

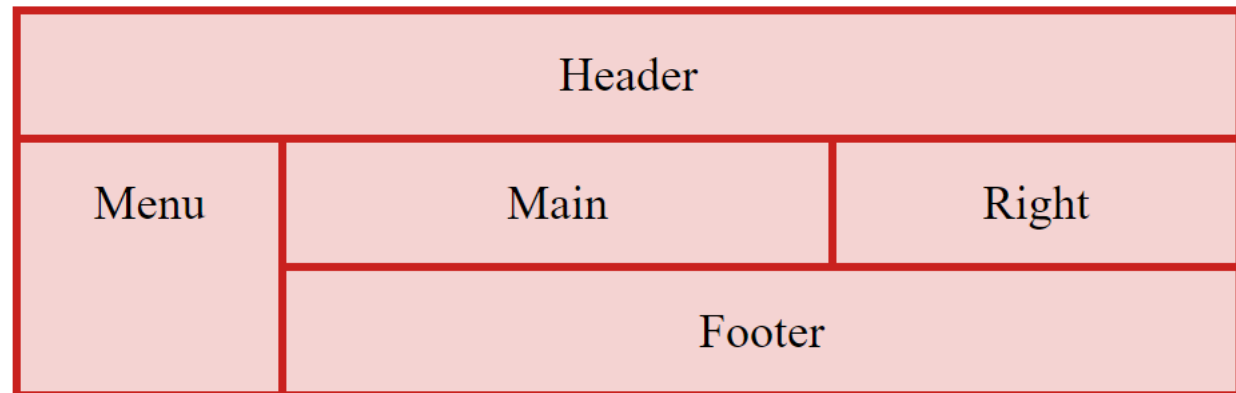


# Grid

Klára Pešková, [Klara.Peskova@mff.cuni.cz](mailto:Klara.Peskova@mff.cuni.cz)  
Katedra softwaru a výuky informatiky, MFF UK  
Základy tvorby webu, ZS 2023/24

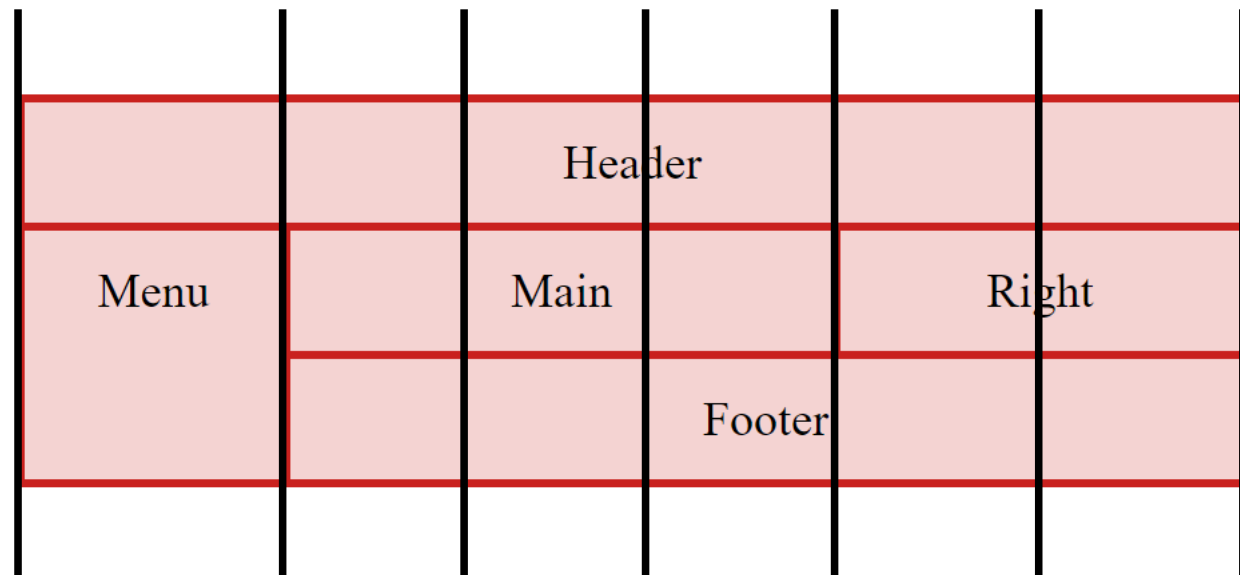
# grid

- rozložení prvků na stránce do mřížky - řádků a sloupců



# grid

- rozložení prvků na stránce do mřížky - řádků a sloupců
- celý layout je rozdělen na sloupce



