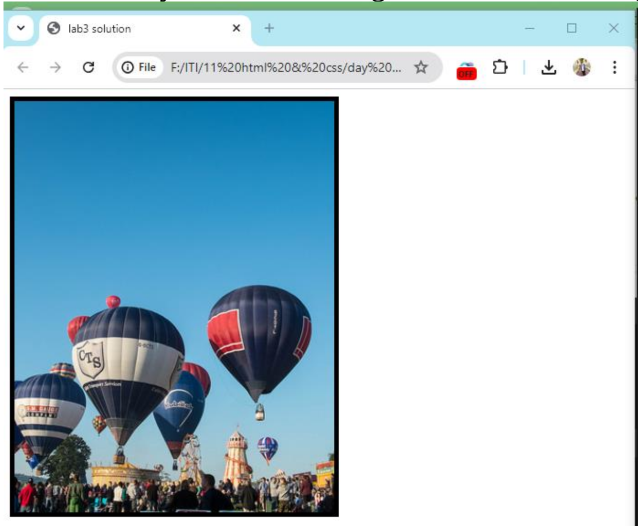
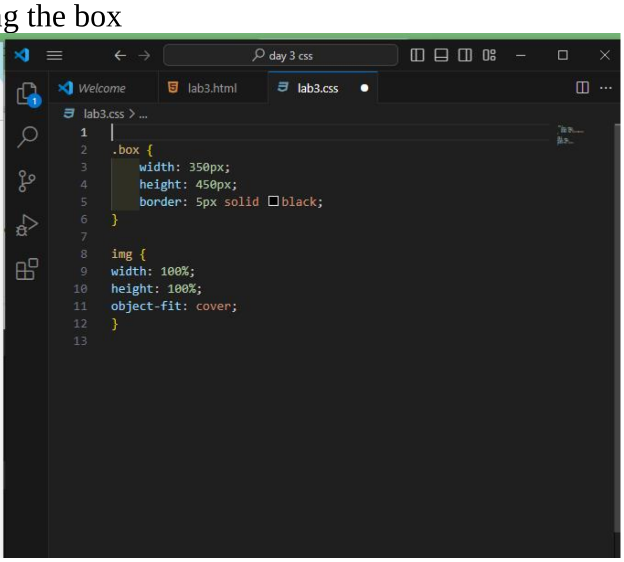
CSS Exercises lab 3

Exercise #1

In this task, you have an image that is overflowing the box

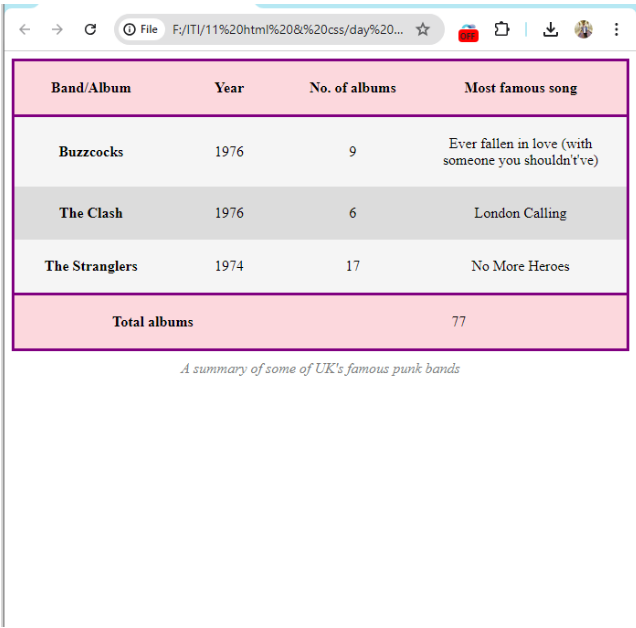




```css

.box {
    width: 350px;
    height: 450px;
    border: 5px solid □black;
}

img {
width: 100%;
height: 100%;
object-fit: cover;
}

```

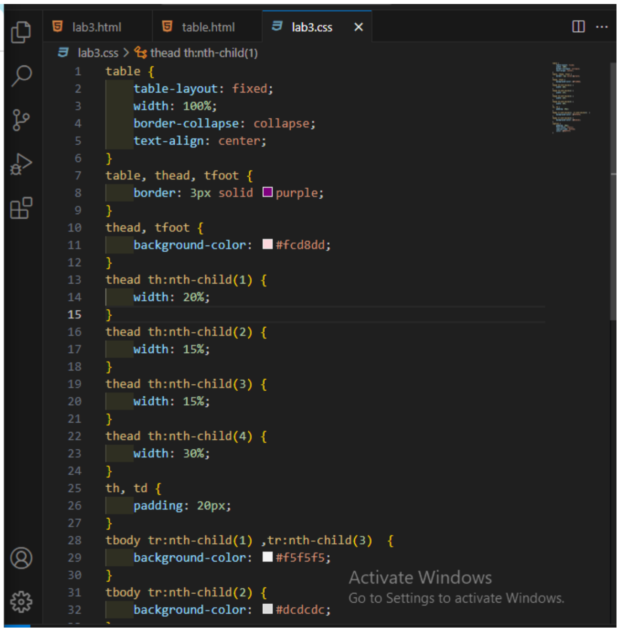
Exercise #2

#2 Create a similar looking table and style it to be as the preview image.

