ЛАБОРАТОРНА РОБОТА № 8
Варіант 2

Робота з Flexbox в CSS

Мета : Навчитися працювати з технологією Flexbox. Використовувати Flexbox при верстці адаптивних веб-сторінок.

Хід роботи:

Завдання 1:

					ДУ«Житомирська політехніка».22.121.02.000–Лр 8	
Змн.	Арк.	№ докум.	Підпис	Дата		
Розроб.		Маньківський В.			Звіт з лабораторної роботи	Літ. / Арк. 1 / Аркушів 14
Перевір.		Чижмотря О.Г.				
Керівник						ФІКТ Гр. ВТ-21-1[2]
Н. контр.						
Зав. каф.						

HTML:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="task_1.css" />
    <title>Task_1</title>
  </head>
  <body>
    <div>
      <a href="#"><div>Головна</div></a>
      <a href="#"><div>Про компанію</div></a>
      <a href="#"><div>Інформація</div></a>
      <a href="#"><div>Про нас</div></a>
    </div>

    <div class="conteiner_b">
      <a href="#"><div>Головна</div></a>
      <a href="#"><div>Про компанію</div></a>
      <a href="#"><div>Інформація</div></a>
      <a href="#"><div>Про нас</div></a>
    </div>

    <div class="conteiner_c">
      <a href="#"><div>Головна</div></a>
      <a href="#"><div>Про компанію</div></a>
      <a href="#"><div>Інформація</div></a>
      <a href="#"><div>Про нас</div></a>
    </div>

    <div class="conteiner_d">
      <a href="#"><div>Головна</div></a>
      <a href="#"><div>Про компанію</div></a>
      <a href="#"
        ><div>Інформація<br />про те,<br />як до нас<br />доїхати</div></a
      >
      <a href="#"><div>Про нас</div></a>
    </div>

    <div class="conteiner_e">
      <a href="#"><div>Головна</div></a>
      <a href="#"><div>Про компанію</div></a>
      <a href="#"
        ><div>Інформація<br />про те,<br />як до нас<br />доїхати</div></a
      >
      <a href="#"><div>Про нас</div></a>
    </div>
```

```
    <div class="conteiner_f">
      <a href="#"><div>Головна</div></a>
      <a href="#"><div>Про компанію</div></a>
      <a href="#"
        ><div>Інформація<br />про те,<br />як до нас<br />доїхати</div></a
      >
      <a href="#"><div>Про нас</div></a>
    </div>

    <div class="conteiner_g">
      <a href="#"><div>Головна</div></a>
      <a href="#"><div>Про компанію</div></a>
      <a href="#"
        ><div>Інформація<br />про те,<br />як до нас<br />доїхати</div></a
      >
      <a href="#"><div>Про нас</div></a>
    </div>

    <div class="conteiner_h">
      <a href="#"><div>Головна</div></a>
      <a href="#"><div>Про компанію</div></a>
      <a href="#"><div>Інформація про те, як до доїхати</div></a>
      <a href="#"><div>Про нас</div></a>
      <a href="#"><div>Продукт</div></a>
      <a href="#"><div>Наші успіхи</div></a>
      <div class="break"></div>
      <a href="#"><div>Наші працівники</div></a>
      <a href="#"><div>Наша зарплата</div></a>
    </div>

