infoShare ACADEMY

CSS Grid

infoShare Academy

infoShareAcademy.com

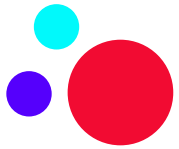
HELLO

Dawid Buliński

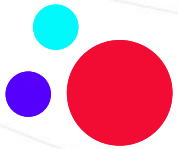
Front End Developer

infoShare ACADEMY

Agenda



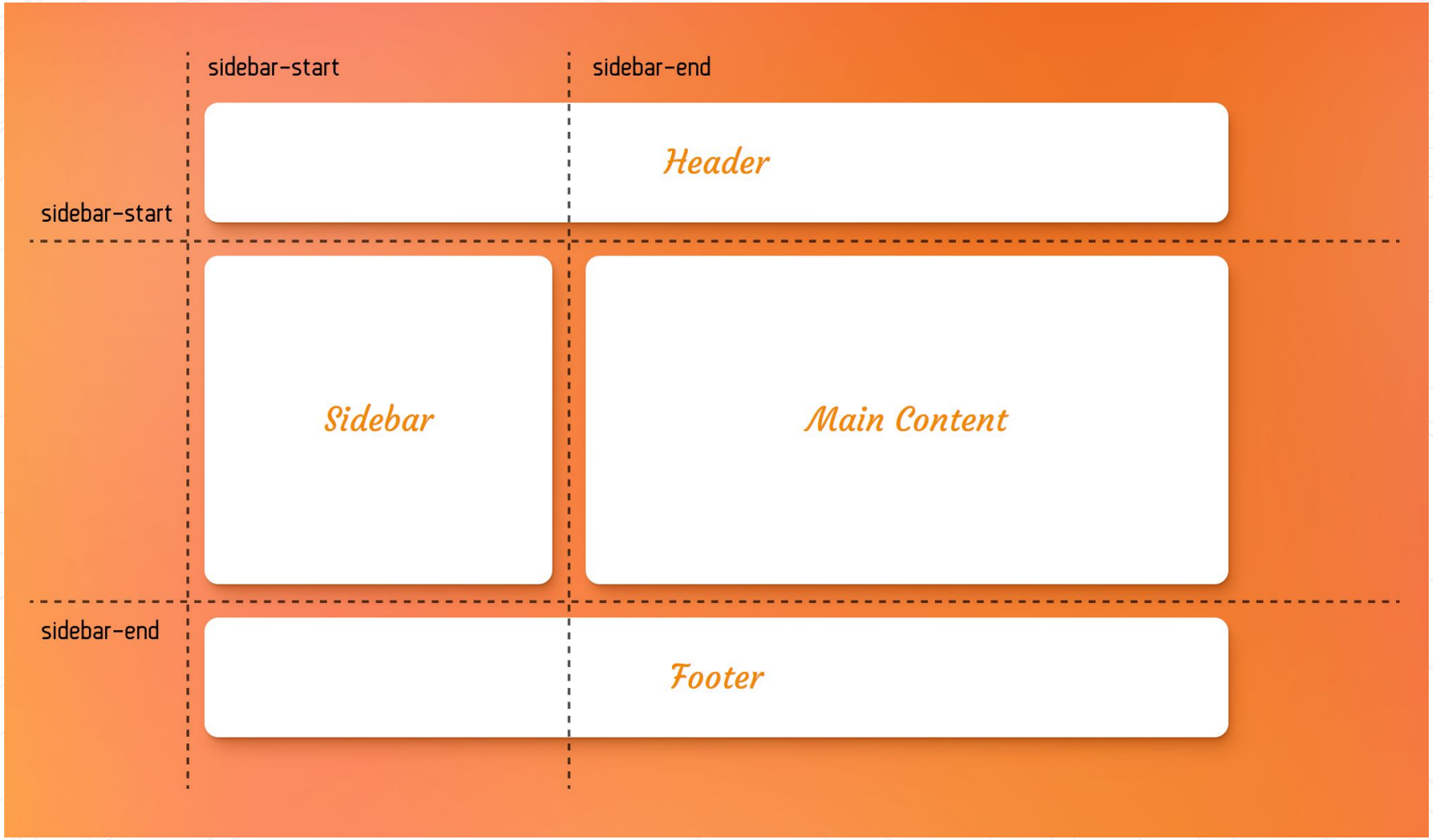
1. Konfiguracja ćwiczeń
2. Koncepcja CSS Grid (2 wymiary).
3. Podstawowe elementy CSS Grid.
4. Grid areas.
5. Pozycjonowanie elementów wewnątrz grida.
6. Budowa fluid grid. (auto-fit vs auto-fill)
7. Q & A.



