* Make sure that all arrows are pointing leftwards \*/

  .container::before {

*left*: 60px;

*border*: medium solid white;

*border-width*: 10px 10px 10px 0;

*border-color*: transparent white transparent transparent;

  }

  /\* Make sure all circles are at the same spot \*/

  .left::after,

  .right::after {

*left*: 15px;

  }

  /\* Make all right containers behave like the left ones \*/

  .right {

*left*: 0%;

  }

}

#html {

*color*: rgb(249, 249, 250);

*background-color*: rgb(13, 13, 13);

  /\* font-family: Roboto; \*/

*font-size*: 10px;

*font-weight*: 500;

*border-radius*: 4px;

*line-height*: 1.6;

*padding*: 4px 8px;

*width*: 10%;

}

#css {

*color*: rgb(249, 249, 250);

*background-color*: rgb(13, 13, 13);

  /\* font-family: Roboto; \*/

*font-size*: 10px;

*font-weight*: 500;

*border-radius*: 4px;

*line-height*: 1.6;

*padding*: 4px 8px;

*width*: 10%;

*margin-top*: -8%;

*margin-left*: 13%;

}

#bootstrap {

*color*: rgb(249, 249, 250);

*background-color*: rgb(13, 13, 13);

  /\* font-family: Roboto; \*/

*font-size*: 10px;

*font-weight*: 500;

*border-radius*: 4px;

*line-height*: 1.6;

*padding*: 4px 8px;

*width*: 17%;

*margin-top*: -8%;

*margin-left*: 46%;

}

#javascript {

*color*: rgb(249, 249, 250);

*background-color*: rgb(13, 13, 13);

  /\* font-family: Roboto; \*/

*font-size*: 10px;

*font-weight*: 500;

*border-radius*: 4px;

*line-height*: 1.6;

*padding*: 4px 8px;

*width*: 17%;

*margin-top*: -8%;

*margin-left*: 26%;

}

#react {

*color*: rgb(249, 249, 250);

*background-color*: rgb(13, 13, 13);

  /\* font-family: Roboto; \*/

*font-size*: 10px;

*font-weight*: 500;

*border-radius*: 4px;

*line-height*: 1.6;

*padding*: 4px 8px;

*width*: 11%;

*margin-top*: -8%;

*margin-left*: 65%;

}

a {

*text-decoration*: none;

*color*: blue;

*font-size*: larger;

*font-weight*: bolder;

}

h1 {

*text-align*: center;

*margin-left*: 3%;

*margin-top*: 2%;

}

#img1 {

*height*: 200px;

*width*: 400px;

}

#img2 {

*height*: 200px;

*width*: 400px;

}

#img3 {

*height*: 200px;

*width*: 400px;

}

#img4 {

*height*: 200px;

*width*: 400px;

}

.scrollbar {

*margin-top*: 5%;

*height*: 700px;

*overflow*: auto;

*background-color*: #f7f3f3;

*width*: 60%;

*margin-left*: 21%;

}

#time {

*margin-top*: -4.9%;

*margin-left*: 5%;

}

#name {

*color*: rgb(23, 31, 70);

*font-family*: "Franklin Gothic Medium", "Arial Narrow", Arial, sans-serif;

*font-size*: 25px;

*font-weight*: bolder;

}

#time {

*font-weight*: bolder;

}

#para {

*color*: #747480;

}

#year1 {

*margin-left*: 55%;

*font-family*: fantasy;

*font-weight*: 500;

*font-size*: larger;

*color*: #0c0c0c;

}

#year2 {

*margin-left*: 35%;

*font-family*: fantasy;

*font-weight*: 500;

*font-size*: larger;

*color*: #0c0c0c;

}

#year3 {

*margin-left*: 55%;

*font-family*: fantasy;

*font-weight*: 500;

*font-size*: larger;

*color*: #0c0c0c;

}

#year4 {

*margin-left*: 33%;

*font-family*: fantasy;

