

ДИСЦИПЛИНА	Фронтенд и бэкенд разработка
ИНСТИТУТ	ИПТИП
КАФЕДРА	Индустриального программирования
ВИД УЧЕБНОГО МАТЕРИАЛА	Методические указания к практическим занятиям
ПРЕПОДАВАТЕЛЬ	Загородних Николай Анатольевич
СЕМЕСТР	1 семестр, 2025/2026 уч. год

Практика 4. CSS-архитектура: БЭМ, переменные, состояния

(накидываем стили на многостраничник из Пр2 и на форму/модалку из Пр3; добавлены изображения, CSS-фон и видео)

В мире фронтенда HTML — это каркас, JS — хитрый план, но порядок на улицах наводит только CSS: заходит, поправляет шляпу :root, раздаёт указания специфичности и шепчет: «стиль — это всё». Flexbox разводит локти, Grid расставляет всех по местам, :hover подмигивает прохожим, а зарвавшийся z-index лезет без очереди — получает по шляпе от .container { position: relative; }. И если соперник прижмёт к стене, у нас есть одна пуля с гравировкой !important — применять редко, но метко



Связь с предыдущими практиками

- **Пр2:** стилизуем index.html, about.html, contacts.html (шапка/навигация, «хлебные крошки», якоря, секции).
- **Пр3:** стилизуем **форму и модальное окно** (<dialog> или «ручную» модалку), состояния и ошибки.

Цель

Единый styles.css, дизайн-токены в :root, БЭМ-структура, корректные состояния (:hover, :focus-visible, :disabled, :target), осознанная блочная модель. Добавляем **изображения, CSS-фон, видео**.

Выход

styles.css подключён ко **всем** страницам Пр2 и Пр3; шапка/навигация/крошки/форма/модалка оформлены по БЭМ; добавлены responsive-изображения и видео. PR dev → main, демо.

1) Теория

1.1. :root, подключение CSS и организация стилей

Подключение (во <head> всех страниц Пр2 и Пр3):

```
<!-- Общая таблица стилей для всех страниц; путь скорректируйте под свою
структуру проекта -->
<link rel="stylesheet" href="styles.css">
```

:root — корень дерева (как правило <html>). Храним **кастомные свойства (CSS-переменные)** — «дизайн-токены», наследуемые по документу и легко переопределяемые (например, на body.theme-dark). Чтение: var(--имя, [фолбэк]).

Организация

- Один базовый styles.css: **токены + общие правила**. При росте — разнос по модулям (напр., blocks/site-nav.css, blocks/form.css) и сборка позже.
- Пишем **по классам**; избегаем #id и !important.
- Вложенность селекторов — **1–2 уровня**.
- При равной специфичности побеждает правило **ниже по файлу** (порядок имеет значение).

- При конфликте используйте `@layer` (когда появится сборка) или структурируйте файл разделами: **tokens** → **base** → **utilities** → **components** → **overrides**.

MDN RU: Каскад/наследование —

<https://developer.mozilla.org/ru/docs/Web/CSS/Cascade>

Специфичность — <https://developer.mozilla.org/ru/docs/Web/CSS/Specificity>

1.2. Мини-стайлгайд (дизайн-токены)

```
:root{
  /* Палитра */
  --color-bg:#fff; --color-fg:#0b0b0b; --color-muted:#666;
  --color-primary:#0a84ff; --color-danger:#b00020;

  /* Типографика и размеры */
  --font-body:system-ui,-apple-system,"Segoe UI",Roboto,Arial,sans-serif;
  --space-1:4px; --space-2:8px; --space-3:16px; --space-4:24px; --space-5:40px;

  /* Скругления и тени */
  --radius:12px;
  --shadow:0 1px 2px rgba(0,0,0,.08), 0 8px 24px rgba(0,0,0,.08);
}

/* Тёмная тема: переопределяем только значения токенов */
body.theme-dark{
  --color-bg:#0b0b0b; --color-fg:#f5f5f5; --color-muted:#aaa;
  --color-primary:#6aa2ff;
}
```

Зачем токены: единая точка правды для цветов/отступов/скруглений → легко поддерживать и темизировать.

