КАИТ 20
МАСТЕРСТВО И ПРОФЕССИОНАЛИЗМ

Департамент образования и науки города Москвы
Государственное бюджетное профессиональное образовательное учреждение
города Москвы «Колледж автоматизации и информационных технологий № 20»

ПРАКТИЧЕСКАЯ РАБОТА

HTML и CSS

Создание страницы сайта

Пошаговая методичка
с учебником Webkit

3 курс · 09.02.07

Информационные системы
и программирование

Квалификация: специалист
по информационным системам

ДУП.01

90 минут · Индивидуально

Преподаватель

Беличенко Александр Андреевич

Студент

Группа

Москва · 2026

01 Задание и план практической работы

ДУП.01 · 3 курс · 09.02.07
Информационные системы и программирование.
Квалификация: специалист по информационным системам.
Преподаватель: Беличенко Александр Андреевич.

Что вы сделаете

Индивидуально создадите страницу с навигацией, карточками, местом и временем, списком подготовки и подвалом. Учебник Webkit КАИТ20 встроен в шаги: два коротких упражнения и справка по HTML/CSS. Маршрут - на стр. 2.

Результат: папка с index.html, styles.css и коротким отчётом. Рекомендуемое время - 90 минут.

Бриф - контрольные сведения

Параметр	Значение
Событие	Учебная мастерская по созданию веб-страниц
Дата и время	15 ноября 2026 года, 12:00
Место	Аудитория 305
Содержание	HTML, оформление CSS, проверка страницы

Сведения о событии вымышлены. При самостоятельных изменениях сохраните дату, время и аудиторию. Публикация в интернете и реальная регистрация посетителей в эту работу не входят.

Порядок работы

Минуты	Что выполнить
0-12	Войти в Webkit, выполнить урок 1, подготовить файлы (стр. 2-4)
12-35	Собрать HTML по шагам (стр. 5-8)
35-65	Урок 4 Webkit, CSS и адаптация (стр. 9-13)
65-75	Выполнить самостоятельные изменения (стр. 14)
75-85	Проверить и исправить ошибки (стр. 15-16)
85-90	Подготовить сдачу и объяснение (стр. 17)

Нужны ПК, редактор обычного текста, браузер и интернет для Webkit. Учётную запись выдаёт преподаватель. Работа с учебником входит в 90 минут. За ПК соблюдайте правила кабинета.

02 Как использовать Webkit в этой работе

Откройте kait20.tech/webkit и войдите под своей учебной учётной записью. Выберите курс **«Webkit - создание веб-сайтов»**. Прямые ссылки ниже ведут именно в этот курс.

Маршрут: изучить → применить → проверить

Когда	Что открыть в Webkit	Что сделать в своей практике
До шага 2 3 минуты	Урок 1. Ваша первая веб-страница https://kait20.tech/webkit/lessons/1	Выполнить упражнение с h1 и p. Проверить решение. Затем создать свой index.html.
Шаг 4 1 минута	Урок 2. Страница о себе https://kait20.tech/webkit/lessons/2	Прочитать объяснение структуры. Найти заголовки, список и href в своём HTML.
До шага 6 4 минуты	Урок 4. Оформите визитку https://kait20.tech/webkit/lessons/4	Выполнить упражнение с .card и тремя свойствами. Проверить. Затем оформить свою страницу.
Шаги 7-8 3 минуты	Уроки 5 / 6 / 7: отступы, Flexbox, Grid. Ссылки есть в соответствующих шагах.	По объяснению найти padding, навигацию и сетку в своих файлах. Полные тренировки этих уроков не обязательны.
Шаг 9 1 минута	Урок 8. Мобильная версия https://kait20.tech/webkit/lessons/8	Сопоставить viewport и @media. Проверить свой макет при ширине 320 и 1280.

Как выполнить две тренировки

В уроках 1 и 4 работайте в стартовом редакторе: измените код → нажмите **«Запустить»** → **«Проверить решение»** → исправьте невыполненные критерии. Запишите итог для отчёта; ориентир автопроверки - 100/100 и все обязательные пункты. Самооценка учебника и оценка этой работы по шкале 0-10 - разные результаты.

Два места работы - разные имена CSS

В Webkit: вкладка **style.css** подключается автоматически; script.js для этих упражнений не нужен. **В своей папке:** файл называется **styles.css**, его подключает строка link в index.html. Правильные имена и пути сохраняйте для каждого места.