*font-weight*: 500;

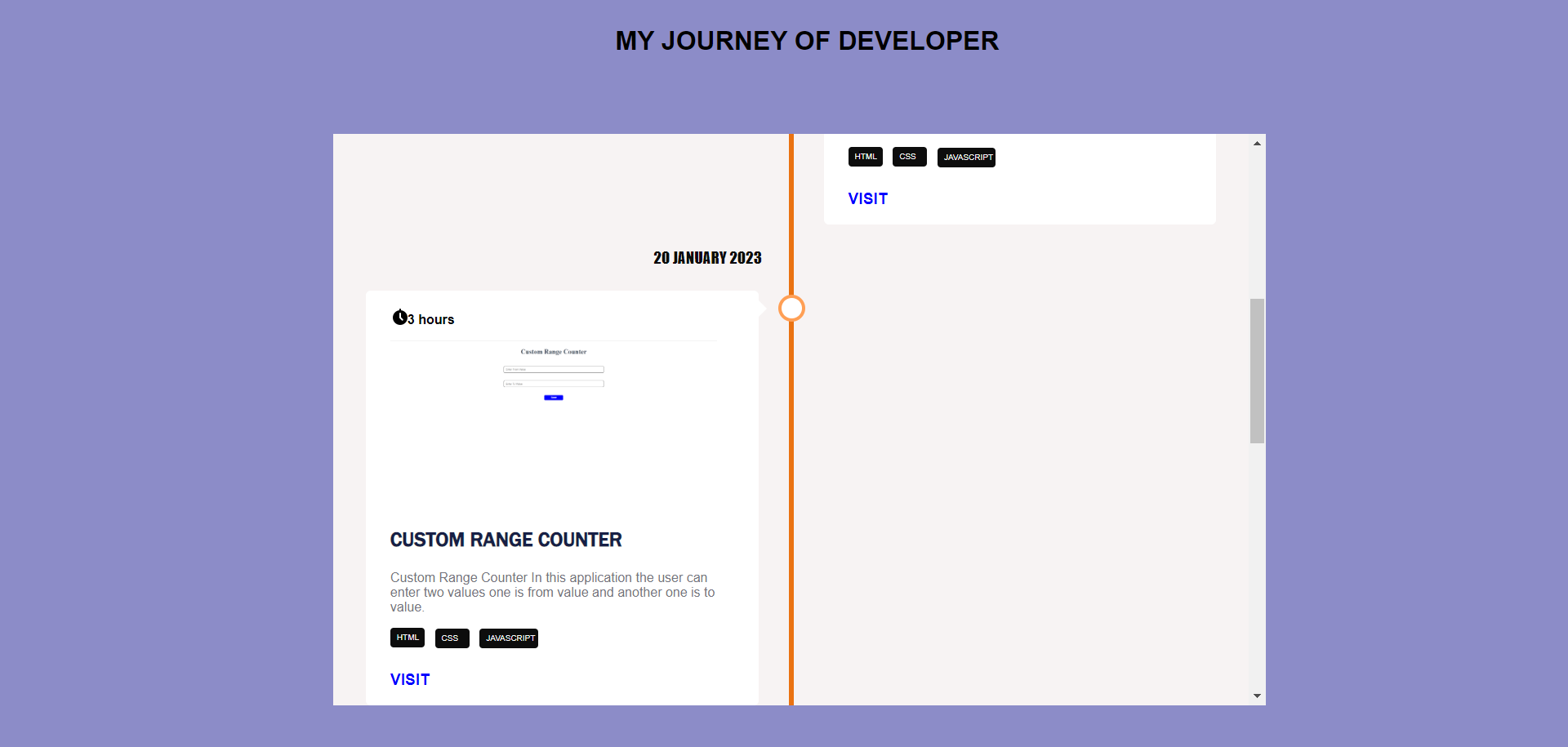
*font-size*: larger;

