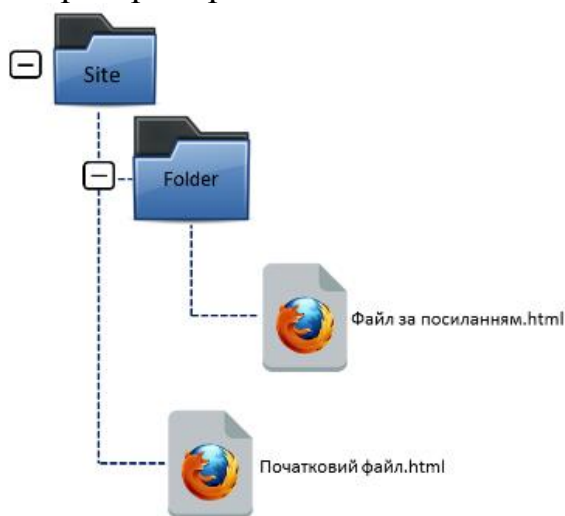


ЛАБОРАТОРНА РОБОТА №2

Основні теги HTML

Практична частина

1. В обраному редакторі коду (на вибір студента) створити новий проєкт.
2. Організувати систему файлів та папок, згідно наведеного рисунку, на якому верхній рівень – це директорія для цілого проєкту лабораторної роботи №2, в якій міститься:
 - index.html – робочий файл проєкту (головна сторінка сайту);
 - директорія img – для збереження картинок та іконок;
 - директорія audio – для збереження аудіо-файлів;
 - директорія project_parts – для збереження html-сторінок кожного завдання лабораторної роботи №2.



Структура index.html:

header

main

footer

- В шапці (тег header) відобразити Заголовок 1-го рівня із текстом: «Лабораторна робота №2 з web-програмування», вирівнювання тексту по центру.
- Контентну частину (тег main) реалізувати згідно зразка із використанням заданих тегів:
 1. кожне посилання – це html-сторінка відповідного індивідуального завдання лабораторної роботи №2.
 2. кожне завдання відкривати в новому вікні браузера.
 3. кегель гіперпосилань – 16px.



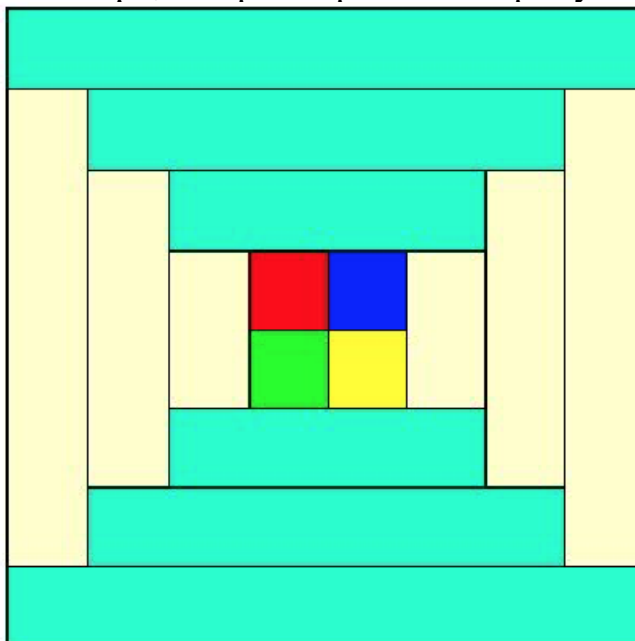
- В підвалі (тег footer) відобразити Заголовок 4-го рівня із текстом: ПІБ студента-виконавця».

Структура .html-сторінок індивідуальних завдань:

- Кожен елемент індивідуального завдання повинен бути розташований по центру вікна браузера.
- На кожній html-сторінці додати гіперпосилання з текстом «Повернутись на головну».

