Module 2: Website Layouts

Video 2.1: Control Layout Using CSS Grid

Introduction

Now that we have an idea of how styling works, let’s look at how we can arrange content in the window. One of the ways is to divide the layout into grids. CSS has a property “display,” for which we can assign the value “grid.” This helps us divide the screen into grids, where we can then place our content.

Starter code (HTML):

```
<!DOCTYPE html>
<html lang="en">
<head>
        <meta charset="UTF-8">
        <title>CSS Grid Lesson</title>
        <link
href="https://fonts.googleapis.com/css2?family=Quicksand:wght@400;600&display=swap
" rel="stylesheet">
        <link href="styles.css" rel="stylesheet">
</head>
<body>
        <h1>CSS Grid Lesson</h1>
        <div class="content">
                <div class="box color1">
                        Alice
                </div>
                <div class="box color2">
                        Bob
                </div>
                <div class="box color1">
                        Charlie
                </div>
                <div class="box color2">
                        Daisy
                </div>
                <div class="box color1">
                        Eva
                </div>
                <div class="box color2">
                        Frank
                </div>
        <div class="box color1">
                        Grace
                </div>
                <div class="box color2">
                        Harry
                </div>
                <div class="box color1">
                        Ivy
                </div>
                <div class="box color2">
                        Jack
                </div>
                <div class="box color1">
                        Kathy
                </div>
                <div class="box color2">
                        Louis
                </div>
        </div>
</body>
</html>
```

CSS:

```
body {
      font-family: "Quicksand", sans-serif;
      background-color: #f0f0f0;
      margin: 0;
      padding: 0;
}
h1 {
      font-size: 36px;
      color: #333;
}

/* Content Wrapper */

.content {
      width: 100%;
}

/* Box Styles */
.box {
      width: 100%;
      padding: 30px;
      text-align: center;
      line-height: 100px;
      font-size: 18px;
}
.color1 {
      background-color: #5d0404;
      color: white;
}
.color2 {
      background-color: #e4f785;
}
```

Exercises

Task 1: In the css file, inside the .content class, add the following lines of code.

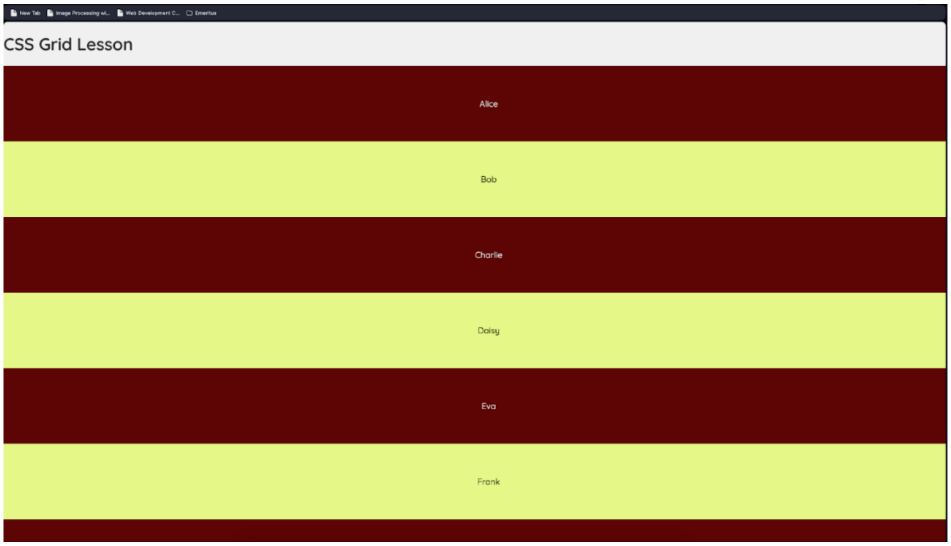
```
display: grid;
Grid-template-columns: 1fr 1fr 1fr;
```

Starter Code (CSS):

```
body {
        font-family: "Quicksand", sans-serif;
        background-color: #f0f0f0;
        margin: 0;
        padding: 0;
}
h1 {
        font-size: 36px;
        color: #333;
}
.content {
        width: 100%;
}

/* Box Styles */
.box {
        width: 100%;
        padding: 30px;
        text-align: center;
        line-height: 100px;
        font-size: 18px;
}
.color1 {
        background-color: #5d0404;
        color: white;
}
.color2 {
        background-color: #e4f785;
}
```