MDN RU: Пользовательские свойства —

https://developer.mozilla.org/ru/docs/Web/CSS/Using_CSS_custom_properties

`var()` — <https://developer.mozilla.org/ru/docs/Web/CSS/var>

1.3. БЭМ: как надо / как не надо

- **Блок** — самостоятельный компонент (`.site-header`, `.site-nav`, `.breadcrumbs`, `.form`, `.modal`).
- **Элемент** — часть блока (`.site-nav__list`, `.form__label`).
- **Модификатор** — вариация (`.site-nav__link--cta`, `.section--alt`).

Хорошо:

```
<header class="site-header">
  <a class="site-header__brand" href="/">Лого</a>
  <nav class="site-nav" aria-label="Основная навигация">
    <ul class="site-nav__list">
      <li class="site-nav__item"><a class="site-nav__link"
href="index.html">Главная</a></li>
      <li class="site-nav__item"><a class="site-nav__link"
href="about.html">О нас</a></li>
      <li class="site-nav__item"><a class="site-nav__link site-nav__link--
cta" href="#contacts">Связаться</a></li>
    </ul>
  </nav>
</header>
```

Плохо: #id, инлайн-стили, .red/.big → взвинчивают специфичность, мешают переопределять.

БЕМ:

Методология — <https://ru.bem.info/methodology/>

Нейминг — <https://ru.bem.info/methodology/naming-convention/>

Быстрый старт — <https://ru.bem.info/methodology/quick-start/>

1.4. Состояния и доступность

Ссылки (LVNA-порядок):

```
/* Порядок важен: link → visited → hover → active */
a:link{} a:visited{} a:hover{} a:active{}
```

Фокус-контуры:

```
/* :where(...) не добавляет специфичности — проще переопределять позже */
:where(a,button,input,select,textarea):focus-visible{
  outline:2px solid var(--color-primary);
  outline-offset:2px;
}
```

```
/* Недоступные элементы визуально «приглушаем» */
:disabled,[aria-disabled="true">{opacity:.6;cursor:not-allowed;}
```

Якоря:

```
html{ scroll-behavior:smooth; } /* Плавная прокрутка к #якорю */
```

```
:target{ outline:2px dashed var(--color-primary); outline-offset:4px; } /*  
Подсветка цели */
```

MDN RU: Стилизация ссылок —

https://developer.mozilla.org/ru/docs/Learn_web_development/Core/Styling_basics/Styling_links

:focus-visible — <https://developer.mozilla.org/ru/docs/Web/CSS/:focus-visible>

:target — <https://developer.mozilla.org/ru/docs/Web/CSS/:target>

1.5. Блочная модель (минимум «на сейчас»)

- Коробка: content + padding + border + margin.
- box-sizing: border-box — размеры считают padding/border → проще.
- Вертикальные margin соседних блоков могут **схлопываться** (возьмётся максимум).
- Типы отображения: block | inline | inline-block | flex | grid.

```
/* Reset-минимум — подробнее в 1.8 */  
*{ box-sizing: border-box; }  
html, body{ margin: 0; padding: 0; }  
body{  
  background: var(--color-bg);  
  color: var(--color-fg);  
  font-family: var(--font-body);  
  line-height: 1.5;  
}
```

MDN: Блочная модель —

https://developer.mozilla.org/ru/docs/Learn_web_development/Core/Styling_basics/Box_model

box-sizing — <https://developer.mozilla.org/ru/docs/Web/CSS/box-sizing>

display — <https://developer.mozilla.org/ru/docs/Web/CSS/display>

1.6. Единицы и «ритм отступов»

