Chapter 7 CSS Grid & Media Queries

A css grid can be initialized using

.container{

Display:grid;

}

All direct children automatically becomes grid items.

The grid-column-gap property

Used to adjust the space between the columns of a css grid.

The grid-row-gap property

Used to adjust the space between the rows of a css grid.

The grid-gap property

Shorthand property for grid-row-gap and grid-column-gap.

.container{

display:grid;

grid-gap:40px 60px;

}

Following are the properties for grid container:

1. The grid-template-columns property can be used to specify the width of columns.

.container{

Display:grid;

Grid-template-columns:40px 60px auto;

}

2. The grid-template-rows property can be used to specify the height of each row.

 .container{

Display:grid;

Grid-template-rows:60px 123px;

}

3. The justify-content property is used to align the whole grid inside the container.

4. The align-content property is used to vertically align the whole grid inside the container.

Following are the properties for grid item:

1. The grid-column property define how many columns an items will span.

.grid-item {

Display:grid;

Grid-column:1/5;

}

2. The grid-row property define how many columns an items will span.

3. We can make an item to start on column 1 and span 3 columns like this.

 .item{

Grid-column:1/span 3;

}

CSS Media Queries

Used to apply css only when a certain condition is true syntax.

@media only screen and (max-width:800px){

Body{

Background-color: blue;

}

}

Source Code:

```
Index.html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Chapter 7 Grid in Css</title>
    <style>
        .container {
            display: grid;
            background-color: blueviolet;
            /*   grid-column-gap: 10px;
            grid-row-gap: 5px;
            */
            grid-gap: 10px;
            /*
            grid-template-columns: auto auto auto;
            grid-template-rows: 321px 223px;
            */
```

```
            grid-template-columns: auto auto auto;
            grid-template-rows: 320px 222px;
        }

        .item {
            background-color: aqua;
            border: 3px solid rgb(23, 147, 41);
        }
    </style>
</head>

<body>
    <div class="container">
        <div class="item">1</div>
        <div class="item">2</div>
        <div class="item">3</div>
        <div class="item">4</div>
        <div class="item">5</div>
        <div class="item">6</div>
        <div class="item">7</div>
        <div class="item">8</div>
        <div class="item">9</div>
    </div>
</body>

</html>

MediaQuery.html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Media Queries </title>
    <style>
        .container {
            background-color: green;
            height: 60vh;
        }

        @media screen and (max-width:800px) {
            .container {
                background-color: greenyellow;
                color: white;
```

```
                }
            }
        </style>
    </head>

    <body>
        <div class="container">
            Lorem ipsum dolor, sit amet consectetur adipisicing elit. Illo
repellendus tempora molestiae cupiditate ipsum maiores impedit fugit, doloribus
asperiores, nam deserunt voluptate, velit repudiandae quas accusamus vel ipsam.
Nisi accusantium consectetur,
            libero amet minus error rem dignissimos tempora quisquam incidunt
mollitia quod? Consectetur atque unde iure odio molestiae, voluptate quisquam.
Officia fuga facilis reiciendis quam.
        </div>
    </body>

</html>
```

Thank You