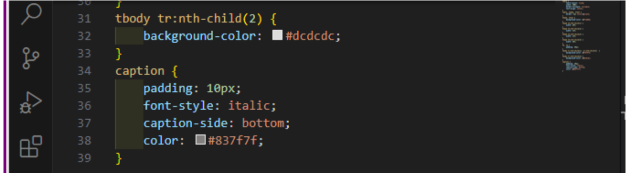


Band/Album	Year	No. of albums	Most famous song
Buzzcocks	1976	9	Ever fallen in love (with someone you shouldn't've)
The Clash	1976	6	London Calling
The Stranglers	1974	17	No More Heroes
Total albums		77	

A summary of some of UK's famous punk bands



```css
table {
    table-layout: fixed;
    width: 100%;
    border-collapse: collapse;
    text-align: center;
}
table, thead, tfoot {
    border: 3px solid ■purple;
}
thead, tfoot {
    background-color: ■#fcd8dd;
}
thead th:nth-child(1) {
    width: 20%;
}
thead th:nth-child(2) {
    width: 15%;
}
thead th:nth-child(3) {
    width: 15%;
}
thead th:nth-child(4) {
    width: 30%;
}
th, td {
    padding: 20px;
}
tbody tr:nth-child(1) ,tr:nth-child(3)  {
    background-color: ■#f5f5f5;
}
tbody tr:nth-child(2) {
    background-color: ■#dcdcdc;
```



```css
}
tbody tr:nth-child(2) {
    background-color: ■#dcdcdc;
}
caption {
    padding: 10px;
    font-style: italic;
    caption-side: bottom;
    color: ■#837f7f;
}
```

Exercise #3

Use the given HTML structure with the following instructions.

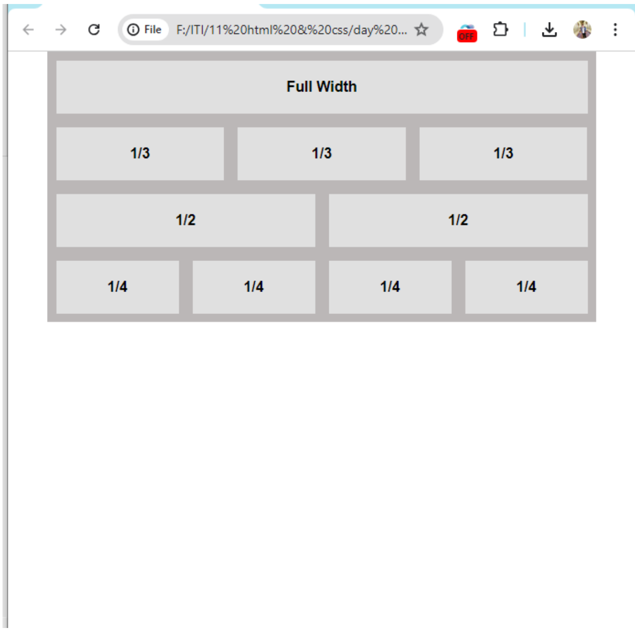
1) There's no padding in the container div element.

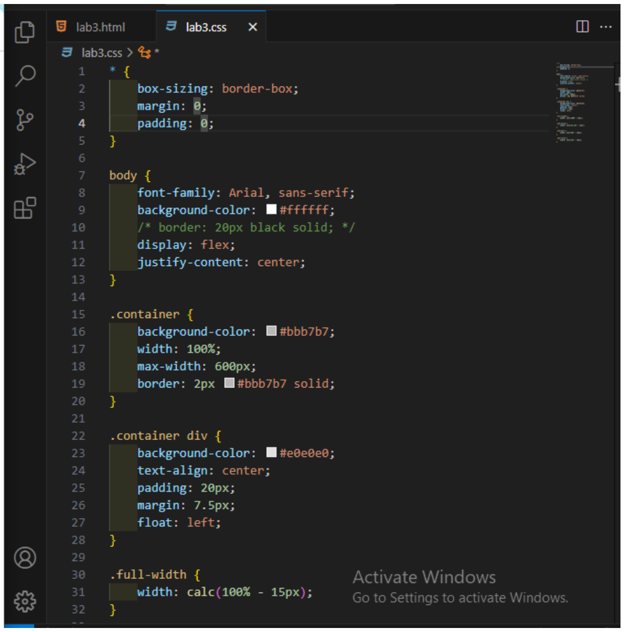
2) Use the float property to make the nested div elements float to the left, you should also clear/fix the layout in the container div CSS styling place somehow after doing so.

3) The horizontal and vertical gaps between the nested div elements is 15px each.

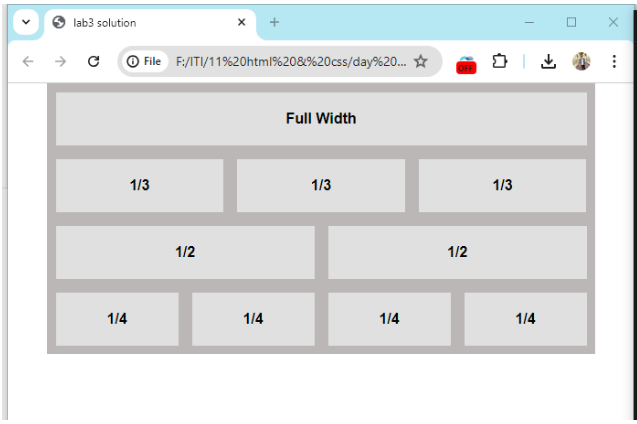
4) The width of the container div element should not be calculated manually and should be dynamic (use 100% unit).

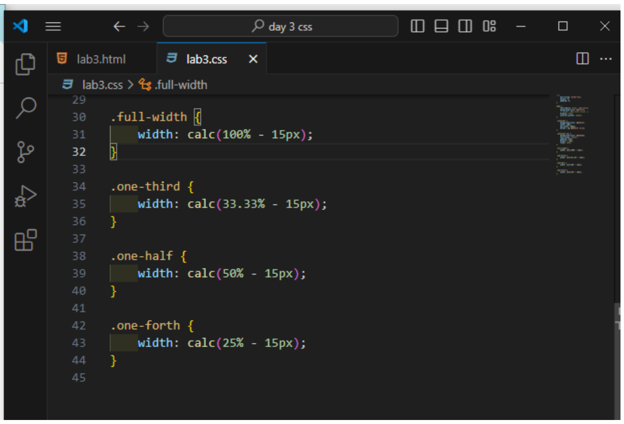
5) Ensure that the layout is not broken when you resize the page.





```css
* {
    box-sizing: border-box;
    margin: 0;
    padding: 0;
}

body {
    font-family: Arial, sans-serif;
    background-color: #ffffff;
    /* border: 20px black solid; */
    display: flex;
    justify-content: center;
}

.container {
    background-color: #bbb7b7;
    width: 100%;
    max-width: 600px;
    border: 2px #bbb7b7 solid;
}

.container div {
    background-color: #e0e0e0;
    text-align: center;
    padding: 20px;
    margin: 7.5px;
    float: left;
}

.full-width {
    width: calc(100% - 15px);
}
```





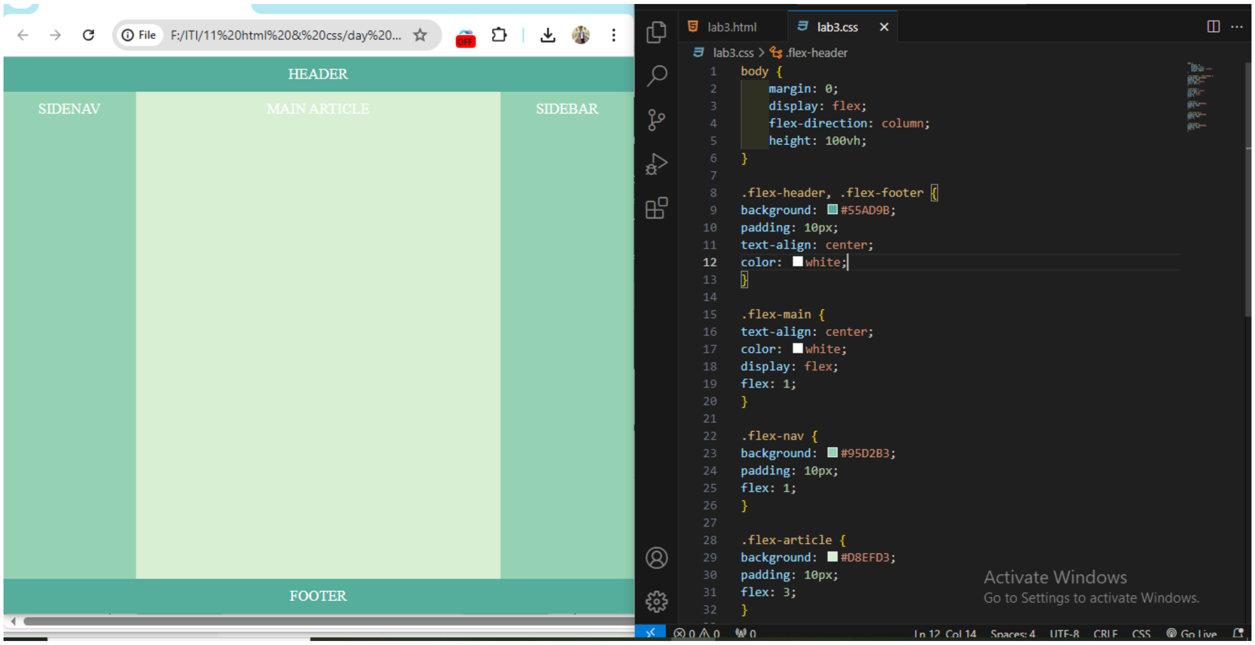
```css
.full-width {
    width: calc(100% - 15px);
}

.one-third {
    width: calc(33.33% - 15px);
}

.one-half {
    width: calc(50% - 15px);
}

.one-forth {
    width: calc(25% - 15px);
}
```