- Абсолютные: px. Относительные: rem, em, %.
- Для отступов и размеров компонентов используйте **шкалу токенов** -- спрасе-* и чаще rem — масштабируется консистентно.

```
/* Центровка и ограничение ширины контейнера */
```

```
.container{ width:min(100% - 2*var(--space-4), 1120px); margin-inline:auto; }

/* Вертикальные отступы секций по токенам */
.section{ padding-block:var(--space-5); }
.section__title{ margin:0 0 var(--space-3); }
```

MDN: Значения и единицы —

https://developer.mozilla.org/ru/docs/Learn_web_development/Core/Styling_basics/Values_and_units

1.7. Мини-таблица специфичности

- селектор типа (div) — **0-0-1**
- класс/атрибут/псевдокласс (.btn, [data-x], :hover) — **0-1-0**
- #id — **1-0-0**
- **инлайн-стиль** — **1-0-0** (особый случай; переопределять трудно)

Как считать: .card .card__title:hover → 0-3-0; #main .card__title → 1-1-0 ⇒ второй сильнее.

Как снижать вес: короткие селекторы по классам; группируйте через :where(...) (нулевая специфичность).

```
/* Антипаттерн: селектор тяжёлый и хрупкий */
#header .menu .item a{ color:#333; }

/* Хорошо: опора на БЭМ-класс компонента */
.site-nav__link{ color:var(--color-fg); }
```

1.8. Reset-минимум

Достаточно для этого этапа:

```
{*{ box-sizing:border-box; } /* Проще считать ширины/высоты */
html,body{ margin:0; padding:0; } /* Убираем стартовые отступы браузера */
```

Полезные дополнения (опционально):

```
/* Медиа не «выпрыгивают» из контейнеров */
img,video{ max-width:100%; height:auto; display:block; }

/* Форм-контролы наследуют шрифт и цвет */
button,input,select,textarea{ font:inherit; color:inherit; }
```

Не делаем: не отключаем глобально фокус-кольца; не используем «жёсткие» ресеты без последующего полного стайлинга всех элементов. Normalize можно добавить позже.

1.9. DevTools-ритуал — (Chrome/Chromium)

Горячие клавиши: F12 / Ctrl+Shift+I (macOS: Cmd+Opt+I).

Порядок диагностики:

1. **Elements** → **Styles**: фильтр по свойству (например, color) — видно, кто проиграл/победил.
2. **Computed**: клик по свойству — откуда «приехало» итоговое значение.
3. Кнопка **:hov**: форсируйте :hover/:focus/:active и проверяйте состояния без действий мышью/клавиатурой.
4. **Box Model**: найдите «прыжки» из-за margin/padding/border.
5. **Layout** (или подсветка сеток): диагностика flex/grid.
6. **Coverage** (Ctrl+Shift+P → «Coverage»): оценка **неиспользуемого CSS**.

Чек-лист перед PR: фокус виден Tab'ом; картинки/видео вписываются в контейнер; нет #id/инлайна; БЭМ-имена читаемы; шкала --space-* соблюдается.

1.10. Переключатель темы body.theme-dark

Принцип: компоненты читают var(--color-*); смена темы — **только** переопределение токенов.

```
:root{ color-scheme: light; } /* Светлые скроллбары/контролы по умолчанию */
body.theme-dark{ color-scheme: dark; } /* Тёмные — при активной теме */
```

Мини-JS (по желанию):

```
<button class="theme-toggle" type="button" aria-pressed="false">Тема</button>
<script defer>
  const KEY='theme', btn=document.querySelector('.theme-toggle');
  const prefersDark = matchMedia('(prefers-color-scheme: dark)').matches;

  // Автовыбор: системная тема или сохранённый выбор пользователя
  if(localStorage.getItem(KEY)=== 'dark' || (!localStorage.getItem(KEY) &&
prefersDark)){
    document.body.classList.add('theme-dark');
    btn?.setAttribute('aria-pressed','true');
  }
}
```

```
// Переключение по кнопке с сохранением в localStorage
btn?.addEventListener('click', ()=>{
  const isDark=document.body.classList.toggle('theme-dark');
  btn.setAttribute('aria-pressed', String(isDark));
  localStorage.setItem(KEY, isDark ? 'dark' : 'light');
});
</script>
```

А11у-заметки: проверяйте контраст $\geq 4.5:1$; фокус-кольца должны быть заметны в обеих темах; SVG-иконки лучше красить через `currentColor`. Анимацию смены темы делайте мягкой и учитывайте `prefers-reduced-motion`.

