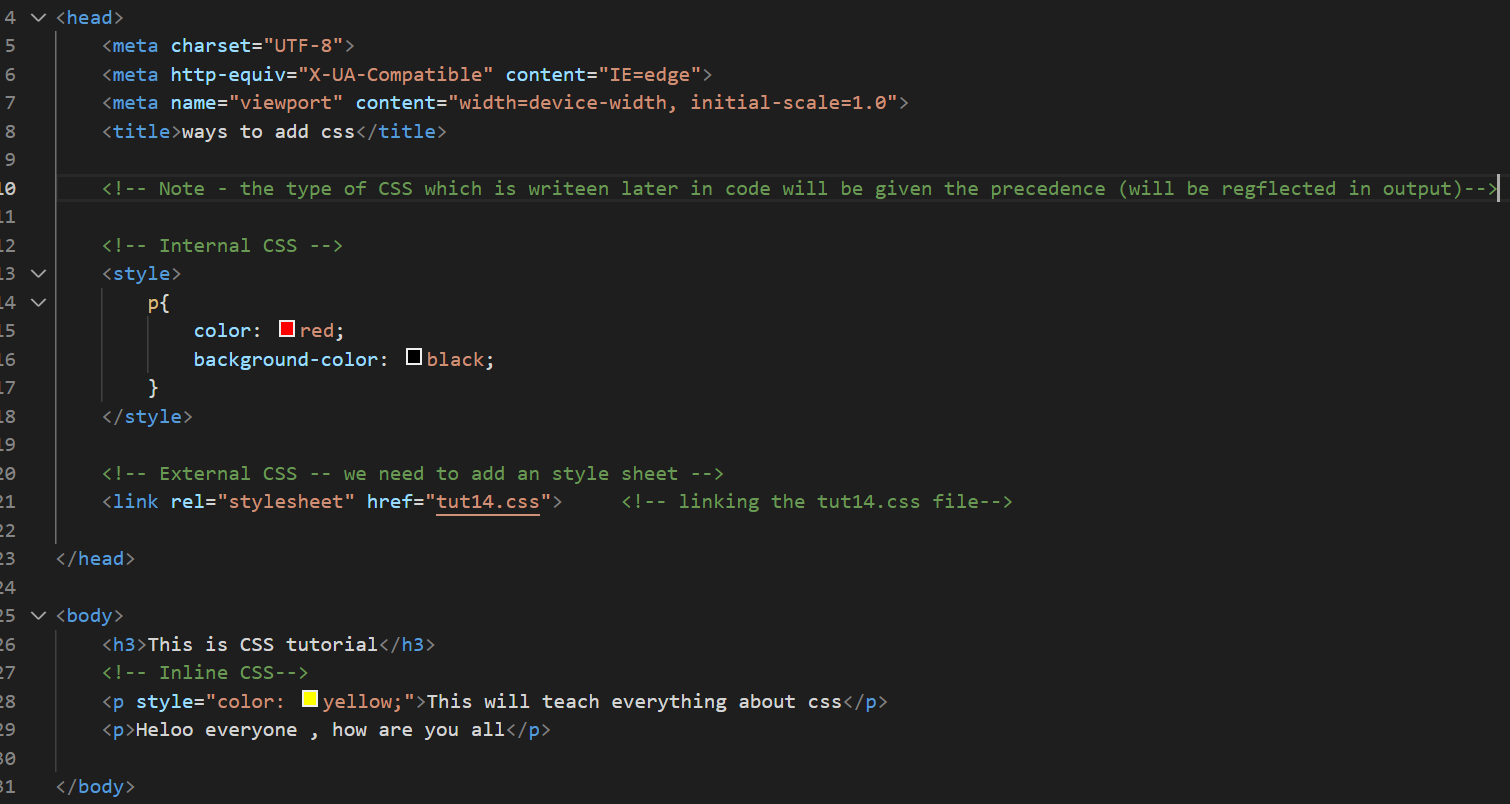
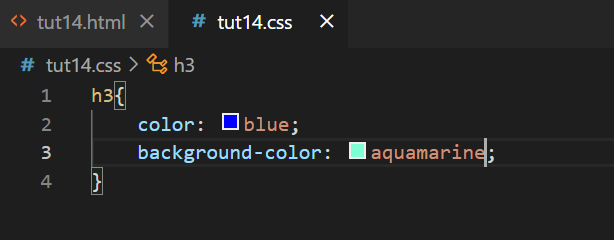
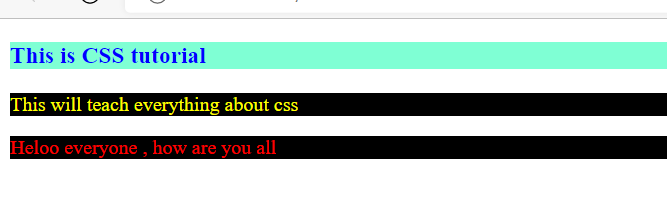
**CSS ALL CODES SS IN ONE**

**Ways to add css in html file**

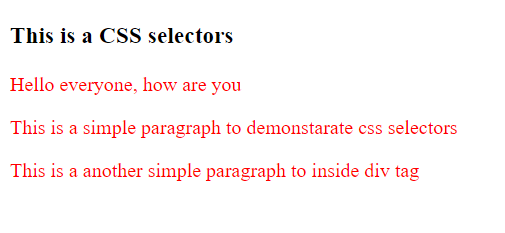
****

****

****

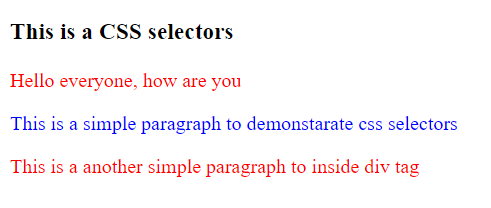
**Css element selector**

****

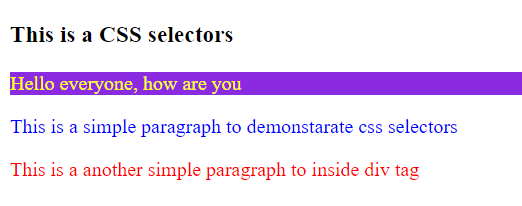
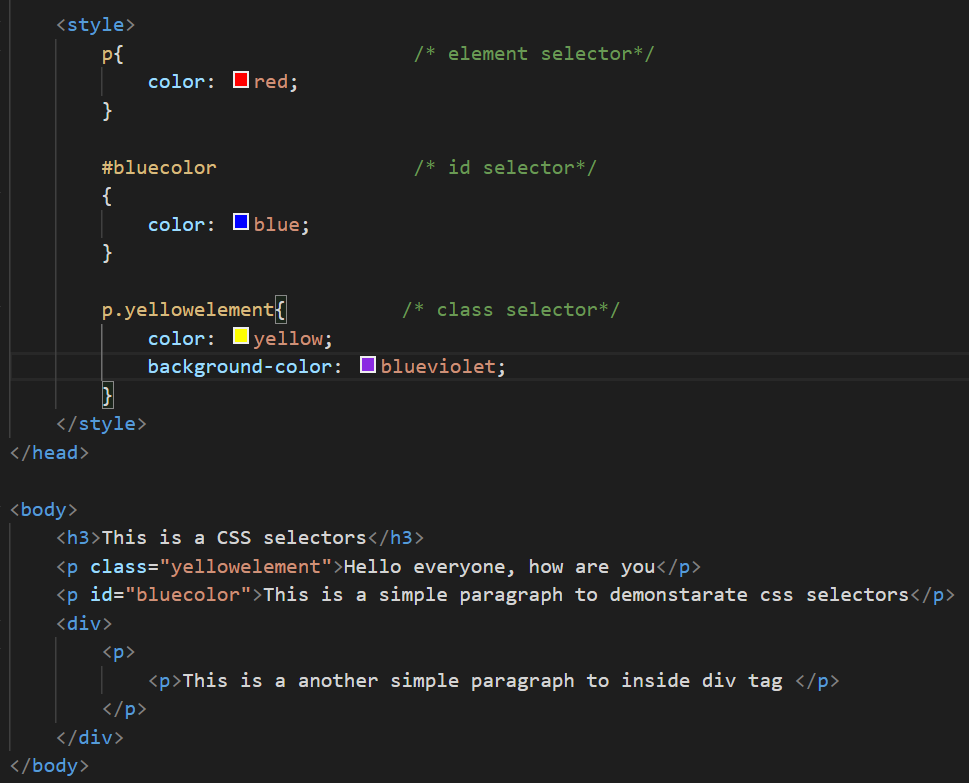
****