Точные селекторы и значения упражнения проверяются его условиями. Например, .gallery и 600px в учебнике отличаются от .cards и 44rem нашего макета. Переносите понимание, а свой файл собирайте по методичке. Весь курс и все упражнения за эту пару проходить не требуется.

Если вход или сайт недоступны, выполняйте те же шаги по PDF; в отчёте укажите причину вместо результата Webkit. Доступ выдаёт преподаватель. Отсутствие доступа само по себе не считается ошибкой HTML/CSS.

03 Коротко об основах: что и как записывать

Понятие	Значение и пример
HTML	Язык разметки. Описывает смысл и структуру содержимого.
Тег	Разметочная запись в угловых скобках: или .
Элемент	Вся конструкция: Привет! . У пустых элементов, например meta, нет закрывающего тега.
Атрибут	Уточнение в открывающем теге: class="lead". Имя - class, значение - lead.
CSS	Язык описания представления: цвет, размер, отступы, расположение.
Селектор	Выбирает элементы. .lead выбирает class="lead", h1 - элементы h1.
Свойство и значение	color: #493786; задаёт конкретный цвет текста.
Класс и id	Класс можно повторять. Значение id в документе должно быть уникальным. #program в ссылке ведёт к id="program".

HTML: имя="значение"

```
<p class="lead">Описание мастерской.</p>
```

Открывающий тег - , содержимое - текст, закрывающий тег - . Атрибут расположен внутри открывающего тега.

CSS: свойство: значение;

```
.lead {
  color: #493786;
  font-size: 1.25rem;
}
```

Селектор находится до фигурной скобки. Объявления пишутся внутри блока. Между свойством и значением ставится двоеточие; объявления разделяются точками с запятой.

Вложенный элемент закрывайте раньше внешнего. Код сохраняйте как обычный текст UTF-8. После изменения: сохранить файл, затем обновить страницу.

04 Шаг 1. Подготовьте папку и два файла

Действия

1. Создайте в своей рабочей папке каталог Web_Фамилия. Например, Web_Ivanov. Положите все файлы этой работы в него.

2. В Проводнике включите расширения имён файлов. Windows 11: Вид → Показать → Расширения имён файлов. Windows 10: вкладка Вид → Расширения имён файлов.

3. Откройте Блокнот. Выберите Файл → Сохранить как. Перейдите в созданную папку. Имя: index.html. Тип: Все файлы. Кодировка: UTF-8. Сохраните.

4. Создайте второй пустой документ и сохраните в ту же папку как styles.css: также Все файлы и UTF-8. Не выбирайте формат Word.

5. Если используете VS Code, откройте эту папку через Файл → Открыть папку, создайте два файла с указанными именами. Установка нового редактора не требуется.

Структура проекта

```
Web_Ivanov/
  index.html
  styles.css
```

Проверка №1

В одной папке должны лежать ровно эти два рабочих файла. Посмотрите полные имена: index.html.txt и styles.css.txt не подходят. Значок файла может отличаться на разных компьютерах, поэтому проверяйте именно имя.

Как работать с кодом в PDF

Выделяйте только строки кода в светлом блоке, копируйте их и вставляйте в нужный файл. Не копируйте заголовки страниц и подписи. Отступы помогают читать код, но нельзя терять скобки, кавычки и имена тегов. В каждом следующем шаге прямо указано: заменить строку или добавить блок.

HTML редактируйте в редакторе текста, результат смотрите в браузере. Это разные действия.

05 Шаг 2. Создайте каркас HTML-документа

Webkit. Ваша первая веб-страница. До этого шага выполните упражнение один раз: добавьте h1 и p в body стартового документа, нажмите «Запустить», затем «Проверить решение». Запишите результат. Далее соберите свой файл по блоку ниже.

Файл: index.html. Вставьте весь блок ниже в пустой файл. Если там уже есть пробный текст, замените его полностью.

```
<!doctype html>
<html lang="ru">
<head>
  <meta charset="utf-8">
  <meta name="viewport"
        content="width=device-width, initial-scale=1">
  <title>Веб-мастерская | Учебная страница</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h1>Моя первая страница</h1>
</body>
</html>
```

Что означают строки

Запись	Назначение
<!doctype html>	Включает стандартный режим HTML.
html lang="ru"	Корень документа и основной язык содержимого.
head	Сведения о документе: кодировка, заголовок вкладки, подключение CSS.
meta viewport	Настройка области просмотра для мобильных устройств.
link ... href="styles.css"	Подключает CSS-файл из этой же папки.
body	Содержимое, отображаемое на странице.