Initial Screenshot:



CSS Grid Lesson

Alice

Bob

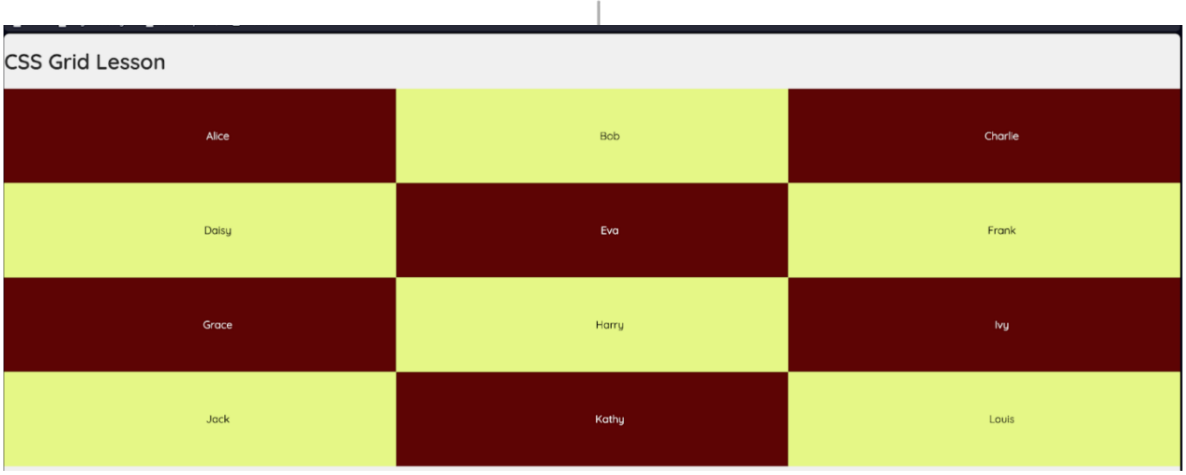
Charlie

Daisy

Eva

Frank

Final Screenshot:



CSS Grid Lesson

Alice	Bob	Charlie
Daisy	Eva	Frank
Grace	Harry	Ivy
Jack	Kathy	Louis

Let’s examine the code. The display is set to grid, and a grid has rows and columns.
By setting the property ‘grid template columns, we get to decide how many columns we want. Since there are 3 instances of ‘1fr’ next to it, we want three columns.

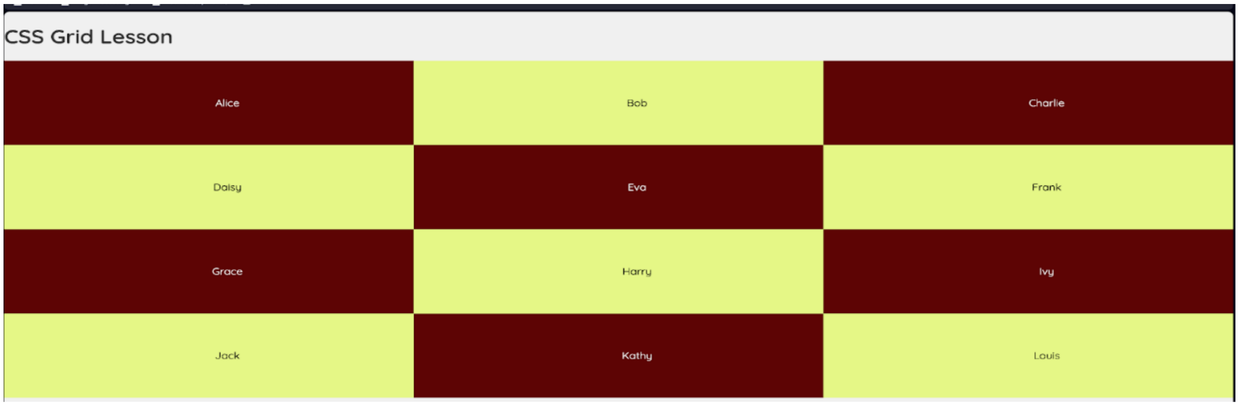
Also, fr refers to ‘fraction’. This means that there are three columns, each occupying one part of the fraction. That means, each column will take up one third of the width. There were 12 names to begin with. If they’re divided into three columns, then there are four such rows, right? Let’s add some gap to make it look neater.

Task 2: Add gap: 5px inside the .content class.

Starter Code (CSS):

```
body {
      width: 100%;
      font-family: "Quicksand", sans-serif;
      background-color: #f0f0f0;
      margin: 0;
      padding: 10px;
}
h1 {
      font-size: 36px;
      color: #333;
}
.content {
      display: grid;
      grid-template-columns: 1fr 1fr 1fr;
}
.box {
      width: 100%;
      padding: 30px;
      text-align: center;
      line-height: 100px;
      font-size: 18px;
}
.color1 {
      background-color: #5d0404;
      color: white;
}
.color2 {
      background-color: #e4f785;
}
```

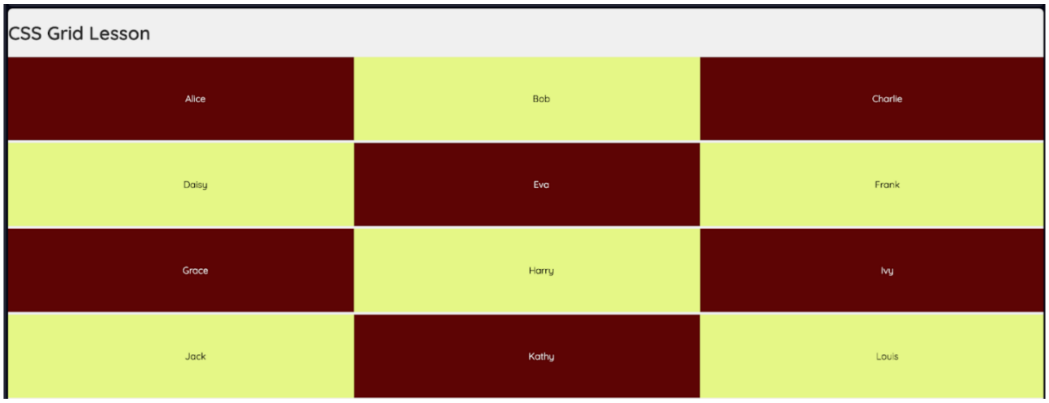
Initial Screenshot:



CSS Grid Lesson

Alice	Bob	Charlie
Daisy	Eva	Frank
Grace	Harry	Ivy
Jack	Kathy	Louis

Final Screenshot:



CSS Grid Lesson

Alice	Bob	Charlie
Daisy	Eva	Frank
Grace	Harry	Ivy
Jack	Kathy	Louis

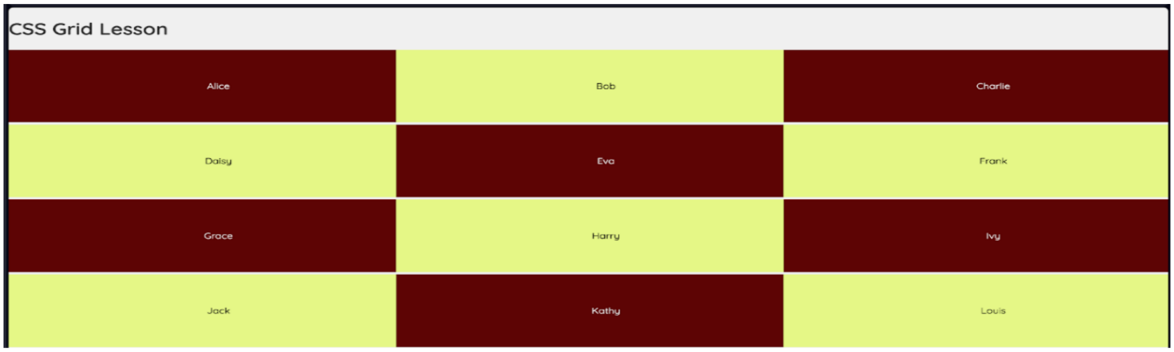
It looks neater now. What happens if we change the size of one of the columns? Let's see.

Task 3: Change the grid-template-columns property to 1fr 2fr 1fr

Starter Code (CSS):

```
body {
	width: 100%;
	font-family: "Quicksand", sans-serif;
	background-color: #f0f0f0;
	margin: 0;
	padding: 10px;
}
h1 {
	font-size: 36px;
	color: #333;
}
.content {
	display: grid;
	grid-template-columns: 1fr 1fr 1fr;
	gap: 5px;
}
.box {
	width: 100%;
	padding: 30px;
	text-align: center;
	line-height: 100px;
	font-size: 18px;
}
.color1 {
	background-color: #5d0404;
	color: white;
}
.color2 {
	background-color: #e4f785;
}
```

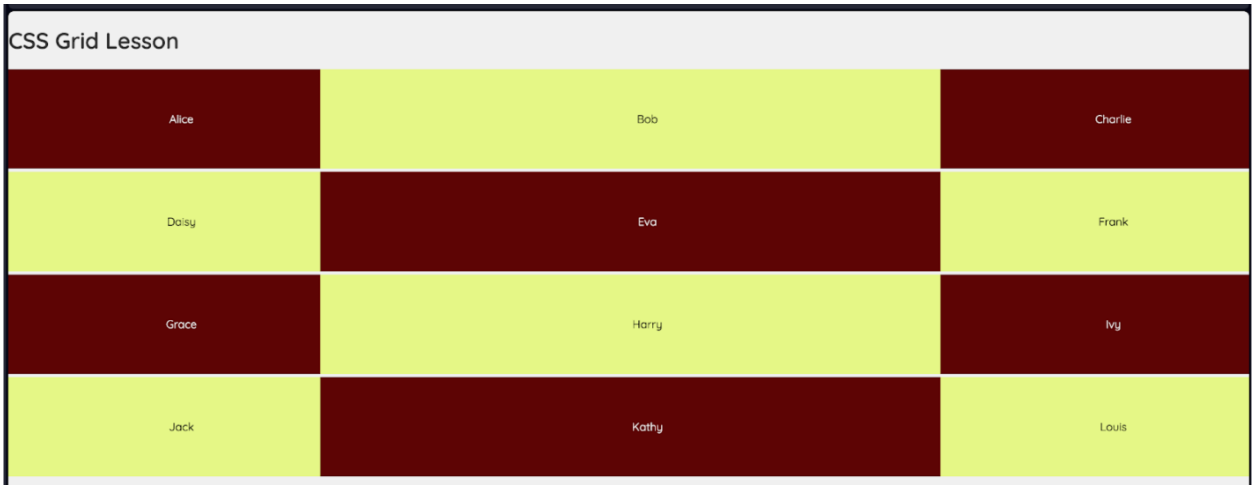
Initial Screenshot:



CSS Grid Lesson

Alice	Bob	Charlie
Daisy	Eva	Frank
Grace	Harry	Ivy
Jack	Kathy	Louis

Final Screenshot:



CSS Grid Lesson

Alice	Bob	Charlie
Daisy	Eva	Frank
Grace	Harry	Ivy
Jack	Kathy	Louis

Can you see the increase in the size of the second column? This is because the columns occupy the frames proportionally. The second column now occupies twice the width as the first and the third columns. Now, the entire width gets divided into four parts instead of three. One part is occupied by the first column, two parts by the second column, and one by the last column.