1.11. Изображения, CSS-фон и медиа

1.11.1. Контентные изображения:

```
<img>
<!-- width/height уменьшают «скачок» макета (CLS); alt обязателен -->

```

Responsive (ширины):

```
<!-- Браузер сам выбирает лучший ресурс по ширине контейнера -->

```

Альтернативные форматы:

```
<!-- <picture> позволяет предложить AVIF/WebP, а <img> — надёжный фолбэк -->
<picture>
  <source type="image/avif" srcset="img/card.avif">
  <source type="image/webp" srcset="img/card.webp">
  
</picture>
```

Кадрирование:

```
/* Изображение в карточке заполняет блок и не искажает пропорции */
.card__img{
```



```
width:100%; height:220px; /* фиксируем высоту карточки */
object-fit:cover; object-position:center; /* аккуратное кадрирование */
border-radius:var(--radius); display:block;
}
```

MDN: — <https://developer.mozilla.org/ru/docs/Web/HTML/Element/img>

Responsive images (srcset/sizes) —

https://developer.mozilla.org/ru/docs/Learn_web_development/Core/Images/Responsive_images

<picture> — <https://developer.mozilla.org/ru/docs/Web/HTML/Element/picture>

object-fit — <https://developer.mozilla.org/ru/docs/Web/CSS/object-fit>

1.11.2. Декоративные изображения: CSS-фон

```
/* Герой-секция: затемняющий градиент поверх фонового изображения */
.hero{
  background:
    linear-gradient(to top, rgba(0,0,0,.55), rgba(0,0,0,.15)), /* оверлей для
контраста текста */
    url("img/hero-1920.jpg") center/cover no-repeat; /* картинка
как декор */
  color:#fff;
  padding: clamp(2rem, 6vw, 6rem) 1rem; /* адаптивные
отступы */
  border-radius:var(--radius);
}
```

Ретина (image-set) и fallback:

```
.hero{
  background-image:
    linear-gradient(to top, rgba(0,0,0,.55), rgba(0,0,0,.15)),
    image-set(
      url("img/hero-1280.jpg") 1x, /* обычные экраны */
      url("img/hero-2560.jpg") 2x /* плотные экраны */
    );
  background-position:center;
  background-size:cover;
  background-repeat:no-repeat;
}
```

MDN: background — <https://developer.mozilla.org/ru/docs/Web/CSS/background>

image-set() — <https://developer.mozilla.org/ru/docs/Web/CSS/image>

1.11.3. Адаптивные медиа-контейнеры и соотношение сторон

```
/* Контейнер с фиксированным соотношением сторон для <iframe>/<video> */
.media{
    aspect-ratio: 16/9;                /* удерживаем пропорцию блока */
    width: min(100%, 960px);           /* не шире 960px, но адаптивен */
    margin-inline: auto;               /* центрируем блок */
}
.media iframe,
.media video{
    width:100%; height:100%;           /* медиа занимает всю площадь контейнера */
    border:0; border-radius:var(--radius);
}
```

MDN: aspect-ratio — <https://developer.mozilla.org/ru/docs/Web/CSS/aspect-ratio>

1.11.4. Видео: локальное <video> и встраиваемые сервисы

А) Локальное видео (контроль и доступность):

```
<!-- Добавьте субтитры через <track> (WebVTT); poster — превью-кадр -->
<video class="media" controls preload="metadata" playsinline
    width="960" height="540" poster="video/poster.jpg">
    <source src="video/teaser-1080.mp4" type="video/mp4"> <!-- основной
формат -->
    <source src="video/teaser-720.webm" type="video/webm"> <!-- альтернатива -
->
    <track kind="captions" srclang="ru" src="video/captions-ru.vtt"
label="Русские субтитры" default>
    Ваш браузер не поддерживает видео.
</video>
```

Примечание: autoplay без muted часто блокируется. Для iOS добавляйте playsinline.

