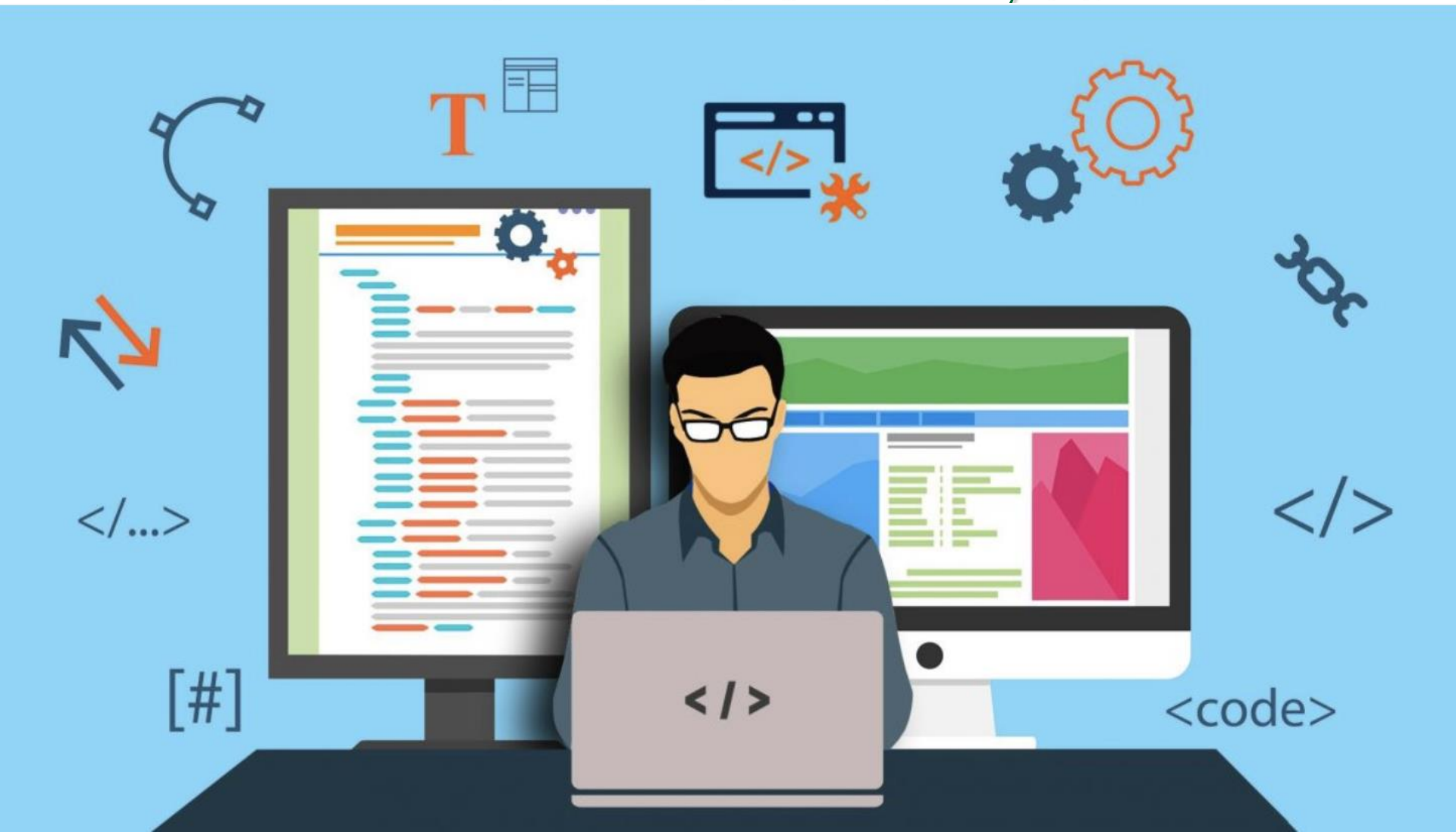
Cascading Style Sheets IV
Flexbox & Grid Layout



Flexbox Layout

Flexbox Layout Module

- The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning.

https://www.w3schools.com/css/css3_flexbox.asp

1. Design a Flex Container

- The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning.