    <div class="conteiner_i">
      <a href="#"><div>Головна</div></a>
      <a href="#"><div>Про компанію</div></a>
      <a href="#"><div>Інформація</div></a>
      <a href="#"><div>Про нас</div></a>
    </div>
  </body>
</html>
```

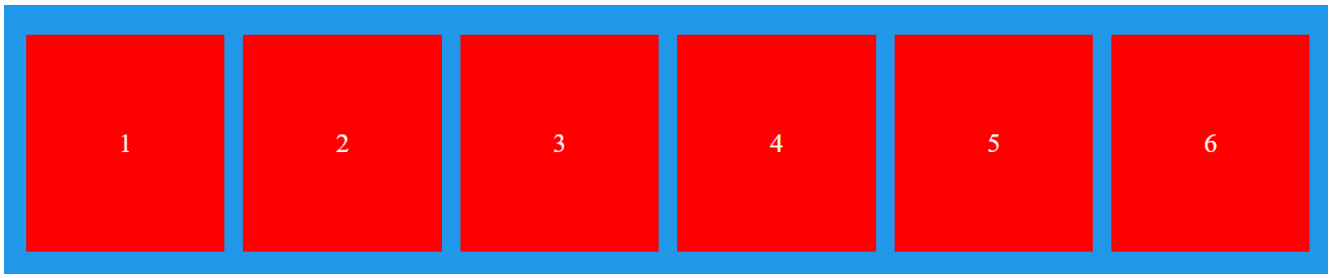
CSS:

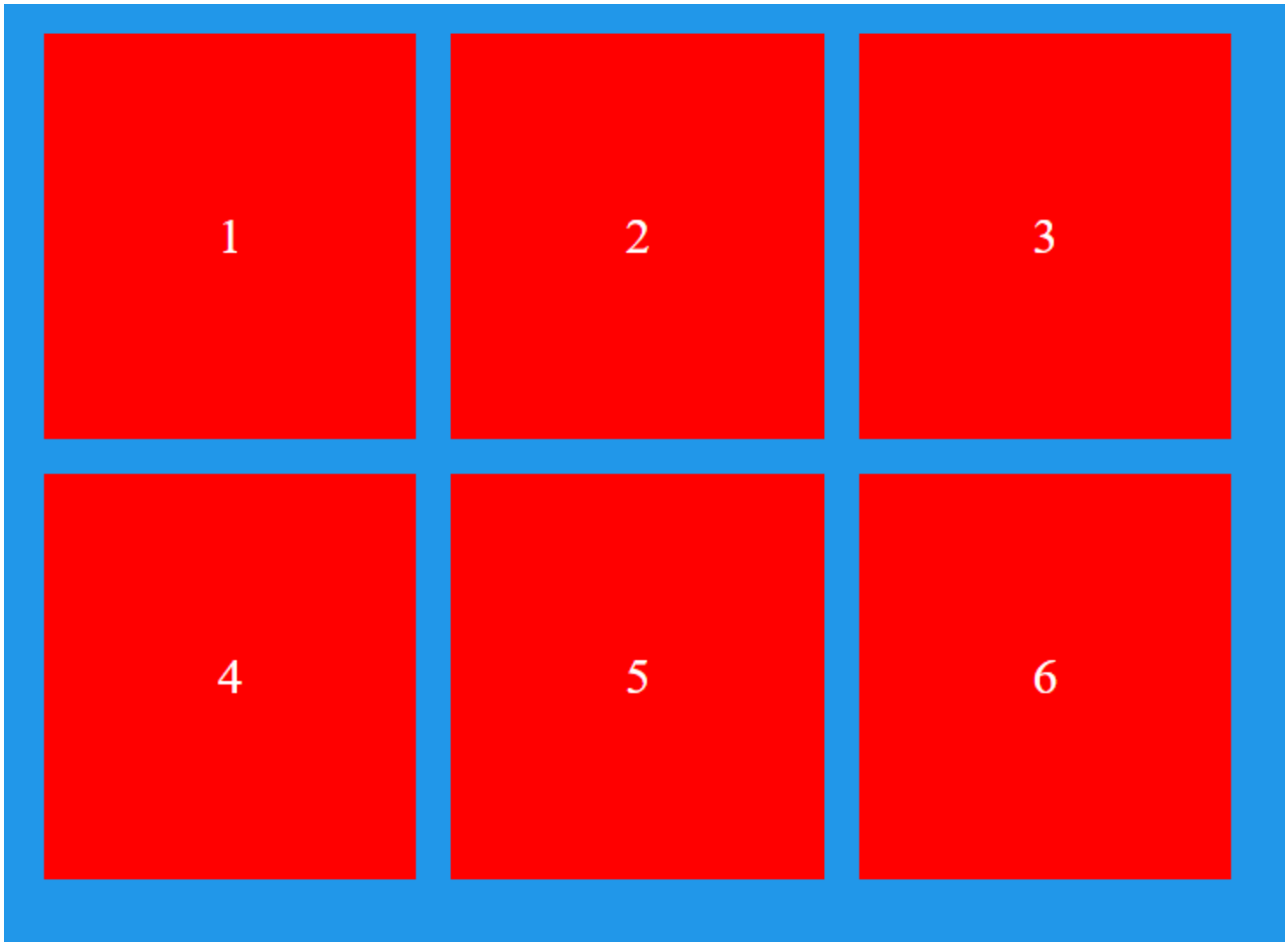
```
body {
  background-color: #2197e9;
  font-size: 1.3em;
}
a {
  display: flex;
  text-decoration: none;
```