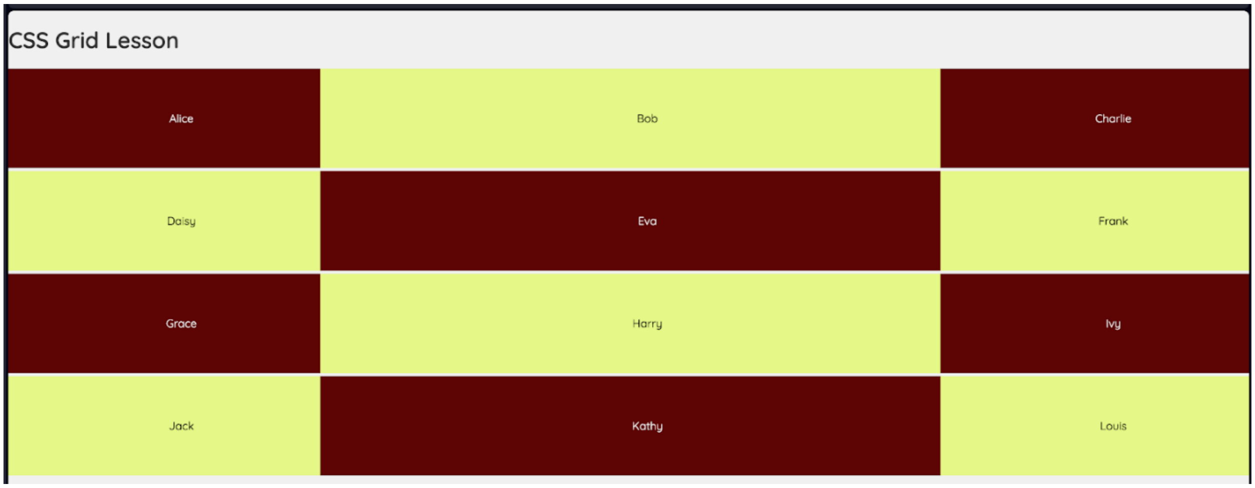
And right now, the rows are equally sized. By using 'grid-template-rows', we can manipulate the size of the rows as well.

Task 4: Add 'grid-template-rows: 1fr 2fr 3fr 2fr;' into the .content class.

Starter Code (CSS):

```
body {
        width: 100%;
        font-family: "Quicksand", sans-serif;
        background-color: #f0f0f0;
        margin: 0;
        padding: 10px;
}
h1 {
        font-size: 36px;
        color: #333;
}
.content {
        display: grid;
        grid-template-columns: 1fr 2fr 1fr;
        gap: 5px;
}
.box {
        width: 100%;
        padding: 30px;
        text-align: center;
        line-height: 100px;
        font-size: 18px;
}
.color1 {
        background-color: #5d0404;
        color: white;
}
.color2 {
        background-color: #e4f785;
}
```

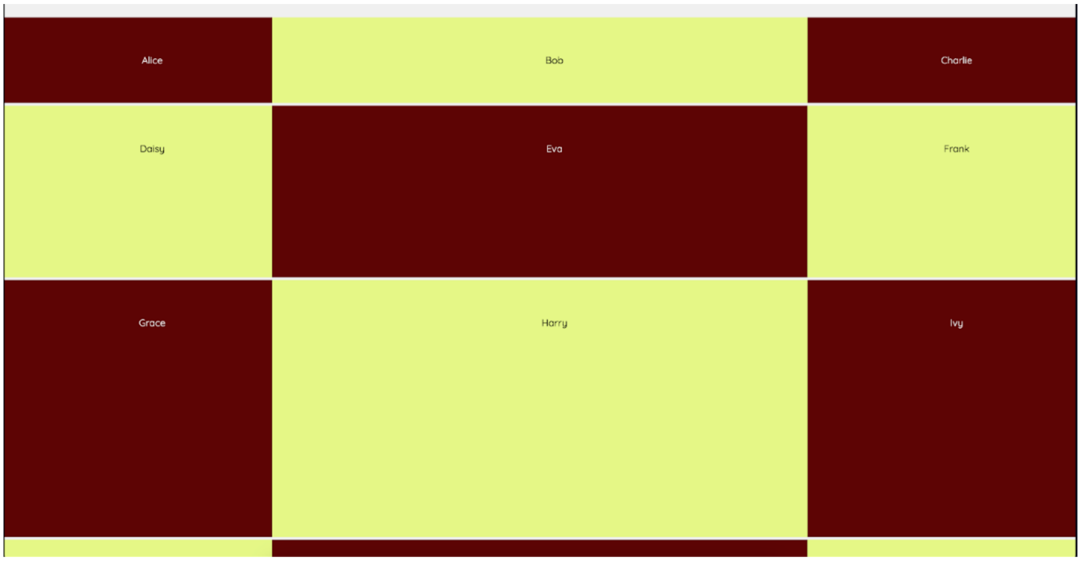
Initial Screenshot:



CSS Grid Lesson

Alice	Bob	Charlie
Daisy	Eva	Frank
Grace	Harry	Ivy
Jack	Kathy	Louis

Final Screenshot:



Alice	Bob	Charlie
Daisy	Eva	Frank
Grace	Harry	Ivy

As you can see, the rows have different sizes now as well.