LESSON 8



</>

CSS Grid

Learn it all @MyWebUniversity



A FREE MYWEBUNIVERSITY COURSE

HTML & CSS

LESSON 8

CSS Grid

grid-template-columns, fr units, repeat, gap, spanning, areas and auto-fit

display: grid

style.css

```css
.gallery {
  display: grid;
  grid-template-columns: 200px 200px 200px;
  gap: 1rem;
}
/* 6 images → 2 rows of 3 */
```

- Grid lays items out in rows and columns at the same time. Define the columns, and items …
- Use flexbox for one direction and grid for whole-page or two-dimensional layouts.

01

The fr Unit and repeat

style.css

```
grid-template-columns: 1fr 1fr 1fr;          /* 3 equal */
grid-template-columns: repeat(3, 1fr);       /* same */
grid-template-columns: 250px 1fr;            /* sidebar + content */
grid-template-columns: 1fr 2fr;              /* 1/3 and 2/3 */
grid-template-columns: repeat(4, minmax(150px, 1fr));
```

- fr means a fraction of the free space. One fr two fr gives the second column twice the …
- repeat avoids typing the same value many times, and minmax sets a minimum and maximum size.

02

Placing and Spanning Items

style.css

```
.banner   { grid-column: 1 / -1; }          /* full width */
.featured { grid-column: span 2;
            grid-row: span 2; }            /* 2 x 2 cells */
.sidebar  { grid-column: 3; grid-row: 1 / 4; }
```

- Grid lines are numbered from one. grid-column one slash minus one stretches an item across …
- span makes an item cover several cells, perfect for a big featured photo in a gallery.

03

Named Template Areas

style.css

```css
.page {
  display: grid;
  grid-template-columns: 220px 1fr;
  grid-template-areas:
    "header  header"
    "sidebar main"
    "footer  footer";
}
header { grid-area: header; }
aside  { grid-area: sidebar; }
main   { grid-area: main; }
footer { grid-area: footer; }
```

- grid-template-areas lets you draw the layout with words. Each string is a row, and each …
- Then assign each element to an area by name. Changing the layout for phones means changing …

04

Responsive Grid without Media Queries

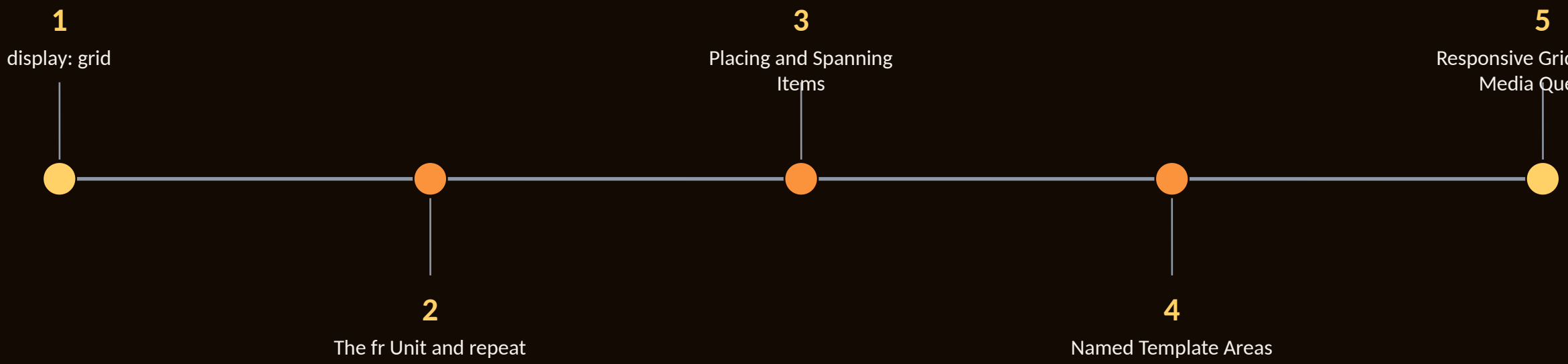
style.css

```
.cards {
  display: grid;
  grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
  gap: 1.5rem;
}
/* phone: 1 column · tablet: 2 · desktop: 4 */
```

- This one line creates as many columns as fit, each at least 250 pixels wide, sharing the …
- On a phone you get one column, on a tablet two, on a desktop four, with no media queries at …

05

Lesson Map


1
display: grid
2
The fr Unit and repeat
3
Placing and Spanning Items
4
Named Template Areas
5
Responsive Grid without Media Queries

Sources and Further Reading

style.css

```
.gallery {
  display: grid;
  grid-template-columns: 200px 200px 200px;
  gap: 1rem;
}
/* 6 images → 2 rows of 3 */
```

HTML & CSS · 8

style.css

```
.banner   { grid-column: 1 / -1; }          /* full width */
.featured { grid-column: span 2;
            grid-row: span 2; }             /* 2 x 2 cells */
.sidebar  { grid-column: 3; grid-row: 1 / 4; }
```

Thank you for watching

MyWebUniversity.com

Free technical education since 2000 — English and Dari

@MyWebUniversity · @MyWebUniversity-Dari