Завдання №1

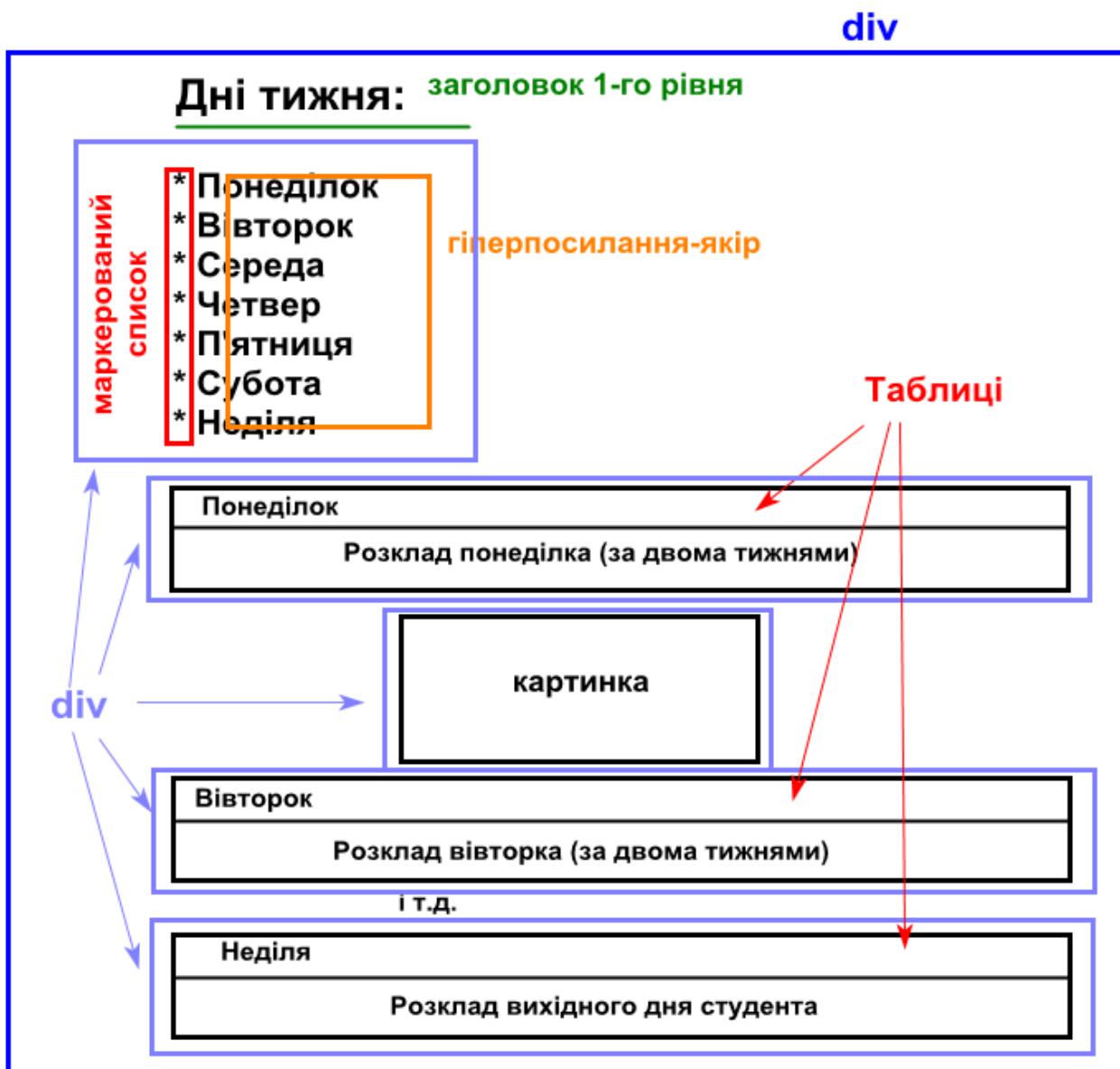
Створити файл table.html, в якому реалізувати таблицю, згідно зразку. Рамки таблиці завтовшки 2px, колір комірок на вибір студента.



Завдання №2

Створити файл schedule.html.

- передбачити маркований список із днів тижня;
- кожен елемент списку – це посилання-якір на розділ поточної веб-сторінки, де розділами виступають таблиці навчального розкладу студента (за двома тижнями) для кожного робочого дня тижня, та таблиці розкладу вільного часу кожного вихідного дня тижня;
- всі елементи на сторінці, окрім заголовку, повинні бути поміщені в блочний елемент div;
- після кожного блочного елемента, який буде містити таблицю, необхідно додати блочний елемент, який буде містити рисунок (на вибір студента) розміром 400 на 400 пікселів;
- розфарбувати таблиці на власний розсуд;
- батьківський блочний елемент повинен бути розташований по центру вікна браузера.



Завдання №3

Створити файл questionnaire.html і реалізувати форму анкети, структуру якої студент формує самостійно, яка буде збирати інформацію від користувача. Форму реалізувати таким чином, щоби були продемонстровані наступні можливості HTML5:

- текстові поля (input type="text");
- текстова область (textarea);
- поле пароллю (input type="password");
- поле email (input type="email");
- поле мобільного номера (input type="tel");
- поле дати (input type="date");
- не менше двох блоків radio-buttons (input type="radio") по три значення в кожному з блоків, передбачивши застосування для одного з значень другого блоку активність за замовчуванням (атрибут checked);
- блок check-buttons (input type="checkbox") із чотирма значеннями;

- випадний список (теги `select` + `option`) із п'ятьма значеннями, в якого третє значення визначене за замовчуванням (атрибут `selected`);
- випадний список із шістьма значеннями з можливістю множинного вибору (атрибут `multiple`);
- випадний список з чотирма групами пунктів по три значення в кожній групі (тег `optgroup`);
- випадний список із підказками (тег `datalist`);
- кнопки `type="submit"` та `type="reset"`;
- кнопку завантаження файлів (`input type="file"`) – з умовою завантаження тільки аудіо-файлів і більше ніж одного файлу (атрибути `accept` + `multiple`);
- продемонструвати структуру запиту, який буде сформовано формою, після натиснення на `button (type="submit")`, в полі `url`-адресу поточної `html`-сторінки через атрибут `method="get"` та `method="post"` + `enctype="multipart/form-data"`.

Додаткові умови:

- до усіх полів додати підписи за допомогою тегу `label`;
- згрупувати елементи форми за тематичними блоками за допомогою тегів `fieldset` і `legend`;
- додати пояснення (підказки) до усіх полів, які підтримують атрибут `placeholder`;
- частину полів зробити обов'язковими для заповнення (атрибут `required`);
- для поля `email` відключити автозаповнення за допомогою атрибуту `autocomplete="off"`;
- для полів `email` та `tel` використати атрибут `pattern` і задати регулярні вирази для перевірки коректності цих полів;
- для поля `password` використати атрибут `pattern` і задати регулярний вираз перевірки пароля за параметрами:
 - складається з 6 символів;
 - містить хоча б одну велику літеру;
 - містить хоча б одну малу літеру;
 - містить хоча б одну цифру;
- врахувати можливість інтерактивного переходу між елементами форми через клавішу `TAB` (атрибут `tabindex`);
- для кожного елемента форми застосувати пару атрибутів `name` + `value`, щоб обробник форми міг ідентифікувати ім'я елемента (`name`), яке відправляється разом з вхідними даними (`value`) на сервер.

Завдання №4

Створити файл `map.html`. Знайти в мережі Інтернет карту України і розмістити у створеному файлі дане зображення. Зображення помістити в блочний елемент.

За допомогою онлайн-сервісу для створення карт зображень `Image Map Generator` (<https://www.image-map.net/>) визначити координати областей та

прив'язати до кожної області посилання на сторінку відповідної області у Вікіпедії.