Проверка №2

1. Сохраните файл сочетанием Ctrl+S. В Проводнике дважды щёлкните index.html. При необходимости выберите Открыть с помощью → браузер.

2. На странице должен появиться заголовок «Моя первая страница». Во вкладке браузера - «Веб-мастерская | Учебная страница».

3. Оставьте браузер и редактор открытыми. После каждого следующего шага сохраняйте файл и обновляйте браузер: Ctrl+R или F5.

Пока styles.css пустой, стандартный вид без оформления - ожидаемый результат.

06

Шаг 3. Добавьте шапку и главный блок

Файл: index.html. Найдите строку <h1>Моя первая страница</h1> внутри body. Замените только эту строку всем блоком ниже. Строки <body> и </body> оставьте.

```
<header class="site-header">
  <div class="container header-row">
    <a class="brand" href="#top">Веб-мастерская</a>
    <nav aria-label="Основная навигация">
      <a href="#program">Программа</a>
      <a href="#details">Место и время</a>
    </nav>
  </div>
</header>

<main id="top" class="container">
  <section class="hero">
    <p class="eyebrow">Открытая учебная мастерская</p>
    <h1>Создай свою первую веб-страницу</h1>
    <p class="lead">
      Разберём HTML и CSS и соберём страницу,
      которая удобно читается на разных экранах.
    </p>
    <a class="button" href="#program">Смотреть программу</a>
  </section>
</main>
```

Назначение: header - шапка; nav - навигация; main - основное содержимое; section - смысловой раздел; h1 - главный заголовок нашей страницы. Класс button оформит ссылку как крупный элемент перехода: она ведёт к разделу, поэтому используется a.

Сохраните и обновите: видны название, два пункта меню, заголовок и ссылка «Смотреть программу». Разделы назначения добавим в следующих шагах.

07 Шаг 4. Добавьте три карточки программы

Webkit. Страница о себе. Прочитайте объяснение заголовков, списков и href. Покажите в своём коде h1, h2, h3 и ссылку к #program. Полное упражнение с изображением сейчас не требуется.

Файл: index.html. Найдите единственный закрывающий тег </main>. Вставьте блок ниже **непосредственно перед ним**. Не вставляйте его после </html>.

```
<section id="program" class="section">
  <h2>Программа мастерской</h2>
  <p>Три шага от текста к готовой странице.</p>
  <div class="cards">
    <article class="card">
      <p class="step">01 / СТРУКТУРА</p>
      <h3>Размечаем HTML</h3>
      <p>Создаём заголовки, абзацы, списки и ссылки.</p>
    </article>
    <article class="card">
      <p class="step">02 / ОФОРМЛЕНИЕ</p>
      <h3>Подключаем CSS</h3>
      <p>Настраиваем цвета, размеры и отступы.</p>
    </article>
    <article class="card">
      <p class="step">03 / ПРОВЕРКА</p>
      <h3>Проверяем результат</h3>
      <p>Испытываем ссылки и узкий экран.</p>
    </article>
  </div>
</section>
```

Почему структура именно такая

h2 открывает раздел программы, h3 - заголовок каждой карточки. article объединяет отдельный пункт программы с собственным заголовком и описанием. div class="cards" группирует карточки для размещения средствами CSS.

Проверка №3

После сохранения появились раздел «Программа мастерской» и три карточки. Пока они выглядят как обычный текст друг под другом. Нажмите «Смотреть программу»: браузер должен перейти к этому разделу, а адрес закончиться на #program.

Связь перехода: href="#program" у ссылки и id="program" у section. Регистр букв и написание должны совпасть.

08 Шаг 5. Укажите место, время и автора

Файл: index.html. Снова найдите </main>. Добавьте перед ним этот раздел, после раздела программы:

```
    <section id="details" class="section details">
      <h2>Место и время</h2>
      <p>
        <strong>15 ноября 2026 года, 12:00.</strong>
        Аудитория 305.
      </p>
      <h3>Что подготовить</h3>
      <ul>
        <li>Редактор обычного текста.</li>
        <li>Браузер для просмотра страницы.</li>
        <li>Папку для сохранения двух файлов.</li>
      </ul>
      <p>Все сведения о мероприятии вымышлены.</p>
    </section>
```

Теперь добавьте подвал **после </main>, но перед </body>**:

```
  <footer class="site-footer">
    <div class="container">
      <p>Учебная страница. Автор: студент группы ИС-31.</p>
      <a href="#top">К началу страницы</a>
    </div>
  </footer>
```

Проверка №4

На странице есть дата 15 ноября 2026 года, время 12:00, аудитория 305 и список из трёх пунктов. Ссылка меню «Место и время» ведёт к #details. Внизу есть автор и ссылка к началу основного содержимого.

ul создаёт список без номеров; li - один пункт. strong выделяет важность. footer содержит заключительную информацию страницы. Закрывающий </main> должен быть один: все три section находятся до него.

