НАВЧАЛЬНА ДИСЦИПЛІНА: ПРОГРАМУВАННЯ В ІНТЕРНЕТ
ЛАБОРАТОРІЯ: КОМП'ЮТЕРНОЇ ТЕХНІКИ

Інструкція
ДО ПРАКТИЧНОЇ РОБОТИ РОБОТИ № 4.
*"*Методологія BEM. Написання якісного коду.*"*

1. Мета роботи

Ознайомитися з методологією BEM (Block Element Modifier) та принципами написання якісного HTML та CSS-коду. Закріпити навички застосування BEM на практиці шляхом рефакторингу частини головної сторінки сайту lpnu.ua відповідно до цієї методології.

2. Теоретичні відомості

2.1 Методологія BEM.

BEM (Block-Element-Modifier) — це методологія для написання чистого, структурованого та зрозумілого CSS-коду. Вона базується на трьох основних концепціях:

- **Блок (Block)** – незалежний компонент, який може існувати самостійно (наприклад, `header`, `menu`, `button`).
- **Елемент (Element)** – частина блоку, яка не може існувати без нього. Позначається через два підкреслення (`__`). Наприклад, `menu__item`, `button__icon`.
- **Модифікатор (Modifier)** – варіант блоку або елемента, який змінює його вигляд або поведінку. Позначається через подвійний дефіс (--). Наприклад, `button--primary`, `menu__item--active`.

Приклад меню:

```
<div class="menu">
  <ul class="menu__list">
    <li class="menu__item menu__item--active">Головна</li>
    <li class="menu__item">Новини</li>
    <li class="menu__item">Контакти</li>
  </ul>
</div>
```

—----------

```
.menu {
  background-color: #f5f5f5;
  padding: 10px;
}
.menu__list {
  list-style: none;
  margin: 0;
  padding: 0;
}
.menu__item {
  display: inline-block;
  margin-right: 10px;
}
.menu__item--active {
  font-weight: bold;
}
```

Приклад новини:

```
<section class="news">
    <article class="news__item news__item--featured">
        <h2 class="news__title">Головна новина</h2>
        <p class="news__date">05 березня 2025</p>
        <p class="news__lead">Короткий опис...</p>
    </article>
    <article class="news__item">
        <h2 class="news__title">Звичайна новина</h2>
        <p class="news__date">04 березня 2025</p>
    </article>
</section>
```

—----------

```
.news {
```

```
    padding: 20px;
}

.news__item {
    border-bottom: 1px solid #ddd;
    padding: 10px;
}

.news__item--featured {
    background: yellow;
}

.news__title {
    font-size: 20px;
}
```

Правильне іменування класів у BEM:

- Не використовувати теги в класах:

  - section.news, h2.news__title

  - .news, .news__title

- Не використовувати вкладеність більше ніж 2 рівні:

  .news .news__item .news__title

  .news__title

- Уникати універсальних класів

  .title (не зрозуміло, до якого блоку належить)

  .news_title

Чому використовують BEM?

1. **Прозорість коду** – легко зрозуміти структуру лише за назвами класів.
2. **Уникнення конфліктів** – немає проблеми з перезаписом стилів.

3. **Модульність** – блоки легко переносити між проектами.
4. **Гнучкість та масштабованість** – легко додавати нові стилі без зміни існуючої структури.

2.2 Принципи написання якісного коду

При написанні HTML та CSS важливо дотримуватися наступних принципів:

1. **Семантичність** – використання тегів за їхнім призначенням (`<header>`, `<section>`, `<article>`, `<nav>`).
2. **Чистий та організований CSS-код** – відсутність дублювання CSS-коду, використання зрозумілих класів за методологією BEM, коментування складних правил CSS, використання CSS-змінних (`--primary-color`)

   Приклад:

```
:root {
  --primary-color: #007bff;
  --text-color: #333;
}

body {
  font-family: Arial, sans-serif;
  color: var(--text-color);
}

.button {
  background-color: var(--primary-color);
  color: white;
  padding: 10px 20px;
 }
```

3. **Оптимізація продуктивності та адаптивність** – завантаження зображень (<picture>, srcset), @media для адаптації під мобільні пристрої, швидкість роботи (https://pagespeed.web.dev/), мінімізація CSS-коду, використання CSS-препроцесора.