MDN: <video> —

<https://developer.mozilla.org/ru/docs/Web/HTML/Element/video>

<track> — <https://developer.mozilla.org/ru/docs/Web/HTML/Element/track>

В) Встраивание — Rutube / VK Видео:

```
<!-- Вставка с Rutube; замените XXXXXXXXXXXXXXX на ID ролика -->
<div class="media">
    <iframe
        src="https://rutube.ru/play/embed/XXXXXXXXXXXX"
        title="Видео о проекте"                                <!-- важно для
доступности -->
        allow="autoplay; encrypted-media; picture-in-picture; web-share"
```

```

        allowfullscreen loading="lazy"></iframe>
</div>
<!-- Вставка с VK Видео; проверьте, разрешён ли embed для ролика -->
<div class="media">
  <iframe
    src="https://vk.com/video_ext.php?oid=XXXX&id=YYYY&hash=ZZZZ"
    title="Видео о проекте"
    allow="autoplay; encrypted-media; picture-in-picture; web-share"
    allowfullscreen loading="lazy"></iframe>
</div>

```

Замечания: учитывайте CSP/политику встраивания; задавайте title; оборачивайте <iframe> в .media с aspect-ratio (см. MDN RU ссылку выше).

1.11.5. Производительность и качество

- **Форматы:** AVIF/WebP/JPEG — для фото; SVG/PNG — для графики/иконок.
- **Атрибуты :** всегда width/height, по возможности loading="lazy" и decoding="async".
- **Критичные изображения** (герой/логотип) — **не** делайте ленивыми.
- Не подменяйте контентные фоном — это ухудшит доступность/SEO.

2) Пошаговая работа (накидываем стили на Пр2 и Пр3)

1. Подключите styles.css ко **всем** страницам Пр2 и Пр3 (<head>).
2. Добавьте токены, reset-минимум и базовую типографику (разд. 1.2, 1.5, 1.6).
3. Оформите header/nav по БЭМ (Пр2), добавьте :hover/:focus-visible, подсветку активного пункта.
4. Оформите «крошки» и якоря (Пр2): БЭМ + scroll-behavior:smooth + :target.
5. Стилизируйте **форму и модальку** (Пр3): блок .form/.modal, поля/кнопки/ошибки/фокус, <dialog>::backdrop.
6. **Изображения:**

— как минимум одно сделайте **responsive** (srcset/sizes или <picture>);

— для героя примените **фон-изображение** с градиент-оверлеем.

7. **Видео:**

— либо <video> с poster и <track>;

— либо <iframe> (Rutube/VK) в контейнере .media с aspect-ratio (MDN RU: <https://developer.mozilla.org/ru/docs/Web/CSS/aspect-ratio>).

8. (Опц.) Переключатель темы (разд. 1.10).
9. Проверьте клавиатурой и через DevTools-ритуал (разд. 1.9).
10. Коммит → PR → демо.

3) Задания

БАЗА:

1. Подключить styles.css ко всем страницам Пр2 и Пр3.
2. Вынести токены в :root; **использовать только шкалу --space-***.
3. БЭМ для шапки/навигации; :hover/:focus-visible; активный пункт.
4. «Крошки» и якоря (:target, плавный скролл).
5. Форма и модалка (поля, кнопки, фокус, ошибки).
6. **Responsive-изображение** (через srcset/sizes или <picture>) + **фон-секция** с градиент-оверлеем.
7. **Видео:** локальное <video> с poster и <track> **или** Rutube/VK <iframe> в .media с aspect-ratio (MDN RU: <https://developer.mozilla.org/ru/docs/Web/CSS/aspect-ratio>).