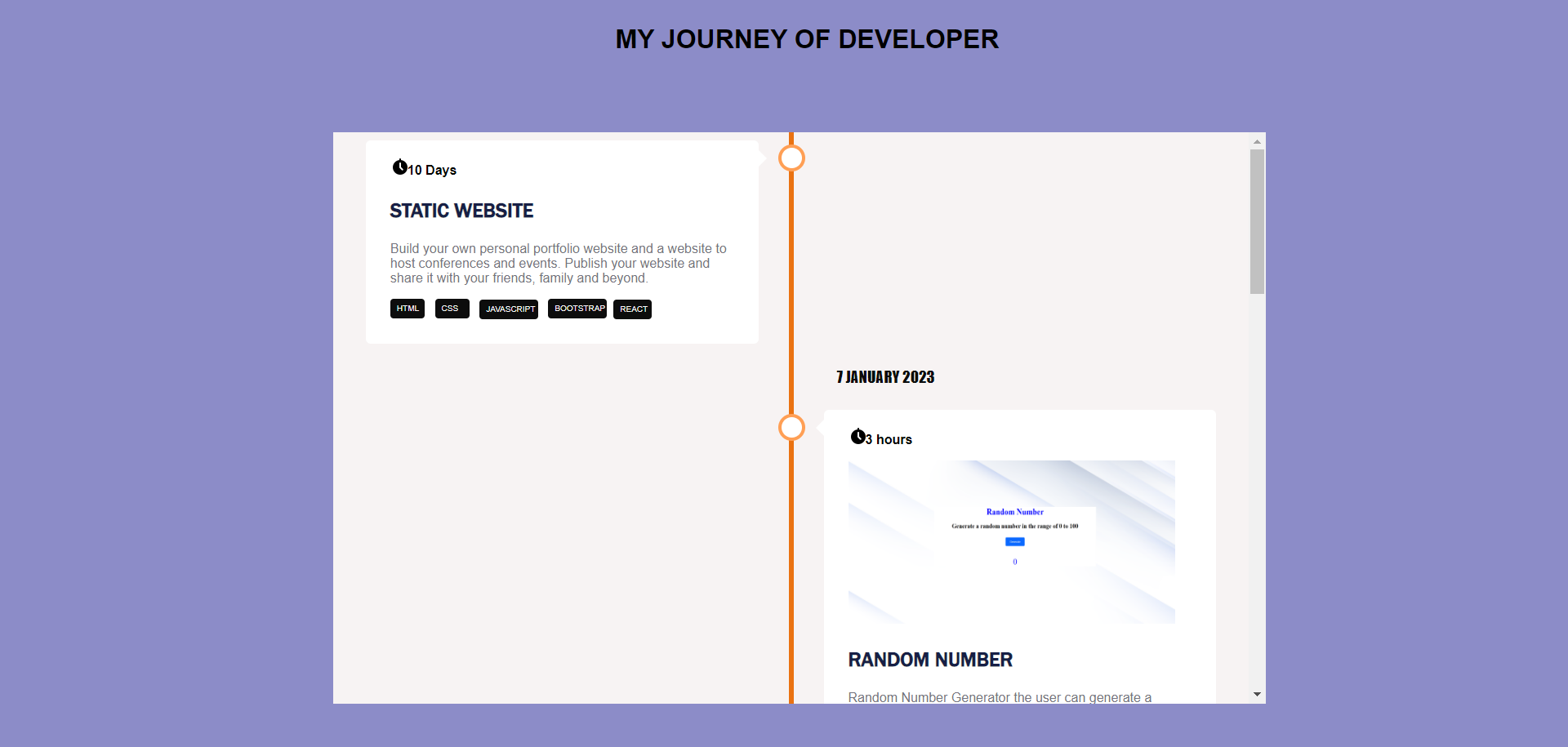
*color*: #0c0c0c;

