

ДПО	Фронтенд и бэкенд разработка
ДИСЦИПЛИНА	Основы фронтенд-разработки
ВИД УЧЕБНОГО МАТЕРИАЛА	Методические указания к практическим занятиям

Практическое занятие 3: Стилизация веб-страницы с использованием современных подходов CSS

Цель: освоить принципы организации CSS-кода с помощью одной из методологий (БЭМ, OOCSS, ACSS). Закрепить навыки работы с типографикой и создания отзывчивой верстки.

Введение и теоретическая база

После создания семантической HTML-структуры наступает этап визуального оформления. CSS превращает структурную разметку в эстетически привлекательный интерфейс. Однако без продуманной архитектуры CSS-код быстро становится трудно поддерживаемым.

Методологии организации CSS (выберите одну):

- **БЭМ (Блок-Элемент-Модификатор)** — строгие правила именования, делающие классы самодокументируемыми:
 - **Блок** — независимый компонент (`header`, `card`, `menu`)
 - **Элемент** — часть блока (`card__title`, `menu__item`)
 - **Модификатор** — внешний вид/состояние (`button--primary`, `card--featured`)
- **OOCSS (Объектно-ориентированный CSS)** — разделение структуры и оформления:
 - **Структура** — общие паттерны расположения (`.container`, `.media`)
 - **Оформление** — визуальный вид (`.bg-primary`, `.rounded`)
- **ACSS (Атомарный CSS)** — каждый класс выполняет одну функцию:
 - Утилитарные классы (`.text-center`, `.mt-20`, `.flex`)
 - Высокая переиспользуемость, минимальная специфичность

Подробнее можно ознакомиться с материалами в Лекции 3 (Вопрос 12).

Последовательность выполнения работы

Шаг 1: Подготовка проекта

1. Продолжайте работу в репозитории с предыдущей практики.
2. Убедитесь, что у вас есть корректная HTML-структура из предыдущей работы.
3. Работайте в одном CSS-файле (`styles/style.css`) для упрощения.

Шаг 2: Сброс стилей и базовые настройки

Добавьте в начало `style.css` сброс стилей

Шаг 3: Выбор и применение методологии

Выберите одну из методологий и применяйте её последовательно во всей работе:

Вариант А: БЭМ

```
<header class="header">
  <div class="header__logo">Логотип</div>
  <nav class="header__nav nav">
    <ul class="nav__list">
      <li class="nav__item nav__item--active">
        <a class="nav__link" href="#">Главная</a>
      </li>
    </ul>
  </nav>
</header>

<section class="cards">
  <div class="card card--featured">
    <h3 class="card__title">Заголовок</h3>
    <p class="card__text">Текст карточки</p>
  </div>
</section>
```

```
.header {
  display: flex;
  justify-content: space-between;
  padding: 1rem;
  background: #fff;
}

.header__logo {
  font-size: 1.5rem;
  font-weight: bold;
}

.nav__list {
  display: flex;
  gap: 1rem;
  list-style: none;
}

.nav__item--active .nav__link {
  color: #2c5530;
  border-bottom: 2px solid #2c5530;
}

.card {
```

```
padding: 1rem;
border: 1px solid #ddd;
border-radius: 8px;
}

.card--featured {
background: #f8f9fa;
border-color: #2c5530;
}
```