```
  color: white;
}
body > div {
  display: flex;
  border: 1.5px solid black;
  justify-content: space-between;
  margin-bottom: 15px;
}

body > div div {
  padding: 10px;
  transition: all 0.5s;
}
body > div div:hover {
  color: #2197e9;
  background-color: white;
}

.conteiner_b {
  direction: rtl;
}

.conteiner_c {
  justify-content: center;
}

.conteiner_d {
  justify-content: center;
  text-align: center;
}
.conteiner_d div {
  border: 1.5px solid black;
}

.conteiner_e {
  align-items: center;
  justify-content: center;
  text-align: center;
}
.conteiner_e div {
  border: 1.5px solid black;
}

.conteiner_f {
  align-items: center;
  justify-content: flex-end;
  text-align: center;
}
.conteiner_f div {
  border: 1.5px solid black;
```

```
}

.conteiner_g {
  flex-direction: row-reverse;
  align-items: center;
  justify-content: flex-start;
  text-align: center;
}
.conteiner_g div {
  border: 1.5px solid black;
}

.conteiner_h {
  flex-wrap: wrap;
  align-items: center;
  justify-content: center;
  text-align: center;
}

.break {
  flex-basis: 100%;
  height: 0;
  visibility: hidden;
}

.conteiner_i {
  flex-direction: column;
}
.conteiner_i div {
  flex-basis: 20%;
  border: 1.5px solid black;
}
```

Завдання 2:





HTML:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="task_2.css" />
    <title>Task_2</title>
  </head>
  <body>
    <ul class="flex-container">
      <li class="flex-item">1</li>
      <li class="flex-item">2</li>
      <li class="flex-item">3</li>
      <li class="flex-item">4</li>
      <li class="flex-item">5</li>
      <li class="flex-item">6</li>
    </ul>
  </body>
</html>
```