**Css id selector:**

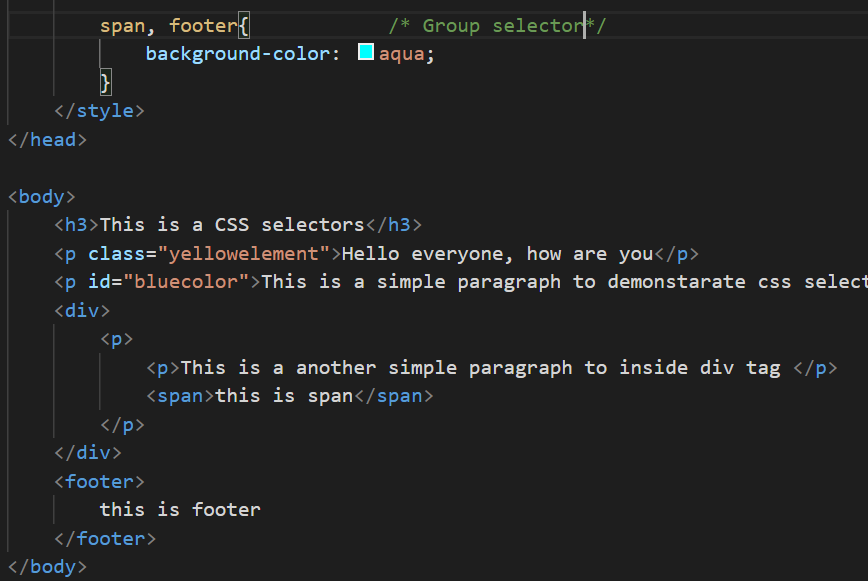
****

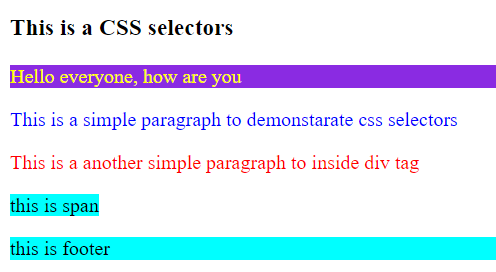
****

**Css class selector**

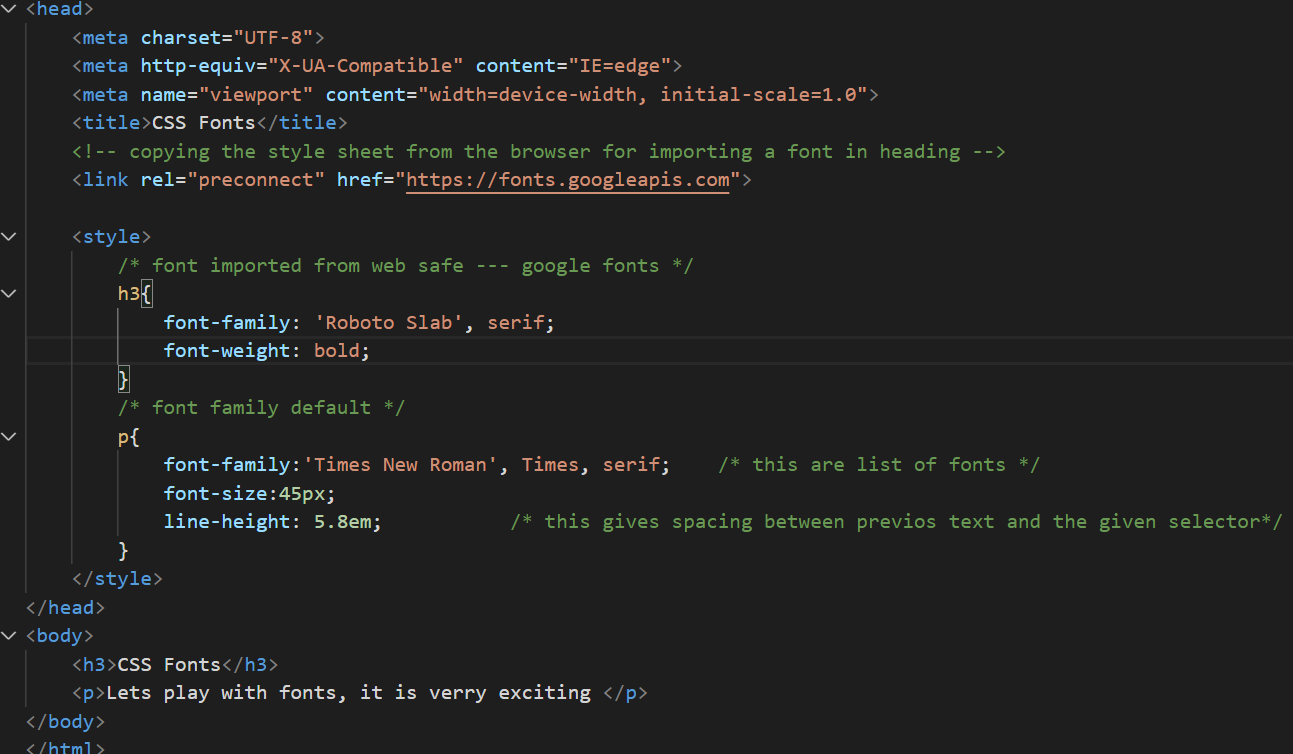
****

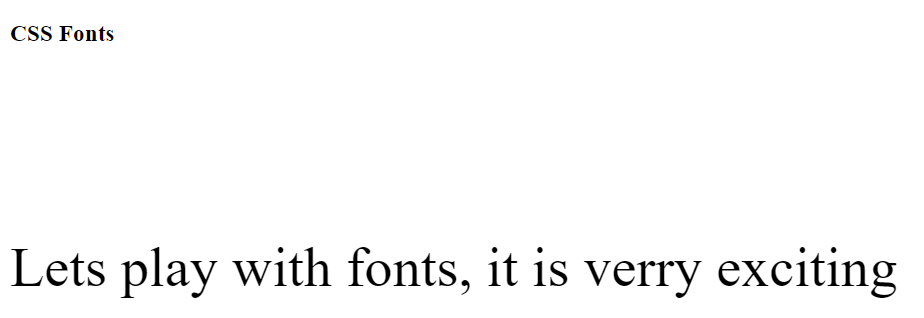
**Group selector**

****

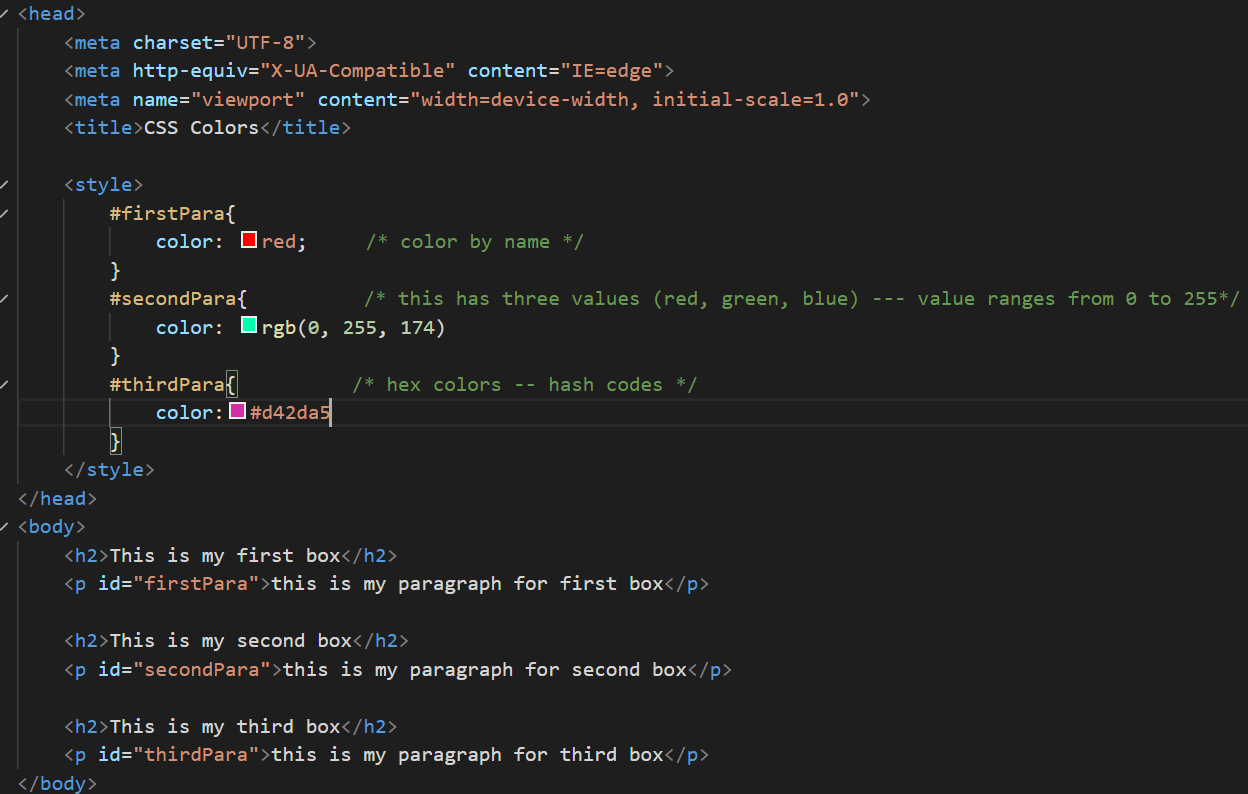
****

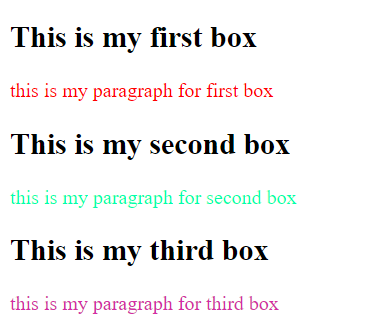
**Fonts**

****

****

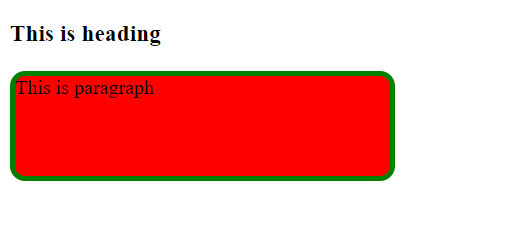
**CSS Colors**

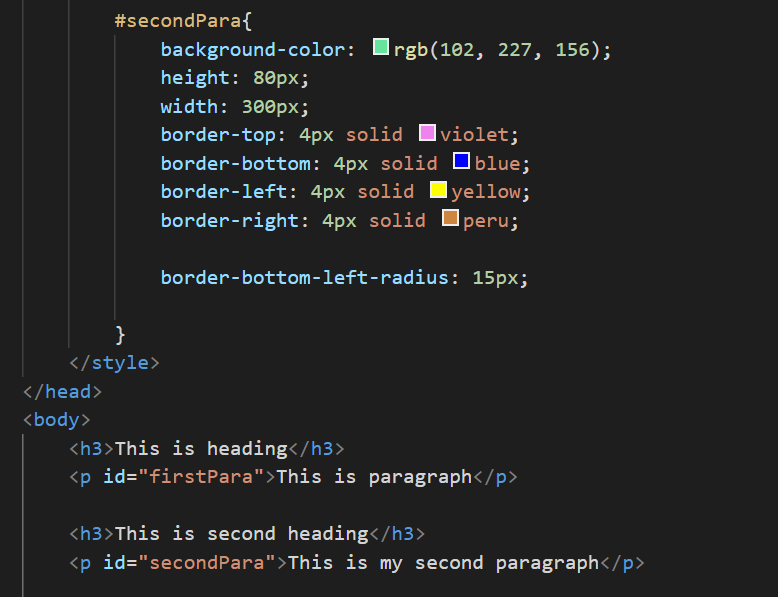
