|  |
| --- |
| **ЗВІТ**  лабораторної работи №3  з курсу « **OсновиWEB/Основи front-end технологій** »  **Тема: Блочна верстка макету сайту за допомогою HTML і CSS** |

|  |  |
| --- | --- |
| Перевірив:  Доц. Голубєв Л. П. | Виконав:  Гр. ЗПІ-ЗП-11  Запорожец Віталій |

**Завдання 1.**

Макет структури сайту виконано в PhotoShop. За допомогою **HTML5** та **CSS3** зробити блочну верстку розробленогомакету використовуючи технологію **float**. Обов’язково використати **сontainer** та **width** у відсотках.

**Завдання 2.**

Створити верстку цього ж макету сайту за допомогою технології **Flex.**

**Хід роботи**

За допомогою текстового редактора Notepad ++, було створено структуру WEB-сторінки мовою HTML-5.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Лабораторна робота №3</title>

<link rel="stylesheet" type="text/css" href="css/style.css">

</head>

<body>

<!-- FLOAT -->

<div class="container"> <!-- створили контейнер з фіксованою ширину, де будуть розміщені наші блоки -->

<div class="header"> <!-- умовна шапка на всю ширину -->

FLOAT

</div>

<div class="img-left"> <!-- контентний блок, який розташований зліва -->

IMG

</div>

<div class="text-right"> <!-- контентний блок, який розташований справа -->

TEXT

</div>

<div class="clear"></div> <!-- службовий тег, якому задається стиль clear: both; для того щоб нижні блок не налазив на верхні блоки у яких є стан float -->

<div class="footer"> <!-- умовний футер на всю ширину -->

IMG

</div>

<div class="footer"> <!-- умовний футер на всю ширину -->

TEXT

</div>

</div>

<!-- FLEX -->

<div class="container-flex container"> <!-- створили контейнер з фіксованою ширину, де будуть розміщені наші блоки і задали блоку display:flex; -->

<div class="header-flex"> <!-- умовна шапка на всю ширину -->

FLEX

</div>

<div class="row-flex-3 flex-left"> <!-- контентний блок, який розташований зліва -->

IMG

</div>

<div class="text-right"> <!-- контентний блок, який розташований -->

TEXT

</div>

<div class="footer-flex"> <!-- умовний футер на всю ширину -->

IMG

</div>

<div class="footer-flex"> <!-- умовний футер на всю ширину -->

TEXT

</div>

</div>

</body>

</html>

.container {

width: 900px; /\* задали фіксовану ширину для контейнера як у прикладі в документі \*/

margin-left: auto; /\* це означає що блок відштовхується зліва і починає прилягати справа \*/

margin-right: auto; /\* це означає що блок відштовхується справа і починає прилягати зліва \*/

/\* коли margin-left: auto; і margin-right: auto; разом то блок фіксується, це працює тільки для блочних елементів, для інлайнових не працює \*/

margin-top: 30px; /\* відступ зверху 30 пікселів \*/

margin-bottom: 30px; /\* відступ знизу 30 пікселів \*/

}

/\* FLOAT \*/

.clear {

clear: both; /\* це задається тоді, коли нам треба щоб нижні блок не обгортав верхні блоки у яких є float \*/

}

.header { /\* шапка у завданні для флоат \*/

width: 100%; /\* блок займає усю ширину \*/

background-color: #74D7F3; /\* фон згідно макету \*/

text-align: center; /\* текст позиціонуємо по середині \*/

padding: 10px; /\* внутрішній відступ 10 пікселів, щоб текст не прилягав до країв згідно макету \*/

box-sizing: border-box; /\* це означає що внутрішній відступ враховується у ширину, тобто внутрішній відступ 10 пікселів враховується у ширину блоку 100% і не збільшує його ще на 10 пікслеів \*/

}

.footer { /\* футер у завданні для флоат \*/

width: 100%; /\* блок займає усю ширину \*/

background-color: #84CDF0; /\* фон згідно макету \*/

text-align: center; /\* текст позиціонуємо по середині \*/

padding: 10px; /\* внутрішній відступ 10 пікселів, щоб текст не прилягав до країв згідно макету \*/

box-sizing: border-box; /\* це означає що внутрішній відступ враховується у ширину, тобто внутрішній відступ 10 пікселів враховується у ширину блоку 100% і не збільшує його ще на 10 пікслеів \*/