CSS:

```css
body {
  background-color: #2197e9;
  font-size: 1.8em;
  color: white;
}

li {
  background-color: red;
  padding: 100px;
  margin: 10px;
  list-style-type: none;
}

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

Завдання 3:

HTML:

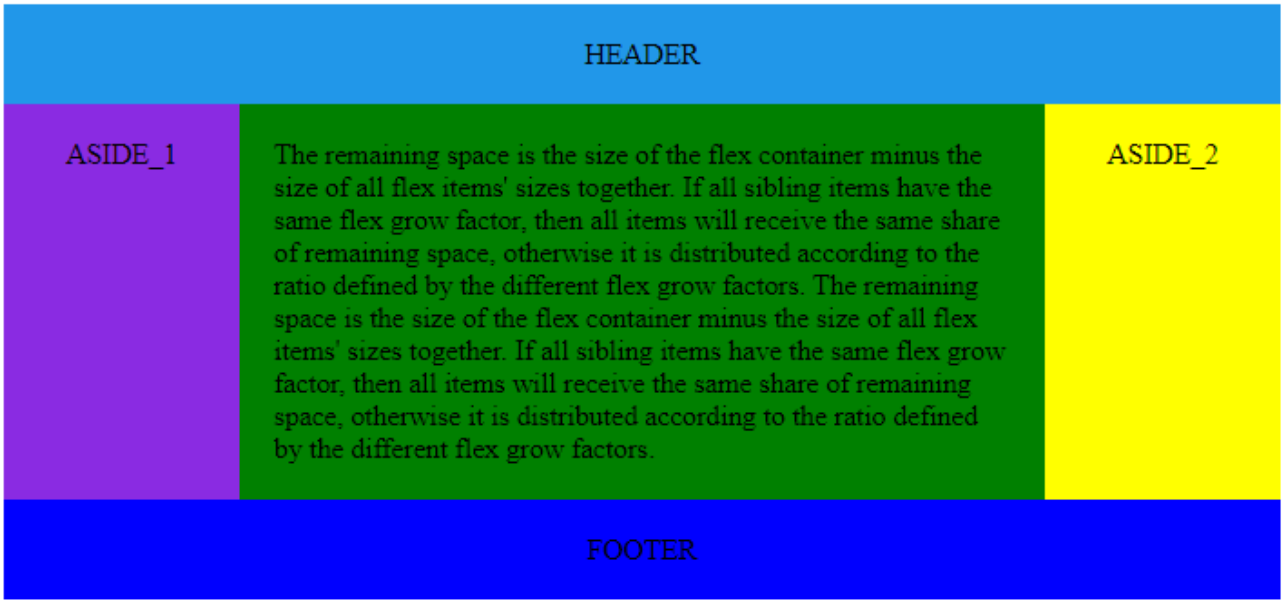
```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=\, initial-scale=1.0" />