****

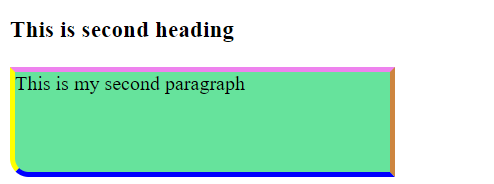
****

**Borders**

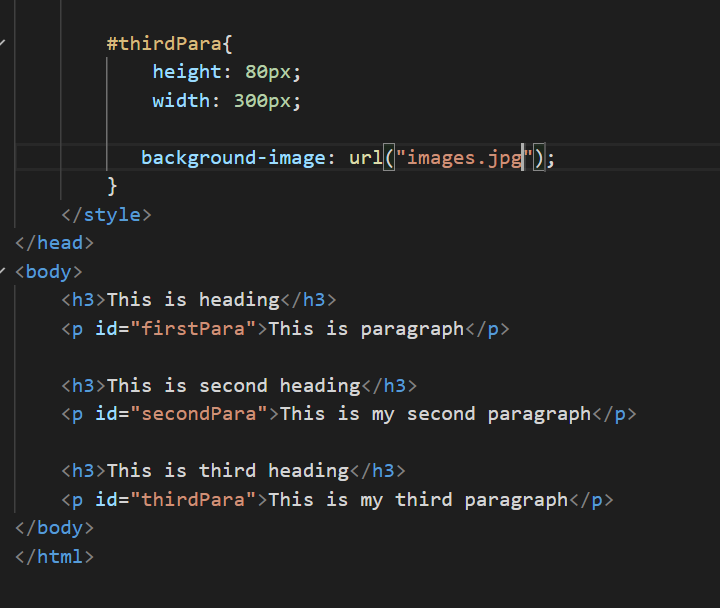
****

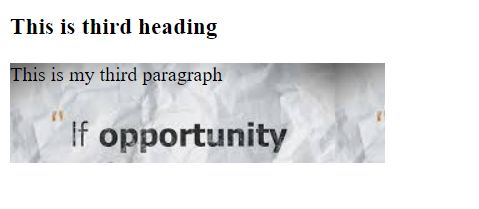
****

****

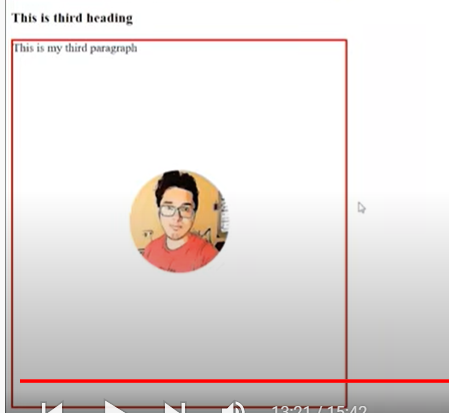
****

**Adding background image to para**

****

****

****

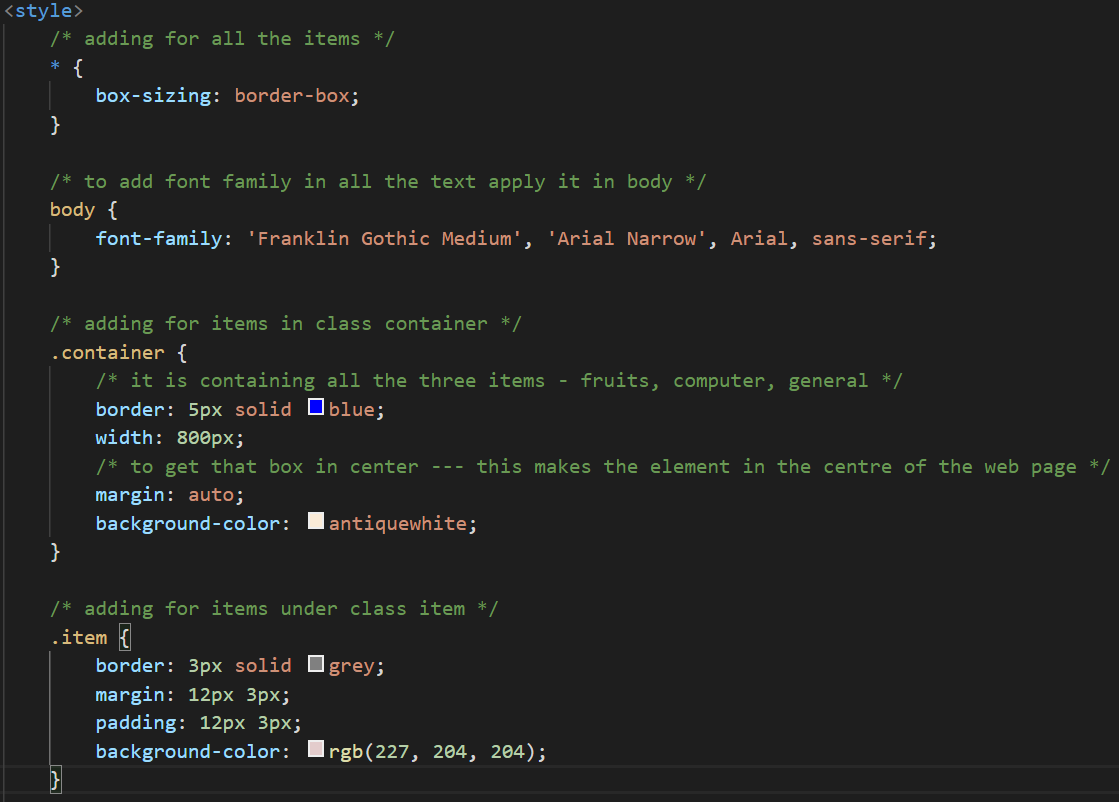
****

**CSS Box model**

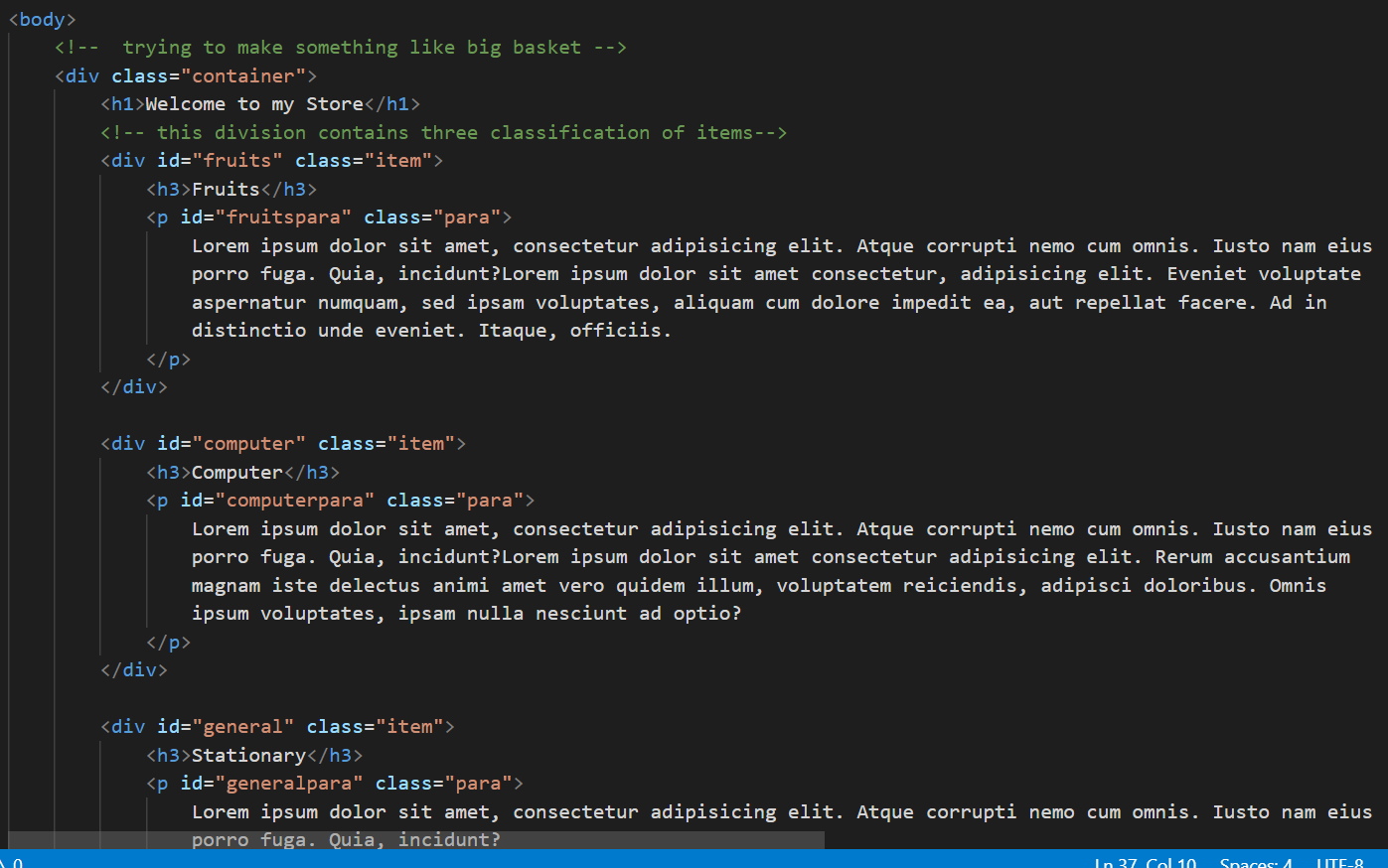
****

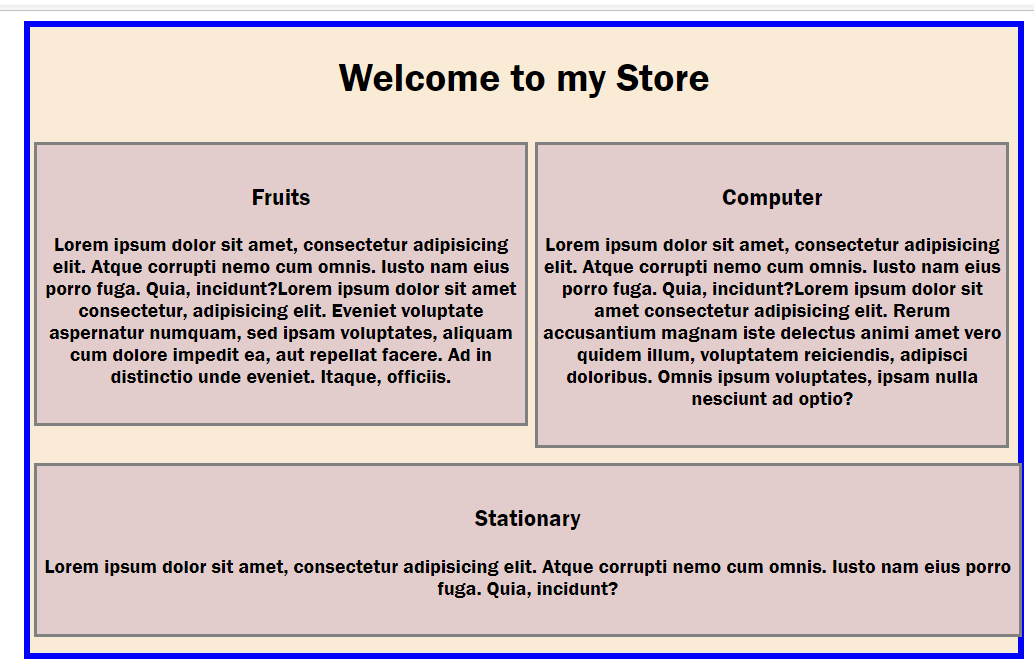
****

**Float and Clear**

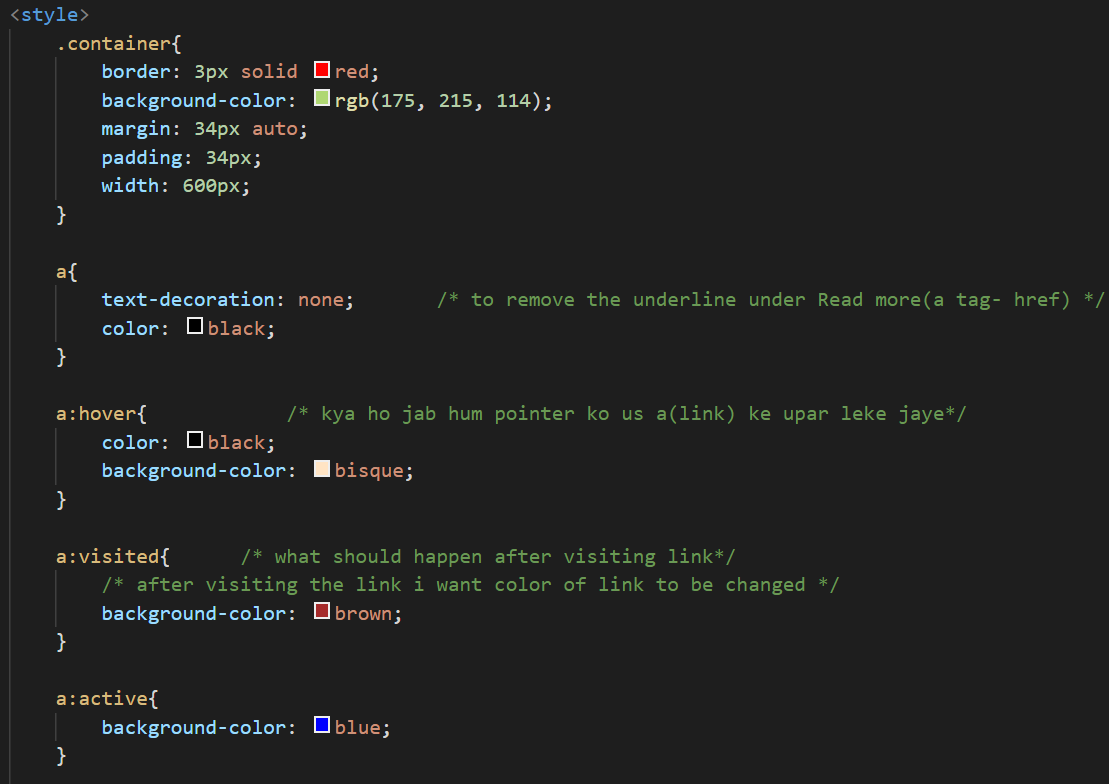