Дополнительно (любой):

- Переключатель темы (body.theme-dark).
- SVG-иконки с currentColor; спрайт.
- «Ленивая» загрузка для карточек каталога; токены для типографики.

4) Критерии оценивания (10 б)

- **Архитектура (4):** один styles.css, токены в :root, без инлайна/#id; стили подключены ко **всем страницам Пр2 и Пр3**.
- **БЭМ (3):** корректный block__element--modifier, минимум вложенности, читаемость классов.
- **UX/медиа (3):** :focus-visible, :hover, :disabled, :target; доступная модалка; есть responsive-изображение и одно видео, корректно вписанное по aspect-ratio (MDN RU: <https://developer.mozilla.org/ru/docs/Web/CSS/aspect-ratio>).

5) Требования к сдаче

PR dev → main, осмысленные коммиты, демо-ссылка в README, комментарии в ключевых местах (БЭМ/токены/состояния/медиа).

6) Контрольные вопросы

1. Когда использовать **модификатор**, а когда — **новый блок**?
2. Как считается **специфичность** и почему #id/инлайн — избыточны?
3. Зачем **дизайн-токены** и как они помогают темизации?
4. content-box vs border-box: чем второй удобнее?
5. Что даёт :focus-visible относительно :focus?
6. Когда лучше, чем background-image, и наоборот?
7. Чем помогают srcset/sizes и почему важно задавать width/height у ?
8. Как корректно встраивать видео (локально и через Rutube/VK), включая контейнер с aspect-ratio (MDN RU: <https://developer.mozilla.org/ru/docs/Web/CSS/aspect-ratio>)?

7) Частые ошибки

- Стили подключены не на всех страницах Пр2/Пр3.
- Глубокие селекторы, #id, инлайн, !important.
- Нет фокус-контуров; Tab «убегает» из модалки.
- Нарушена шкала --space-*.
- Текст на фоне без оверлея → низкий контраст.
- Видео без title у <iframe> или без controls; нет субтитров (<track>).

8) Источники

- Каскад/наследование: <https://developer.mozilla.org/ru/docs/Web/CSS/Cascade>
- Специфичность: <https://developer.mozilla.org/ru/docs/Web/CSS/Specificity>
- Блочная модель: https://developer.mozilla.org/ru/docs/Learn_web_development/Core/Styling_basics/Box_model
- box-sizing: <https://developer.mozilla.org/ru/docs/Web/CSS/box-sizing>
- display: <https://developer.mozilla.org/ru/docs/Web/CSS/display>

- Пользовательские свойства:
https://developer.mozilla.org/ru/docs/Web/CSS/Using_CSS_custom_properties
- var(): <https://developer.mozilla.org/ru/docs/Web/CSS/var>
- Стилизация ссылок (LVHA):
https://developer.mozilla.org/ru/docs/Learn_web_development/Core/Styling_basics/Styling_links
- :focus-visible: <https://developer.mozilla.org/ru/docs/Web/CSS/:focus-visible>
- :target: <https://developer.mozilla.org/ru/docs/Web/CSS/:target>
- aspect-ratio: <https://developer.mozilla.org/ru/docs/Web/CSS/aspect-ratio>
- : <https://developer.mozilla.org/ru/docs/Web/HTML/Element/img>
- Responsive images (srcset/sizes):
https://developer.mozilla.org/ru/docs/Learn_web_development/Core/Images/Responsive_images
- <picture>:
<https://developer.mozilla.org/ru/docs/Web/HTML/Element/picture>
- object-fit: <https://developer.mozilla.org/ru/docs/Web/CSS/object-fit>
- background-*: <https://developer.mozilla.org/ru/docs/Web/CSS/background>
- image-set(): <https://developer.mozilla.org/ru/docs/Web/CSS/image>
- <video>: <https://developer.mozilla.org/ru/docs/Web/HTML/Element/video>
- <track>/WebVTT:
<https://developer.mozilla.org/ru/docs/Web/HTML/Element/track>