# grid technicky

- Layout je tvořen jedním rodičovským elementem s jedním nebo více potomků
  - potomci jsou automaticky součástí mřížky

1	2
3	4
5	6

# grid technicky

- Layout je tvořen jedním rodičovským elementem s jedním nebo více potomků
  - potomci jsou automaticky součástí mřížky

1	2
3	4
5	6

```
<div class="grid-container">  
  <div class="grid-item">1</div>  
  <div class="grid-item">2</div>  
  <div class="grid-item">3</div>  
  <div class="grid-item">4</div>  
  <div class="grid-item">5</div>  
  <div class="grid-item">6</div>  
</div>
```

# grid technicky

- Layout je tvořen jedním rodičovským elementem s jedním nebo více potomků
  - potomci jsou automaticky součástí mřížky

1	2
3	4
5	6

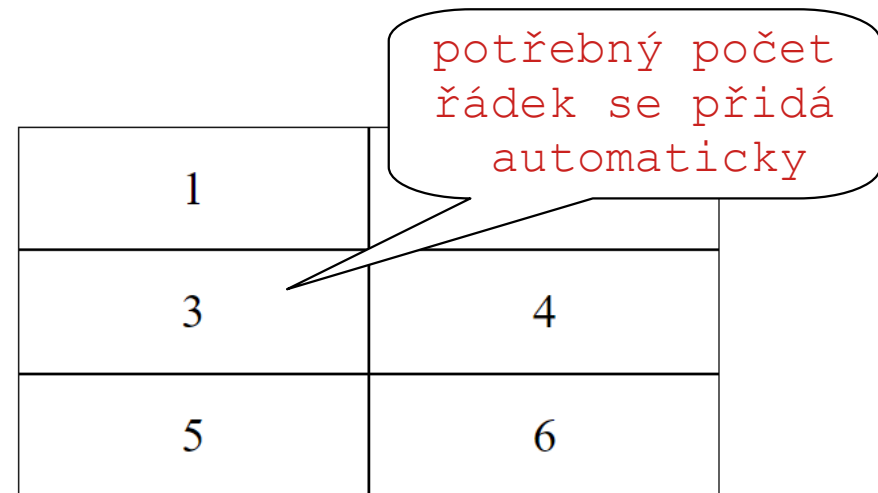
```
<div class="grid-container">  
  <div class="grid-item">1</div>  
  <div class="grid-item">2</div>  
  <div class="grid-item">3</div>  
  <div class="grid-item">4</div>  
  <div class="grid-item">5</div>  
  <div class="grid-item">6</div>  
</div>
```

```
.grid-container {  
  display: grid;  
  grid-template-columns: auto auto;  
}  
.grid-item {  
}
```

# grid technicky

- Layout je tvořen jedním rodičovským elementem s jedním nebo více potomků
  - potomci jsou automaticky součástí mřížky

```
<div class="grid-container">  
  <div class="grid-item">1</div>  
  <div class="grid-item">2</div>  
  <div class="grid-item">3</div>  
  <div class="grid-item">4</div>  
  <div class="grid-item">5</div>  
  <div class="grid-item">6</div>  
</div>
```



```
.grid-container {  
  display: grid;  
  grid-template-columns: auto auto;  
}  
.grid-item {  
}
```

# grid – příklad

```
<div class="grid-container">
  <div class="item1">1</div>
  <div>2</div>
  <div>3</div>
  <div>4</div>
  <div>5</div>
</div>
```