****

****

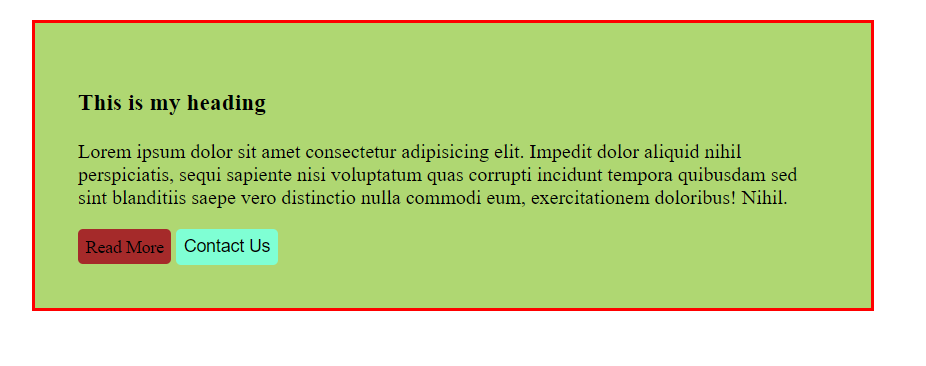
****

****

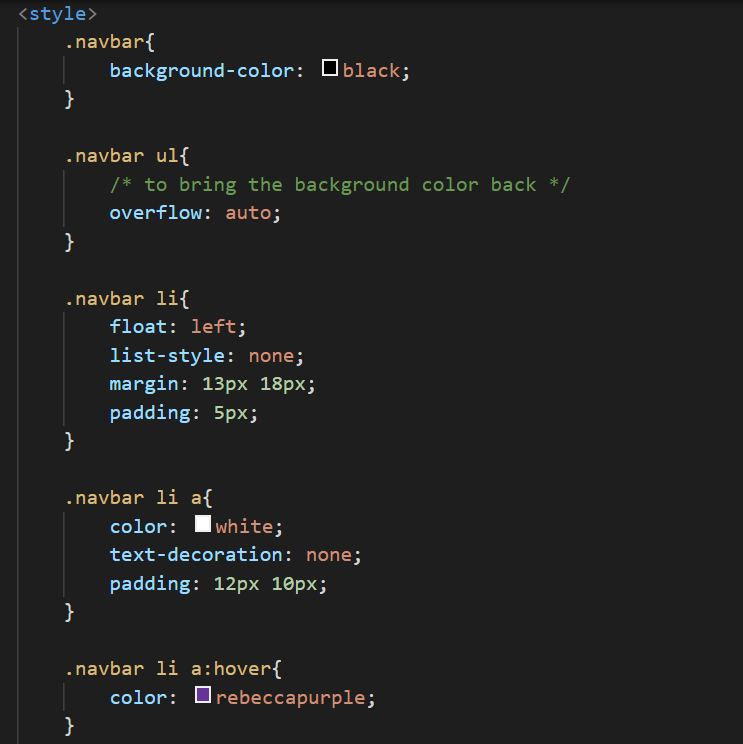
**Styling links and buttons**

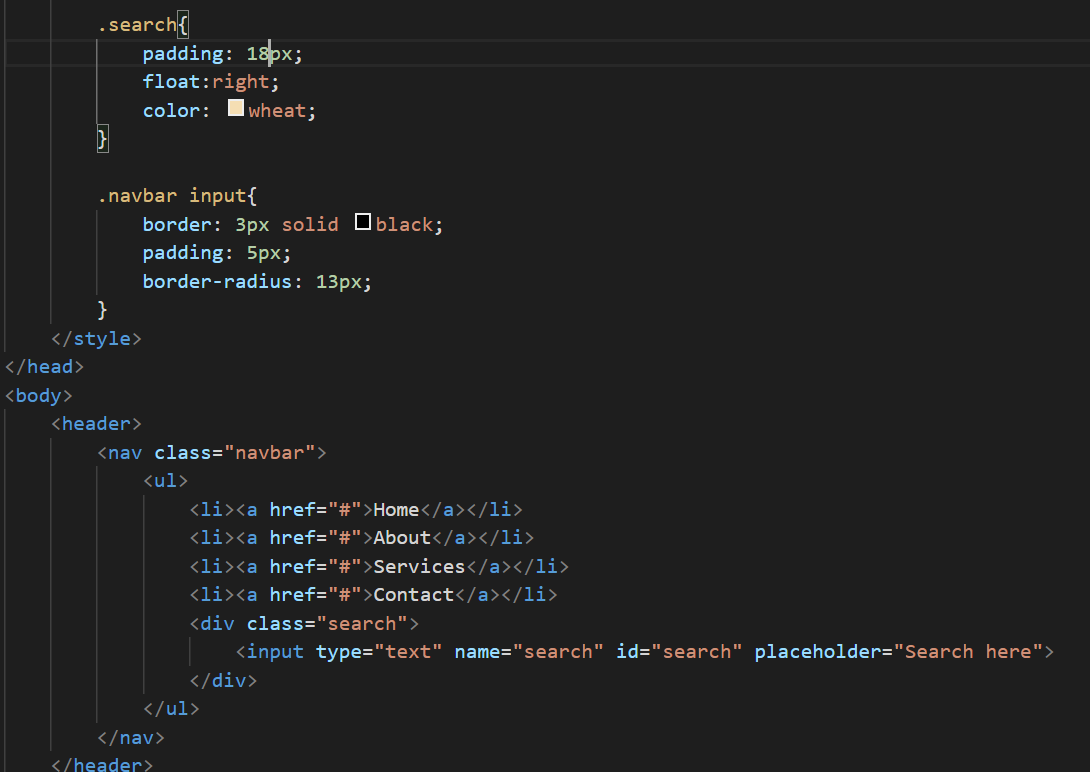
****

****

****

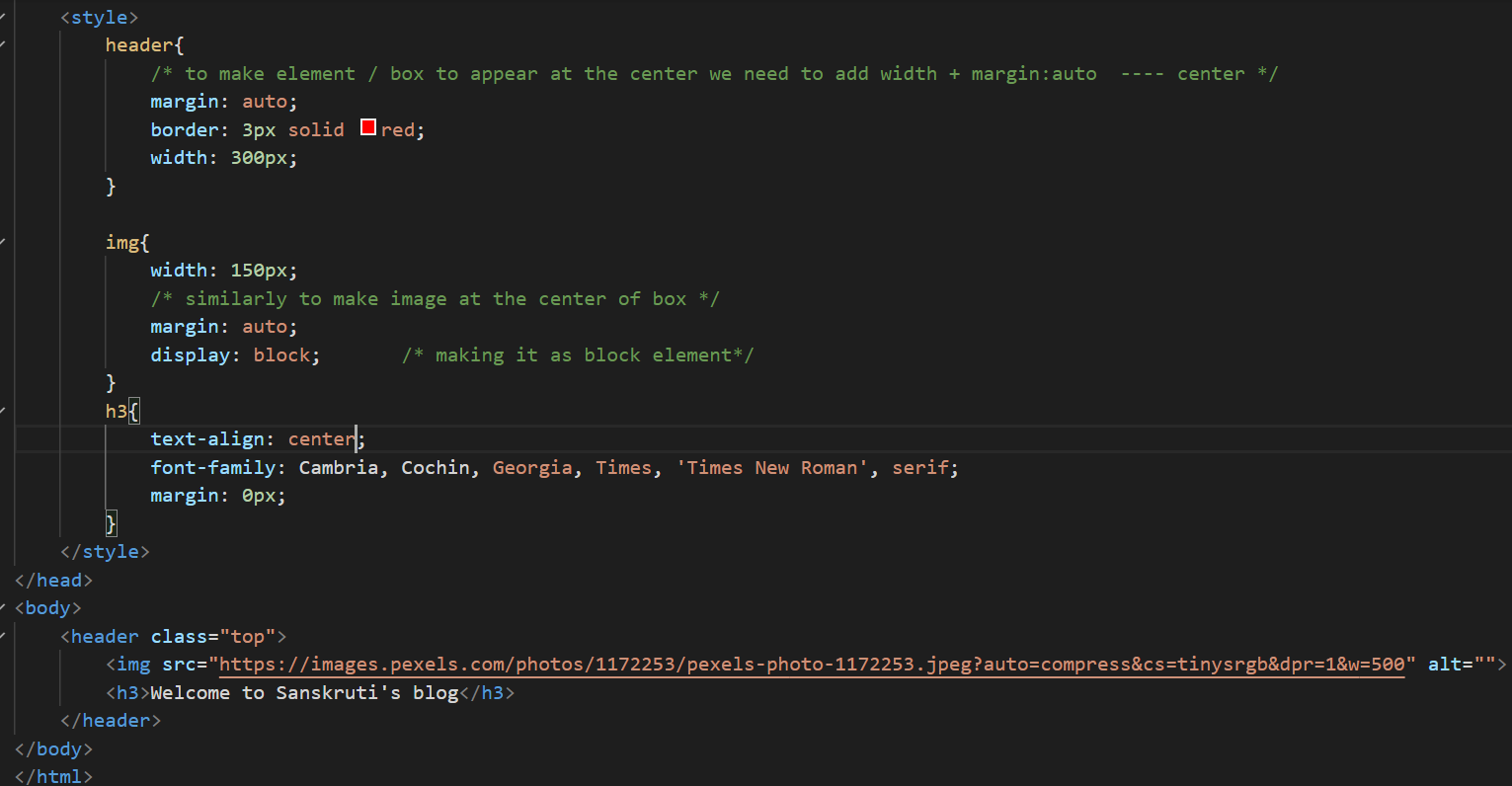
**Navigation bar**

****

****

****

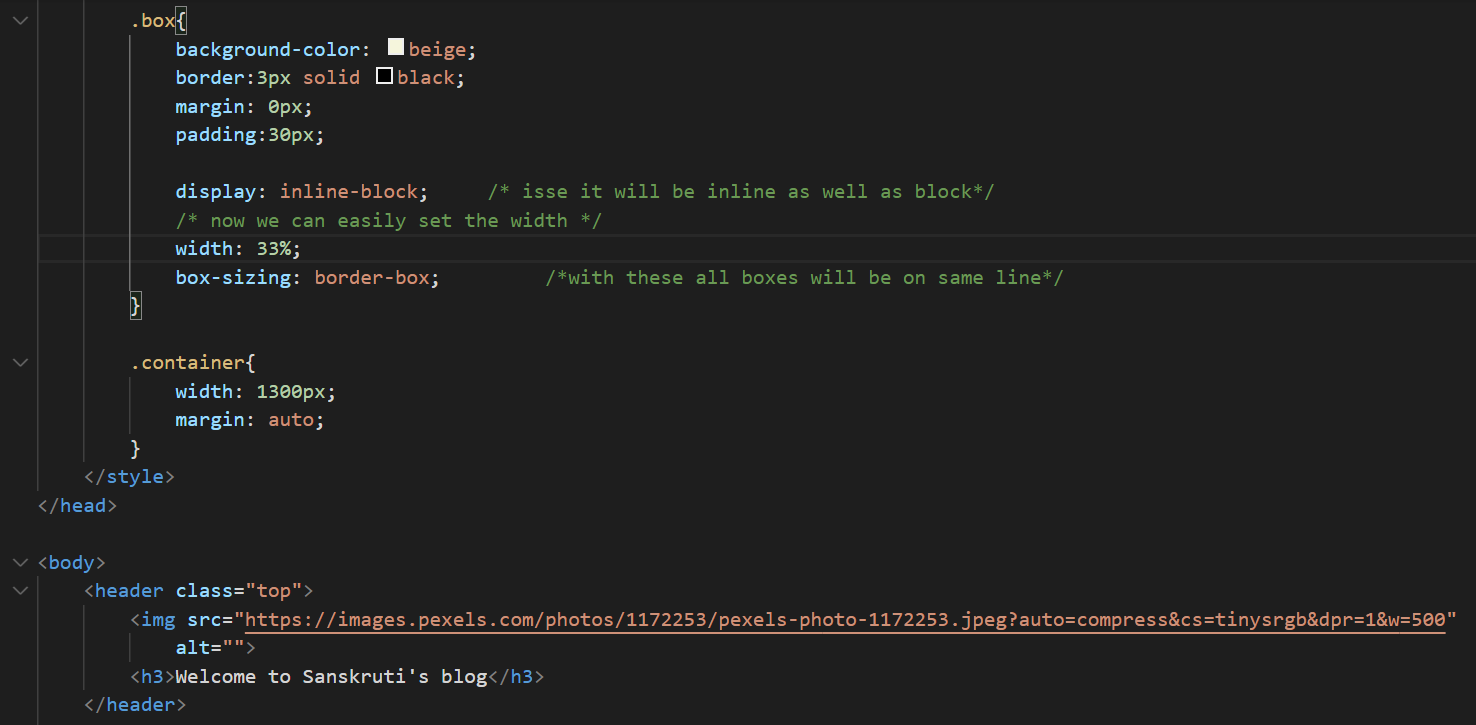
**Display property**

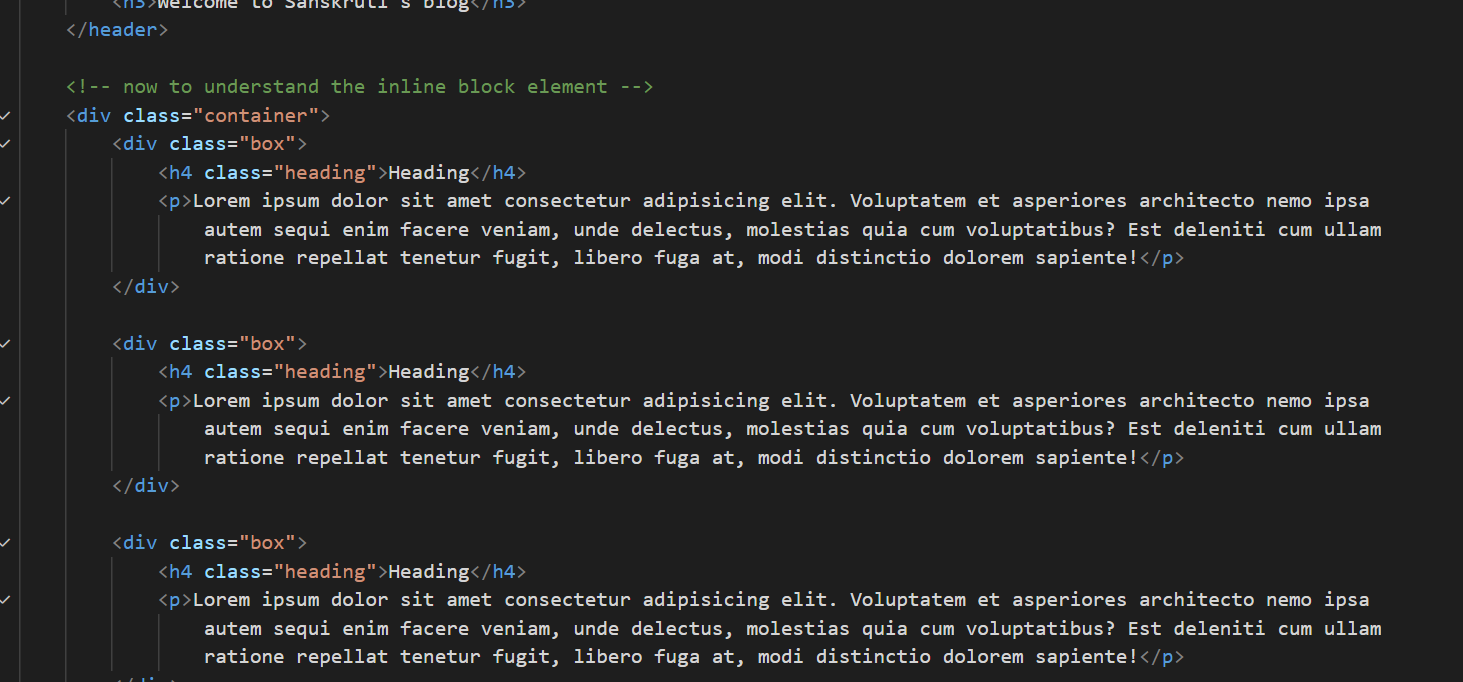
****

****

**Inline block**

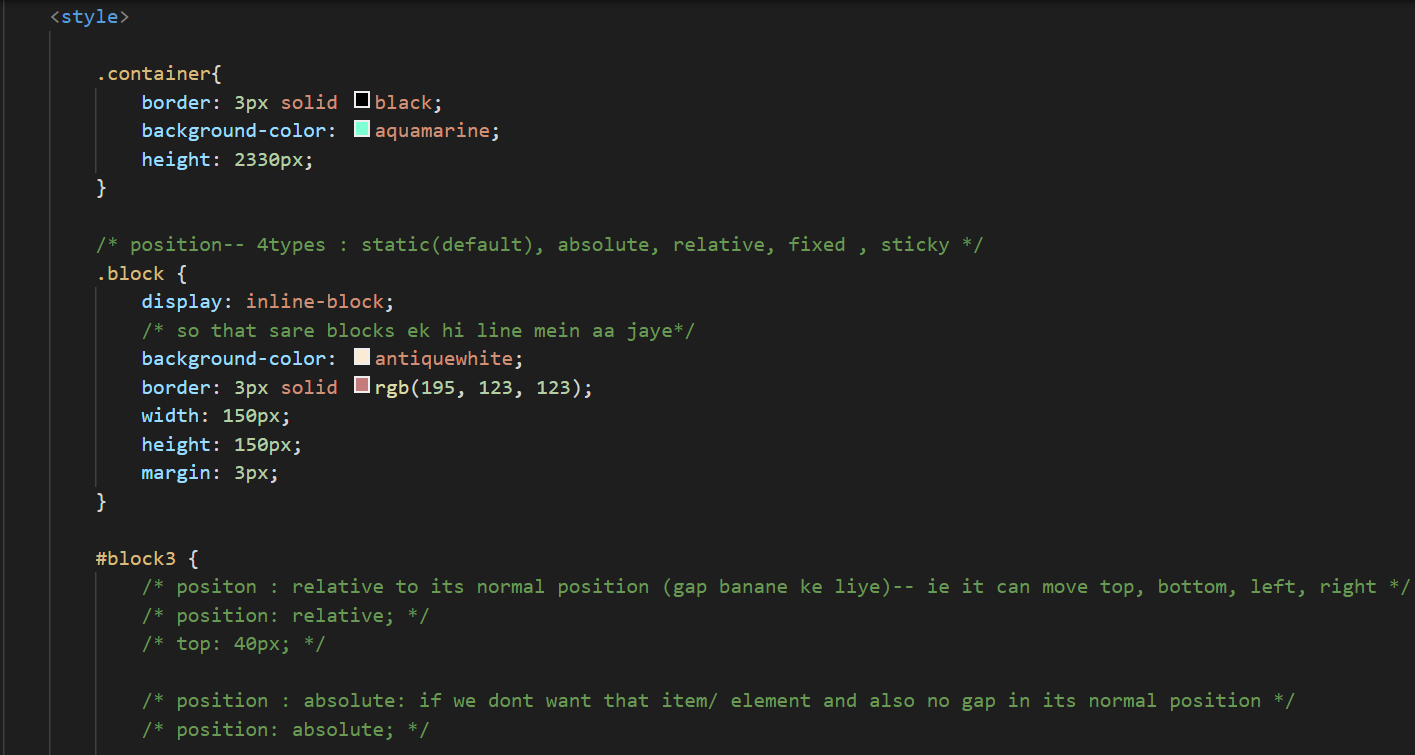