The element above represents a flex container (the blue area) with three flex items.

```
<div class="flex-container">
  <div>1</div>
  <div>2</div>
  <div>3</div>
</div>
```

```
.flex-container {
  display: flex;
}
```

2. Add Properties

- The flex container properties are:
  - flex-direction
  - flex-wrap
  - flex-flow
  - justify-content
  - align-items
  - align-content

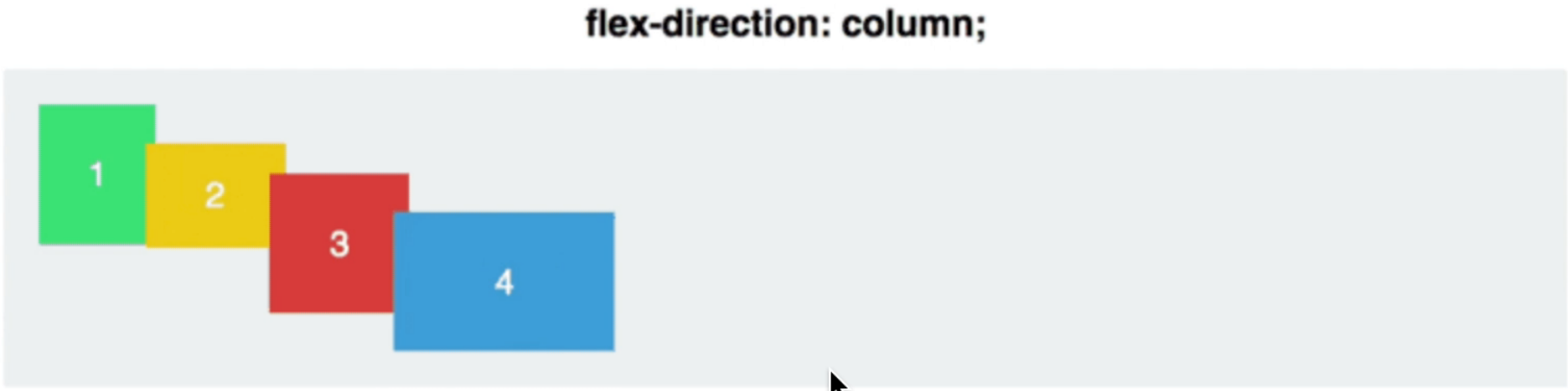
2. Add Properties

- flex-direction


```
.flex-container {
  display: flex;
  flex-direction: column;
}
```

2. Add Properties

- flex-direction



2. Add Properties

- justify-content


```
.flex-container {
  display: flex;
  justify-content: center;
}
```

2. Add Properties

- justify-content

2. Add Properties

flex-direction: row;