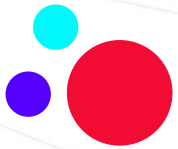
Podstawowe informacje

- Jedyny dwuwymiarowy design system w CSS
- Pozwala w łatwy sposób budować layouty całych stron z zachowaniem responsywności
- Szeroko wspierany (ponad 96% aktualnie używanych przeglądarek)
- Uzupełnia, a nie zastępuje pozostałe *layout modes* (flexbox, flow, positioned)

Podstawowe informacje - praktyka



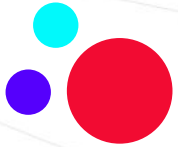
https://css-tricks.com/almanac/properties/g/grid-template-areas/



Podstawowe informacje – składnia

display: grid

Definiuje element jako grid container. Po jego użyciu dzieci tego elementu używają grida jako swojego layout mode.



Podstawowe informacje – składnia

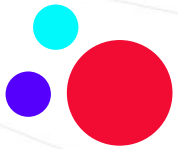
grid-template-columns: 10% 20% 30% 40%;

Pozwala zdefiniować szablon kolumn dla grida (używany na poziomie grid kontenera, czyli elementu, na którym definiujemy *display: grid*). Wartości oddzielone spacjami określają kolejne kolumny szablonu. W tym przypadku tworzymy 4 kolumny o wymiarach kolejno 10, 20, 30 i 40%.

W przypadku definiowania kilku kolumn (lub wierszy) o tych samych wymiarach możemy użyć funkcji **repeat(ilość powtórzeń, wymiar).** Np. możemy stworzyć 4 kolumny o szerokości 25% na dwa sposoby:

grid-template-columns: 25% 25% 25% 25%;

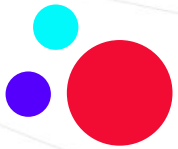
grid-template-columns: repeat(4, 25%);



Podstawowe informacje - składnia

grid-template-rows: 40px 60px;

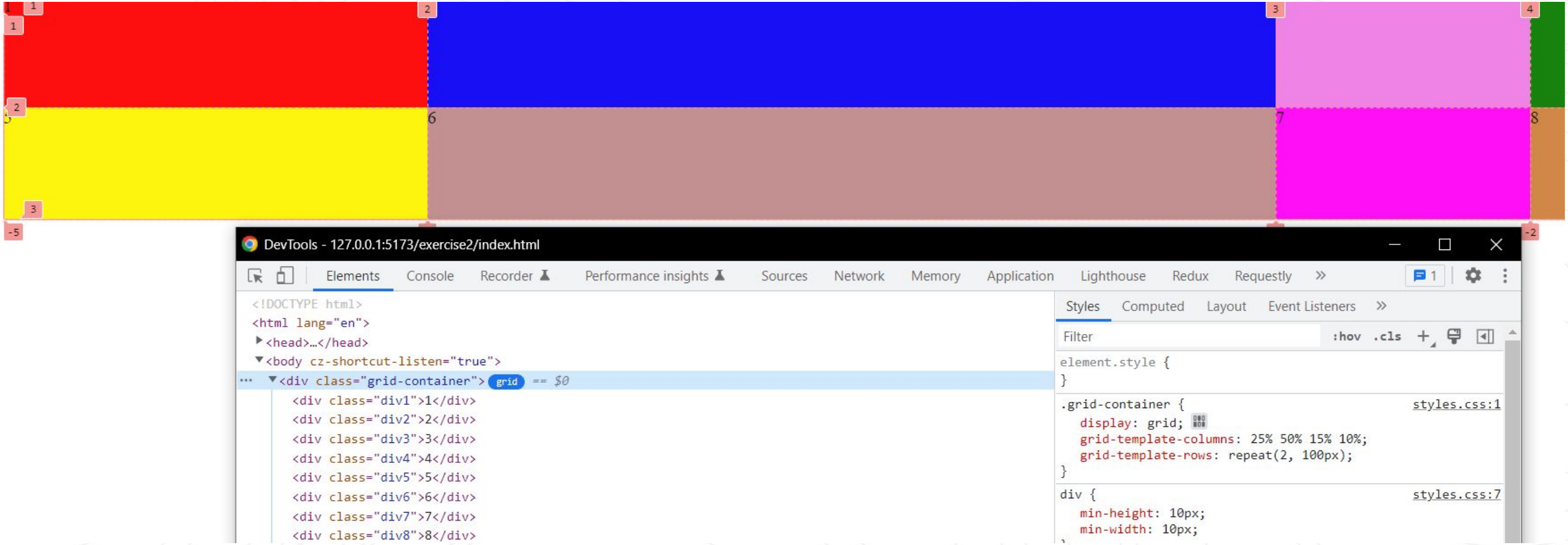
Pozwala zdefiniować szablon wierszy dla grida (używany na poziomie grid kontenera, czyli elementu, na którym definiujemy *display: grid*). Wartości oddzielone spacjami określają kolejne wiersze szablonu. W tym przypadku tworzymy 2 wiersze o wymiarach kolejno 40 i 60 px.