****

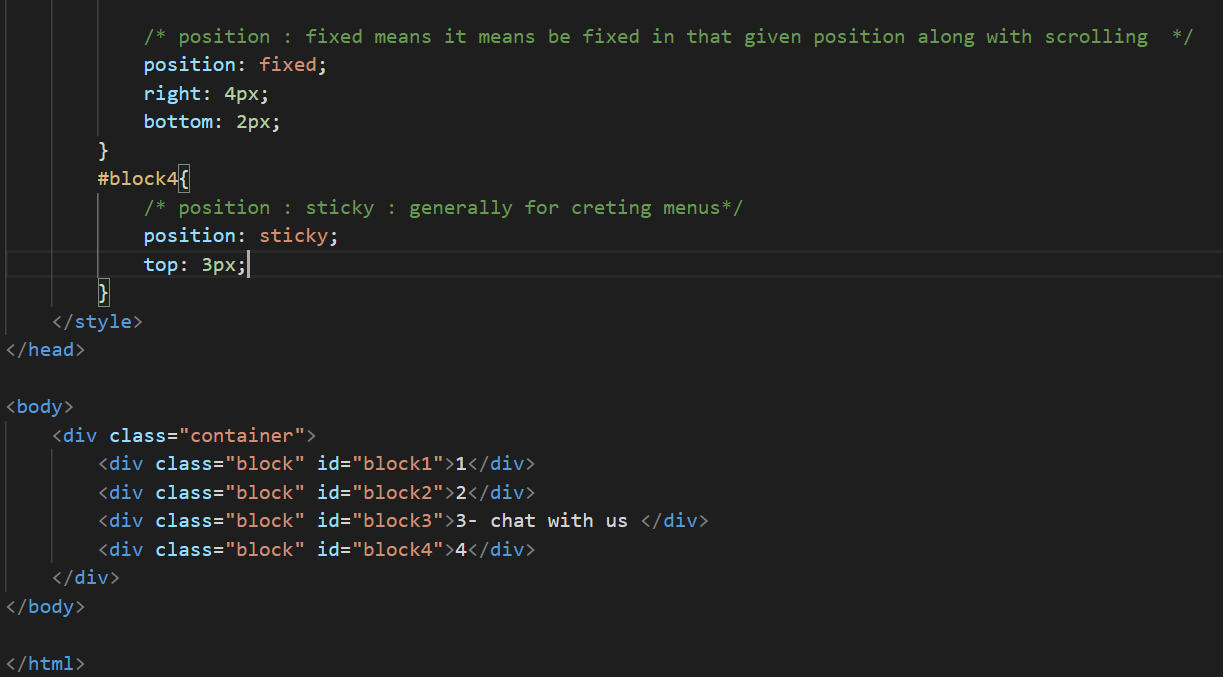
****

****

****

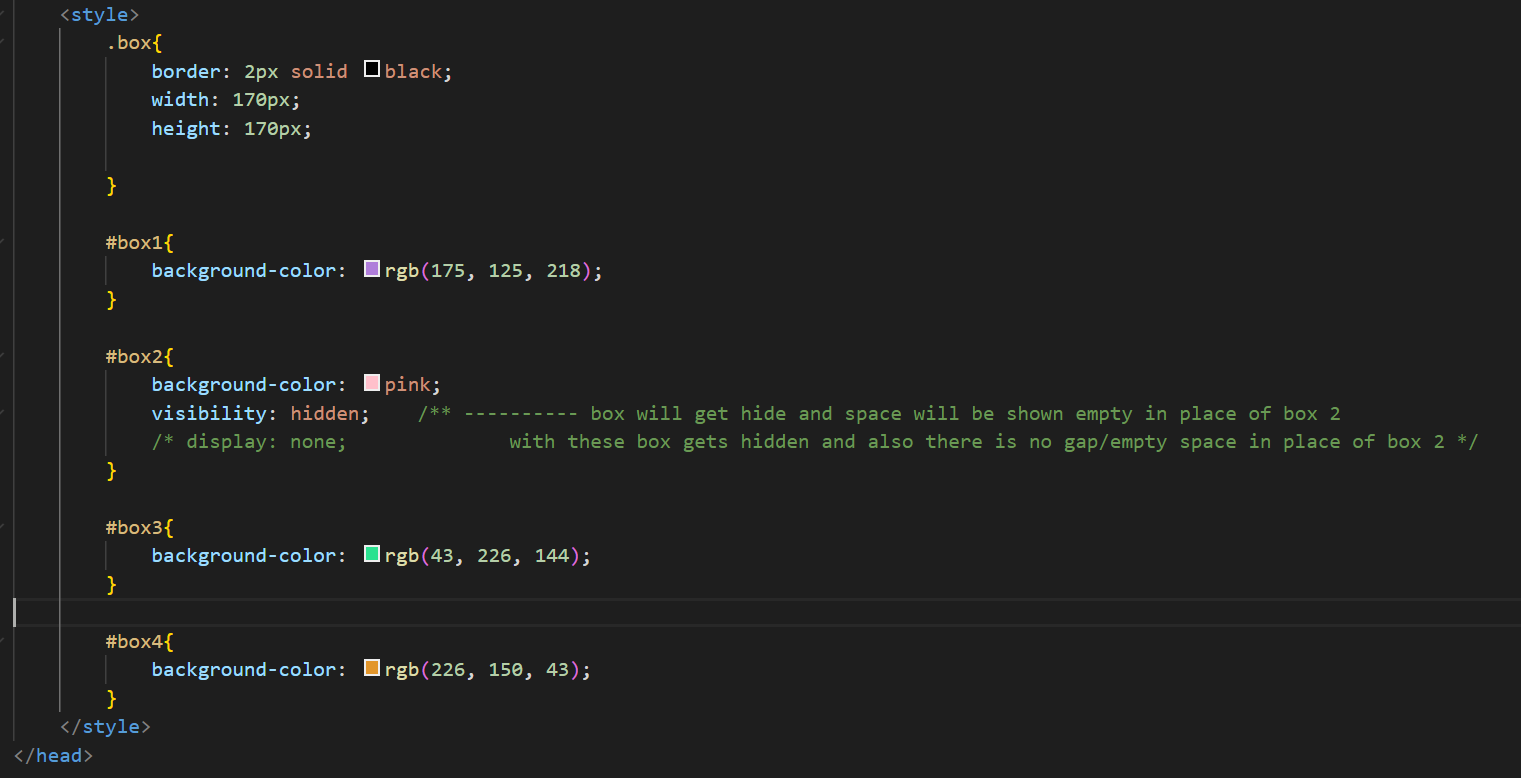
**CSS positions**

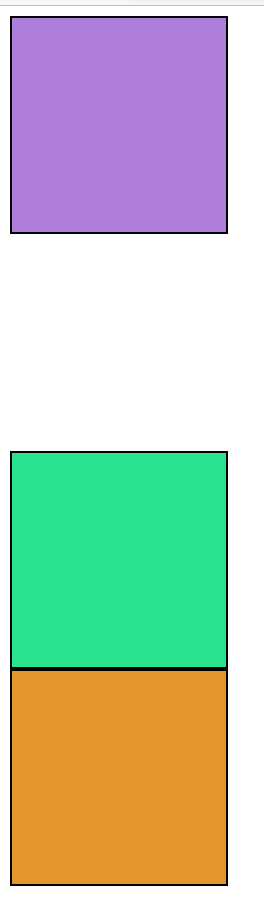
****

****

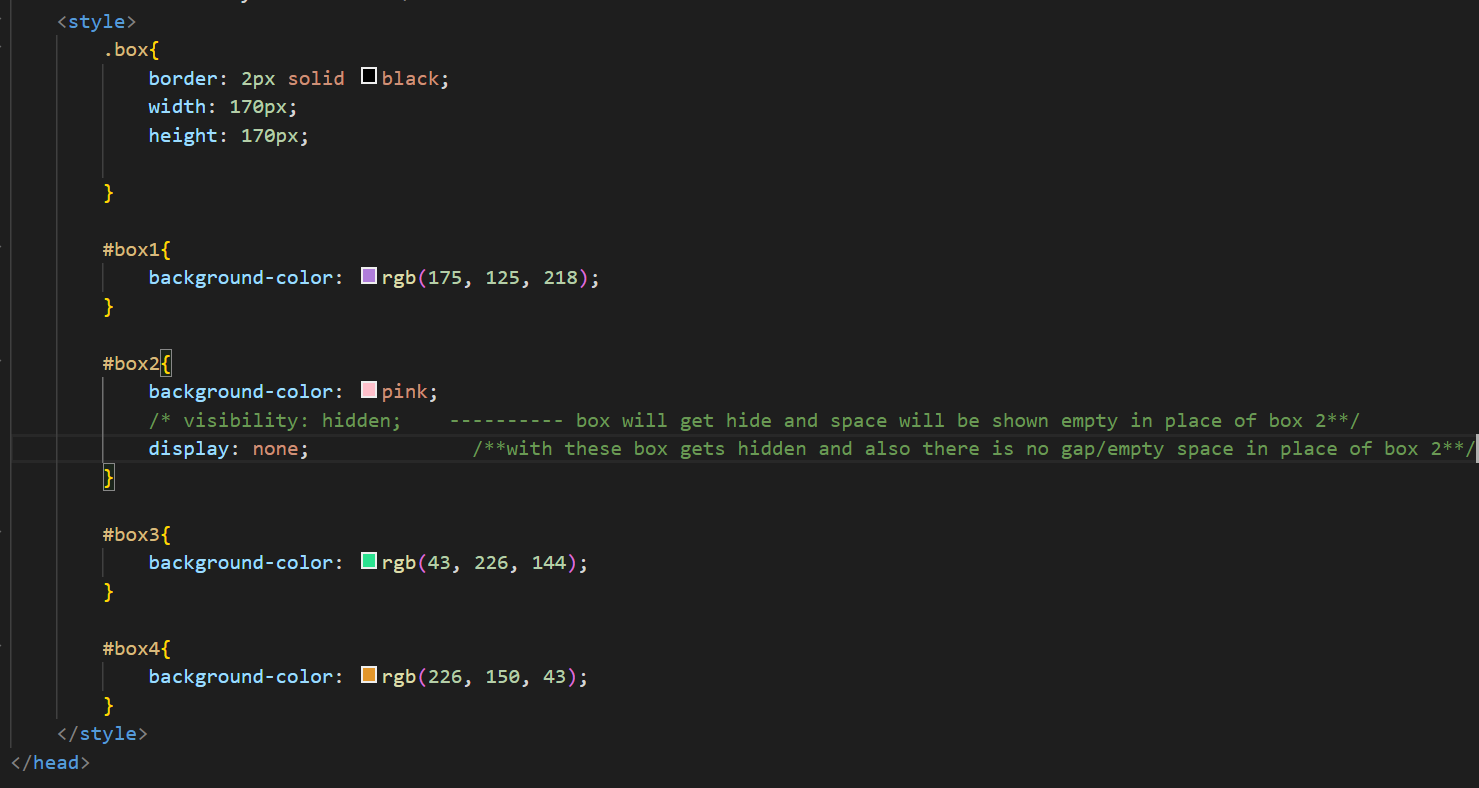
****

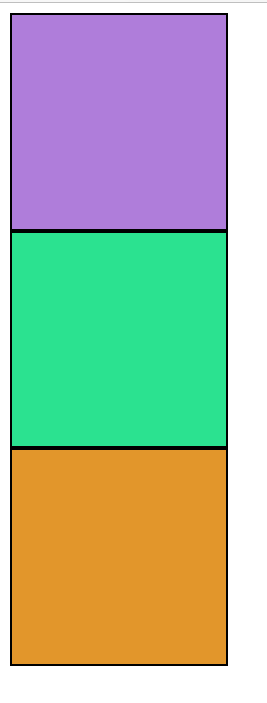
**Visibility**

****

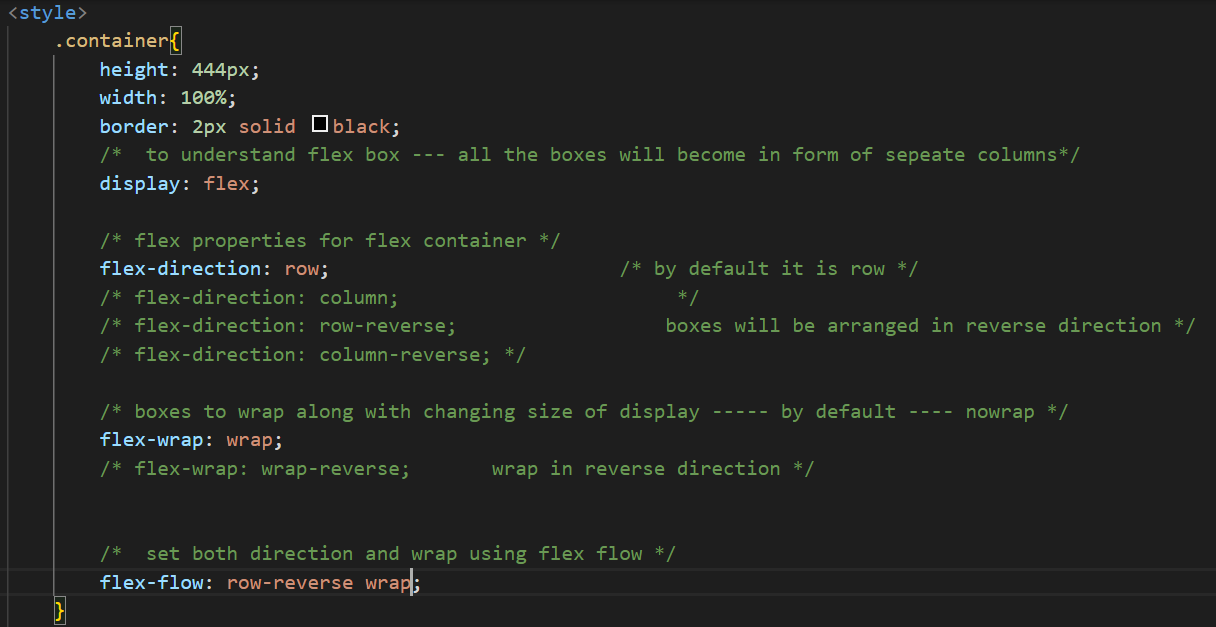
****

**Display none**

****

****

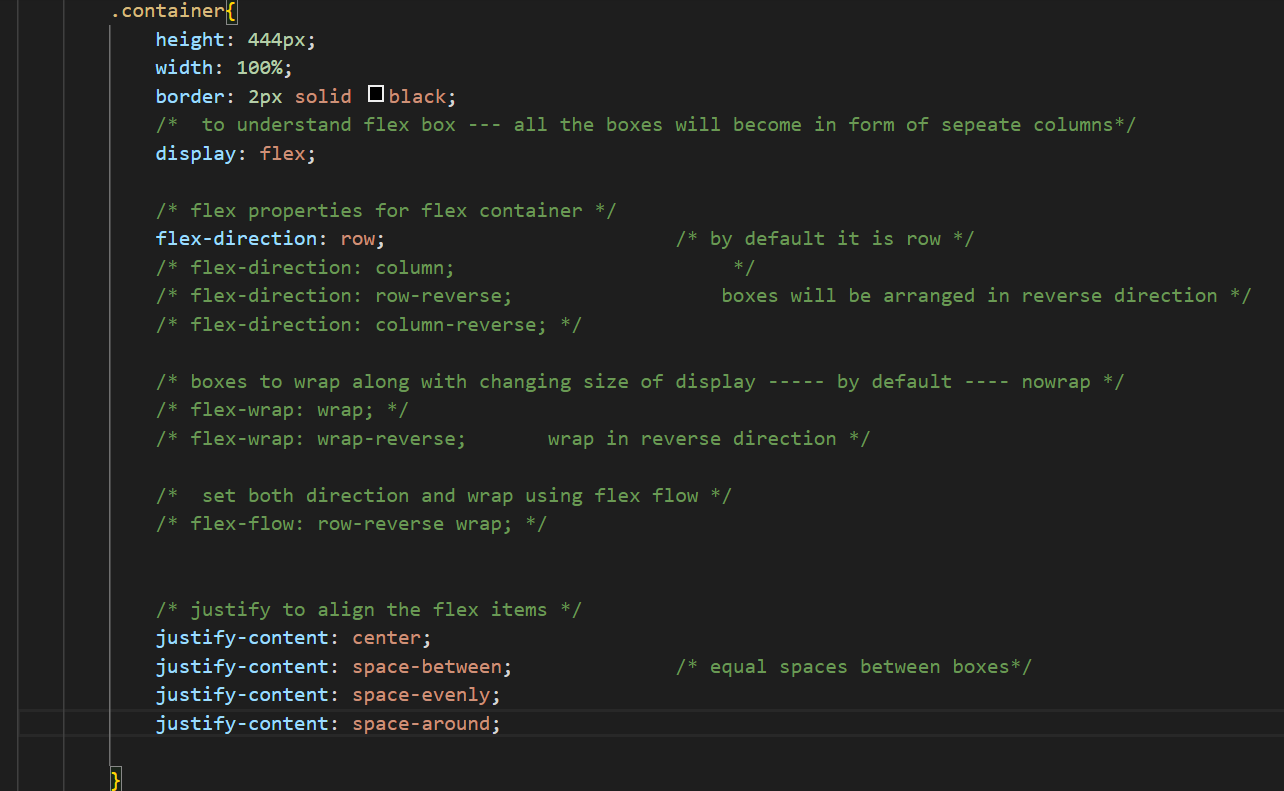
**Flex**

****

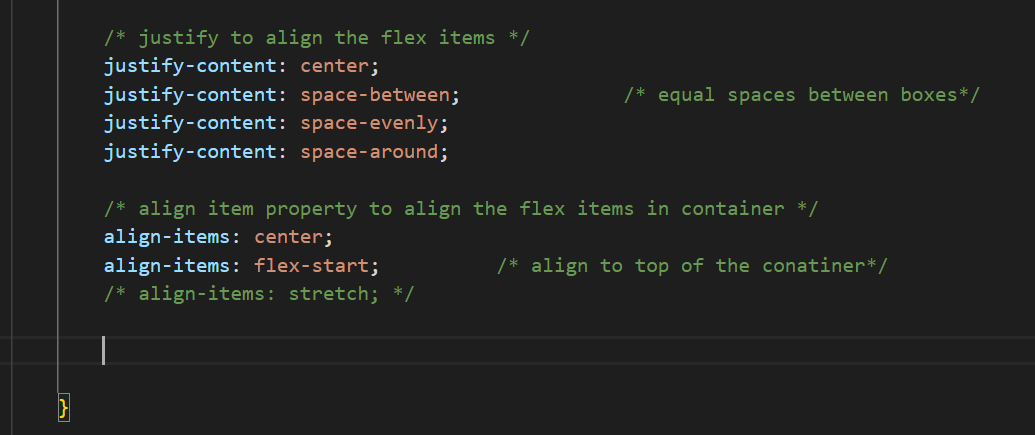