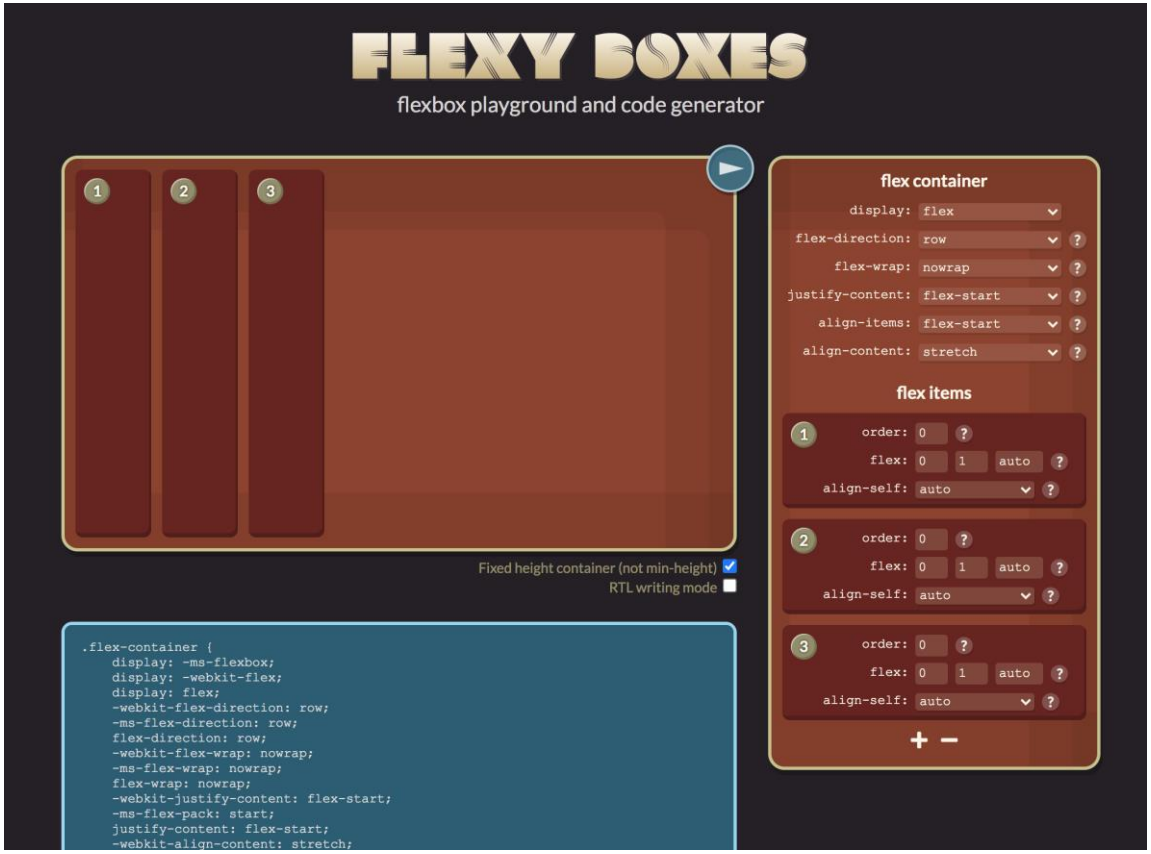
justify-content: center; align-items: center;