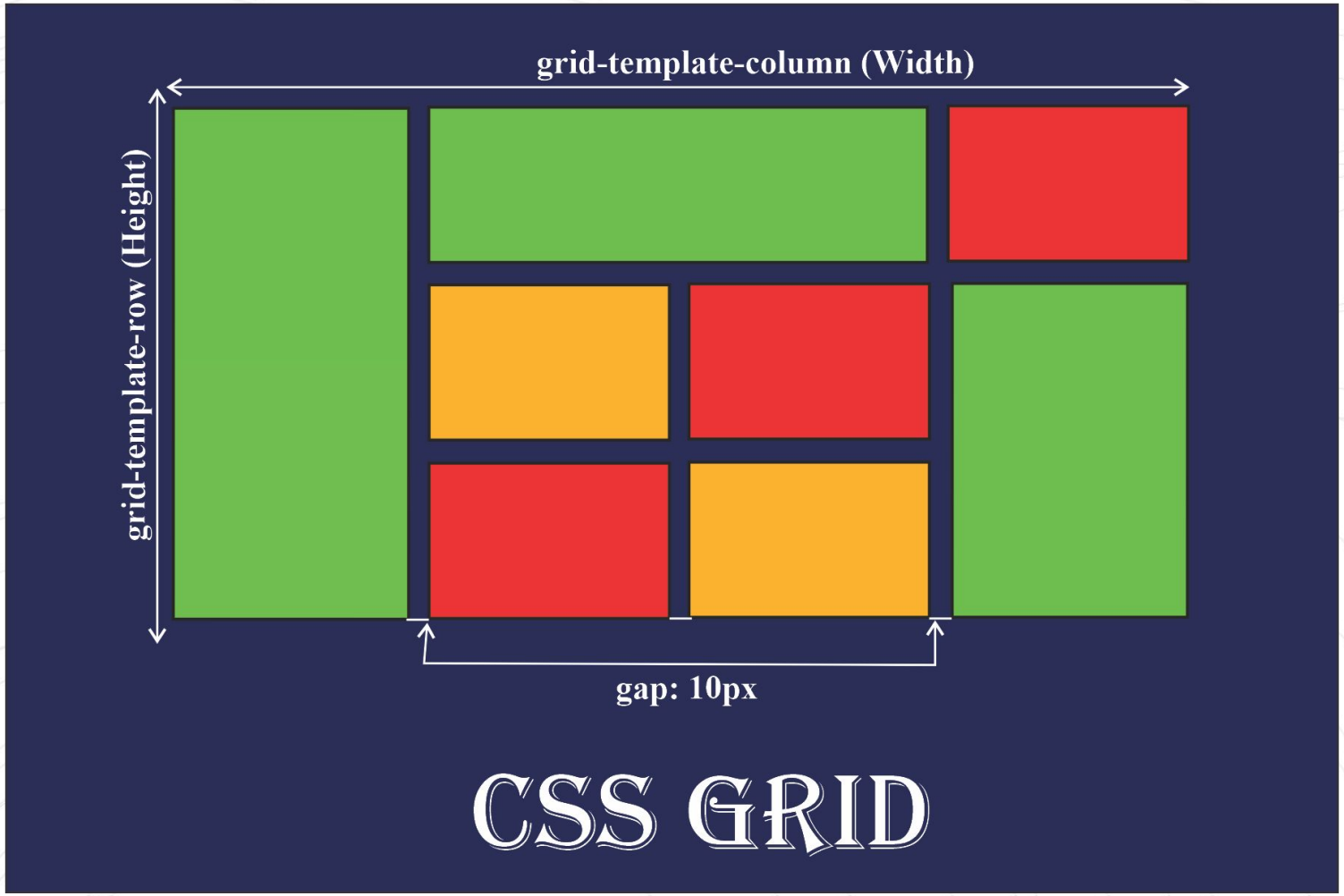
Ćwiczenie nr 1.

1. Uruchom ćwiczenie z folderu **exercise1**
2. W pliku README.md znajduje się instrukcja wykonania zadania
3. Jeżeli uda Ci się wykonać zadanie poinformuj o tym na czacie (idealnie zamieść również screen)
4. Jeżeli po kilku minutach prób masz problem z wykonaniem zadania poinformuj prowadzącego aby otrzymać wskazówkę :)

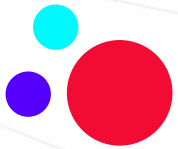
Używanie narzędzi developerskich



Rozmieszczenie elementów w gridzie



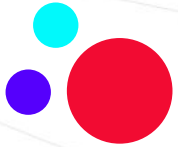
https://www.freecodecamp.org/news/how-to-use-css-grid-layout/



Rozmieszczenie - kolumna

grid-column: [początek] / [koniec]
grid-column: [początek] / span [ile kolumn]

Pozwala umieścić element w konkretnym miejscu grida (zamiast domyślnego, automatycznego rozmieszczenia).



Rozmieszczenie - wiersz

grid-row: [początek] / [koniec]
grid-row: [początek] / span [ile kolumn]

Pozwala umieścić element w konkretnym miejscu grida (zamiast domyślnego, automatycznego rozmieszczenia).

Ćwiczenie nr 2.

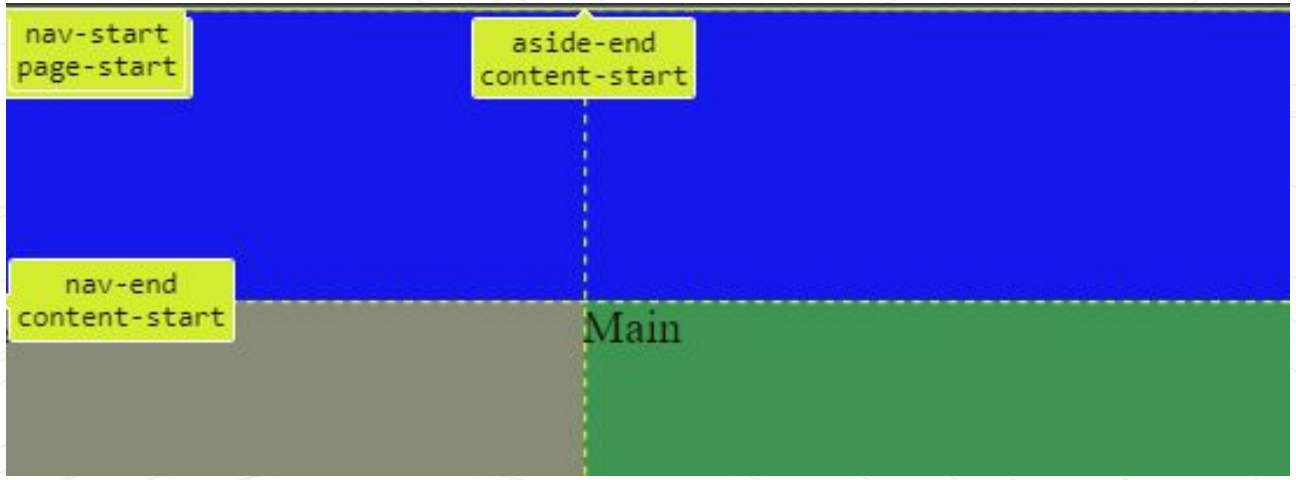
1. Uruchom ćwiczenie z folderu **exercise2**
2. W pliku README.md znajduje się instrukcja wykonania zadania
3. Jeżeli uda Ci się wykonać zadanie poinformuj o tym na czacie (idealnie zamieść również screen)
4. Jeżeli po kilku minutach prób masz problem z wykonaniem zadania poinformuj prowadzącego aby otrzymać wskazówkę :)

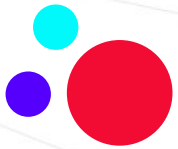
Tworzenie prostego layoutu

Live coding

Nazywanie wierszy i kolumn

CSS Grid umożliwia nam nazywanie poszczególnych linii w wierszach i kolumnach. Dzięki temu możemy tworzyć znacznie czytelniejsze rozwiązania, a sama praca z layoutem staje się znacznie przyjemniejsza.



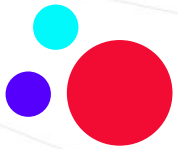


Nazywanie linii - składnia

grid-template-columns: [nazwa1 nazwa2] wymiar
grid-template-rows: [nazwa] wymiar

CSS Grid umożliwia nam nazywanie linii. Robimy to poprzez dodanie własnych nazw przed wymiarem. Przykładowo:

```
grid-template-columns: [page-start nav-start] 150px [page-end nav-end] 200px;
grid-template-rows: [page-start nav-start] 150px [page-end nav-end];
```

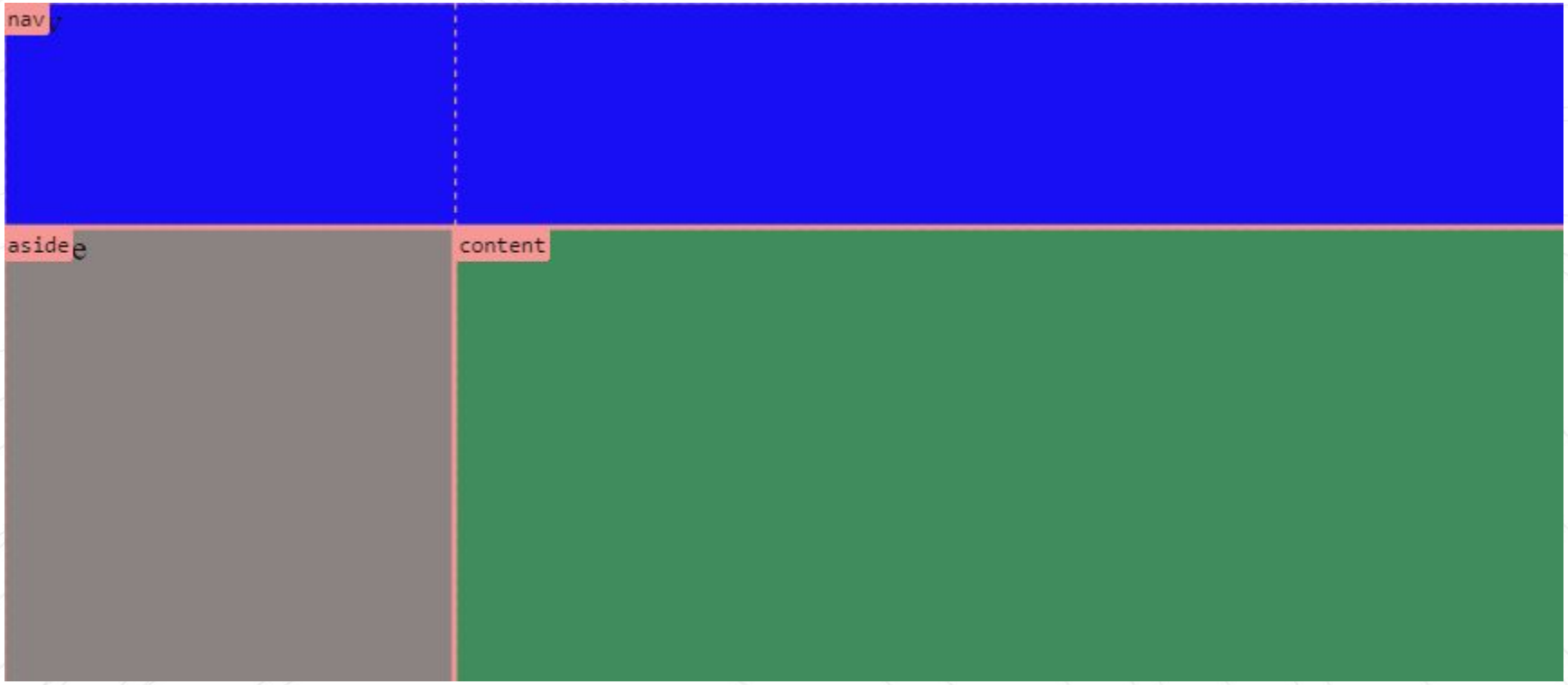


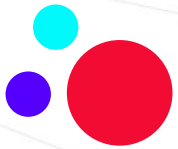
Ćwiczenie nr 3.

1. Uruchom ćwiczenie z folderu **exercise3**
2. W pliku README.md znajduje się instrukcja wykonania zadania
3. Jeżeli uda Ci się wykonać zadanie poinformuj o tym na czacie (idealnie zamieść również screen)
4. Jeżeli po kilku minutach prób masz problem z wykonaniem zadania poinformuj prowadzącego aby otrzymać wskazówkę :)

Grid areas

W CSS Grid oprócz definiowania nazw linii, mamy również możliwość nazywania całych obszarów pomiędzy liniami (w dwóch wymiarach, zarówno wiersze jak i kolumny).