Exercise #4

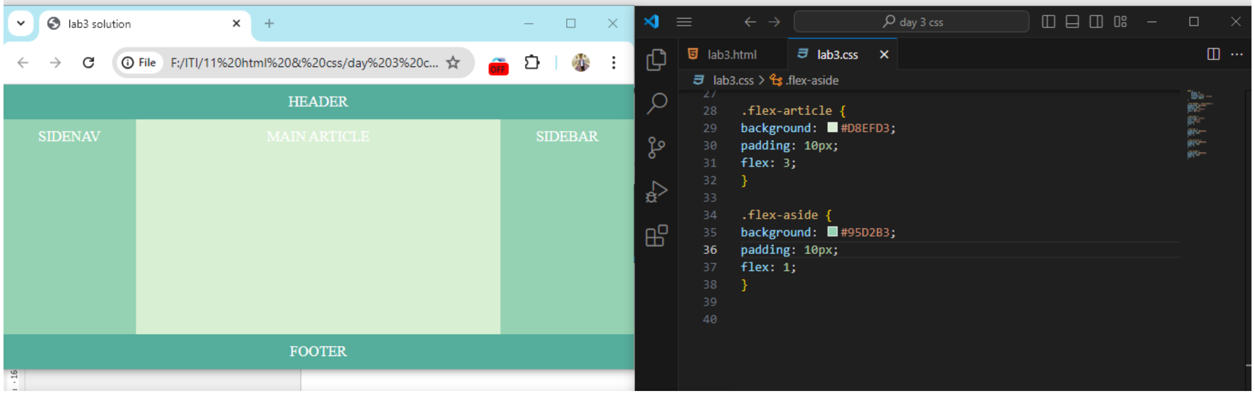
Use the given HTML structure with the following instructions.

1) Using flex-box properties, create a holy-grail design similar to the preview image below.

2) Use media queries to adjust the direction of the middle part to be column when the screen size reaches 500px or less.


F:/ITI/11%20html%20&%20css/day%20...
HEADER
SIDENAV
MAIN ARTICLE
SIDEBAR
FOOTER
lab3.html
lab3.css
lab3.css > .flex-header
Activate Windows
Go to Settings to activate Windows.
Ln 12, Col 14 Spaces: 4 UTF-8 CRLF CSS Go Live

```css
body {
        margin: 0;
        display: flex;
        flex-direction: column;
        height: 100vh;
}

.flex-header, .flex-footer {
background: #55AD9B;
padding: 10px;
text-align: center;
color: white;
}

.flex-main {
text-align: center;
color: white;
display: flex;
flex: 1;
}

.flex-nav {
background: #95D2B3;
padding: 10px;
flex: 1;
}

.flex-article {
background: #D8EFD3;
padding: 10px;
flex: 3;
}
```



```css
.flex-article {
background: #D8EFD3;
padding: 10px;
flex: 3;
}

.flex-aside {
background: #95D2B3;
padding: 10px;
flex: 1;
}
```