2. Add Properties

- align-items

Flexbox

- Play with this tool to understand more about how flexbox works



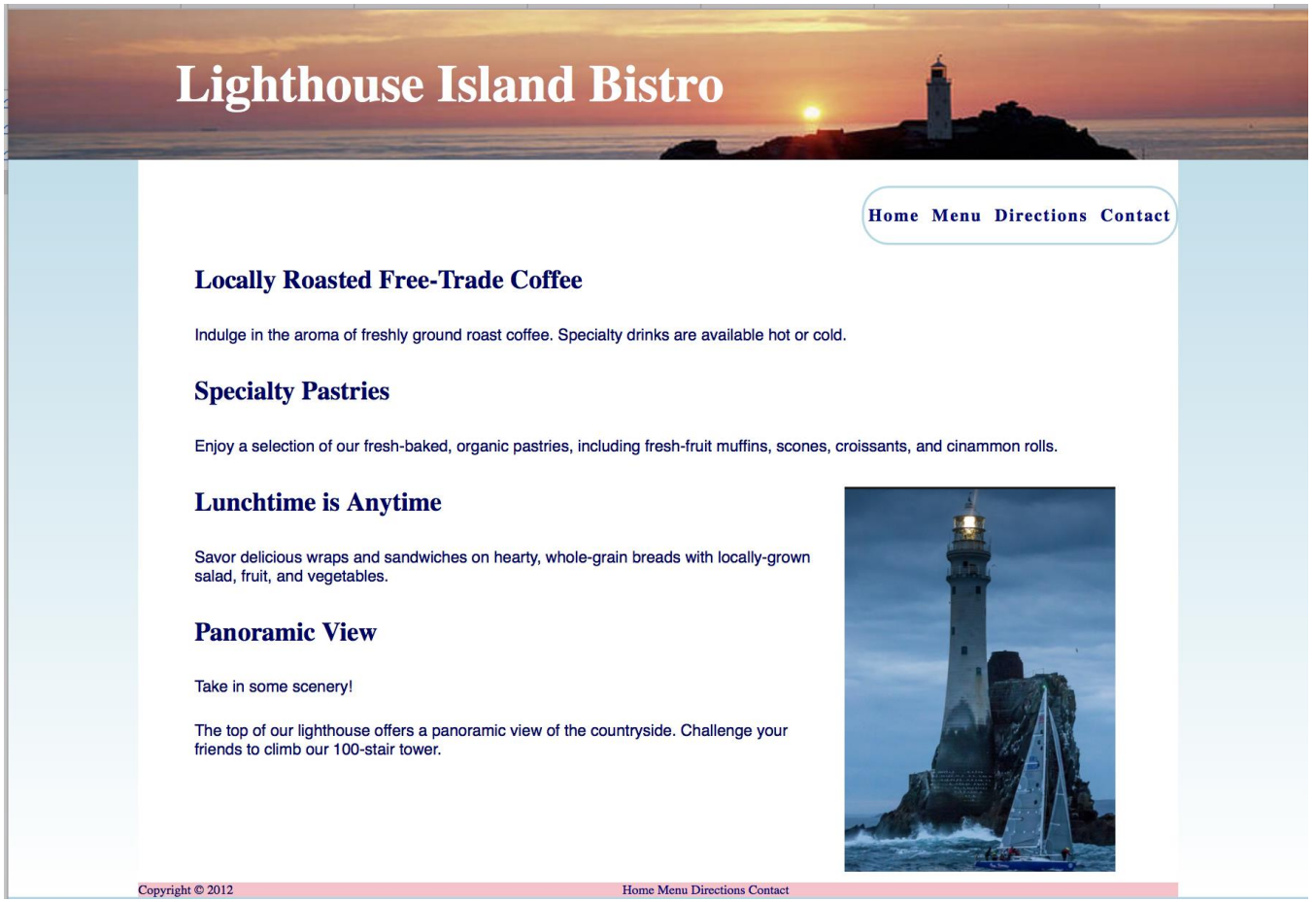
https://the-echoplex.net/flexyboxes/
https://loading.io/flexbox/ *(another tool!)*

Let's apply Flexbox to the Bistro website

The footer is aligned badly. How can flexbox help me here?

Let's apply Flexbox to the Bistro website

```
footer{
…
…
display: flex;
align-content: center;
}
```



