**CSS Grid**

**Lab report #02**

****

Fall 2023

CSE-423L Web Engineering

Submitted by: **Ashfaq Ahmad**

Registration No: **19PWCSE1795**

Class Section: **B**

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work.”

Student Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Submitted to:

**Eng. Usman Malik**

**March** 26, 2023

**Department of Computer Systems Engineering**

**University of Engineering and Technology, Peshawar**

**Task 01:**

For this task, you'll be working with `Grid.html` and `Grid.css`. Create a page layout as

shown below:

**HTML 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>Task01</title>

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

</head>

<body>

    <div class="container">

        <div class="item0">Header</div>

        <div class="item1">Side bar</div>

        <div class="item2">Main</div>

        <div class="item3">item1</div>

        <div class="item4">item2</div>

        <div class="item5">Footer</div>

    </div>

</body>

</html>

**CSS Code:**

  \*{

    margin: 0;

    padding: 0;

  }

  .container{

    display: grid;

    gap: 8px;

    grid-template-areas: 'header header header header header header'

    'sidebar sidebar main main main main'

    'sidebar sidebar item1 item1 item2 item2'

    'footer footer footer footer footer footer';

    padding: 10px; /\* this padding is container only.\*/

    background-color: black ;

}

.container >div{

    /\* make each div inside container class as grid and apply grid properties to bring each grid elements to the center. in this case each grid has single element as "simple text" \*/

    display: grid;

    align-items: center;

    justify-content: center;

    color: white;

    font-weight: bold;

}

.item0{

    background-color: rgb(73, 9, 126);

    grid-area: header;

    height: 80px;

}

.item1{

    background-color: rgb(132, 20, 238);

    height: 400px;

    width: 300px;

    grid-area: sidebar;

}

.item2{

    background-color:rgb(10, 127, 243);

    height: 270px;

    width:1035px;

    grid-area: main;

}

.item3{

    background-color: rgb(163, 107, 216);

    height: 117px;

    grid-area: item1;

}

.item4{

    background-color: rgb(127, 111, 141);

    height: 117px;

    grid-area: item2;

}

.item5{

    background-color: rgb(44, 1, 85);

    height: 140px;

    grid-area: footer;

}

**Result:**

****

**Task 02:**

For this task, you'll be working with `Labs\_Page.html` and `Labs\_Page.css`. Design a page

for your labs as shown below:

**HTML 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>Labs page</title>

    <link rel="stylesheet" href="labs\_page.css">

</head>

<body>

    <ul>

        <li><a href="#">Home</a></li>

        <li><a href="#">About Me</a></li>

        <li><a href="#">Labs</a></li>

      </ul>

    <h1>Labs</h1>

    <div class="container">

        <div class="item1"></div>

        <div class="item2">

            <h3>Lab 1</h3>

            <h4>My First Webpage</h4>

            <br>

            <p>In this lab assignment, I created a valid HTML5 webpage where I introduced myself to my instructor and my

                classmates.</p>

        </div>

        <div class="item1"></div>

        <div class="item2">

            <h3>Lab 2</h3>

            <h4>Side Architecture- First Steps</h4>

            <br>

            <p>In this second lab assignment, I set up the file-folder structure of this website and created a new

                homepage using HTML5 sectioning elements and CSS layout using box model.</p>

        </div>

        <div class="item1"></div>

        <div class="item2">

            <h3>Lab 3</h3>

            <h4>Inline-block Gallery</h4>

            <br>

            <p>In this third lab assignment, I built a mock-up of a web gallery, using inline-block CSS display

                property. I also explored the use of dummy content generator.</p>

        </div>

        <div class="item1"></div>

        <div class="item2">

            <h3>Lab 4</h3>

            <h4>CSS Grid and Flexbox</h4>

            <br>

            <p>In this fourth lab assignment, I stayled the page you're currently viewing using both grid and flexbox

                layout strategies.</p>

        </div>

        <div class="item1"></div>

        <div class="item2">

            <h3>The Rest</h3>

            <h4>TED</h4>

            <br>

            <p>The rest of my labs have yet to be completed but will go here.</p>

        </div>

    </div>

    <br>

    <hr>

    <div class="footer">

        <h4>&copy;2023</h4>

        <a href="#"><b><u>Computer Systems Engineering</u></b></a>

        <a href="#"><b><u>Uet Peshawar</b></u></a>

    </div>

</body>

</html>

**CSS Code:**

/\* in case of grid-tamplete-area property justify-content property don't work \*/

\*{

    margin: 0px;

    padding: 0px;

}

ul{

    height: 80px;

    background-color:rgb(38, 36, 33);

    display: grid;

    grid-template-columns:13fr 1fr 0.5fr;

    align-items: center;

    list-style: none;

    padding: 0px 30px;

}

a{

    color: white;

    text-decoration: none;

}

h1{

    padding: 20px;

}

.container{

    display: grid;

    grid-template-columns:1fr 2fr 1fr 2fr 1fr 2fr;

    row-gap: 10px;

    padding: 0px 12px;

}

h3{

    color:rgb(39, 132, 239);

}

.item1{

    height: 300px;  /\* height must be same of all grids in each row\*/

    width: 170px;

    background-color: rgba(26, 24, 24, 0.735);

}

.item2{

    background-color:rgb(204, 202, 202);

    font-size: 17px;

    height: 300px;

    width: 245px;

    padding: 0px 10px;

}

.footer{

    height: 80px;

    display: grid;

    grid-template-columns: 1fr 1fr 0.3fr;

    align-items: center;  /\*veritcally alignment \*/

    background-color: rgb(246, 241, 241);

    padding: 0px 30px;

}

u{

    color: blue;

}

**Result:**

 