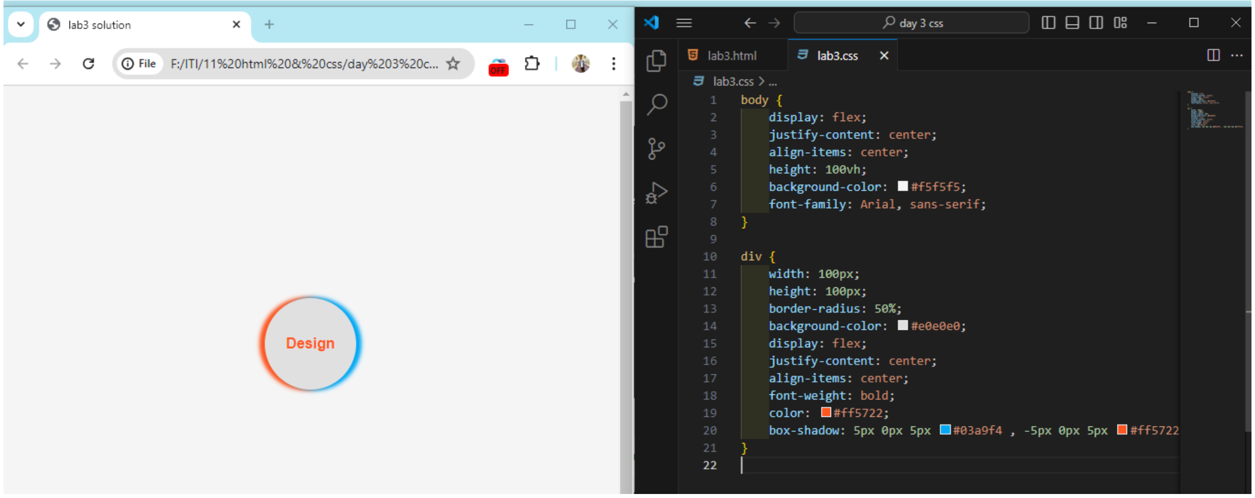
Exercise #5

Use the given HTML structure with the following instructions.

1) The width and height of the div element is 100px.

2) Center the div element horizontally and vertically on the page using flex properties.

3) Center the contents (i.e., the word "Design") both horizontally and vertically using flex



```css
body {
        display: flex;
        justify-content: center;
        align-items: center;
        height: 100vh;
        background-color: #f5f5f5;
        font-family: Arial, sans-serif;
}

div {
        width: 100px;
        height: 100px;
        border-radius: 50%;
        background-color: #e0e0e0;
        display: flex;
        justify-content: center;
        align-items: center;
        font-weight: bold;
        color: #ff5722;
        box-shadow: 5px 0px 5px #03a9f4 , -5px 0px 5px #ff5722
}
```

Exercise #6

Use the given HTML structure with the following instructions.

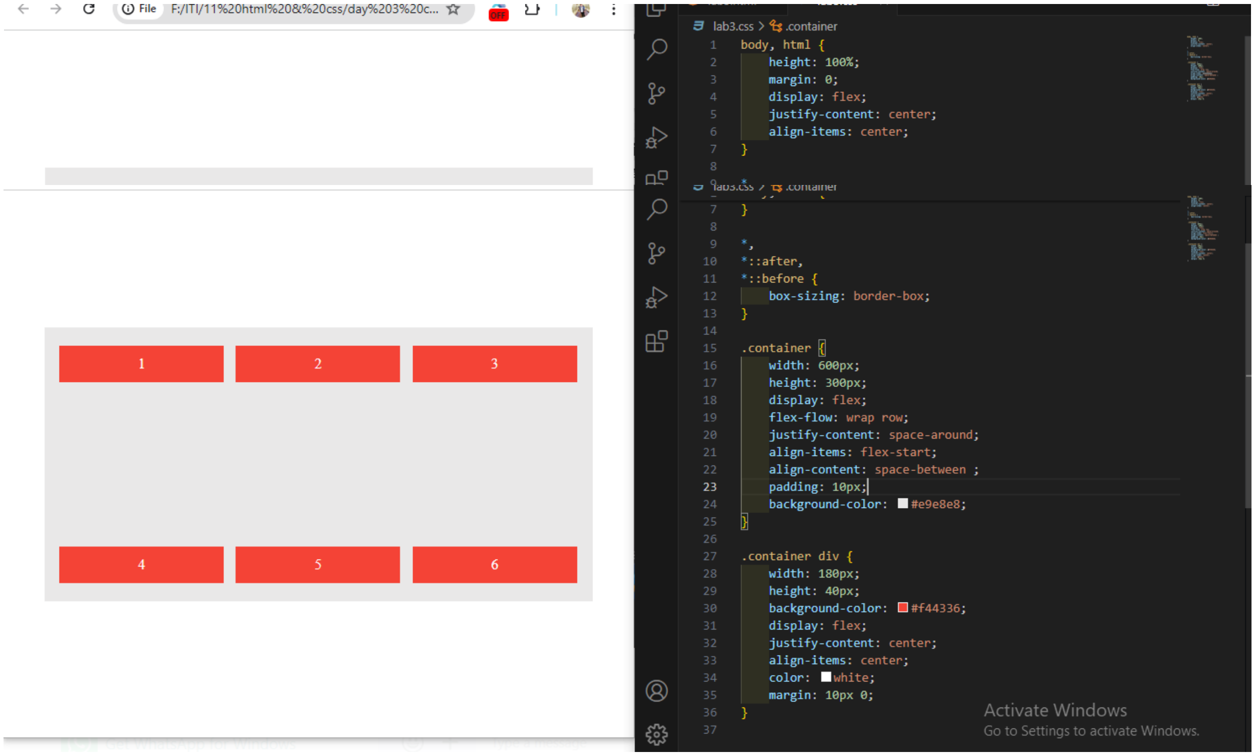
1) The width of the container div element is 600px and the height is 300px.