    <link rel="stylesheet" href="task_3.css" />
    <title>Task_3</title>
  </head>
  <body>
    <div></div>
    <div class="conteiner">
      <a href="#"><div>Home</div></a>
      <a href="#"><div>About</div></a>
      <a href="#"><div>Products</div></a>
      <a href="#"><div>Contact</div></a>
    </div>
```

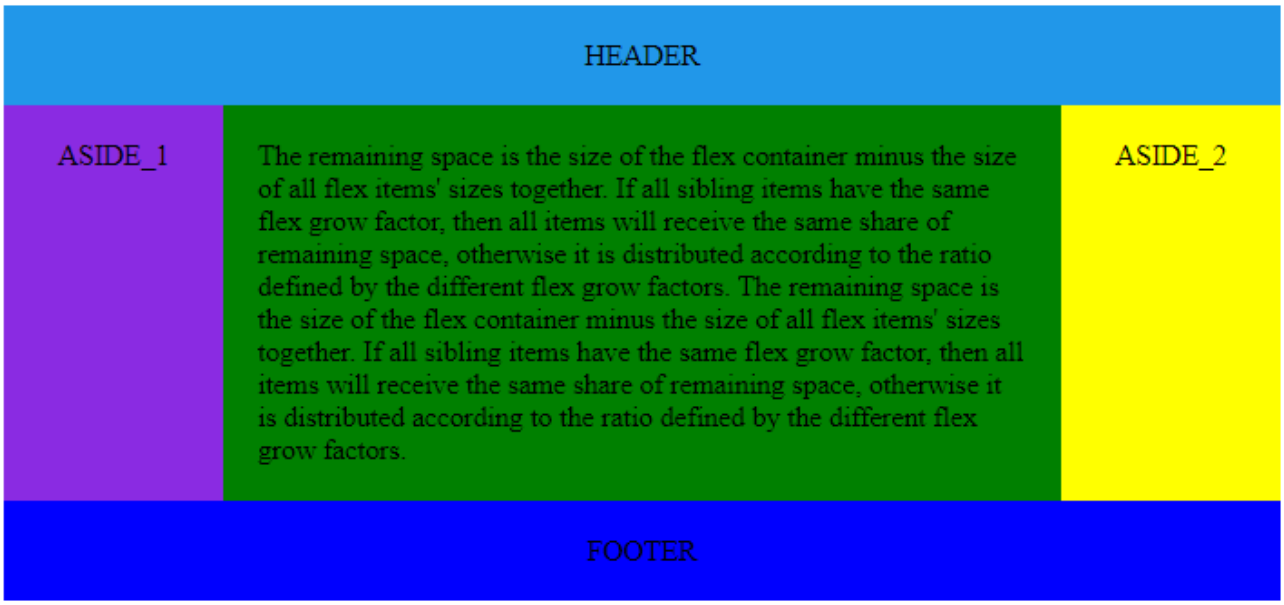
```
  </body>
</html>
```

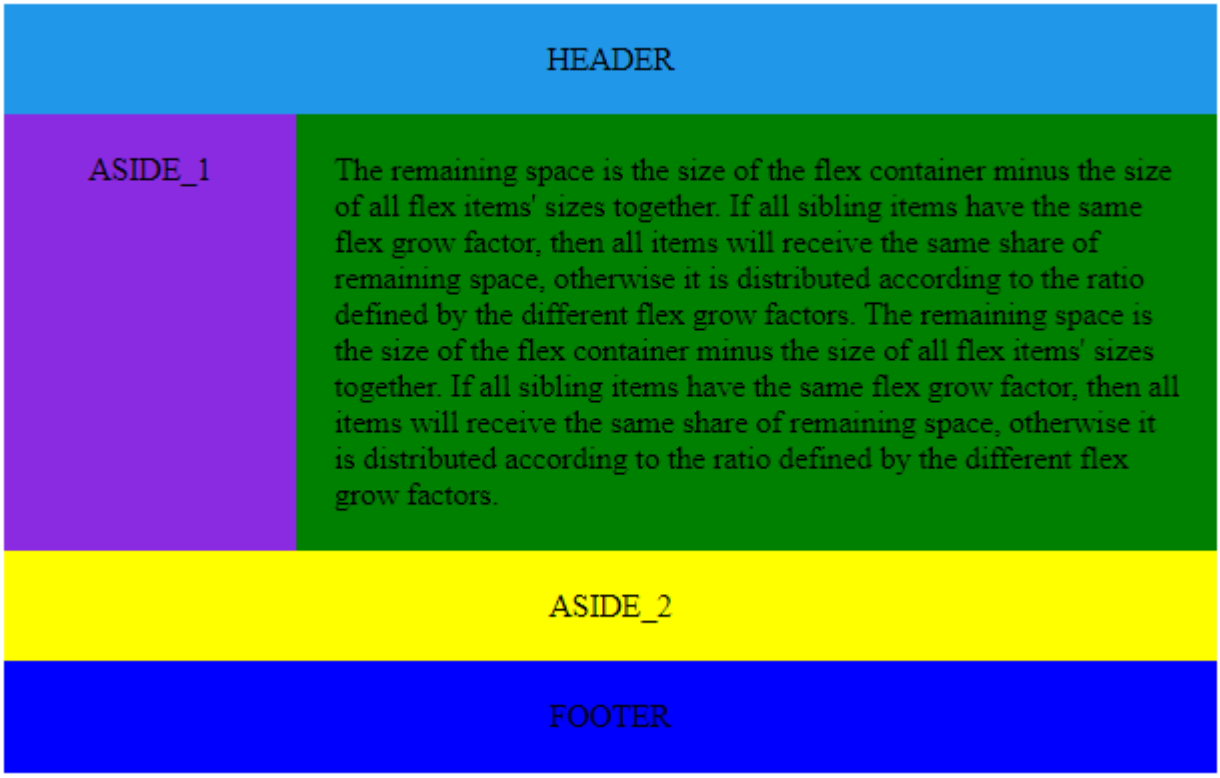