Flexbox Froggy

- A nice game to get more familiar with flexbox

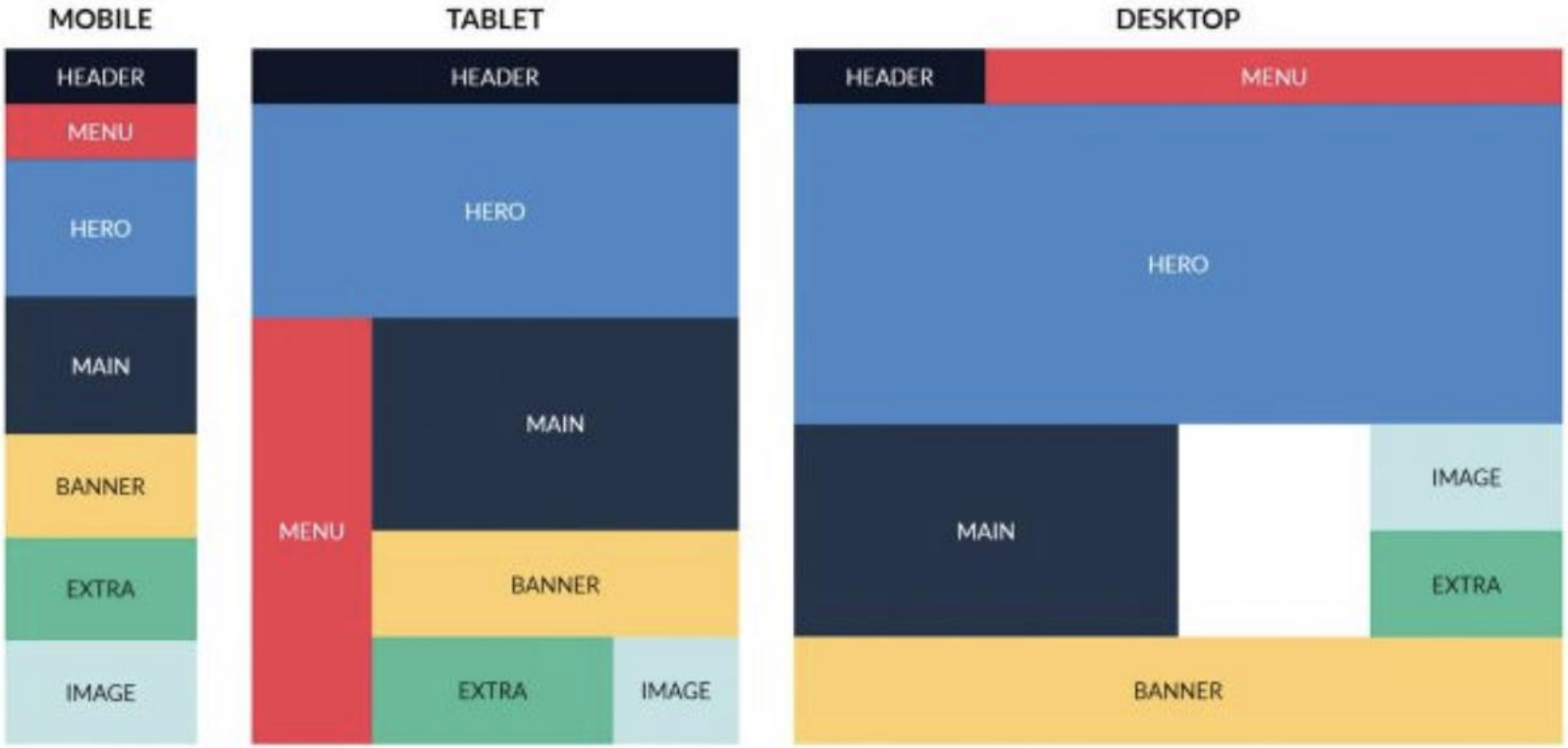
Grid Layout

Grid Layout Module

- A new and versatile system for positioning elements within a website layout through the use of a highly customisable grid.
- It introduces unprecedented flexibility in layout, using just pure CSS and without absolutely positioning elements.
- CSS Grid enables us to achieve extremely diverse and device-specific layouts from the same exact HTML markup.

https://www.creativebloq.com/advice/a-comprehensive-guide-to-using-css-grid

Grid Layout Module



https://www.creativebloq.com/advice/a-comprehensive-guide-to-using-css-grid

1. Define the Grid

- **display: grid**. Columns and rows are determined by the number of space-separated sizes assigned to **grid-template-columns** and **grid-template-rows respectively**.