2) Center the container div element horizontally and vertically in the page using flex properties.

3) The minimum width and minimum height of each nested div element inside the container div is 180px and 40px, respectively.

4) Use flex properties to arrange the nested div elements like the attached image.

5) The gap between the nested div elements should be 10px.

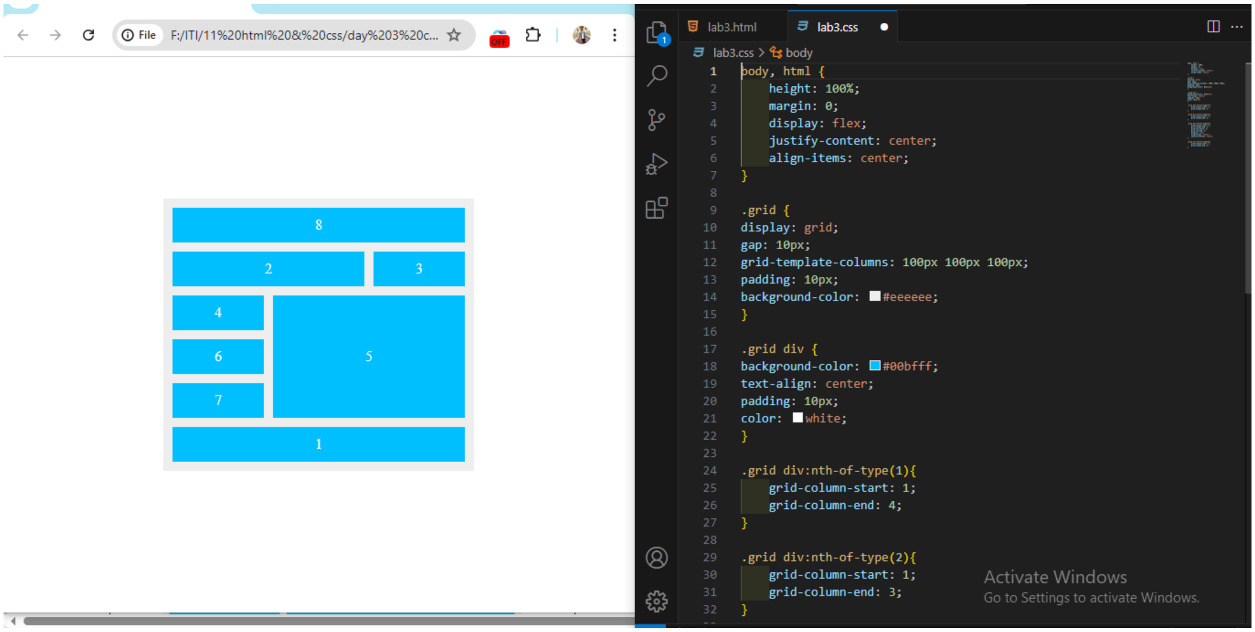

File
F:/ITI/11%20html%20&%20css/day%203%20c...
OFF
1
2
3
4
5
6
lab3.css > .container
Activate Windows
Go to Settings to activate Windows.

```css
body, html {
    height: 100%;
    margin: 0;
    display: flex;
    justify-content: center;
    align-items: center;
}

```

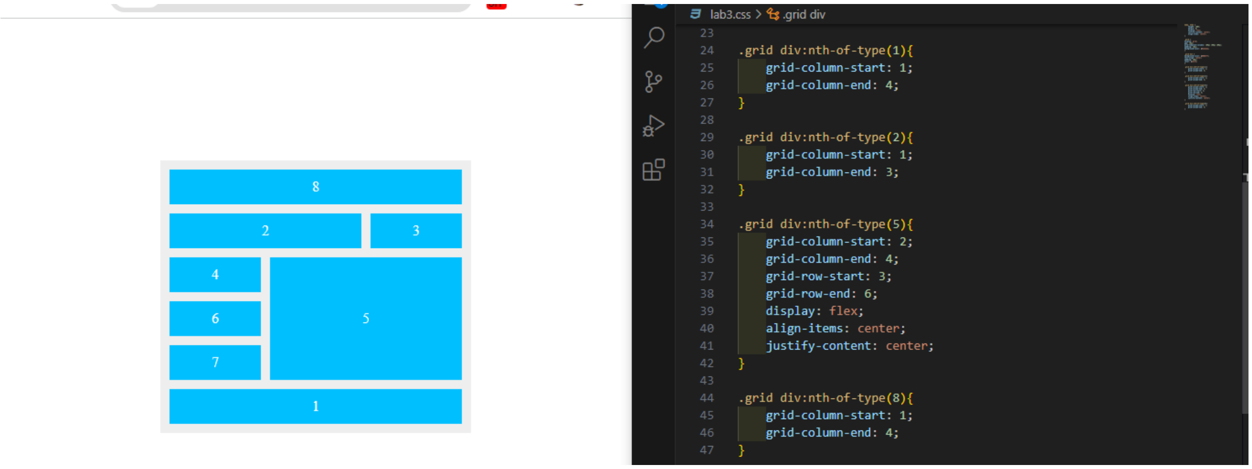
```css
}

*,
*::after,
*::before {
    box-sizing: border-box;
}

.container {
    width: 600px;
    height: 300px;
    display: flex;
    flex-flow: wrap row;
    justify-content: space-around;
    align-items: flex-start;
    align-content: space-between ;
    padding: 10px;
    background-color: #e9e8e8;
}

.container div {
    width: 180px;
    height: 40px;
    background-color: #f44336;
    display: flex;
    justify-content: center;
    align-items: center;
    color: white;
    margin: 10px 0;
}