4. **Валідація коду та доступність** – CSS Validator, перевірити HTML-код через W3C Validator

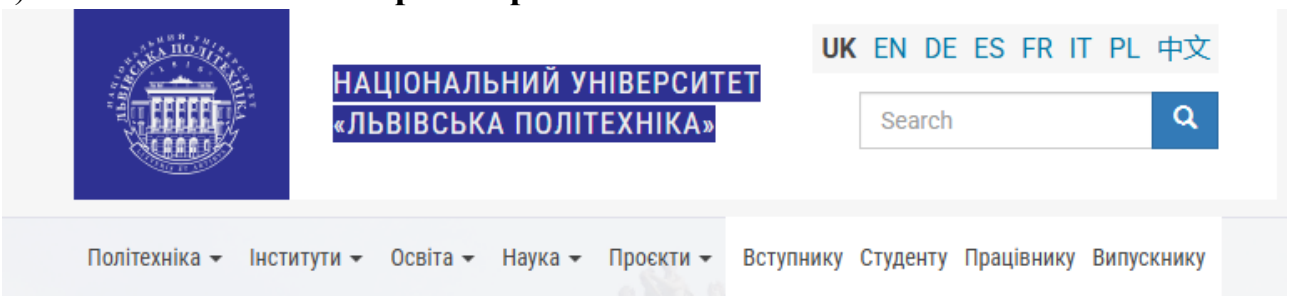
Отже, щоб HTML/CSS-код був **якісним**, потрібно:

- Використовувати **семантичну розмітку**;
- Організовувати CSS-код (змінні, BEM);
- Оптимізувати продуктивність (швидке завантаження);
- Робити **адаптивний** і **доступний** дизайн
- Перевіряти код через **валідацію**.

3. Порядок виконання роботи

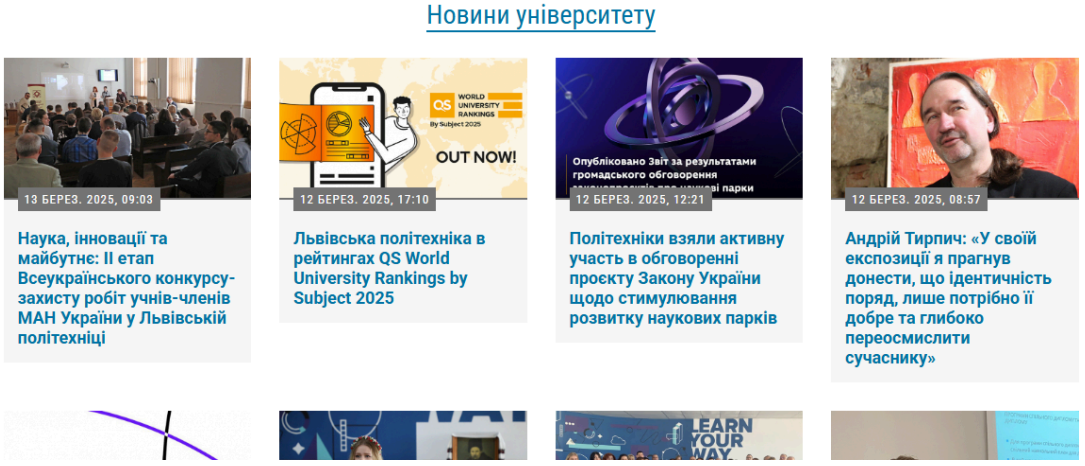
3.1 Відкрийте проект labs з проектом навчальної дисципліни чи клонуйте його з Вашого репозиторію.
3.2 Створіть нову вітку в git - pw4.
3.3 Додайте нову сторінку pw4/index.html, використовуючи семантичний html.
3.4 У браузері Chrome відкрийте головну сторінку сайту https://lpnu.ua/ та дослідіть структуру веб-сторінки за допомогою панелі Developer tools.
3.5 Умовно розбийте сторінку на основні блоки за методологією BEM.
3.6 **Ваша сторінка повинна містити**:

a) **Header з меню першого рівня**:



b) **Секцію з вмістом елемента header**, в залежності від Вашого варіанту (варіант визначається порядковим номером як остача від ділення на 4):
   - 1-ий варіант: селектор header#hero_container
   - 2-ий варіант: селектор header#logos_container
   - 3-ий варіант: селектор header#featured_container
   - 4-ий варіант: селектор header#events_container

c) Секцію з вмістом **“Новини університету”**.



d) **Footer з трьома колонками**, у кожній по 3 посилання.

3.7 Для правильного відображення сайту на різних пристроях **додайте стилі для 3 стандартних брейкпоінтів**:

- **Mobile** (мобільні пристрої): до 767px;
- **Tablet** (планшети): 768px – 1023px;
- **Desktop** (настільні ПК і ноутбуки): 1024px і більше.

3.8 На головній сторінці курси labs/index.html, додайте у меню посилання на роботу pw4/index.html.

3.9 За допомогою команди merge, додайте роботу з вітки pw4 до основної вітки проекту та виконайте push у віддалений репозиторій.

3.10 Переконайтесь, що проект оновився на хостингу https://www.netlify.com/. Скопіюйте посилання у звіт.

3.11 Перевірте швидкість та продуктивність сторінки на сервісі PageSpeed Insights

3.12 Оформити звіт по роботі, відповісти на питання вихідного контролю, зробити висновки.

4. Питання вихідного контролю

4.1 Які три основні складові методології BEM?

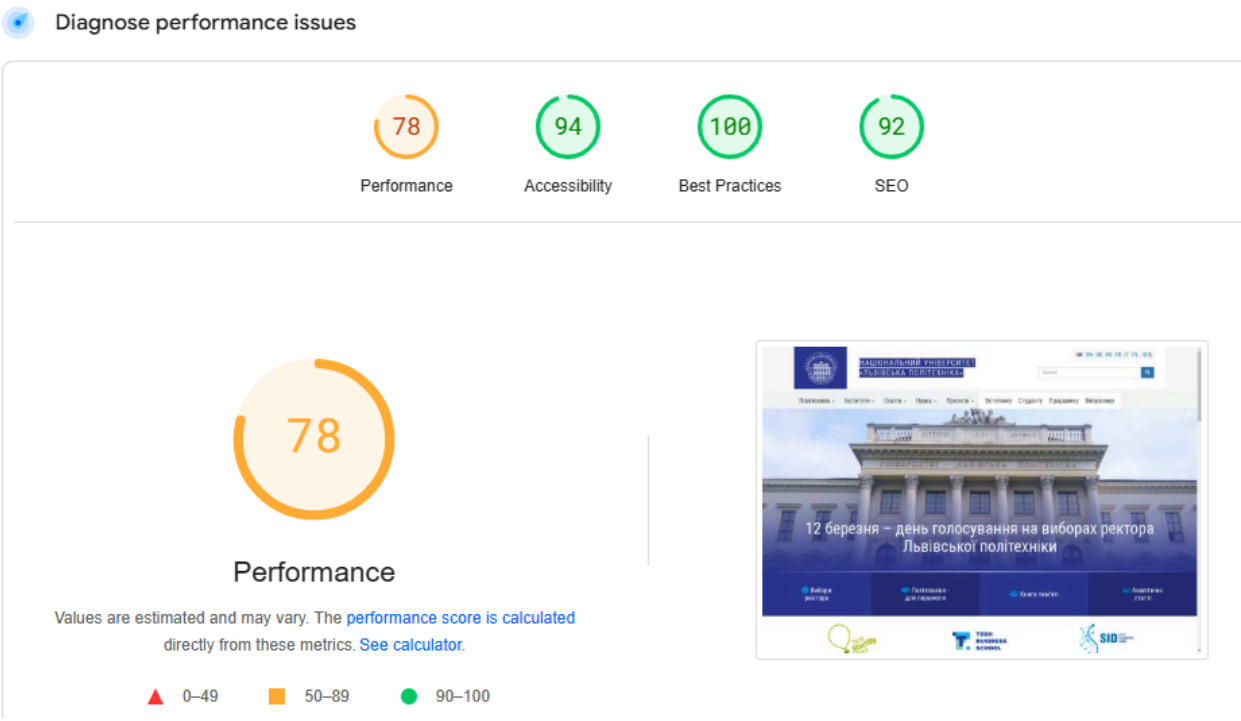
4.2 Як правильно іменувати класи за BEM?

4.3 Чому важливо використовувати семантичну розмітку?

4.4 Які медіа-запити використали у роботі?

4.5 Які значення продуктивності для вашої сторінки видає сервіс PageSpeed Insights для мобільної та десктопної версій? Наведіть скріншоти. Наприклад:

4.6



5. Оформлення звіту

5.1 Мета роботи.
5.2 Теоретичні відомості.
5.3 Завдання.
5.4 Звіт по роботі.

Перелік посилань

1. https://developer.chrome.com/docs/devtools
2. https://developer.mozilla.org/en-US/docs/Web/CSS
3. https://lpnu.ua/
4. Сервіс PageSpeed Insights
5. CSS Validator
6. W3C Validator