****

****

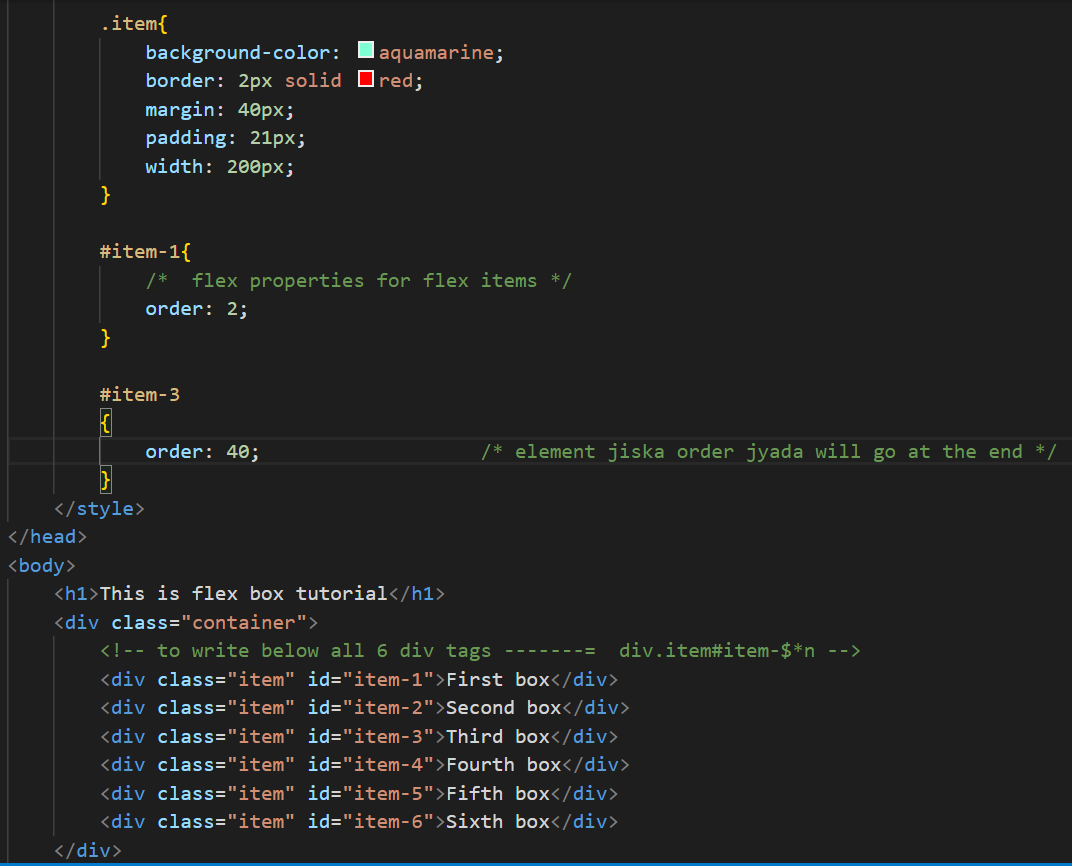
**Justify-content**

****

**Align item**

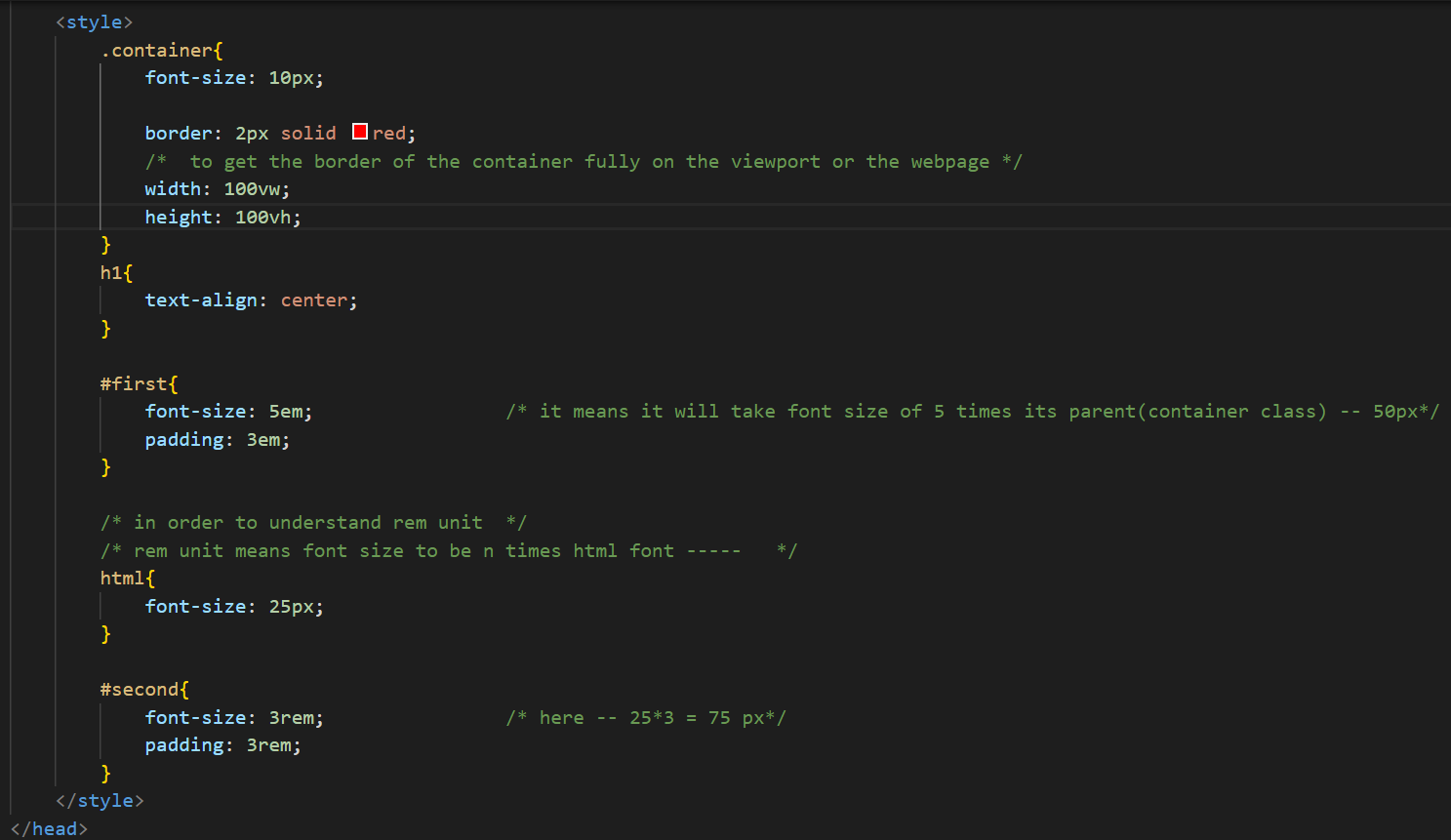
****

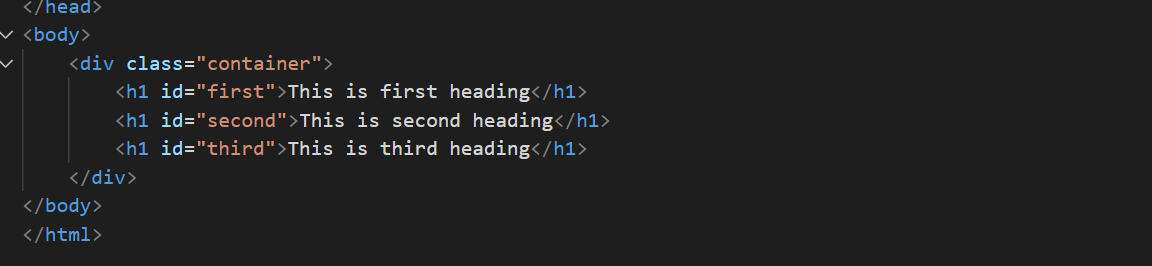
**Flex item properties**

****

****

**Responsive design**

****

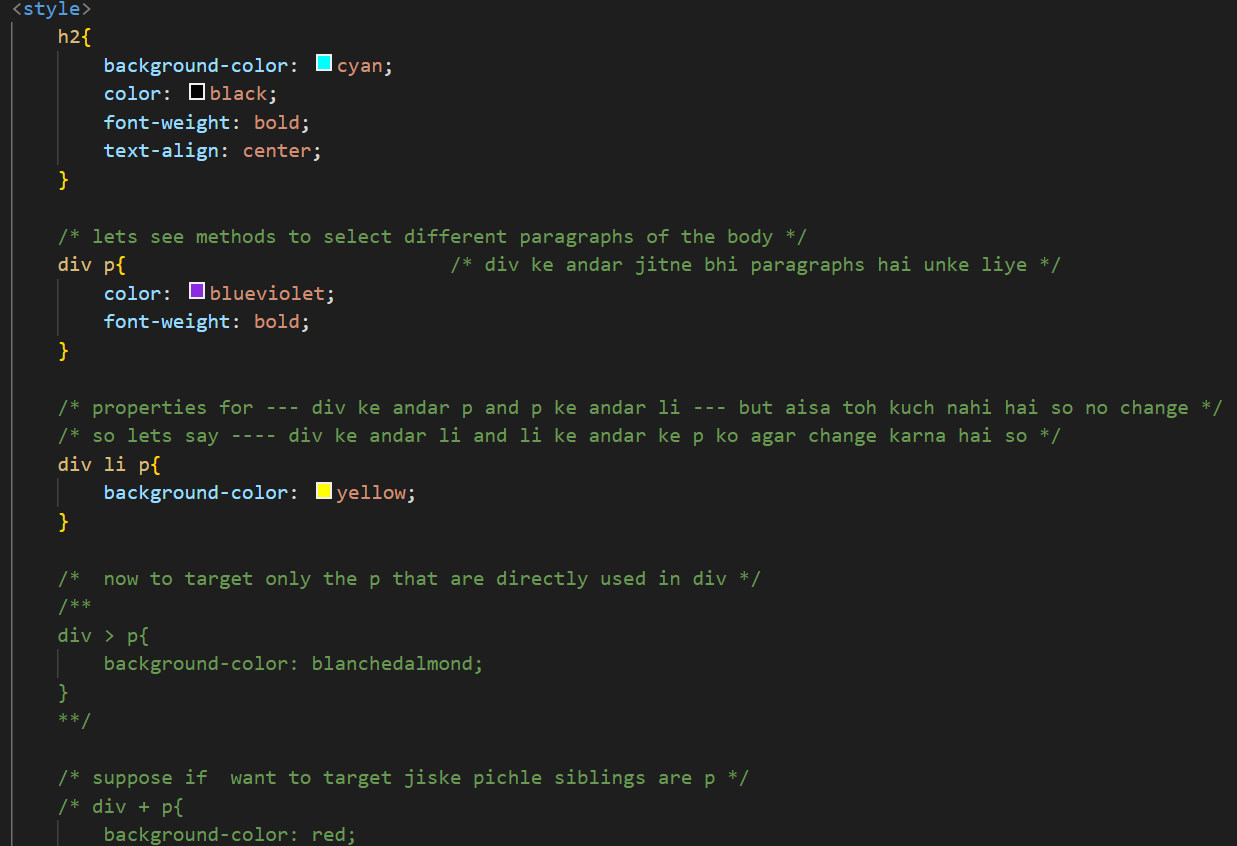
****

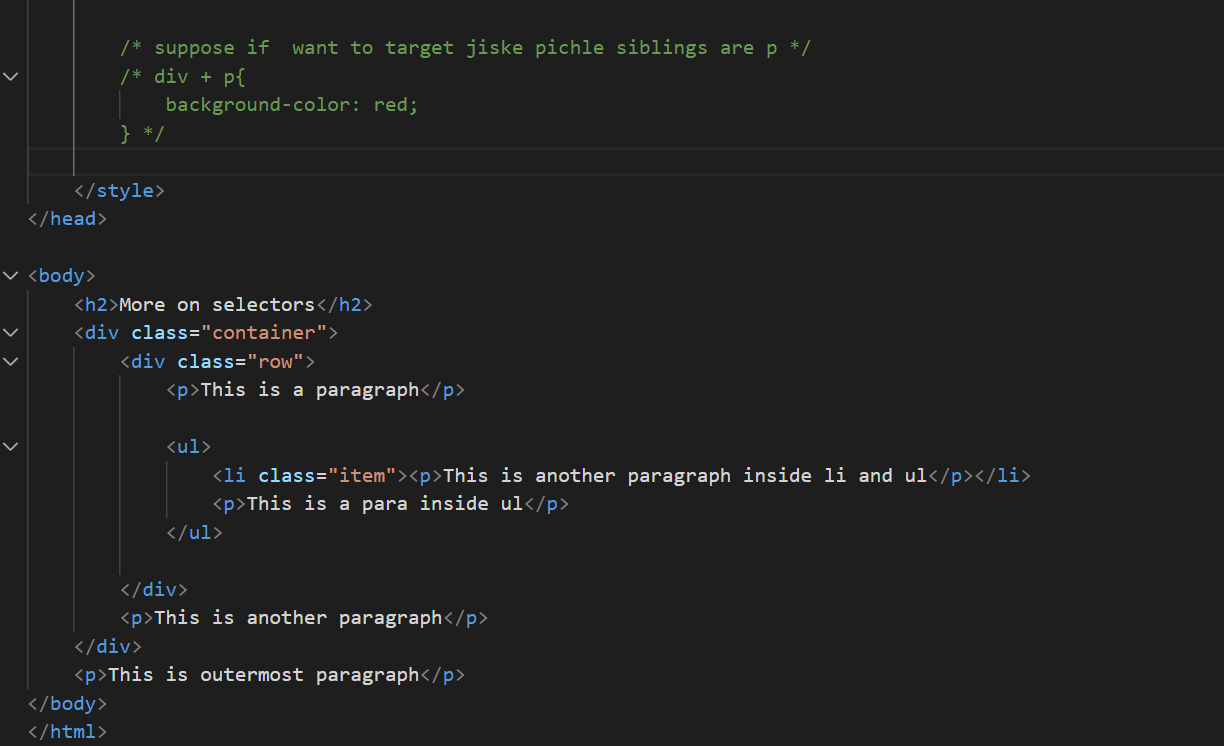
**Media querry**

****

****

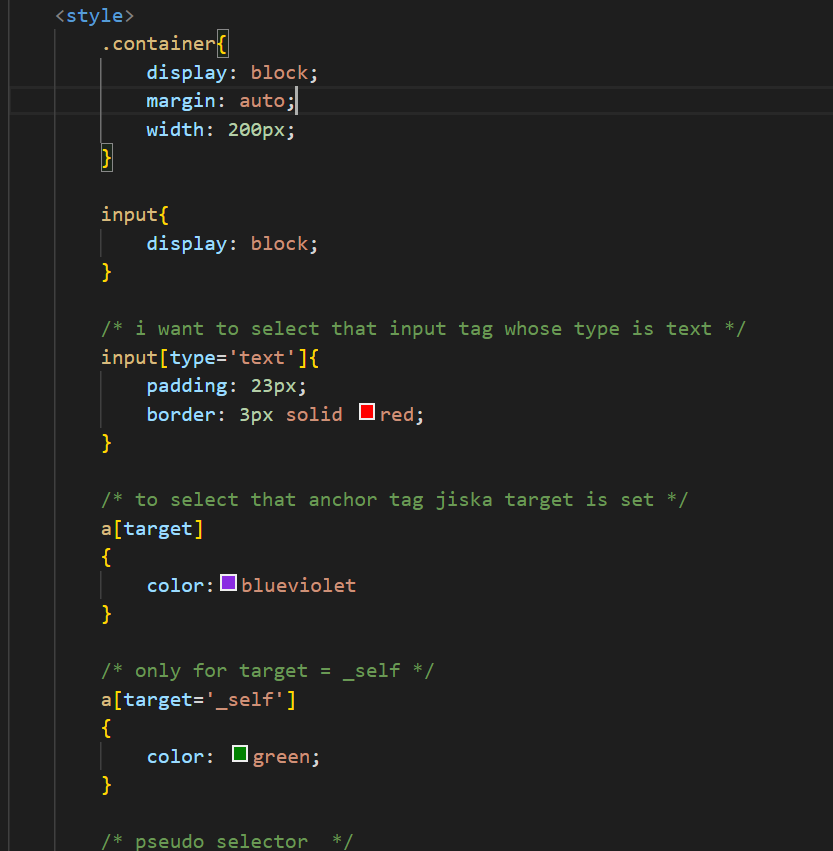
**More on css selectors**

****

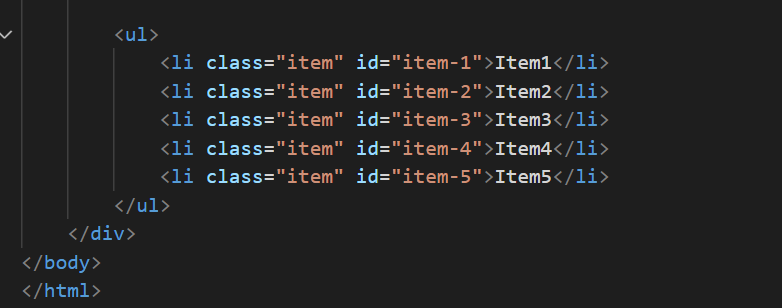