```

Exercise #7

#7 Use the given HTML structure with the following instructions.
1) Center the div element with the class of ‘grid’ horizontally and vertically on the page using grid properties.
2) Use the grid properties to arrange the nested div elements to create a design close to the attached preview image.


File
F:/ITI/11%20html%20&%20css/day%203%20c...
8
2
3
4
5
6
7
1
lab3.html
lab3.css
lab3.css > body
Activate Windows
Go to Settings to activate Windows.

```css
body, html {
    height: 100%;
    margin: 0;
    display: flex;
    justify-content: center;
    align-items: center;
}

.grid {
display: grid;
gap: 10px;
grid-template-columns: 100px 100px 100px;
padding: 10px;
background-color: #eeeeee;
}

.grid div {
background-color: #00bfff;
text-align: center;
padding: 10px;
color: white;
}

.grid div:nth-of-type(1){
    grid-column-start: 1;
    grid-column-end: 4;
}

.grid div:nth-of-type(2){
    grid-column-start: 1;
    grid-column-end: 3;
}
```



```css

.grid div:nth-of-type(1){
    grid-column-start: 1;
    grid-column-end: 4;
}

.grid div:nth-of-type(2){
    grid-column-start: 1;
    grid-column-end: 3;
}

.grid div:nth-of-type(5){
    grid-column-start: 2;
    grid-column-end: 4;
    grid-row-start: 3;
    grid-row-end: 6;
    display: flex;
    align-items: center;
    justify-content: center;
}

.grid div:nth-of-type(8){
    grid-column-start: 1;
    grid-column-end: 4;
}
```

Exercise #8

#8 Use the given HTML structure with the following instructions.

1) Using both grid and flex properties, create a holy-grail design similar to the preview image below.

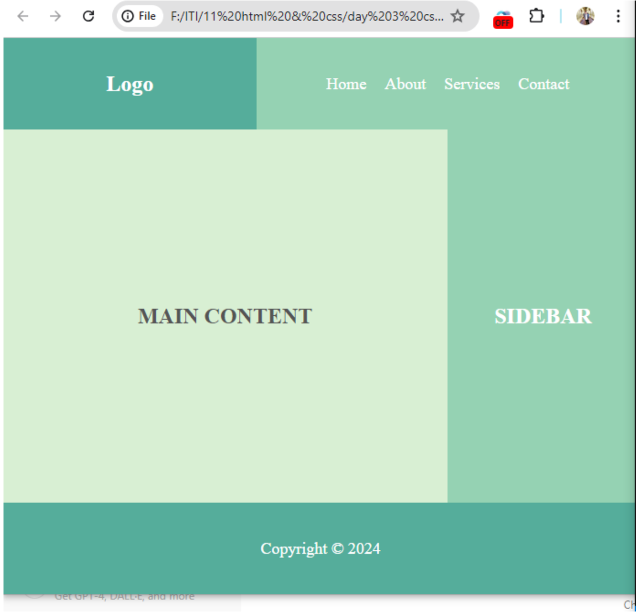
2) Define the grid canvas to have 3 grid rows with values of 50px, automatic, and 50px, respectively.

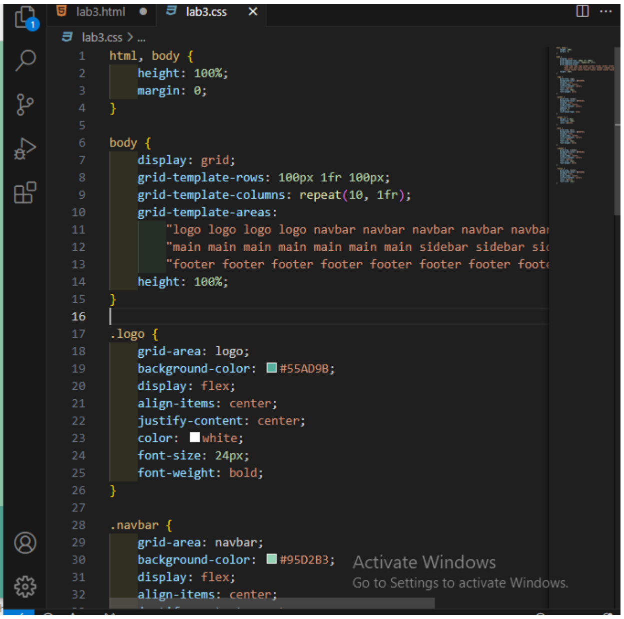
3) The grid canvas should also have 10 grid columns each filling the whole width greedily.