CSS:

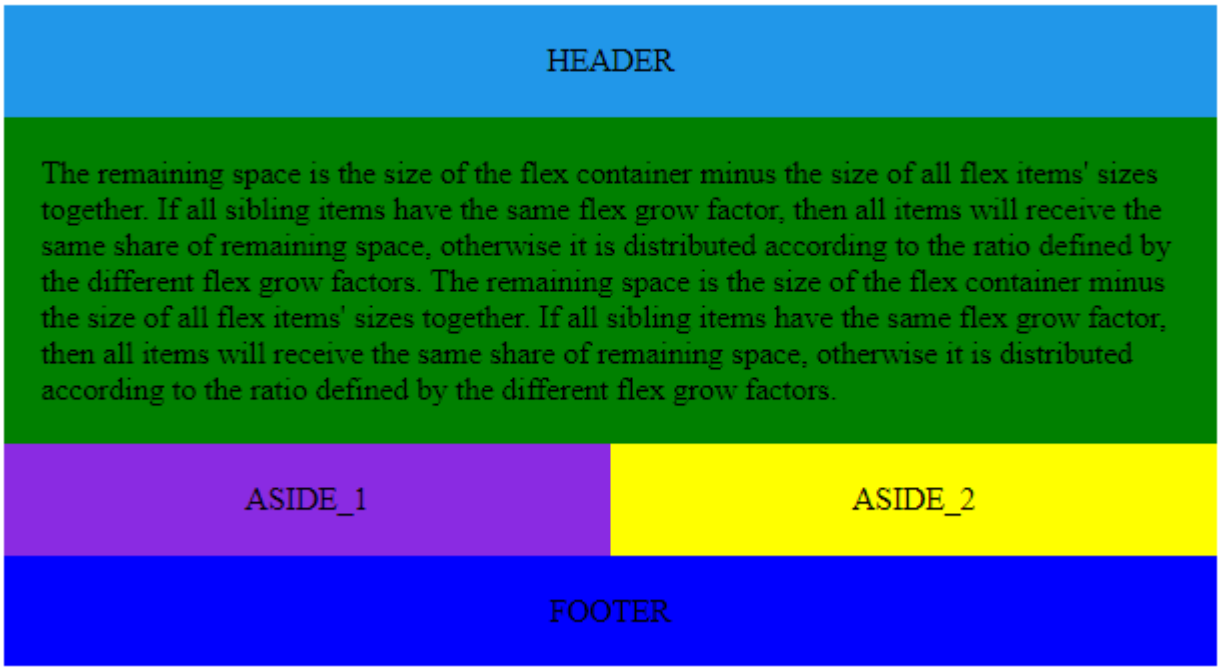
```
body {
  display: flex;
  flex-wrap: wrap;
  background-color: #2197e9;
  font-size: 1.5em;
  text-align: center;
}
body > div {
  width: 1000px;
  flex-grow: 1;
}
a {
  display: flex;
  text-decoration: none;
  color: white;
  flex-grow: 1;
}
.conteiner {
  width: 600px;
  display: flex;
  justify-content: space-around;
  flex-wrap: wrap;
}

.conteiner div {
  padding: 10px;
  transition: all 0.5s;
  flex-grow: 1;
}
.conteiner div:hover {
  color: #2197e9;
  background-color: white;
}
```

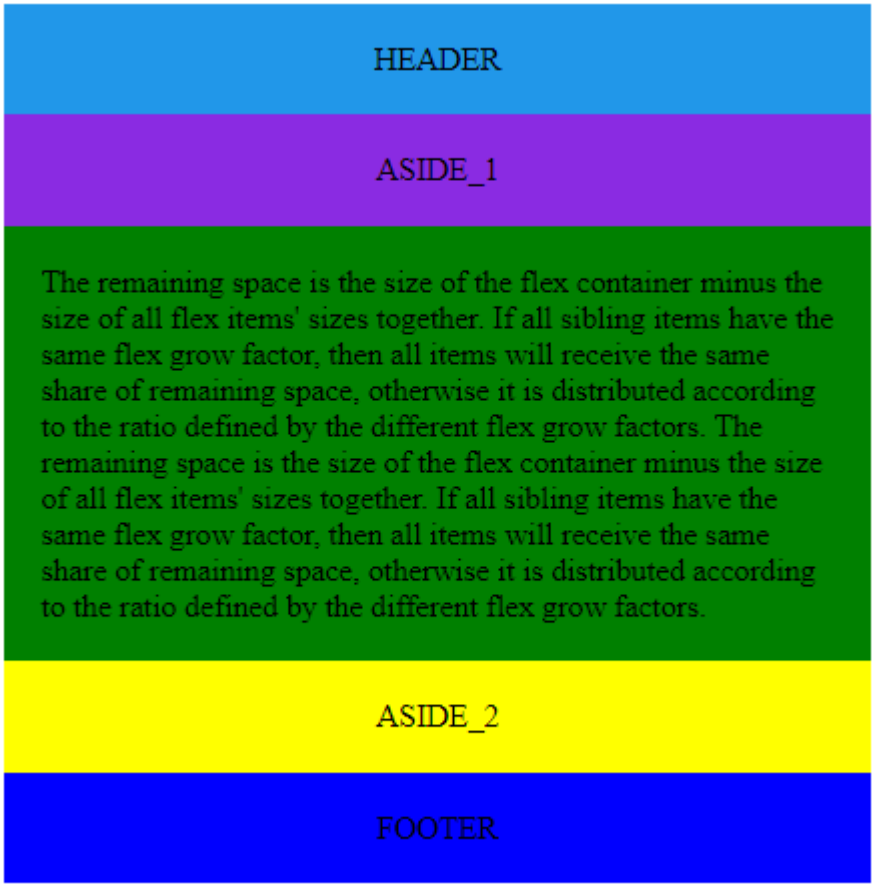
Завдання 5:

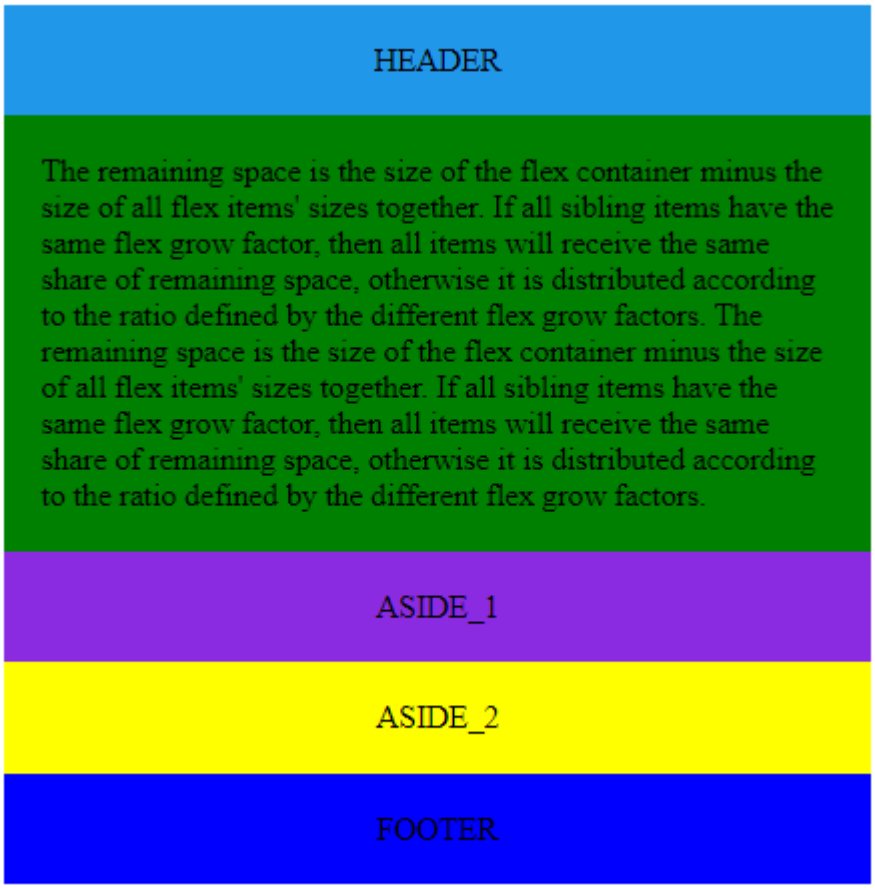

HEADER
ASIDE_1
The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors. The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors.
ASIDE_2
FOOTER


HEADER
ASIDE_1
The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors. The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors.
ASIDE_2
FOOTER


HEADER
ASIDE_1
The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors. The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors.
ASIDE_2
FOOTER


HEADER
The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors. The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors.
ASIDE_1
ASIDE_2
FOOTER


HEADER
ASIDE_1
The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors. The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors.
ASIDE_2
FOOTER


HEADER
The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors. The remaining space is the size of the flex container minus the size of all flex items' sizes together. If all sibling items have the same flex grow factor, then all items will receive the same share of remaining space, otherwise it is distributed according to the ratio defined by the different flex grow factors.
ASIDE_1
ASIDE_2
FOOTER

HTML:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="task_4.css" />
    <title>Task_4</title>
  </head>
  <body>
    <div class="flexbox">
      <div>HEADER</div>
      <div>ASIDE_1</div>
      <div>
        The remaining space is the size of the flex container minus the size of
        all flex items' sizes together. If all sibling items have the same flex
        grow factor, then all items will receive the same share of remaining
        space, otherwise it is distributed according to the ratio defined by the
        different flex grow factors. The remaining space is the size of the flex
        container minus the size of all flex items' sizes together. If all
        sibling items have the same flex grow factor, then all items will
        receive the same share of remaining space, otherwise it is distributed
        according to the ratio defined by the different flex grow factors.
      </div>
      <div>ASIDE_2</div>
      <div>FOOTER</div>
    </div>

    <div class="media">
      <div>HEADER</div>
      <div>ASIDE_1</div>
      <div>
        The remaining space is the size of the flex container minus the size of
        all flex items' sizes together. If all sibling items have the same flex
        grow factor, then all items will receive the same share of remaining
        space, otherwise it is distributed according to the ratio defined by the
        different flex grow factors. The remaining space is the size of the flex
        container minus the size of all flex items' sizes together. If all
        sibling items have the same flex grow factor, then all items will
        receive the same share of remaining space, otherwise it is distributed
        according to the ratio defined by the different flex grow factors.
      </div>
      <div>ASIDE_2</div>
      <div>FOOTER</div>
    </div>
  </body>
</html>
```