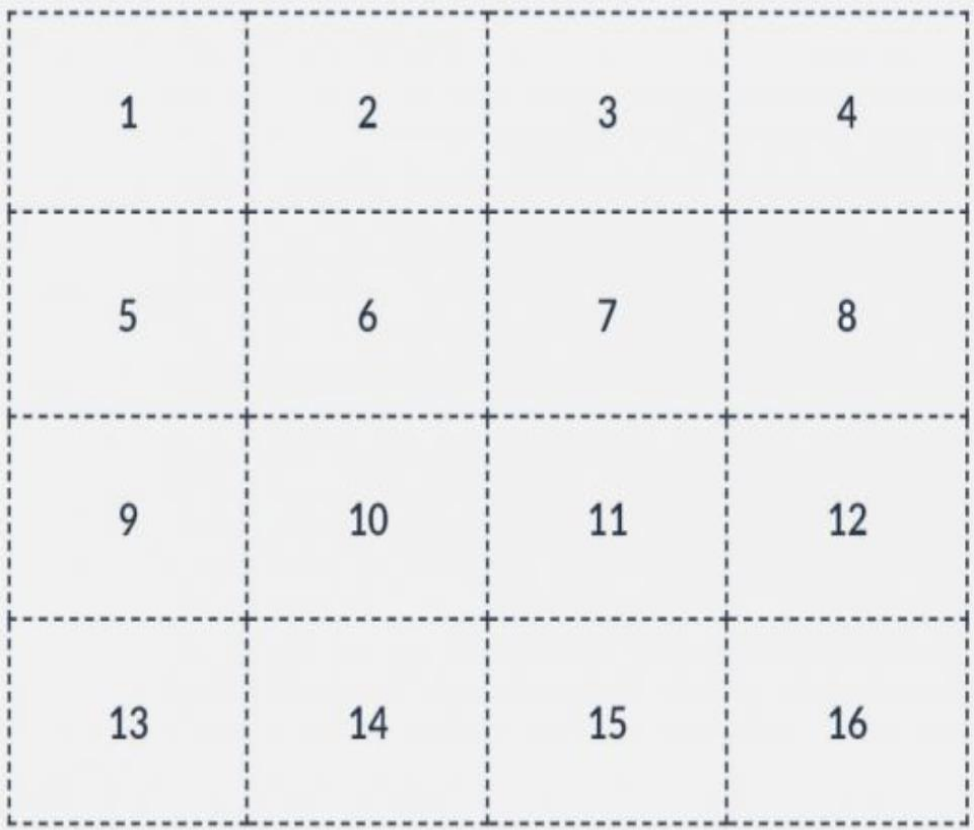
- Sizes can be any valid CSS unit such as **px** or **vw**, or the **auto** keyword that enables columns or rows to stretch across available space.

- For instance, **grid-template-columns: 10px auto** leads to a 10px column followed by a second column that fills all available space.

2. Using fr

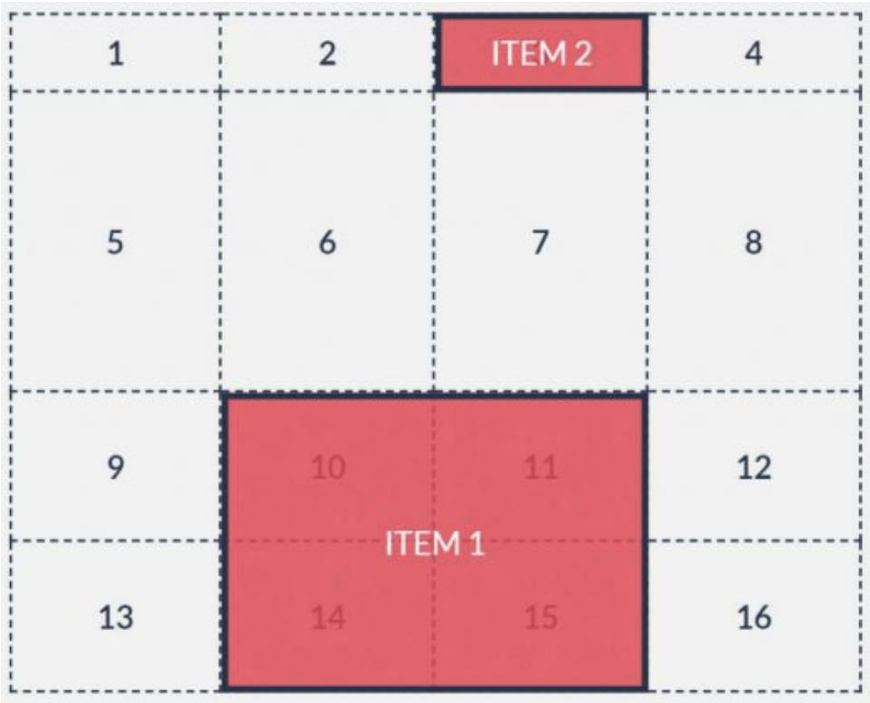
- Grid also uses a 'fractional' unit **fr** that causes any remaining space to be distributed to columns or rows based on the ratios of these units.
- **grid-template-rows: 1fr 2fr** creates two dynamic rows with the second twice the size of the first,
- **grid-template-columns: 1fr 1fr 1fr 1fr** defines four equal-sized columns.
- The latter can be simplified using the new **repeat()** function to **grid-template-columns: repeat(4, 1fr)**.