1		2
3	4	5

```
.grid-container {
  display: grid;
  grid-template-columns:
    auto auto auto;
}

.item1 {
  grid-column-start: 1;
  grid-column-end: 3;
}
```



# container - šířka sloupců, výška řádek

- Počet a šířka sloupců se zadává pomocí vlastnosti  
`grid-template-columns: 80px auto 40px;`
- hodnoty mohou být zadány:
  - v pixelech (px),
  - v procentech (%),
  - dopočítané automaticky (auto),
  - pomocí **fraction (fr)** - část dostupného místa (po vypořádání px a %) - [příklady na digitalocean.com](https://digitalocean.com)
- Výška jednotlivých řádek:  
`grid-template-rows: 200px auto`

# container - zarovnání

- Zarovnání mřížky uvnitř containeru:

`justify-content`

- Vertikální zarovnání

`align-content`

- Hodnoty

`space-evenly`  
`space-around`  
`space-between`  
`center`  
`start`  
`end`

# Container – mezery mezi elementy

- vlastnosti:

gap

row-gap

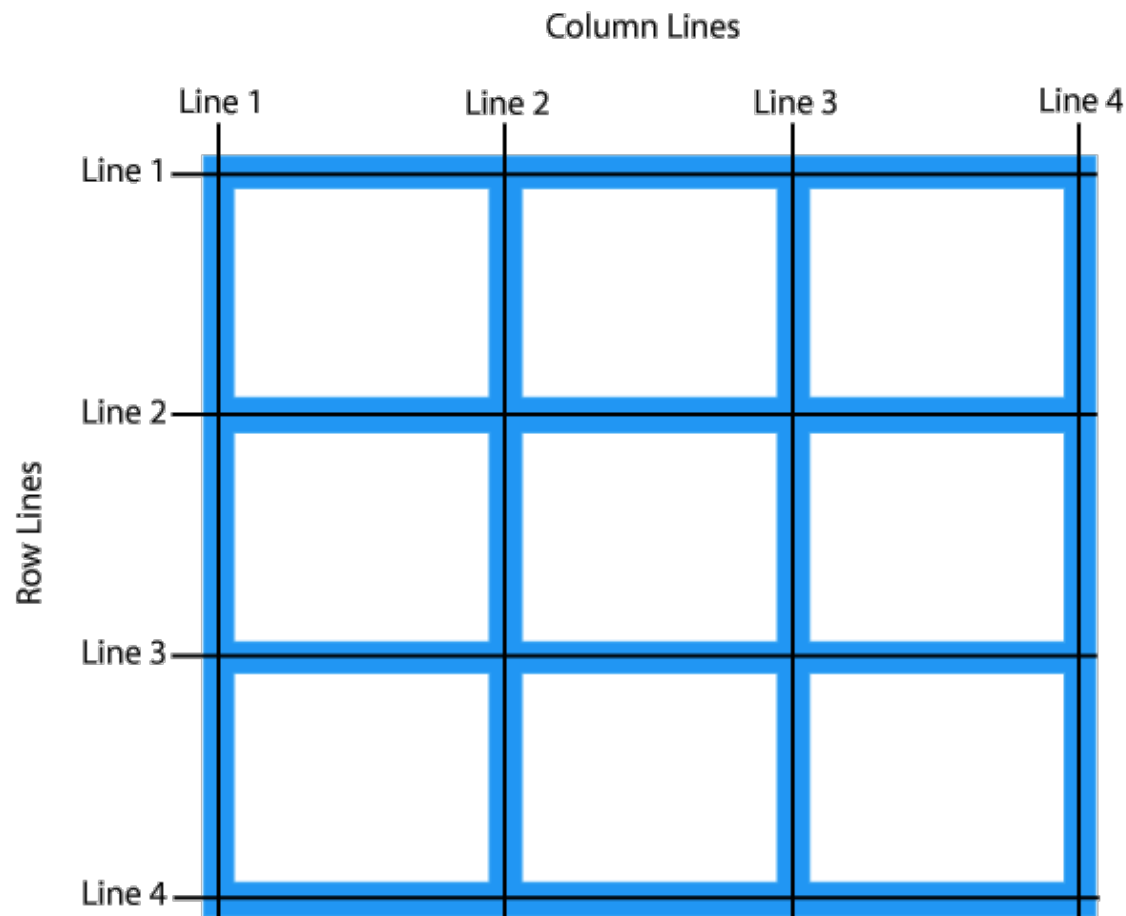
column-gap

## item – umístění ve mřížce

- `grid-column: 1 / 5;`
- `grid-column: 1 / span 3;`
  - určuje sloupeček, ve kterém buňka začne a před kterým skončí
- `grid-row`
  - to samé pro řádky
- Buňka přes více řádek a sloupců  
`grid-area: grid-row-start / grid-column-start / grid-row-end / grid-column-end`
  - `grid-area` je možné použít také ke změně pořadí buněk v tabulce