CSS:

```css
body {
  font-size: 1.3em;
  text-align: center;
}
.flexbox {
  display: flex;
  flex-wrap: wrap;
}
div {
  padding: 25px;
}
.flexbox div:nth-child(1) {
  width: 100%;
  background-color: #2197e9;
}
.flexbox div:nth-child(2) {
  background-color: blueviolet;
  flex-grow: 1;
}
.flexbox div:nth-child(3) {
  background-color: green;
  text-align: left;
  flex-grow: 1;
  width: 500px;
}
.flexbox div:nth-child(4) {
  background-color: yellow;
  flex-grow: 1;
}
.flexbox div:nth-child(5) {
  width: 100%;
  background-color: blue;
}

.media {
  display: grid;
  grid-template-columns: 0.5fr 0.95fr 0.95fr 0.5fr;
}
.media div:nth-child(1) {
  background-color: #2197e9;
  grid-column-start: span 4;
  order:1;
}
.media div:nth-child(2) {
  background-color: blueviolet;
  order:2;
}
.media div:nth-child(3) {
```

```
  background-color: green;
  grid-column-start: span 2;
  text-align: left;
  order:3;
}
.media div:nth-child(4) {
  background-color: yellow;
  order:4;
}
.media div:nth-child(5) {
  background-color: blue;
  grid-column-start: span 4;
  order:5;
}

@media only screen and (max-width: 900px) {
  .media div:nth-child(3) {
    grid-column-start: span 4;
    order:2;
  }
  .media div:nth-child(2) {
    grid-column-start: span 2;
    order:3;
  }
  .media div:nth-child(4) {
    grid-column-start: span 2;
  }
}
@media only screen and (max-width: 765px) {
  .media div:nth-child(2) {
    grid-column-start: span 4;
  }
  .media div:nth-child(4) {
    grid-column-start: span 4;
  }
}
```

Висновки: я навчився працювати з технологією Flexbox. Використовувати Flexbox при верстці адаптивних веб-сторінок.