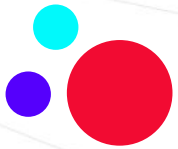




Grid-areas - składnia

```
grid-template-areas: “nav nav”
                     “aside main”
                     “- main”
```

Umożliwia zdefiniowanie nazwa poszczególnych przestrzeni wewnątrz siatki. Zdefiniowane nazwy możemy później przypisać do konkretnego (lub konkretnych) elementów za pomocą **grid-area.**



Grid-area - składnia

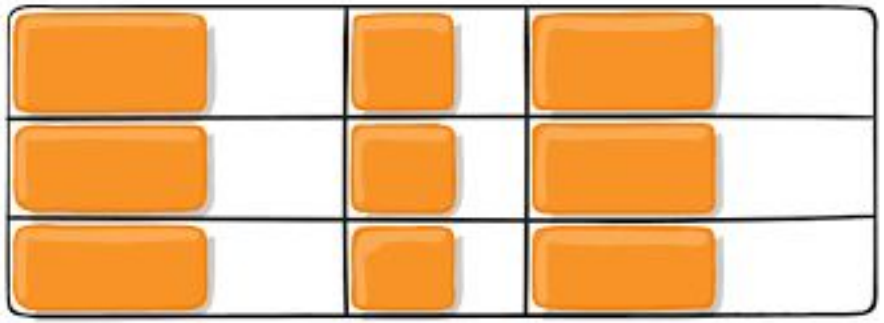
grid-area: nav;

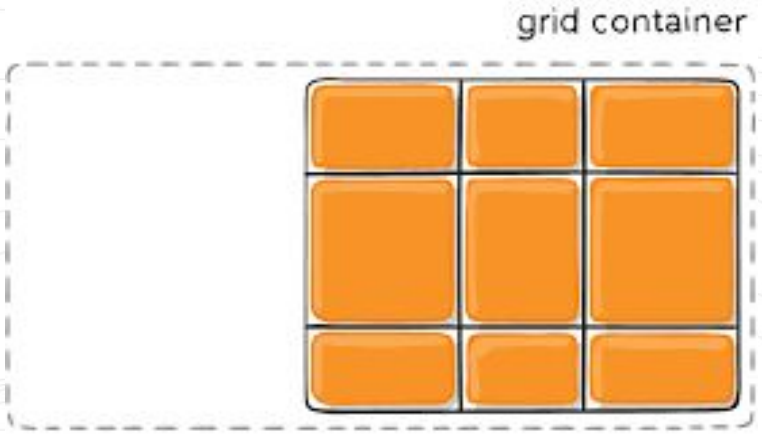
Przypisuje element do konkretnej przestrzeni (w tym przypadku nav).

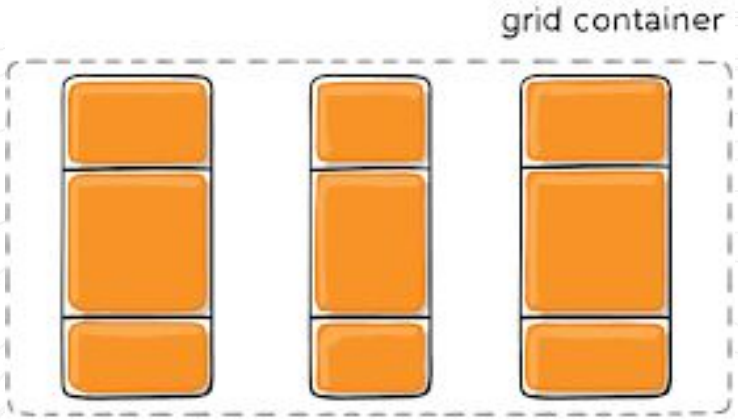
Ćwiczenie nr 4.

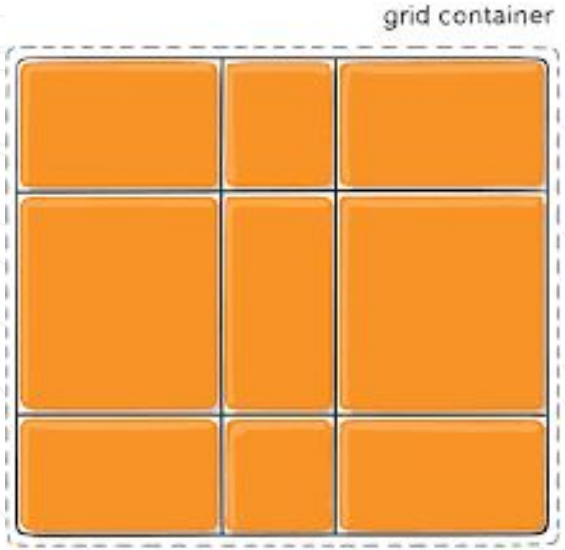
1. Uruchom ćwiczenie z folderu **exercise4**
2. W pliku README.md znajduje się instrukcja wykonania zadania
3. Jeżeli uda Ci się wykonać zadanie poinformuj o tym na czacie (idealnie zamieść również screen)
4. Jeżeli po kilku minutach prób masz problem z wykonaniem zadania poinformuj prowadzącego aby otrzymać wskazówkę :)

Rozmieszczenie elementów wewnątrz siatki

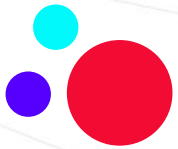








https://css-tricks.com/snippets/css/complete-guide-grid/

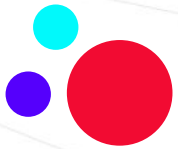


Rozmieszczenie elementów - składnia

justify-items: start;

Wyrównuje elementy siatki w osi poprzecznej (row). Dopuszczalne wartości to:

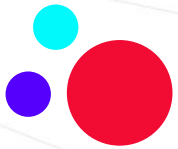
- start - wyrównanie do lewej krawędzi
- end - wyrównanie do prawej krawędzi
- center - wyrównanie do środka
- stretch - rozciągnięcie od lewej do prawej krawędzi



Rozmieszczenie elementów - składnia

gap: value
row-gap: value
column-gap: value

Wprowadza odstęp pomiędzy wierszami / kolumnami w siatce.

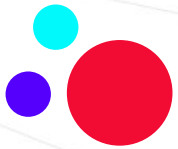


Rozmieszczenie elementów - składnia

justify-self: start;

Wyrównuje element w osi poprzecznej (row). Dopuszczalne wartości to:

- start - wyrównanie do lewej krawędzi
- end - wyrównanie do prawej krawędzi
- center - wyrównanie do środka
- stretch - rozciągnięcie od lewej do prawej krawędzi

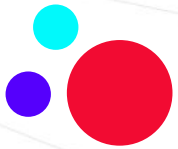


Rozmieszczenie elementów - składnia

align-items: start;

Wyrównuje elementy siatki w osi pionowej (column). Dopuszczalne wartości to:

- start - wyrównanie do górnej krawędzi
- end - wyrównanie do dolnej krawędzi
- center - wyrównanie do środka
- stretch - rozciągnięcie od górnej do dolnej krawędzi
- baseline - wyrównuje elementy do text baseline.

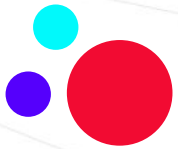


Rozmieszczenie elementów - składnia

align-self: start;

Wyrównuje element siatki w osi pionowej (column). Dopuszczalne wartości to:

- start - wyrównanie do górnej krawędzi
- end - wyrównanie do dolnej krawędzi
- center - wyrównanie do środka
- stretch - rozciągnięcie od górnej do dolnej krawędzi
- baseline - wyrównuje elementy do text baseline.

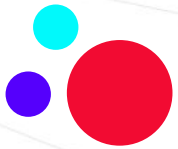


Rozmieszczenie elementów - składnia

justify-content: start;

Wyrównuje elementy siatki w osi pionowej (column). Dopuszczalne wartości to:

- start - wyrównanie do lewej krawędzi kontenera
- end - wyrównanie do prawej krawędzi kontenera
- center - wyrównanie do środka kontenera
- stretch - rozciągnięcie od lewej do prawej krawędzi kontenera
- space-between - rozmieszcza komórki siatki w równomiernych odstępach od siebie (bez odstępu pomiędzy krawędzią a elementem)
- space-around - rozmieszcza komórki siatki w równomiernych odstępach od siebie (z odstępem pomiędzy krawędzią a elementem)
- space-evenly - rozmieszcza komórki siatki w równomiernych odstępach od siebie (z połową standardowego odstępu pomiędzy krawędzią a elementem)



Rozmieszczenie elementów - składnia

align-content: start;

Wyrównuje elementy siatki w osi pionowej (column). Dopuszczalne wartości to:

- start - wyrównanie do górnej krawędzi kontenera
- end - wyrównanie do dolnej krawędzi kontenera
- center - wyrównanie do środka kontenera
- stretch - rozciągnięcie od górnej do dolnej krawędzi kontenera
- space-between - rozmieszcza komórki siatki w równomiernych odstępach od siebie (bez odstępu pomiędzy krawędzią a elementem)
- space-around - rozmieszcza komórki siatki w równomiernych odstępach od siebie (z odstępem pomiędzy krawędzią a elementem)
- space-evenly - rozmieszcza komórki siatki w równomiernych odstępach od siebie (z połową standardowego odstępu pomiędzy krawędzią a elementem)