}

.img-left { /\* блок зліва \*/

background-color: #75798A; /\* фон згідно макету \*/

padding: 10px; /\* внутрішній відступ 10 пікселів, щоб текст не прилягав до країв згідно макету \*/

box-sizing: border-box; /\* це означає що внутрішній відступ враховується у ширину, тобто внутрішній відступ 10 пікселів враховується у ширину блоку 100% і не збільшує його ще на 10 пікслеів \*/

text-align: center; /\* текст позиціонуємо по середині \*/

height: 600px; /\* фіксована висоту блоку приблизно згідно макету \*/

float: left; /\* наш блок починає прилягати до лівої сторони і стає плавучим для сусідніх блоків, які його оточують \*/

width: calc(100% / 3); /\* ширина у відсотках, щоб блок займав рівно третину ширини контейнера, calc() це математична функція яка рахує всереді себе і вертає значення, у даному випадку 33.33....% \*/

}

.img-right { /\* блок справа \*/

background-color: #15278A; /\* фон згідно макету \*/

padding: 10px; /\* внутрішній відступ 10 пікселів, щоб текст не прилягав до країв згідно макету \*/

box-sizing: border-box; /\* це означає що внутрішній відступ враховується у ширину, тобто внутрішній відступ 10 пікселів враховується у ширину блоку 100% і не збільшує його ще на 10 пікслеів \*/

text-align: center; /\* текст позиціонуємо по середині \*/

height: 600px; /\* фіксована висоту блоку приблизно згідно макету \*/

width: calc(100% / 3); /\* ширина у відсотках, щоб блок займав рівно третину ширини контейнера, calc() це математична функція яка рахує всереді себе і вертає значення, у даному випадку 33.33....% \*/

float: right; /\* наш блок починає прилягати до правої сторони і стає плавучим для сусідніх блоків, які його оточують \*/

}

.text-right { /\* блок справа \*/

background-color: #6CCDF0; /\* фон згідно макету \*/

padding: 10px; /\* внутрішній відступ 10 пікселів, щоб текст не прилягав до країв згідно макету \*/

box-sizing: border-box; /\* це означає що внутрішній відступ враховується у ширину, тобто внутрішній відступ 10 пікселів враховується у ширину блоку 100% і не збільшує його ще на 10 пікслеів \*/

text-align: center; /\* текст позиціонуємо по середині \*/

height: 600px; /\* фіксована висоту блоку приблизно згідно макету \*/

width: calc(100%/1.5); /\* ширина у відсотках, щоб блок займав рівно третину ширини контейнера, calc() це математична функція яка рахує всереді себе і вертає значення, у даному випадку 33.33....% \*/

float: right; /\* наш блок починає прилягати до правої сторони і стає плавучим для сусідніх блоків, які його оточують \*/

}

/\* FLEX \*/

.container-flex {

display: flex; /\* наш контейнер стає flex і всі його діти стають флекс-елементами і починають піддаватися правилам флексів \*/

flex-wrap: wrap; /\* означає, що флекс елементи можуть переноситися на новий рядок, якщо вони не вміщаються у відповідному рядку \*/

}

.header-flex {

width: 100%; /\* блок займає усю ширину \*/

background-color: #74D7F3; /\* фон згідно макету \*/

text-align: center; /\* текст позиціонуємо по середині \*/

padding: 10px; /\* внутрішній відступ 10 пікселів, щоб текст не прилягав до країв згідно макету \*/

box-sizing: border-box; /\* це означає що внутрішній відступ враховується у ширину, тобто внутрішній відступ 10 пікселів враховується у ширину блоку 100% і не збільшує його ще на 10 пікслеів \*/

}

.row-flex-3 {

padding: 10px; /\* внутрішній відступ 10 пікселів, щоб текст не прилягав до країв згідно макету \*/

box-sizing: border-box; /\* це означає що внутрішній відступ враховується у ширину, тобто внутрішній відступ 10 пікселів враховується у ширину блоку 100% і не збільшує його ще на 10 пікслеів \*/

text-align: center; /\* текст позиціонуємо по середині \*/

height: 600px; /\* фіксована висоту блоку приблизно згідно макету \*/

width: calc(100% / 3); /\* ширина у відсотках, щоб блок займав рівно третину ширини контейнера, calc() це математична функція яка рахує всереді себе і вертає значення, у даному випадку 33.33....% \*/