Вариант Б: OOCSS

```
<header class="container flex justify-between align-center bg-white">
  <div class="logo brand">Логотип</div>
  <nav class="nav">
    <ul class="list flex gap-md">
      <li class="list-item">
        <a class="link text-primary border-bottom" href="#">Главная</a>
      </li>
    </ul>
  </nav>
</header>

<section class="container grid cols-3 gap-lg">
  <div class="card padding-lg border rounded bg-light">
    <h3 class="title-md">Заголовок</h3>
    <p class="text">Текст карточки</p>
  </div>
</section>
```

```
/* Структурные классы */
.container {
width: 100%;
max-width: 1200px;
margin: 0 auto;
}

.flex {
display: flex;
}

.grid {
display: grid;
}

.justify-between {
justify-content: space-between;
}
```

```
.align-center {
  align-items: center;
}

.gap-md {
  gap: 1rem;
}

.gap-lg {
  gap: 2rem;
}

.cols-3 {
  grid-template-columns: repeat(3, 1fr);
}

/* Стиливые классы */
.bg-white {
  background: #fff;
}

.bg-light {
  background: #f8f9fa;
}

.text-primary {
  color: #2c5530;
}

.border {
  border: 1px solid #ddd;
}

.border-bottom {
  border-bottom: 2px solid;
}

.rounded {
  border-radius: 8px;
}

.padding-lg {
  padding: 1.5rem;
}

/* Компоненты */
.logo.brand {
  font-size: 1.5rem;
  font-weight: bold;
}

.card {
  transition: transform 0.2s;
}
```

```
.card:hover {  
  transform: translateY(-2px);  
}
```

Вариант В: ACSS (Атомарный CSS)

```
<header class="d-flex justify-between align-center p-20 bg-white">  
  <div class="fs-24 fw-bold">Логотип</div>  
  <nav>  
    <ul class="d-flex gap-16 list-none">  
      <li>  
        <a class="td-none c-text hover:c-primary" href="#">Главная</a>  
      </li>  
    </ul>  
  </nav>  
</header>  
  
<section class="d-grid grid-cols-3 gap-24">  
  <div class="p-16 border-1 border-solid border-gray rounded-8">  
    <h3 class="fs-18 fw-semibold mb-8">Заголовок</h3>  
    <p class="fs-14 lh-150">Текст карточки</p>  
  </div>  
</section>
```

```
/* УТИЛИТЫ КОМПОНОВКИ */  
.d-flex {  
  display: flex;  
}  
  
.d-grid {  
  display: grid;  
}  
  
.justify-between {  
  justify-content: space-between;  
}  
  
.align-center {  
  align-items: center;  
}  
  
.grid-cols-3 {  
  grid-template-columns: repeat(3, 1fr);  
}  
  
/* УТИЛИТЫ ОТСТУПОВ */  
.p-16 {  
  padding: 1rem;  
}
```

```
}

.p-20 {
  padding: 1.25rem;
}

.mb-8 {
  margin-bottom: 0.5rem;
}

.gap-16 {
  gap: 1rem;
}

.gap-24 {
  gap: 1.5rem;
}

/* Утилиты оформления */
.bg-white {
  background: #fff;
}

.border-1 {
  border-width: 1px;
}

.border-solid {
  border-style: solid;
}

.border-gray {
  border-color: #ddd;
}

.rounded-8 {
  border-radius: 8px;
}

.c-text {
  color: #333;
}

.c-primary {
  color: #2c5530;
}

.fs-14 {
  font-size: 0.875rem;
}

.fs-18 {
  font-size: 1.125rem;
}
```

```
.fs-24 {  
  font-size: 1.5rem;  
}  
  
.fw-bold {  
  font-weight: bold;  
}  
  
.fw-semibold {  
  font-weight: 600;  
}  
  
.lh-150 {  
  line-height: 1.5;  
}  
  
.td-none {  
  text-decoration: none;  
}  
  
.list-none {  
  list-style: none;  
}  
  
.hover\:c-primary:hover {  
  color: #2c5530;  
}
```

Шаг 4: Подключение шрифтов

Добавьте в style.css подключение шрифтов:

```
/* Способ 1: Google Fonts */  
@import url('https://fonts.googleapis.com/css2?family=Inter:wght@400;500;600;700&display=swap');  
  
/* Способ 2: Загрузка и импорт */  
@font-face {  
  font-family: 'CustomFont';  
  src: url('../assets/fonts/custom-font.woff2') format('woff2');  
  font-weight: 400;  
  font-display: swap;  
}  
  
/* Применение шрифтов */  
body {  
  font-family: 'Inter', system-ui, sans-serif;  
}
```

Шаг 5: Добавление интерактивности и состояний элементов

Создайте интерактивные элементы, которые реагируют на действия пользователя. Используйте псевдоклассы для различных состояний.