****

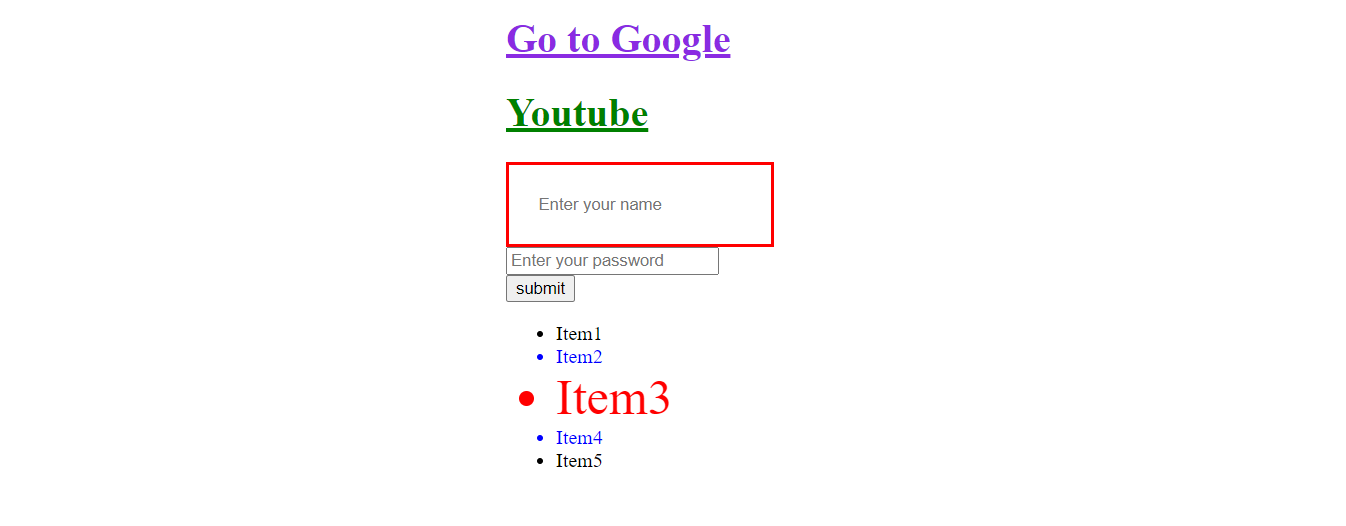
****

Attribute And nth child pseudo selector

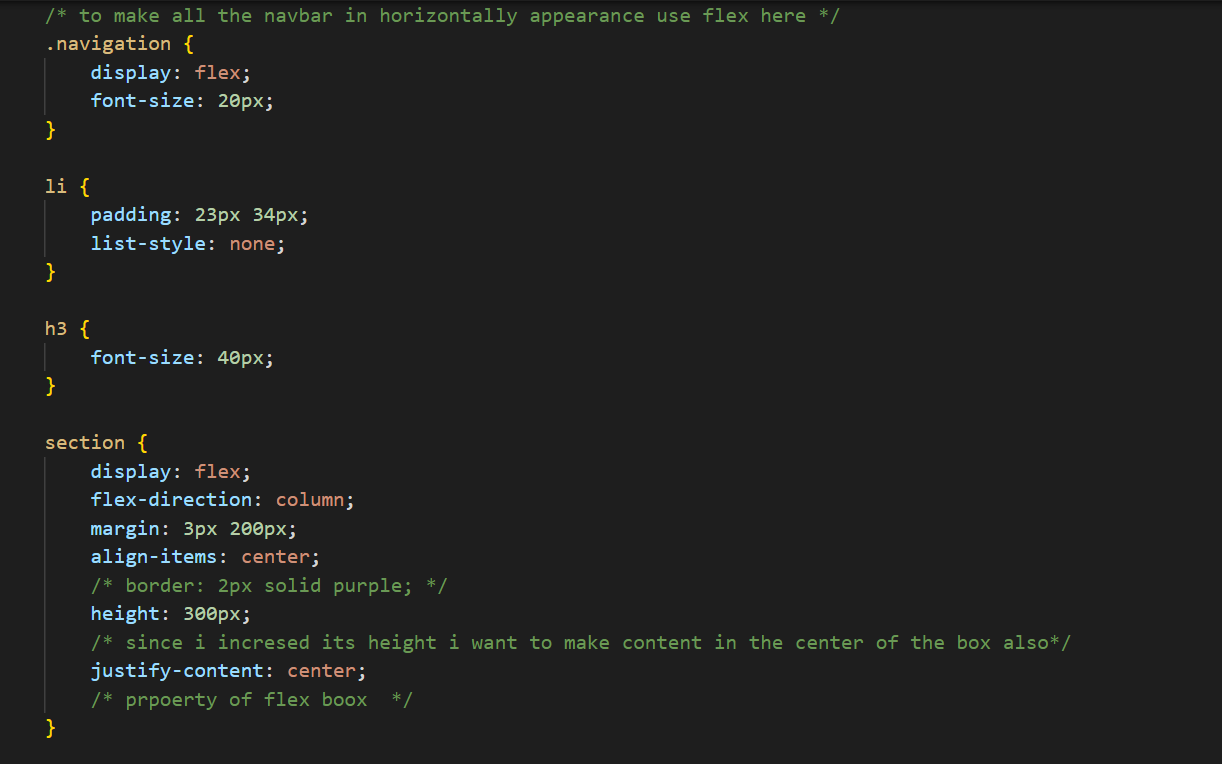
****

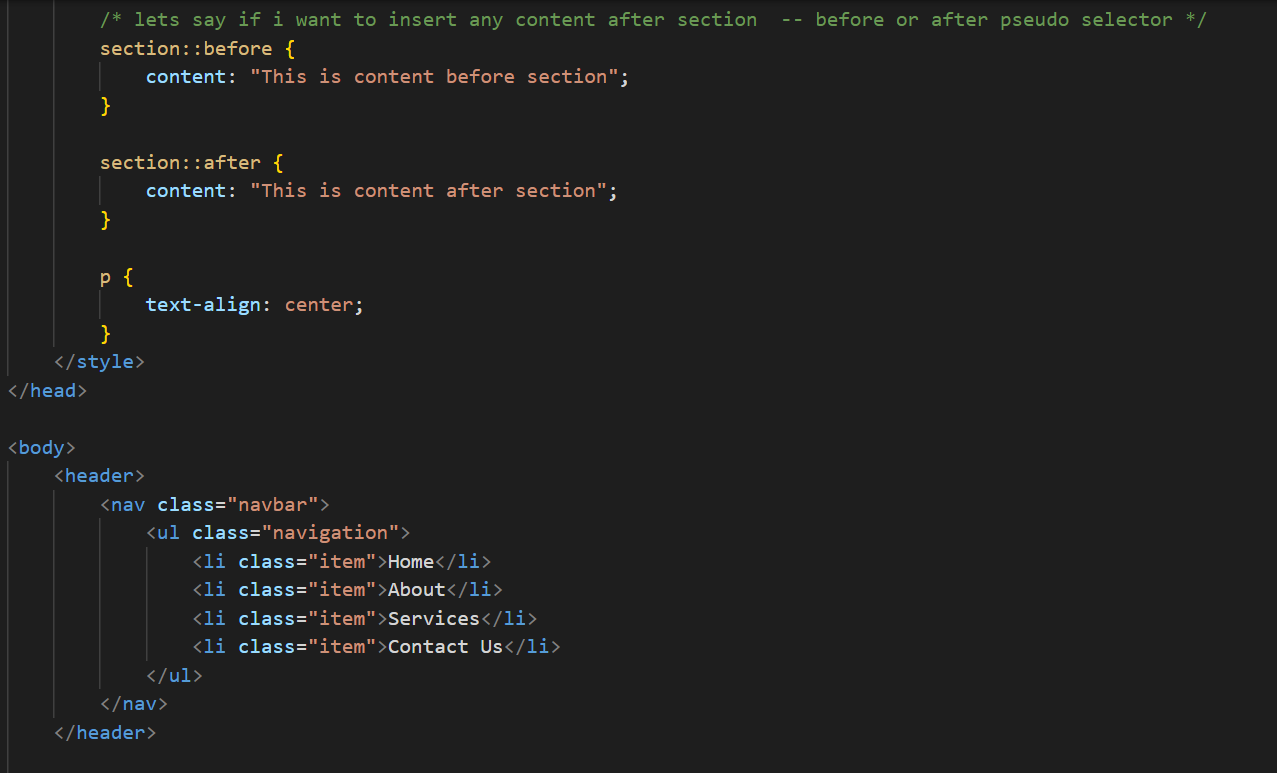
****

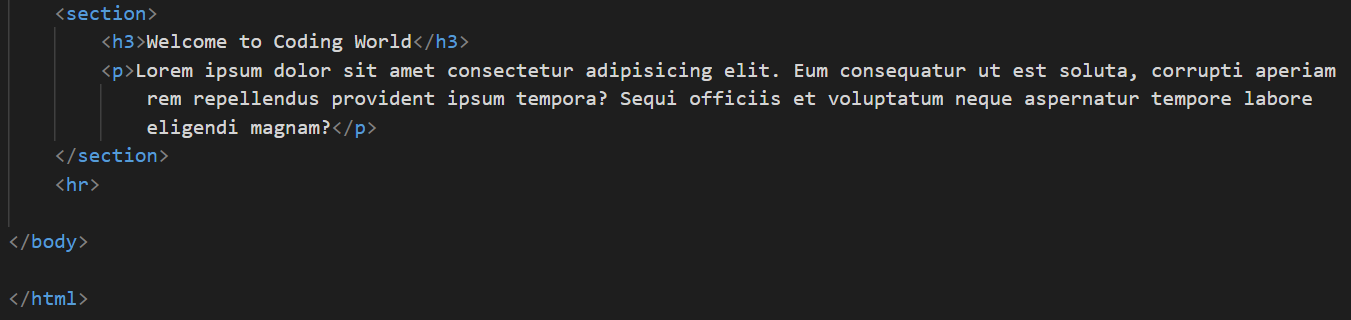
****

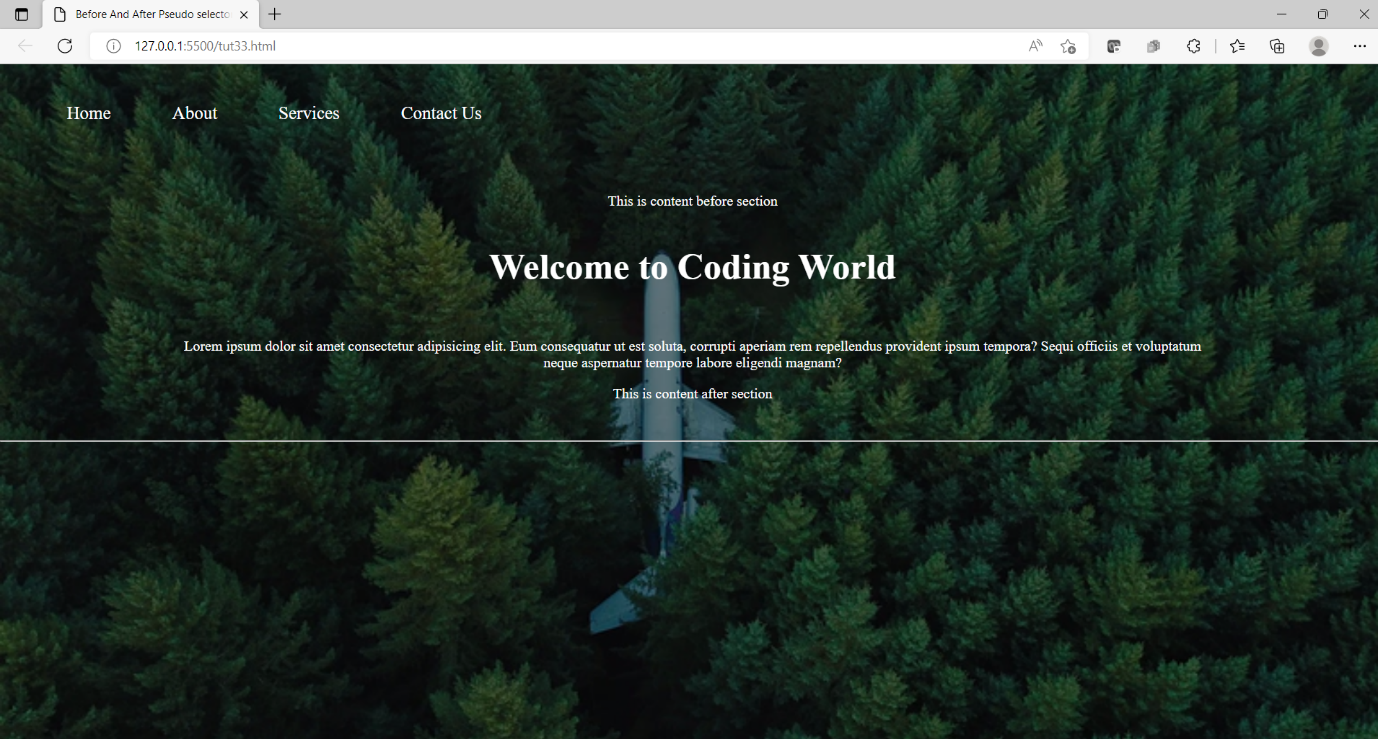
****

**Buliding effective page and background image**

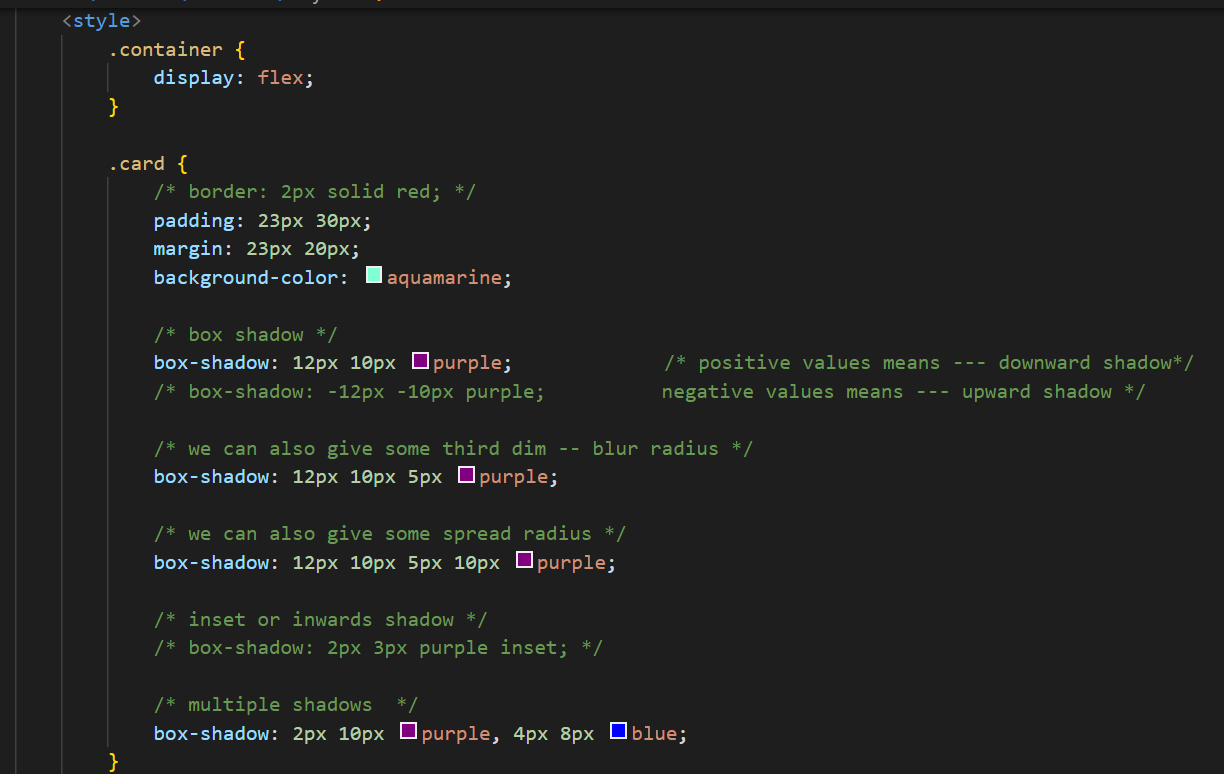


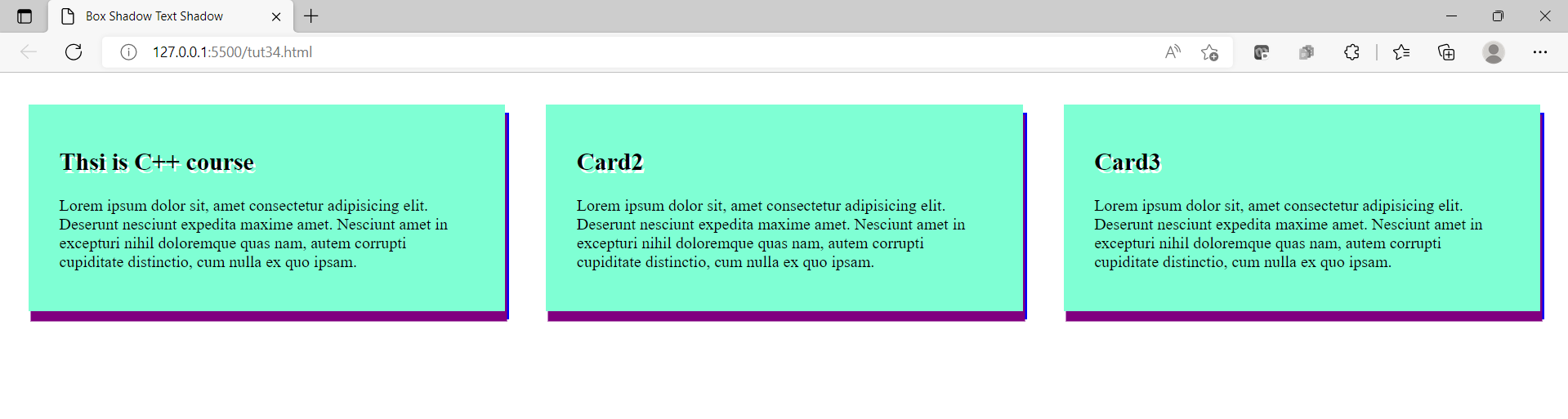
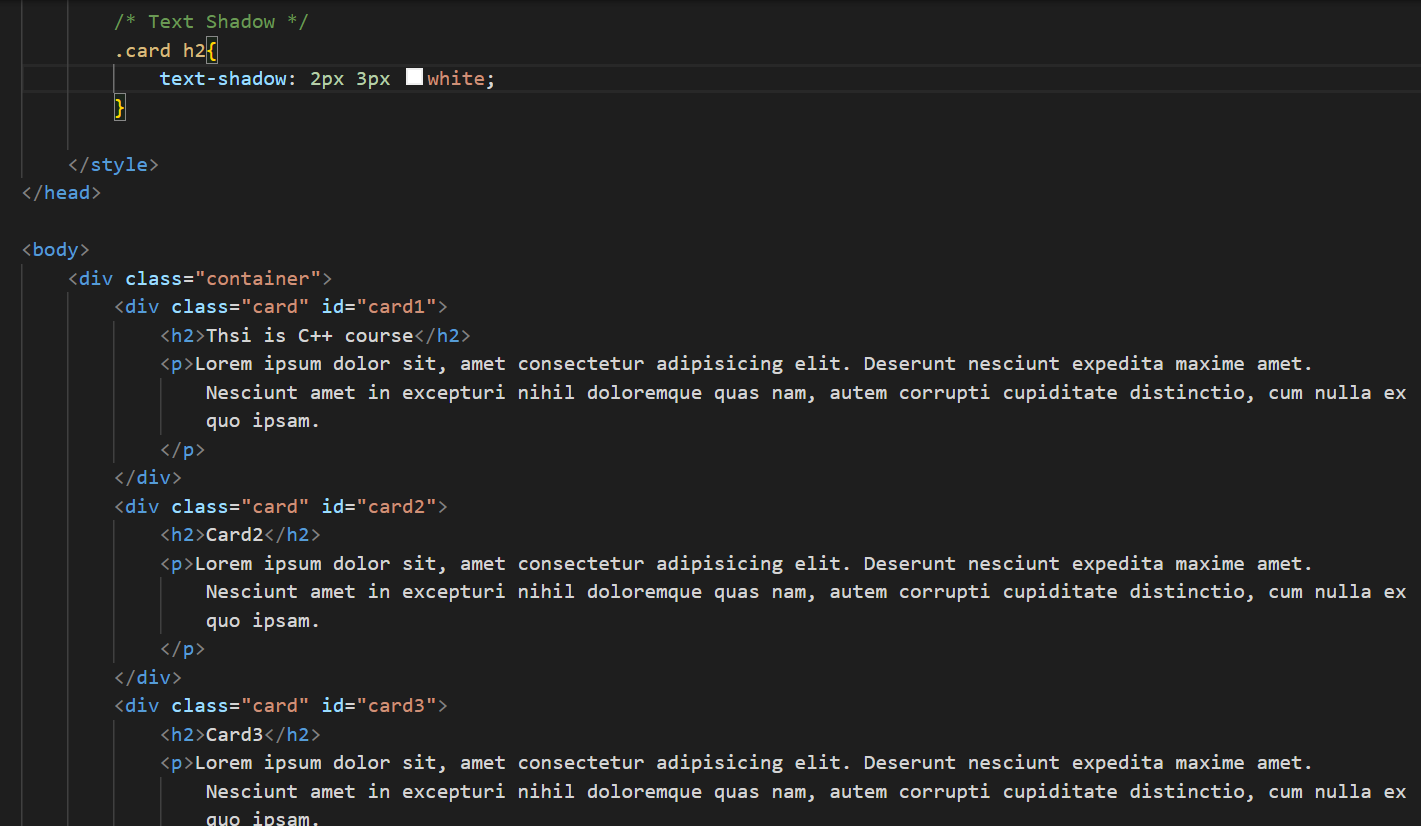