}

.footer-flex {

width: 100%; /\* блок займає усю ширину \*/

background-color: #F8E2B0; /\* фон згідно макету \*/

text-align: center; /\* текст позиціонуємо по середині \*/

padding: 10px; /\* внутрішній відступ 10 пікселів, щоб текст не прилягав до країв згідно макету \*/

box-sizing: border-box; /\* це означає що внутрішній відступ враховується у ширину, тобто внутрішній відступ 10 пікселів враховується у ширину блоку 100% і не збільшує його ще на 10 пікслеів \*/

}

.flex-left {

background-color: #75798A; /\* фон згідно макету \*/

}

.flex-center {

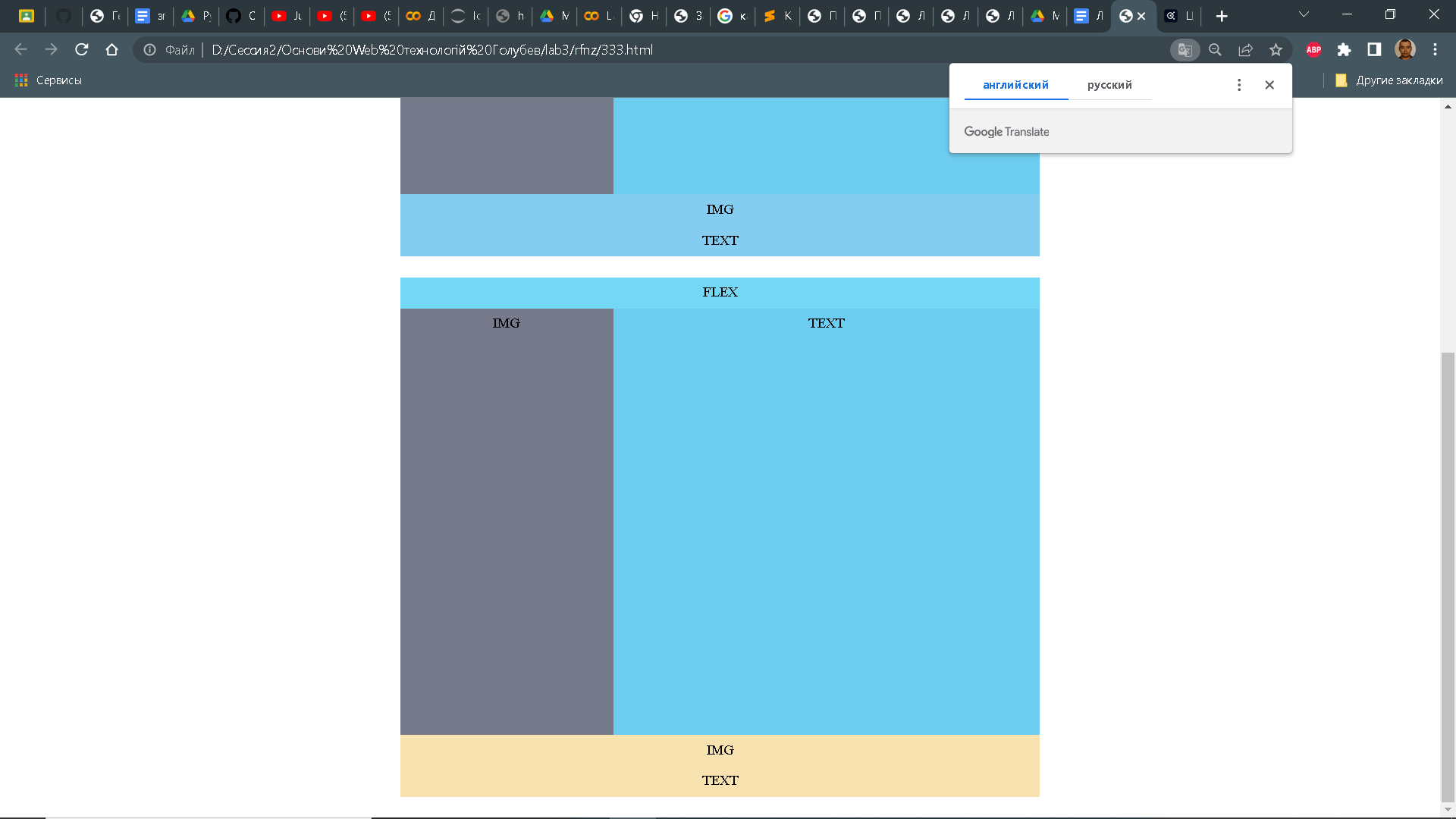
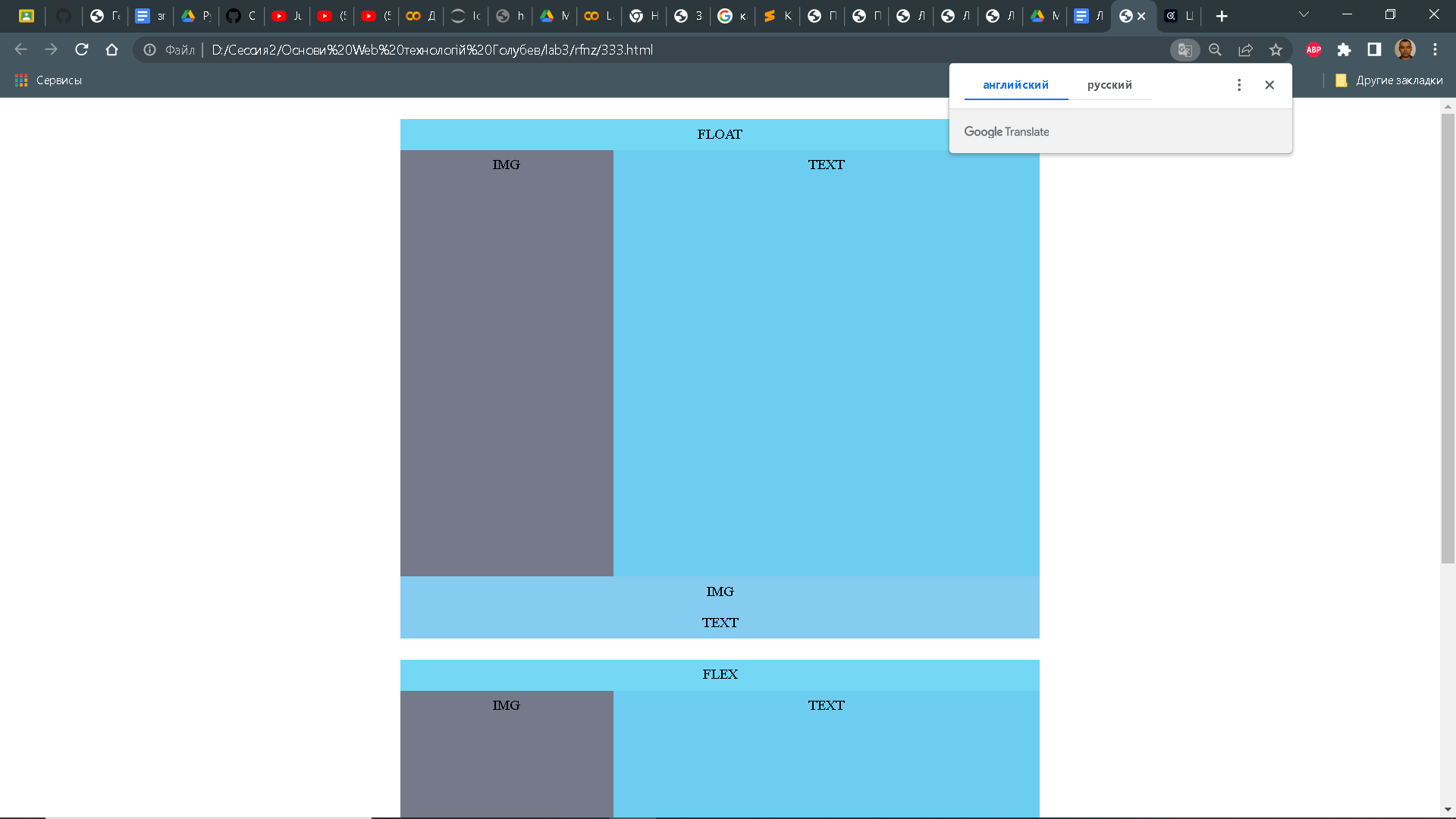
background-color: #F7F894; /\* фон згідно макету \*/

}

.flex-right {

background-color: #F094F8; /\* фон згідно макету \*/

}



**Висновок**

Під час виконання даної лабораторної роботи я оволодів навичками блочної верстки макету сайту за допомогою HTML і CSS. Всі завдання лабораторної роботи були успішно виконані та всі результати представлені у звіті.

.

Посилання на WEB-сторінку:  [**https://github.com/Vitalik2022/WEB**](%20https://github.com/Vitalik2022/WEB)