Базовые псевдоклассы для интерактивности:

- `:hover` — при наведении курсора
- `:focus` — при фокусе (клавиатура, мышь)
- `:active` — в момент активации (нажатие)
- `:disabled` — для заблокированных элементов

Примеры для каждой методологии:

Вариант А: БЭМ

```
.nav__link {
  transition: all 0.3s ease;
  padding: 0.5rem 1rem;
  border-radius: 4px;
}

.nav__link:hover {
  background-color: #f0f0f0;
  transform: translateY(-1px);
}

.nav__link:focus {
  outline: 2px solid #2c5530;
  outline-offset: 2px;
}

.nav__link:active {
  transform: translateY(0);
}

.card {
  transition: transform 0.3s ease, box-shadow 0.3s ease;
}

.card:hover {
  transform: translateY(-5px);
  box-shadow: 0 10px 20px rgba(0,0,0,0.1);
}

.button {
  background-color: #2c5530;
  color: white;
  padding: 0.75rem 1.5rem;
  border: none;
  border-radius: 6px;
  cursor: pointer;
}
```



```
    transition: all 0.3s ease;
}

.button:hover {
    background-color: #1e3a22;
    transform: scale(1.05);
}

.button:active {
    transform: scale(0.98);
}

.button:disabled {
    background-color: #ccc;
    cursor: not-allowed;
    transform: none;
}
```

Вариант Б: OOCSS

```
.hover-lift:hover {
    transform: translateY(-2px);
    transition: transform 0.2s ease;
}

.hover-grow:hover {
    transform: scale(1.02);
    transition: transform 0.2s ease;
}

.hover-shadow:hover {
    box-shadow: 0 4px 12px rgba(0,0,0,0.15);
    transition: box-shadow 0.3s ease;
}

.focus-outline:focus {
    outline: 2px solid #2c5530;
    outline-offset: 2px;
}

.active-scale:active {
    transform: scale(0.98);
}

.transition {
    transition: all 0.3s ease;
}

.bg-hover-light:hover {
    background-color: #f8f9fa;
}
```

```
.bg-hover-primary:hover {
  background-color: #2c5530;
  color: white;
}
```

Вариант В: ACSS (Атомарный CSS)

```
.transition-300 { transition: all 0.3s ease; }
.transition-200 { transition: all 0.2s ease; }

.hover\:translate-y--2:hover { transform: translateY(-2px); }
.hover\:scale-105:hover { transform: scale(1.05); }
.hover\:scale-102:hover { transform: scale(1.02); }

.hover\:bg-gray-50:hover { background-color: #f9fafb; }
.hover\:bg-gray-100:hover { background-color: #f3f4f6; }
.hover\:bg-primary:hover { background-color: #2c5530; }
.hover\:text-white:hover { color: white; }

.hover\:shadow-md:hover {
  box-shadow: 0 4px 6px -1px rgba(0, 0, 0, 0.1);
}

.hover\:shadow-lg:hover {
  box-shadow: 0 10px 15px -3px rgba(0, 0, 0, 0.1);
}

.focus\:outline-2:focus { outline: 2px solid; }
.focus\:outline-primary:focus { outline-color: #2c5530; }
.focus\:outline-offset-2:focus { outline-offset: 2px; }

.active\:scale-98:active { transform: scale(0.98); }

.disabled\:bg-gray-300:disabled { background-color: #d1d5db; }
.disabled\:cursor-not-allowed:disabled { cursor: not-allowed; }
```

HTML пример с интерактивными элементами

```
<!-- Для БЭМ -->
<button class="button">Обычная кнопка</button>
<button class="button disabled">Неактивная кнопка</button>

<!-- Для OOCSS -->
<button class="bg-primary text-white padding-md rounded hover-grow focus-outline
active-scale transition">
  Интерактивная кнопка
</button>

<!-- Для ACSS -->
```

```
<button class="bg-primary c-white p-16 rounded-6 hover:scale-105 focus:outline-2
focus:outline-primary transition-300">
  Атомарная кнопка
</button>
```

Дополнительные эффекты

```
/* Плавное появление элементов */
.fade-in {
  animation: fadeIn 0.5s ease-in;
}

@keyframes fadeIn {
  from { opacity: 0; transform: translateY(10px); }
  to { opacity: 1; transform: translateY(0); }
}

/* Подсветка текущего раздела */
.section-highlight {
  transition: all 0.3s ease;
}

.section-highlight:hover {
  border-left: 4px solid #2c5530;
  padding-left: 1rem;
}

/* Интерактивные карточки */
.interactive-card {
  cursor: pointer;
  transition: all 0.3s ease;
}

.interactive-card:hover {
  transform: translateY(-5px);
  box-shadow: 0 8px 25px rgba(0,0,0,0.1);
}
```

Шаг 6: Реализация навигации по якорным ссылкам

Добавьте навигацию по разделам страницы с помощью якорных ссылок. Это улучшит пользовательский опыт, особенно на длинных страницах.

Пример навигации представлен ниже

```
<!-- Навигационное меню с якорями -->
<nav class="navigation">
  <ul>
    <li><a href="#about">О нас</a></li>
```

```
<li><a href="#services">Услуги</a></li>
<li><a href="#portfolio">Портфолио</a></li>
<li><a href="#contact">Контакты</a></li>
</ul>
</nav>

<!-- Разделы с соответствующими якорями -->
<section id="about">
  <h2>О нас</h2>
  <p>Содержание раздела о компании...</p>
</section>

<section id="services">
  <h2>Услуги</h2>
  <p>Описание предоставляемых услуг...</p>
</section>

<section id="portfolio">
  <h2>Портфолио</h2>
  <p>Примеры выполненных работ...</p>
</section>

<section id="contact">
  <h2>Контакты</h2>
  <p>Контактная информация...</p>
</section>
```

Шаг 7: Публикация и отчётность

Добавьте в style.css подключение шрифтов:

1. Зафиксируйте изменения в Git.
2. Убедитесь, что GitHub Pages отображает стилизованную страницу.

Важное примечание

Адаптивное и корректное отображение элементов с помощью Flexbox и Grid Layout будет подробно рассмотрено в Практическом занятии 4. В текущей практике основное внимание уделяется методологиям организации CSS-кода и базовой стилизации.

Критерии оценивания

- **Выбор и применение методологии** — последовательно применена одна из методологий (БЭМ/OOCSS/ACSS) во всех элементах страницы
- **Качество именования классов** — классы следуют правилам выбранной методологии, названия понятные и логичные
- **Работа с типографикой** — подключены и корректно применены шрифты, соблюдена иерархия заголовков

- **Интерактивность элементов** — реализованы состояния `:hover`, `:focus`, `:active` для интерактивных элементов
 - **Навигация по странице** — реализована навигация по якорным ссылкам между разделами
 - **Работоспособность** — проект доступен по ссылке GitHub Pages, код не содержит ошибок
-

Дополнительные задания (по желанию)

- Создайте систему CSS-переменных для цветов, шрифтов и отступов
 - Добавьте микро-анимации для интерактивных элементов
 - Реализуйте плавную прокрутку к якорным ссылкам
 - Проверьте контрастность цветов для обеспечения доступности
-

Формат отчета

- В область для загрузки прикреплены ссылка на GitHub Pages.
- В комментарии к ответу добавлено обоснование выбора методологии (БЭМ, OOCSS, ACSS).