3. An example ..



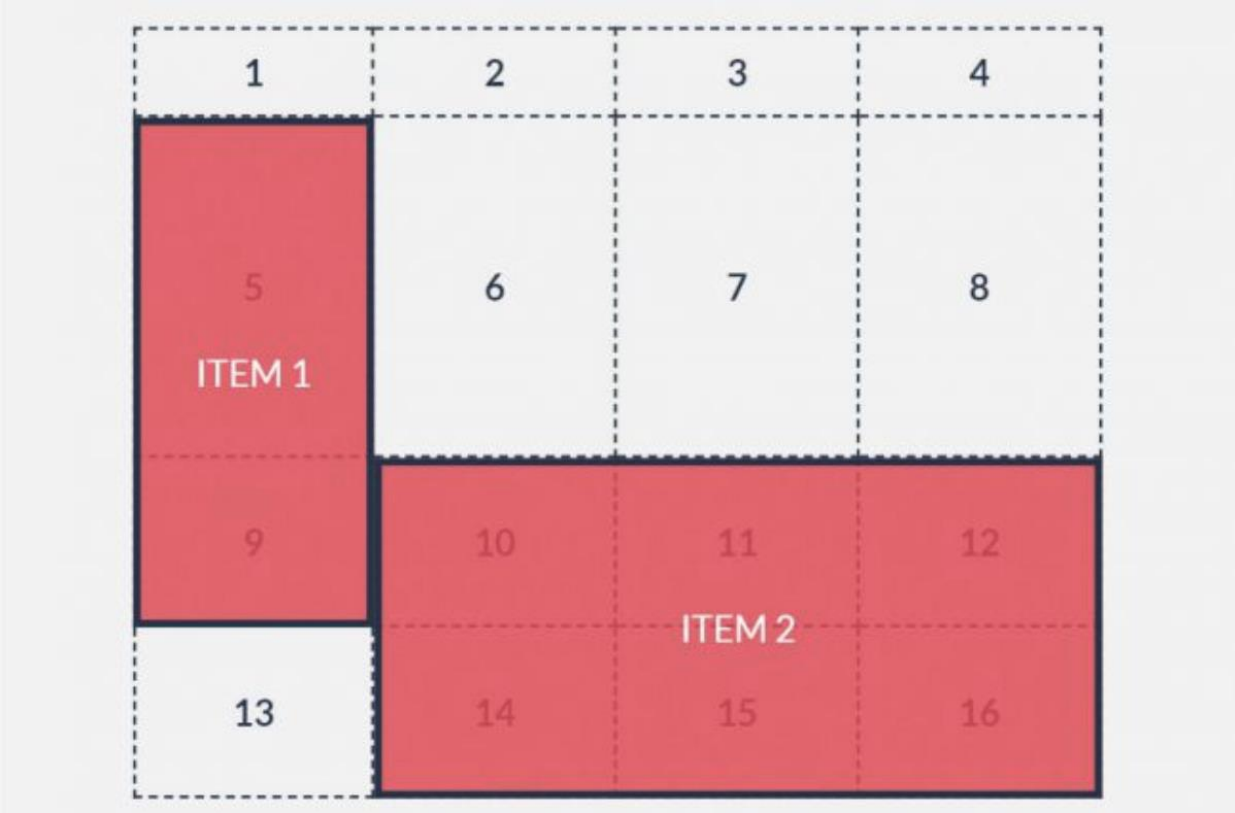

```
.grid {
 display: grid;
 grid-template-columns: repeat(4, 1fr);
 grid-template-rows: repeat(4, 75px);
}
```