HTML готов. Содержание должно быть понятно и без CSS. Сохраните оба файла перед оформлением.

09 Шаг 6. Подключите базовое оформление

Webkit. Оформите визитку. В редакторе учебника создайте .card и задайте три свойства из его задания. Проверьте решение и запишите результат. Затем вернитесь к своему styles.css: значения для макета приведены ниже.

Файл: styles.css. Вставьте весь блок в пока пустой CSS-файл. Теги <style> сюда не нужны: их используют внутри HTML, а у нас отдельный файл.

```
* {
  box-sizing: border-box;
}
body {
  margin: 0;
  font-family: Arial, sans-serif;
  font-size: 1rem;
  line-height: 1.6;
  color: #202434;
  background: #f5f3fa;
}
.container {
  width: 100%;
  max-width: 68rem;
  margin-inline: auto;
  padding-inline: 1rem;
}
h1, h2, h3 {
  line-height: 1.2;
}
a {
  color: #493786;
}
a:focus-visible {
  outline: 3px solid #9a3412;
  outline-offset: 4px;
}
```

Правило	Что оно меняет
box-sizing: border-box	В заданную ширину входят padding и border; margin остаётся снаружи.
.container	Ограничивает ширину контента и оставляет внутренние поля.
1rem	Единица относительно размера шрифта корневого html.
a:focus-visible	Заметная рамка при клавиатурном фокусе.

Проверка №5: фон стал светлым, текст - шрифтом Arial, ссылки - фиолетовыми. Если ничего не изменилось, сначала проверьте сохранение, имя styles.css и строку link в HTML.

10 Шаг 7. Оформите навигацию и главный блок

Файл: styles.css. Добавьте блок **в конец файла**, после закрывающей скобки предыдущего правила. Предыдущий CSS сохраняйте.

```
.site-header {
  background: #ffffff;
  border-bottom: 1px solid #d9d3e7;
}
.header-row {
  display: flex;
  flex-wrap: wrap;
  align-items: center;
  justify-content: space-between;
  gap: 1rem;
  padding-block: 1rem;
}
.brand {
  font-weight: 700;
  text-decoration: none;
}
nav {
  display: flex;
  flex-wrap: wrap;
  gap: 1rem;
}
.hero {
  padding-block: 3rem;
}
.eyebrow, .step {
  color: #493786;
  font-weight: 700;
}
h1 {
  max-width: 18ch;
  font-size: 2.5rem;
}
.lead {
  max-width: 56ch;
}
.button {
  display: inline-block;
  padding: 0.75rem 1.25rem;
  background: #493786;
  color: #ffffff;
  border-radius: 0.6rem;
  text-decoration: none;
}
.button:hover {
  background: #342462;
}
```

Webkit. Панель навигации. Сверьте смысл display: flex, gap и flex-wrap с объяснением учебника. В нашем макете Flex задаёт .header-row и nav. Полное упражнение Webkit здесь не требуется.

11 Шаг 8. Оформите карточки и нижнюю часть

Webkit. Карточка с отступами. Прочитайте разницу между padding и margin. В своём макете найдите внутренний отступ .card и внешний отступ .details. Точные размеры упражнения Webkit в этот файл переносить не нужно.

Файл: styles.css. Добавьте этот блок в конец, после правила .button:hover. Не помещайте новые правила внутрь его фигурных скобок.

```
.section {
  padding-bottom: 2rem;
}
.cards {
  display: grid;
  grid-template-columns: repeat(3, minmax(0, 1fr));
  gap: 1rem;
}
.card {
  padding: 1.25rem;
  background: #ffffff;
  border: 1px solid #d9d3e7;
  border-radius: 0.75rem;
}
.card h3 {
  margin-top: 0;
}
.details {
  padding: 1.5rem;
  margin-bottom: 2rem;
  background: #ffffff;
  border-left: 0.3rem solid #493786;
  border-radius: 0.75rem;
}
.site-footer {
  padding-block: 1.5rem;
  background: #eae6f3;
}
```

Как читается сетка

display: grid включает сетку. **repeat(3, minmax(0, 1fr))** создаёт три равные колонки, которым разрешено сжиматься. **gap** задаёт промежутки. В Webkit: Галерея карточек тот же принцип показан для .gallery; у нас сетка называется .cards.