# Roztáhnutí prvků

- ... přes více sloupců nebo řádek



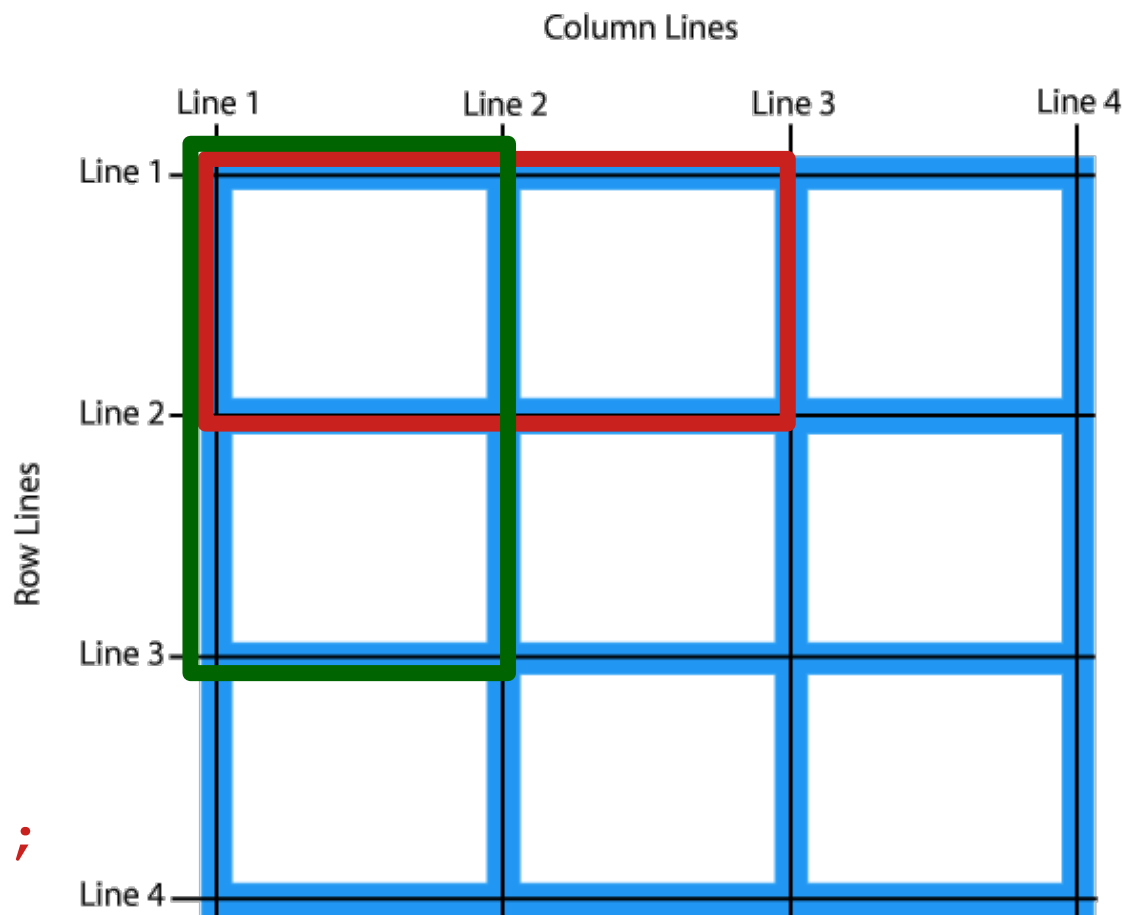
Zdroj: w3schools

# Roztáhnutí prvků

- ... přes více sloupců nebo řádek
- hranice roztažené buňky se zadávají pomocí následujících vlastností a čísel čar:

```
grid-column-start: 1;  
grid-column-end: 3;
```

```
grid-row-start: 1;  
grid-row-end: 3;
```



```
Grid-column: 1 / 3;  
Grid-column: 1 / span 2;
```

Zdroj: w3schools

# Pojmenované **items**

- `grid-area` je možné použít také k pojmenování oblasti

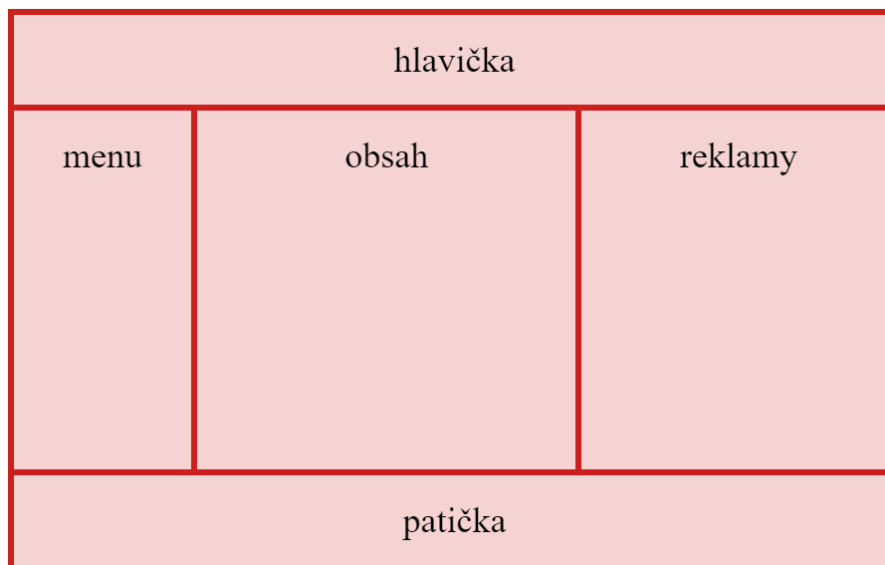
```
.item1 {  
    grid-area: zahlavi;  
}  
  
.grid-container {  
    grid-template-areas:  
    'zahlavi zahlavi zahlavi zahlavi zahlavi';  
}
```

- ostatní sloupečky se zarovnají jako obvykle
- tečka znamená sloupeček, který nemá jméno

# "Svatý grál" pomocí **grid**

```
.item1 { grid-area: header; }
.item2 { grid-area: menu; }
.item3 { grid-area: main; }
.item4 { grid-area: right; }
.item5 { grid-area: footer; }

.grid-container {
  grid-template-areas:
    'header header header header header header'
    'menu main main main right right'
    'footer footer footer footer footer
  footer';
}
```





# Co můžeme nastavit pro **container**

- Počet a šířku sloupců `grid-template-columns`
- Výšku řádek `grid-template-rows`
- Zarovnání obsahu `justify-content`
- Vertikální rozmístění řádek `align-content`
- Mezery `gap`

# Co můžeme nastavit pro **item**

- Roztažení buněk `grid-column`, `grid-row`, `grid-area`
- Jednotlivé buňky můžeme pojmenovat `grid-area`
- Můžeme změnit pořadí buněk `grid-area`