4. Positioning items using line numbers..




```
#item1 {
 grid-column-start: 2;
 grid-column-end: 4;
 grid-row-start: 3;
 grid-row-end: -1;
}
#item2 {
 grid-column: 3;
 grid-row: 1;
}
```

4. Positioning items using line numbers..




```
#item1 {
 grid-column: 1;
 grid-row: 2 / 4;
}
#item2 {
 grid-column: 2 / span 3;
 grid-row: 3 / span 2;
}
```

grid-row-start *grid-column-start* *grid-row-end* *grid-column-end*

```
#item1 {
 grid-area: 2 / 1 / span 2 / span 3;
}
#item2 {
 grid-area: 4 / 4;
}
```

OR

5. Positioning items using area names..

```
#item1 {
 grid-area: item1;
}
#item2 {
 grid-area: item2;
}
.grid {
 grid-template-areas:
   ". . . ."
   ". . . item1"
   "item2 item2 item2 item1"
   "item2 item2 item2 .";
}
```

What would this grid look like?

HTML

```
<div class="grid">
 <div class="header"></div>
 <div class="menu"></div>
 <div class="hero"></div>
 <div class="main"></div>
 <div class="banner"></div>
 <div class="extra"></div>
 <div class="image"></div>
</div>
```

CSS

```
.header { grid-area: header; }
.menu { grid-area: menu; }
.hero { grid-area: hero; }
.main { grid-area: main; }
.banner { grid-area: banner; }
.extra { grid-area: extra; }
.image { grid-area: image; }
.grid {
  display: grid;
  grid-template-columns: repeat(4, 1fr);
  grid-template-rows: 40px 2fr repeat(4, 1fr);
  grid-template-areas:
    "header header header header"
    "hero hero hero hero"
    "menu main main main"
    "menu main main main"
    "menu banner banner banner"
    "menu extra image image";
```

Write the *grid-template-area* (CSS code) for the following 4cols x6rows..

Write the *grid-template-area* (CSS code) for the following ..


```
.grid {
  grid-template-areas:
    "header menu menu menu"
    "hero hero hero hero"
    "hero hero hero hero"
    "main main . image"
    "main main . extra"
    "banner banner banner banner";
}
```

Differences between Flexbox & Grid

- **Flexbox** is made for one dimensional layouts and **Grid** is made for two dimensional layouts.

- **Flexbox** will give you more flexibility than **Grid**. It'll also be easier to maintain and require less code.

- **Flexbox** is content-first, **Grid** is layout-first

https://hackernoon.com/the-ultimate-css-battle-grid-vs-flexbox-d40da0449faf

Applying Flexbox & Grid to Fish Creek

Fish Creek Animal Hospital

- Home
- Services
- Ask the Vet
- Contact

Full Services Facility

Verinarians and staff are on duty 24 hours a day,7 days a week.

Years of Experience

Fish Creek Veterinarians have provided quality,dependable care for your beloved animals since 1984.

Open Door Policy.

Our professional welcome owners to stay with their pets during any medical procedure.

1-800-555-5555
1242 Grassy Lane
Fish Creek,WI 55534

Some good resources..

- A Complete Guide to FlexBox - https://css-tricks.com/snippets/css/a-guide-to-flexbox/
- CSS Grid Layout - https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout