

MODULE 3: FRONT END

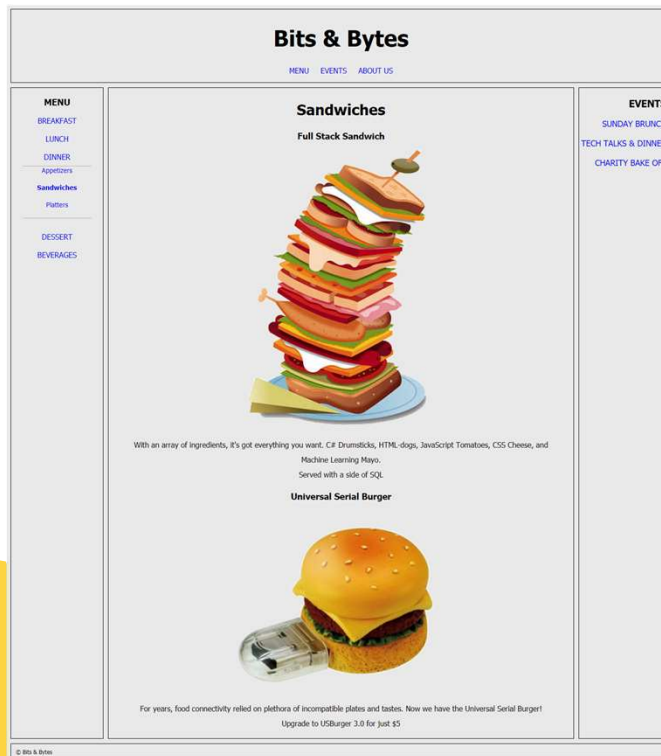
CSS Layouts with Grid



Yesterday

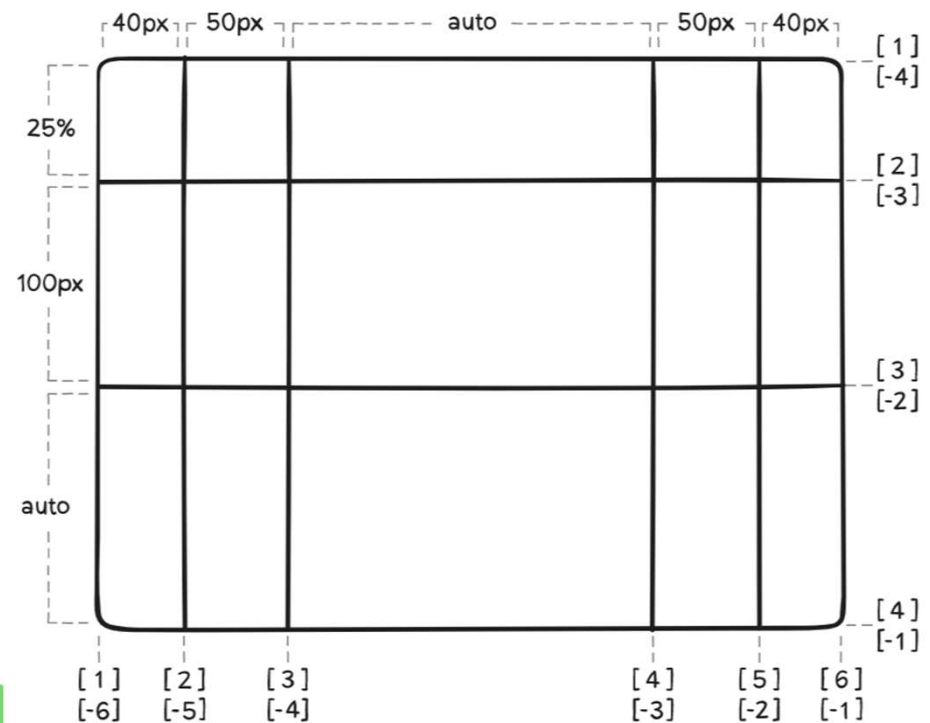
- What does HTML stand for?
- How is HTML organized?
- What does CSS stand for?
- What is a selector?
- What are some selectors?
- What is the difference between block and inline?

Modern Layout



CSS Grid Layout

A two-dimensional grid-based layout system



grid-template-columns

- Lets you define the size of the columns of your grid
- Grid lines are defined as fractional units (fr)
- `grid-template-columns: 1fr 2fr 4fr 1fr;`



grid-template-areas

- Names for different sections of the defined grid

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

```
grid-template-areas:
```