Не путайте отступы

padding отделяет текст от собственной рамки карточки; **margin** отделяет блок от соседей. border рисует границу, border-radius скругляет углы. Значения здесь выбраны для учебного оформления, их можно осмысленно менять.

Проверка №6: на широком окне три карточки стоят в ряд, ниже виден белый блок места и времени, затем подвал. На узком окне пока может быть тесно - это исправит следующий шаг.

12 Шаг 9. Сделайте страницу адаптивной

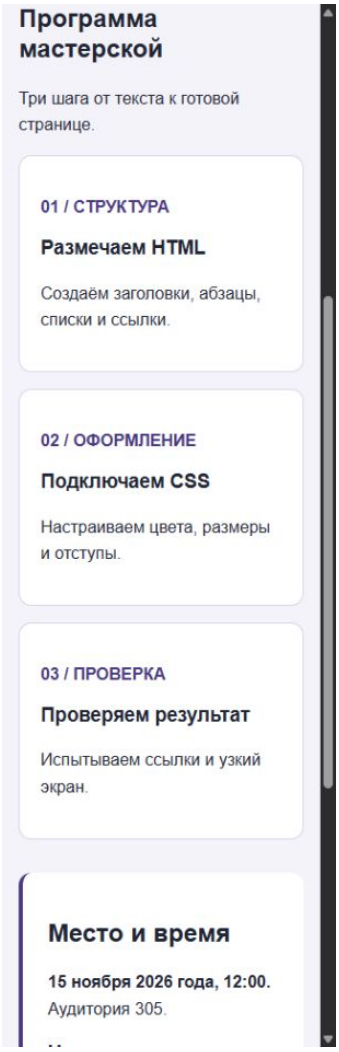
Webkit. Мобильная версия. Прочитайте объяснение viewport и @media. В своём файле найдите такое же назначение строк. В учебнике пример использует 600px, здесь - 44rem: это разные настройки макетов.

Файл: styles.css. Добавьте этот блок в самый конец, после всех остальных правил:

```
@media (max-width: 44rem) {
  .cards {
    grid-template-columns: 1fr;
  }
  h1 {
    font-size: 2rem;
  }
}
```

Медиазапрос @media применяет вложенные правила при выполнении условия. При ширине области просмотра не больше 44rem остаётся одна колонка, а главный заголовок уменьшается. 44rem - точка перестроения нашего макета, а не универсальная норма. Проценты задают гибкую ширину, max-width ограничивает её сверху. Содержание должно оставаться читаемым между контрольными размерами.

Проверка: сузьте окно. Карточки идут друг под другом; меню переносится. Справа не должно появляться обрезанного текста. Фрагмент справа показывает мобильную версию.