4) You should use grid areas property to define the 10 columns.

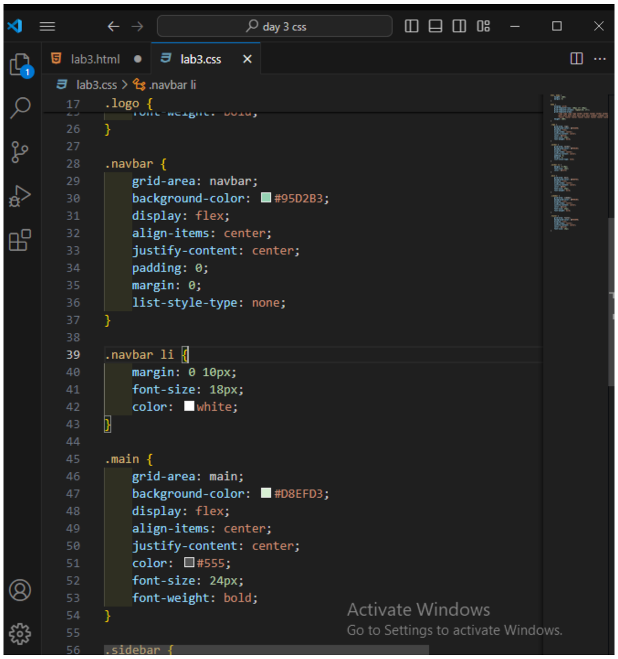
5) You may use flexbox for minor tasks (e.g., centering text or items).

6) Use CSS variables to define the three used colors.

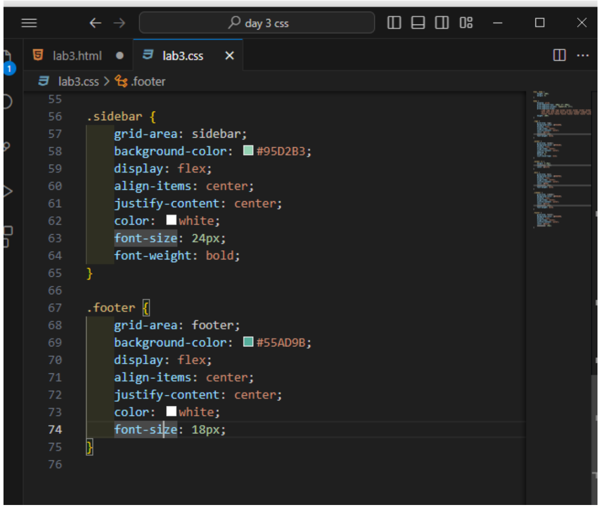




```css
html, body {
    height: 100%;
    margin: 0;
}

body {
    display: grid;
    grid-template-rows: 100px 1fr 100px;
    grid-template-columns: repeat(10, 1fr);
    grid-template-areas:
        "logo logo logo logo navbar navbar navbar navbar navbar
        "main main main main main main main sidebar sidebar sic
        "footer footer footer footer footer footer footer foote
    height: 100%;
}

.logo {
    grid-area: logo;
    background-color: #55AD9B;
    display: flex;
    align-items: center;
    justify-content: center;
    color: white;
    font-size: 24px;
    font-weight: bold;
}

.navbar {
    grid-area: navbar;
    background-color: #95D2B3;
    display: flex;
    align-items: center;
```



```css
.logo {
}

.navbar {
    grid-area: navbar;
    background-color: #95D2B3;
    display: flex;
    align-items: center;
    justify-content: center;
    padding: 0;
    margin: 0;
    list-style-type: none;
}

.navbar li {
    margin: 0 10px;
    font-size: 18px;
    color: white;
}

.main {
    grid-area: main;
    background-color: #D8EFD3;
    display: flex;
    align-items: center;
    justify-content: center;
    color: #555;
    font-size: 24px;
    font-weight: bold;
}

.sidebar {
```



```css

.sidebar {
    grid-area: sidebar;
    background-color: #95D2B3;
    display: flex;
    align-items: center;
    justify-content: center;
    color: white;
    font-size: 24px;
    font-weight: bold;
}

.footer {
    grid-area: footer;
    background-color: #55AD9B;
    display: flex;
    align-items: center;
    justify-content: center;
    color: white;
    font-size: 18px;
}

```