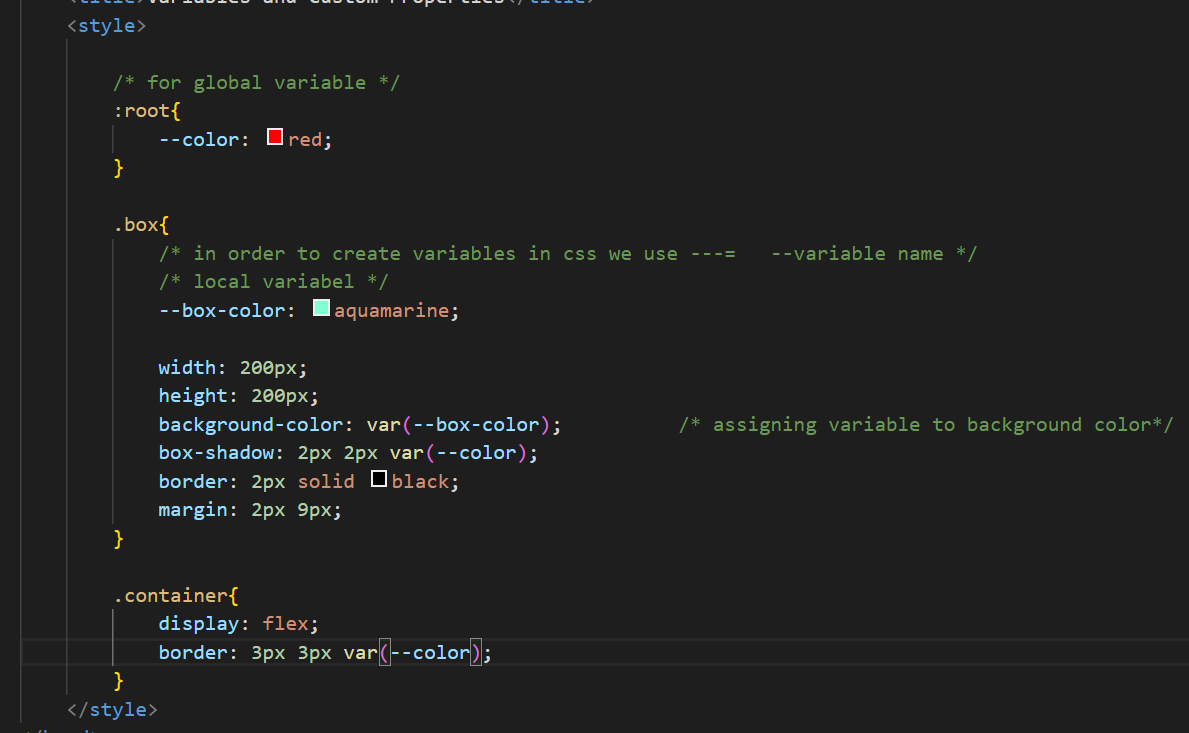




**Box and Text Shadow**









<!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>Animations ans Key Frames</title>

    <style>

        .container{

            background-color: blanchedalmond;

        }

        .box{

            background-color: aqua;

            width: 250px;

            height: 250px;

            text-align: center;

            /\* to make it animated \*/

            position: relative;

            animation-name: sanskruti1;             /\* set any name \*/

            animation-duration: 2s;

            /\* to run it forever --- use ----- infinite \*/

            animation-iteration-count: 3;         /\* no of times we want animation to run in single relaod of web page \*/

            animation-delay: 2s;

            /\* if i want animation result to sustain after completion \*/

            animation-fill-mode: forwards;

            /\* animation slow to fast (ease-in) \*/

            animation-timing-function: ease-in;

            /\* to give the directions of the animationm \*/

            animation-direction: alternate;

        }

        /\* now for creating animation add keyframe\*/

        /\* @keyframes sanskruti1{

            from{

                width: 200px;

            }

            to{

                width: 1000px;

            }

        } \*/

        /\* other way \*/

        @keyframes sanskruti1{

            0%{

                top: 0px;

                left: 0px;

                background-color: blue;

                width: 100px;

            }

            25%{

                top: 250px;

                left: 0px;

                background-color: blueviolet;

                width: 300px;

            }

            75%{

                top: 250px;

                left: 250px;

                background-color: aquamarine;

                width: 800px;

            }

            100%{

                top: 0px;

                left: 250px;

                background-color: red;

                width: 1200px;

            }

        }

    </style>

</head>

<body>

    <div class="container">

        <div class="box">

            This is box

        </div>

    </div>

</body>

</html>

Transition

<!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>Transitions in CSS</title>

    <style>

        .box{

            display: flex;

            align-items: center;

            justify-content: center;

            height: 200px;

            width: 200px;

            background-color: red;

            /\* transition property \*/

            transition-property: background-color;

            transition-duration: 2s;

            /\* transition-timing-function: ease-in; \*/

            transition-delay: 2s;

            /\* lets try shorthand on transition for all properties :  \*/

            transition: all 1s ease-in-out 2s;

        }

        .box:hover{

            background-color: purple;

            height: 400px;

            width: 400px;

            border-radius: 45px;

            font-size: 50px;

        }

    </style>

</head>

<body>

    <h3>This is CSS Transition tutorials </h3>

    <div class="container">

        <div class="box">

            This is box

        </div>

    </div>

</body>

</html>

Transform in CSS

<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>Transform in CSS</title>

    <style>

        \*{

            margin: 0px;

            padding: 0px;

        }

        .container{

            background-color: bisque;

            display: flex;

            justify-content: center;

            align-items: center;

            height: 80vh;

        }

        .box{

            display: flex;

            justify-content: center;

            align-items: center;

            background-color: brown;

            border: 2px solid black;

            border-radius: 8px;

            height: 400px;

            width: 400px;

            /\* transform proprty whenever someone hovers on box \*/

            /\* transform: rotate(45deg); \*/

            /\* transform + transition \*/

            transition: all 1s ease-in-out;

        }

        .box:hover{

            /\* transform: rotate(360deg); \*/

            /\* transform: skew(40deg); \*/

            /\* transform: scale(2); \*/

            /\* transform: translate(123px 123px);                         to move in both x and y direction \*/

            transform: translateX(200px);                     /\*\* it will move 200px in x direction \*\*/

        }

    </style>

</head>

CSS Grid

<!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>CSS Grid Tutorial</title>

    <style>

        .container{

            display: grid;

            /\* so container becomes grid and all the items inside it will be -- grid items \*/

            /\* grid-template-columns: 300px 100px 100px;  there will be three columns made of width(300, 100, 100) \*/

            /\* grid-template-columns: 300px auto 100px; \*/

            /\* below : giving width as 1th to first part 4th part of web page to 2nd part etc \*/

            grid-template-columns: 1fr 4fr 1fr;

            /\* for large number of columns --- use repeat(no of columns , width) method\*/

            /\* grid-template-columns: repeat(3, auto); \*/

            /\* grid gap for gap or padding between grid items \*/

            grid-gap: 2rem;

        }

        .item{

            background-color: aquamarine;

            border: 3px solid black;

            padding: 5px;

        }

    </style>

</head>

<body>

    <div class="container">

        <div class="item">This is item-1</div>

        <div class="item">This is item-2</div>

        <div class="item">This is item-3</div>

        <div class="item">This is item-4</div>

        <div class="item">This is item-5</div>

        <div class="item">This is item-6</div>

        <div class="item">This is item-7</div>

        <div class="item">This is item-8</div>

        <div class="item">This is item-9</div>

    </div>

</body>

</html>



<!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>Grid Row and gaps</title>

    <style>

        .grid{

            display: grid;

            /\* Rows \*/

            /\*                box1 box2 box3    and rest box will be 1fr by default \*/

            grid-template-rows: 1fr 3fr 1fr;

            /\* to set height of other boxes also  --- so all rest boxes except first 3 will now be 2fr\*/

            grid-auto-rows: 2fr;

            /\* Columns \*/

            grid-template-columns: 1fr 4fr;

            grid-gap: 1rem;

        }

        .box{

            background-color: aquamarine;

            border: 2px solid black;

        }

    </style>

</head>

<body>

    <div class="grid">

        <div class="box">This is box-1</div>

        <div class="box">This is box-2</div>

        <div class="box">This is box-3</div>

        <div class="box">This is box-4</div>

        <div class="box">This is box-5</div>

        <div class="box">This is box-6</div>

        <div class="box">This is box-7</div>

        <div class="box">This is box-8</div>

        <div class="box">This is box-9</div>

        <div class="box">This is box-10</div>

    </div>

</body>

</html>



<!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>Spanning Rows and Columns</title>

    <style>

        .container{

            display: grid;

            grid-template-columns: repeat(5, 1fr);

            grid-template-rows: repeat(4, 1fr);

            grid-gap: 2rem;

            /\* types of grid gap \*/

            /\* grid-column-gap: 7rem;

            grid-row-gap: 2rem; \*/

        }

        .box{

            border: 2px solid black;

            background-color: aquamarine;

            padding: 23px;

        }

        /\*  to merge box 1 2 6 7--- into one single box \*/

        /\* to target specific child of the box class --- child pseudo selector \*/

        .box:first-child{

            /\* merging item 1 and 2 --- start 1 and end at 3 \*/

            grid-column-start: 1;

            grid-column-end: 3;

            /\* merging rows 1st and 2nd row \*/

            grid-row-start: 1;

            grid-row-end: 3;

            /\* short hand approach for above spanning \*/

            /\* spanning item 1 2 times larger \*/

            /\* grid-column: 1 / span 2;

            grid-row: 1 / span 2; \*/

        }

    </style>

</head>

<body>

    <div class="container">

        <div class="box">Item-1</div>

        <div class="box">Item-2</div>

        <div class="box">Item-3</div>

        <div class="box">Item-4</div>

        <div class="box">Item-5</div>

        <div class="box">Item-6</div>

        <div class="box">Item-7</div>

        <div class="box">Item-8</div>

        <div class="box">Item-9</div>

        <div class="box">Item-10</div>

        <div class="box">Item-11</div>

        <div class="box">Item-12</div>

        <div class="box">Item-13</div>

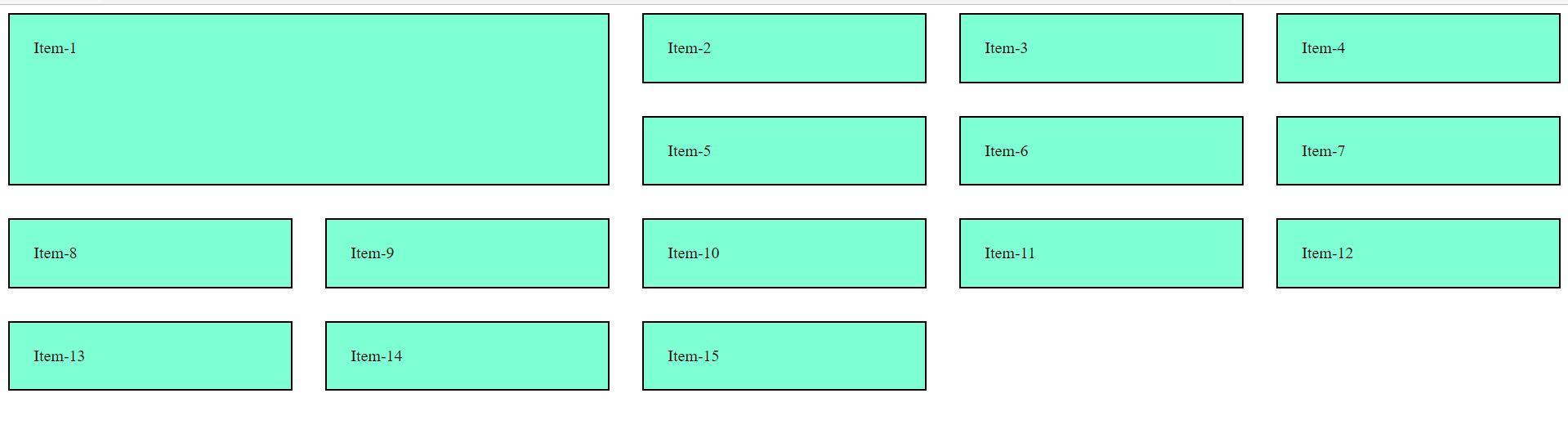
        <div class="box">Item-14</div>

        <div class="box">Item-15</div>

    </div>

</body>

</html>



Autofit

<style>

        .container{

            display: grid;

            grid-gap: 2rem;

            /\* grid-template-columns: 112px 112px 112px; \*/

            /\* grid-template-columns: 1fr 1fr 1fr; \*/

            /\* for making responsive websites -- as the size of the device decrease or phone it will divide as per that \*/

            /\*                                             min  , max  \*/

            grid-template-columns: repeat(auto-fit, minmax(400px, 1fr));

            /\* we can align grid items in the center ---  \*/

            /\* justify-content: center;

            align-items: center; \*/

        }

        .box{

            border: 2px solid black;

            background-color: aquamarine;

            padding: 34px;

        }

    </style>



<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>Creating Layout Using grid Template area</title>

    <style>

        .container {

            display: grid;

            grid-gap: 1rem;

            /\* Grid Template Areas - layout will be having 3 rows and these four columns of the type specified\*/

            grid-template-areas:

                'navbar navbar navbar navbar'     /\* 1st row will be taken by the navbar \*/

                'section section section aside'  /\* 2nd row 3th space - by section and 1th space - by aside tag\*/

                'footer footer footer footer';   /\* 3rd row for the footer\*/

        }

        .item {

            background-color: aquamarine;

            border: 2px solid black;

        }

        /\* Grid area  \*/

        #navbar {

            grid-area: navbar;

        }

        #section {

            grid-area: section;

        }

        #aside {

            grid-area: aside;

        }

        footer{

            grid-area: footer;

        }

    </style>

</head>

<body>

    <div class="container">

        <div id="navbar" class="item">

            Home About Contact-Us

        </div>

        <div id="section" class="item">

            Lorem ipsum dolor sit amet consectetur adipisicing elit. Quaerat, laboriosam cupiditate debitis ut fuga

            recusandae nam delectus molestiae ea mollitia. Fuga, odit, quam eveniet perferendis fugit atque itaque,

            minus magnam molestiae repellendus vero. Adipisci, consectetur harum corporis esse quae asperiores!.

        </div>

        <div id="aside" class="item">

            Lorem ipsum dolor sit amet consectetur adipisicing elit. Repellat officiis neque consequatur!

            Lorem ipsum dolor sit, amet consectetur adipisicing elit. Illum, soluta voluptatem impedit sint quo

            molestiae facilis neque eligendi aspernatur. Laudantium illum placeat expedita itaque illo, cupiditate ut

            delectus culpa nam.

        </div>

        <!-- footer -->

        <footer class="item">Lorem ipsum dolor sit, amet consectetur adipisicing elit. Qui, ut.</footer>

    </div>

</body>

</html>