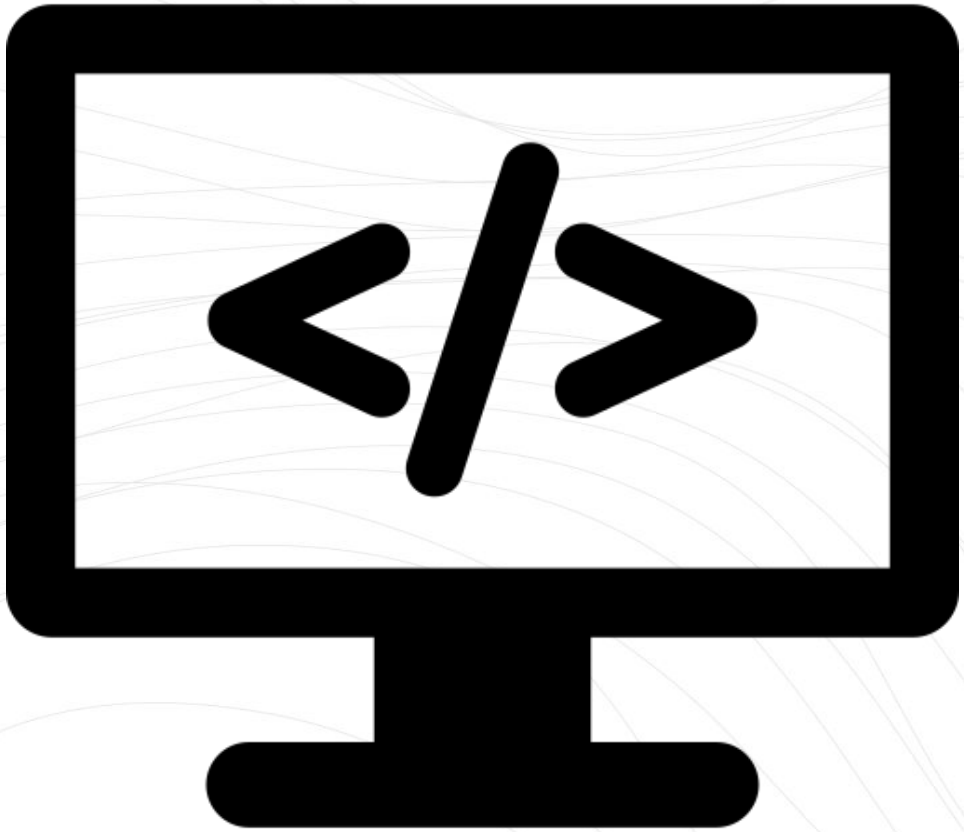
Ćwiczenie nr 5.

1. Uruchom ćwiczenie z folderu **exercise5**
2. W pliku README.md znajduje się instrukcja wykonania zadania
3. Jeżeli uda Ci się wykonać zadanie poinformuj o tym na czacie (idealnie zamieść również screen)
4. Jeżeli po kilku minutach prób masz problem z wykonaniem zadania poinformuj prowadzącego aby otrzymać wskazówkę :)

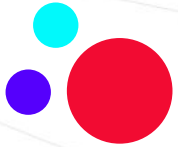
Fluid Grid

CSS Grid umożliwia nam również tworzenie responsywnych layoutów bez używania media queries dzięki zastosowaniu wartości takich jaki **auto-fill, auto-fit** oraz funkcji **repeat.**

Live coding



infoShare ACADEMY

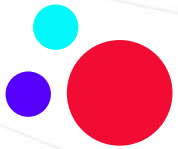


Fluid-grid - składnia

auto-fit

Używając tej wartości w funkcji **repeat**, przeglądarka umieści tak dużo kolumn w jednym rzędzie jak to możliwe. W przypadku gdy wszystkie kolumny zmieszczą się w rzędzie, będą one rozszerzane, tak aby wypełnić całe dostępne miejsce np:

```
grid-template-columns: repeat(auto-fit, minmax(150px, 1fr));
```



Fluid-grid - składnia

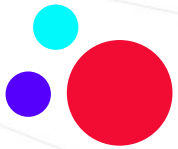
auto-fill

Używając tej wartości w funkcji **repeat**, przeglądarka umieści tak dużo kolumn w jednym rzędzie jak to możliwe. W przypadku gdy wszystkie kolumny zmieszczą się w rzędzie, przeglądarka będzie tworzyć kolejne, puste kolumny, tak aby wypełnić całą dostępną przestrzeń.

```
grid-template-columns: repeat(auto-fill, minmax(150px, 1fr));
```

Ćwiczenie nr 6.

1. Uruchom ćwiczenie z folderu **exercise6**
2. W pliku README.md znajduje się instrukcja wykonania zadania
3. Jeżeli uda Ci się wykonać zadanie poinformuj o tym na czacie (idealnie zamieść również screen)
4. Jeżeli po kilku minutach prób masz problem z wykonaniem zadania poinformuj prowadzącego aby otrzymać wskazówkę :)



Q&A

infoShare ACADEMY

infoShare ACADEMY

Koniec

infoShare Academy

infoShareAcademy.com