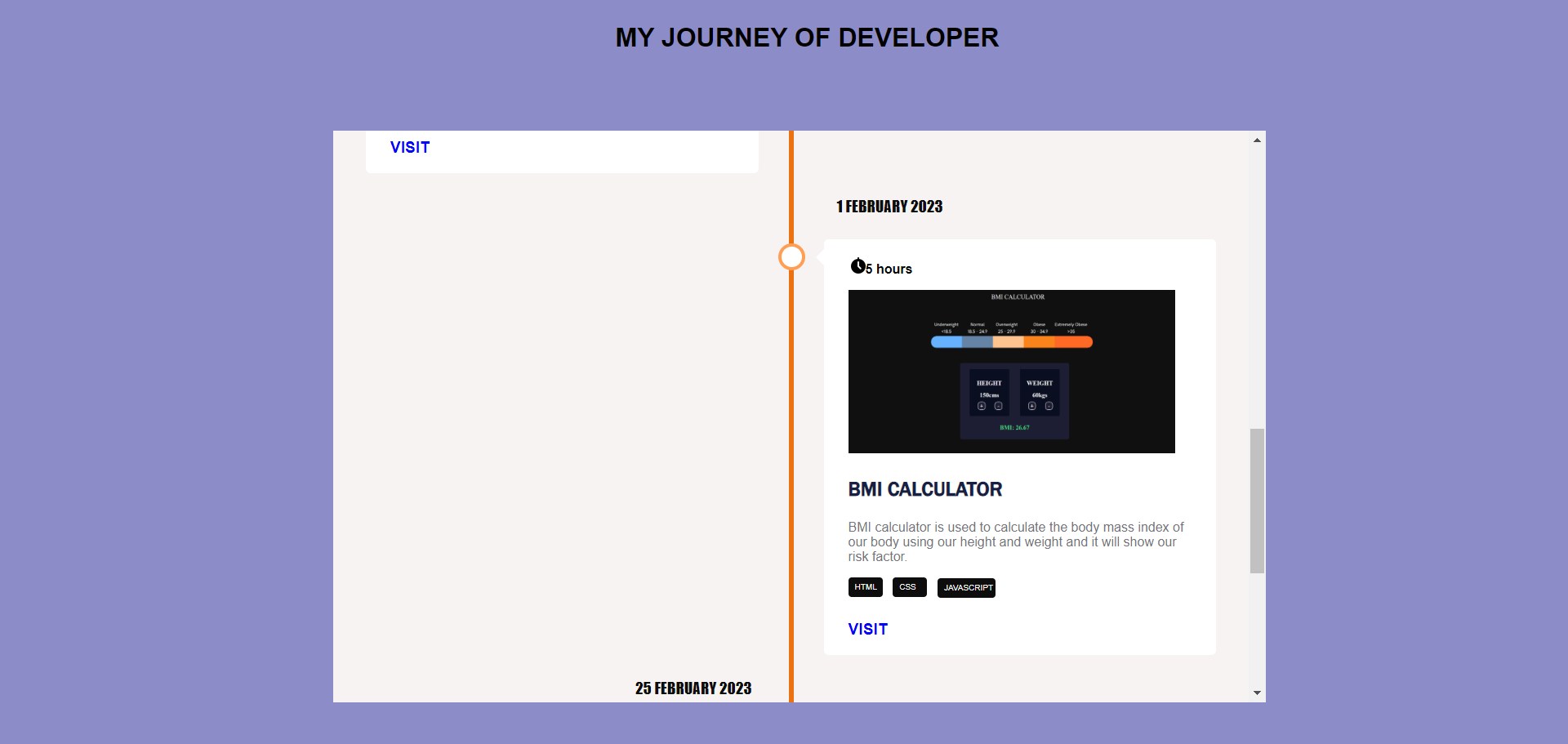
}

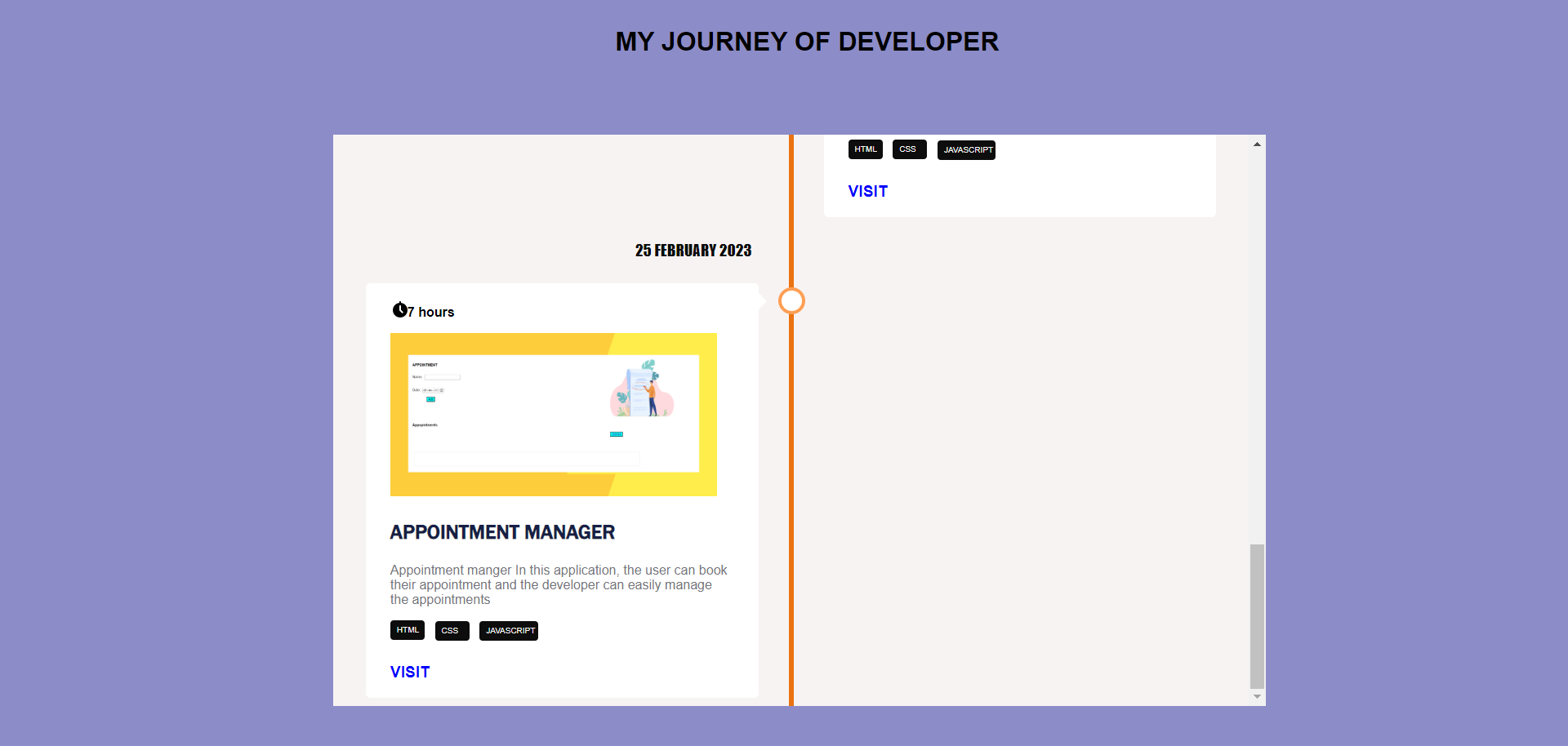
Arrows are added to each container using the ::before pseudo-element. The CSS also includes media queries for responsive design on screens less than 600px wide. Additionally, there are styles for individual elements with ids "html", "css", "bootstrap", "javascript", and "react". These styles control the appearance of individual elements within the timeline.

**FINAL OUTPUT:**

****

****

****

****