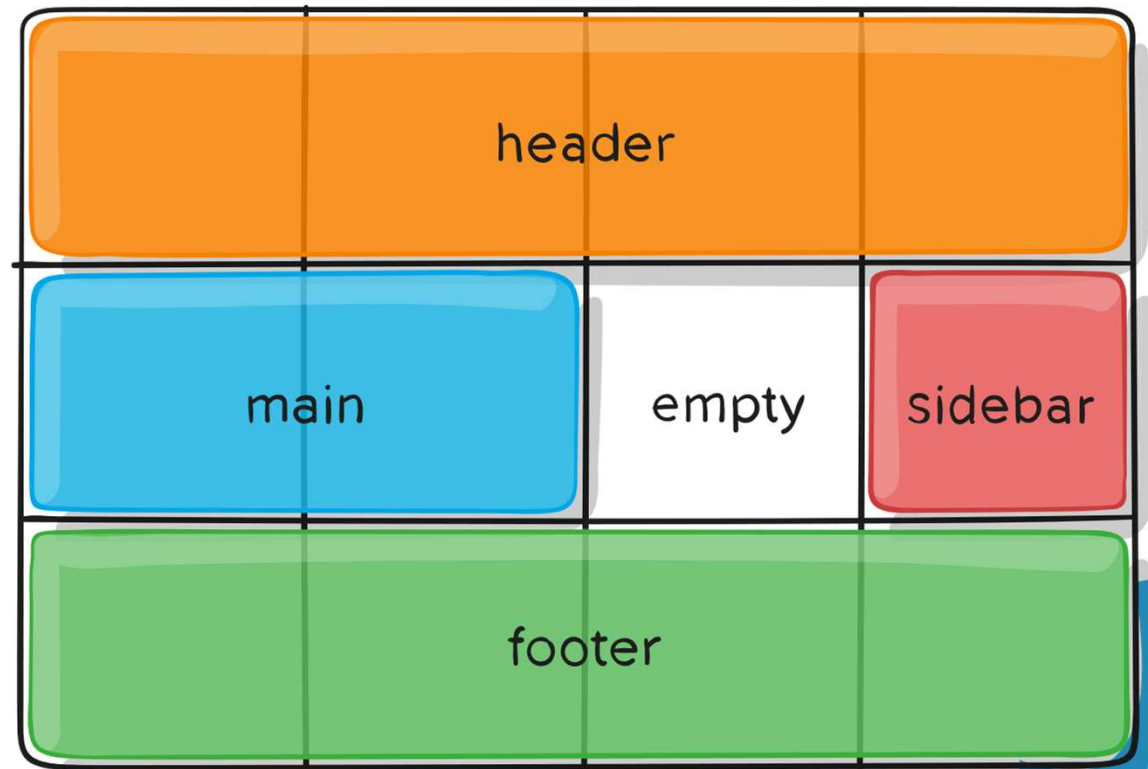
```
  "header header header header"
```

```
  "main main . sidebar"
```

```
  "footer footer footer footer";
```

grid-template-areas

```
grid-template-columns: 1fr 1fr 1fr 1fr;  
grid-template-areas:  
  "header header header header"  
  "main main . sidebar"  
  "footer footer footer footer";
```



grid-area

- Where a certain element will go in the grid
- The names will match the names that were defined in that element's grid-template-areas

```
.container {  
  display: grid;  
  grid-template-columns: 4fr 1fr 2fr;  
  grid-template-areas:  
    "header header header"  
    "main . sidebar"  
    "footer footer footer";  
}  
  
.item-a {  
  grid-area: header;  
}  
  
.item-b {  
  grid-area: main;  
}  
  
.item-c {  
  grid-area: sidebar;  
}  
  
.item-d {  
  grid-area: footer;  
}
```



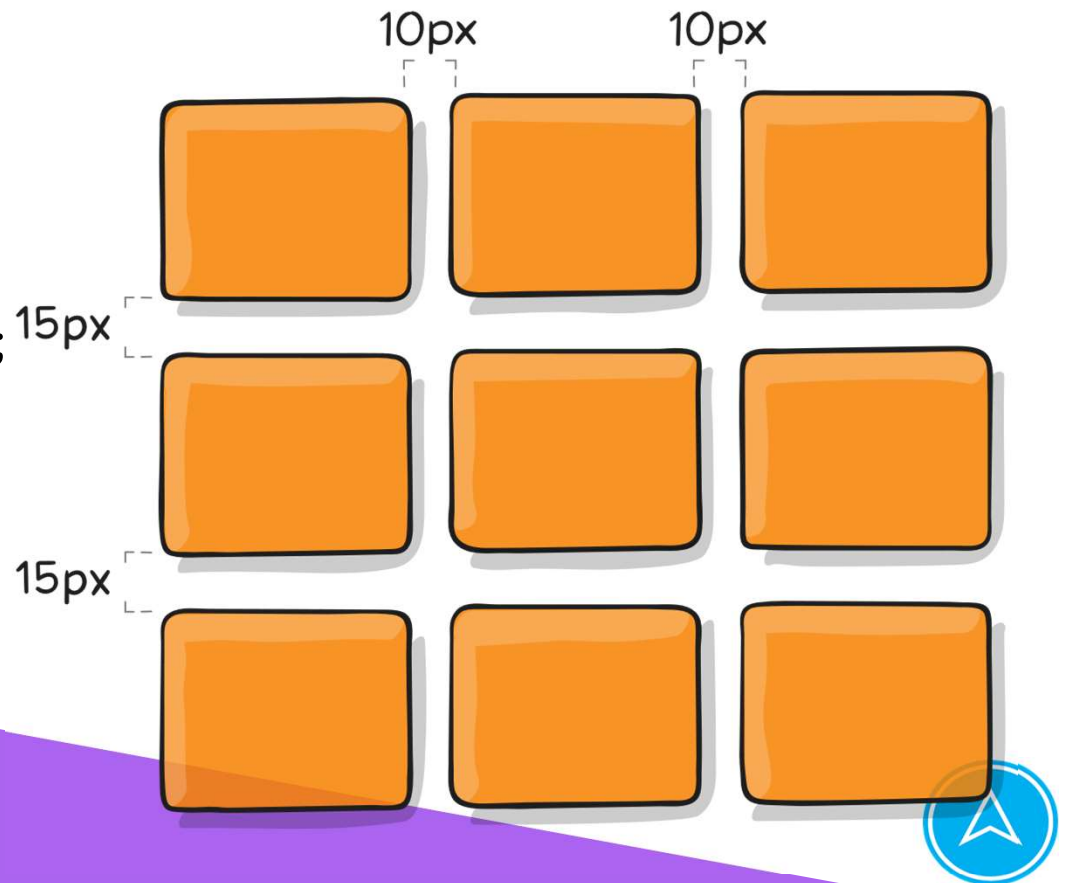
Height

- Set the height of the grid in units of vh
- vh is **view**port **height**
- Each vh is 1% of the viewport height
- **height: 100vh** is 100% of the viewport height

grid-gap

- The space between the grid cells

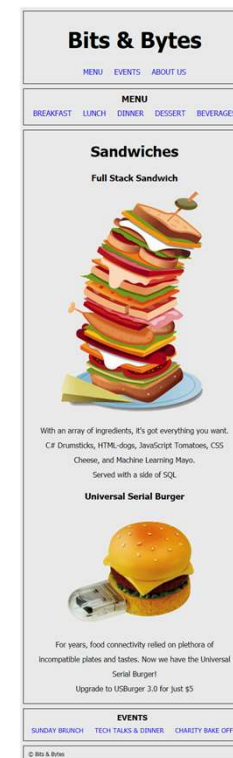
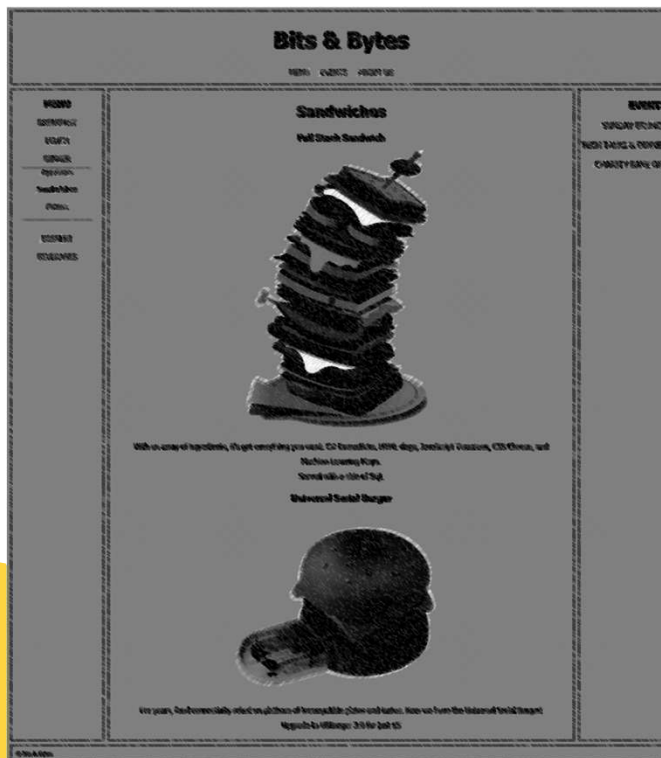
```
.container {  
  grid-template-columns: 1fr 1fr 1fr;  
  grid-column-gap: 10px;  
  grid-row-gap: 15px;  
}
```



LET'S CODE!



Modern Layout



What's the difference?



ELEVATE  YOURSELF

Media Queries

- “Queries” you add to css that change layout
- @media then you query
- Media queries can be used to check many things, such as:
 - width and height of the viewport
 - width and height of the device
 - orientation (is the tablet/phone in landscape or portrait mode?)
 - resolution



Media Queries

```
@media not|only mediatype and (mediafeature and|or|not mediafeature) {  
  CSS-Code;  
}
```

Example:

```
body {  
  background-color: lightblue;  
}
```

```
@media screen and (min-width: 400px) {  
  body {  
    background-color: lightgreen;  
  }  
}
```

LET'S CODE!



WHAT QUESTIONS DO
YOU HAVE?



Reading for tonight:

CSS Flexbox