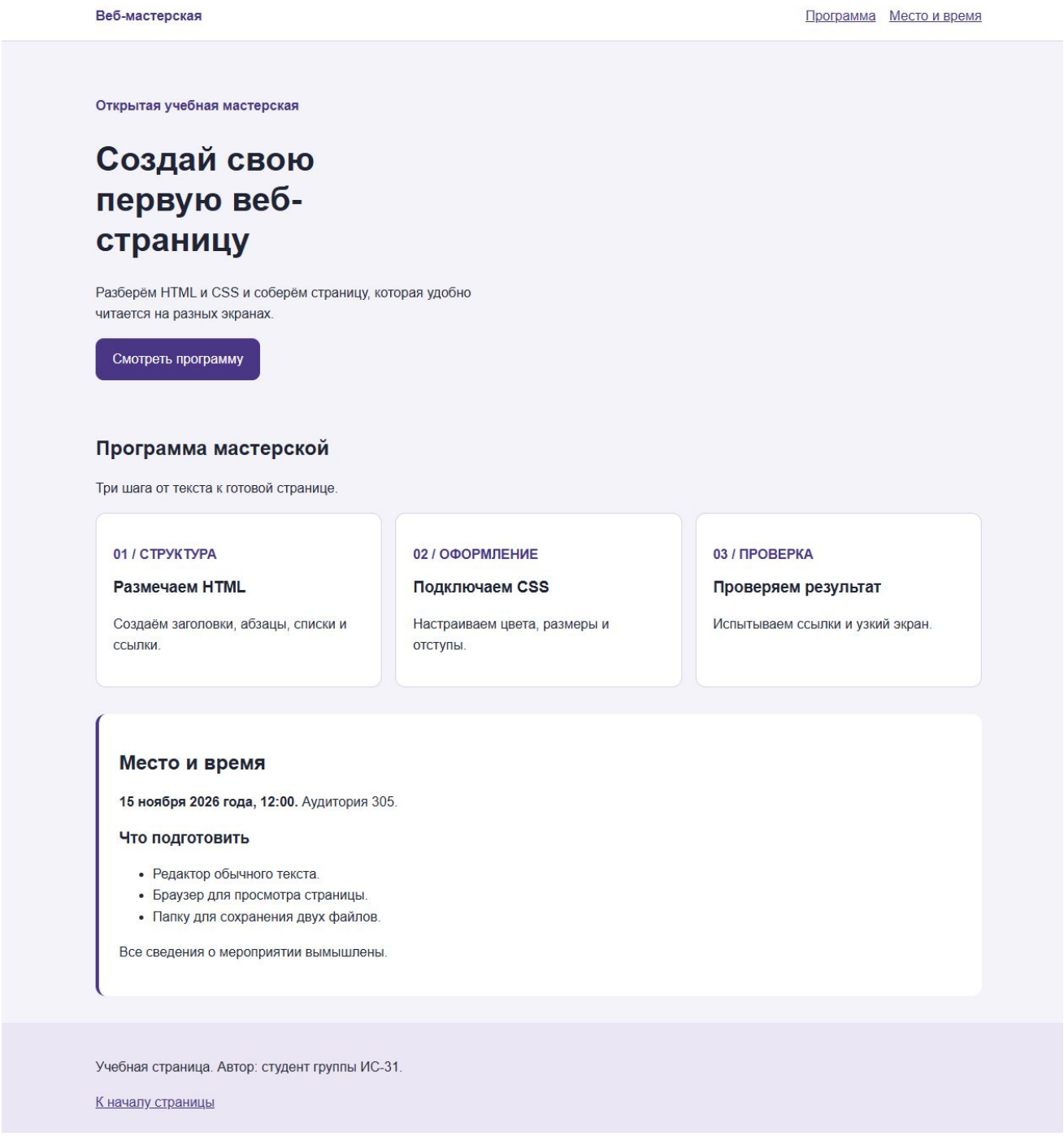
Как проверить точную ширину

В Edge или Chrome откройте инструменты разработчика клавишей F12. Нажмите Ctrl+Shift+M или значок телефона/планшета. Выберите изменяемый размер (Responsive / «Адаптивный»), в поле ширины введите 320, масштаб страницы оставьте 100%. Прокрутите страницу целиком. Затем проверьте ширину 1280. Названия пунктов могут различаться.

Закройте инструменты. Отдельно увеличьте страницу до 200% через меню браузера или Ctrl++. Проверьте чтение и ссылки, затем Ctrl+0 верните 100%. Это учебные ручные проверки, а не полный аудит доступности.

13 Ориентир: готовая базовая страница

Так выглядит образец до самостоятельных изменений при ширине окна 1280 CSS-пикселей. Шрифты и переносы могут немного отличаться в другом браузере. Сравнивайте структуру и работоспособность.



Что должно совпасть

Шапка с двумя пунктами меню; главный заголовок и переход к программе; три карточки; точные дата, время и аудитория; список подготовки; подвал с автором. В узком окне карточки перестраиваются в одну колонку.

Теперь выполните самостоятельную часть. Простого копирования образца недостаточно: итог должен содержать ваши изменения и результаты проверок.

14 Шаг 10. Выполните самостоятельные изменения

Индивидуально, 10 минут. Выберите одну тему: клуб веб-разработки, мастерская цифрового дизайна или встреча начинающих разработчиков. Все сведения по-прежнему учебные. Дату, время и аудиторию из брифа сохраните.

Обязательные изменения

1. Замените название в шапке, title вкладки, h1 и вводный абзац под выбранную тему. Измените подпись автора: свои фамилия и группа. Не убирайте отметку об учебном характере страницы.

2. Переименуйте третью карточку и напишите собственное содержательное описание из одного-двух предложений. Добавьте в список подготовки четвёртый пункт, который связан с вашей темой.

3. Добавьте в навигацию ссылку «Подготовка», которая ведёт к заголовку «Что подготовить». Самостоятельно согласуйте href ссылки и id заголовка. Образец записи - ниже.

4. В CSS измените внутренний отступ карточки с 1.25rem на 1.5rem. Перед обновлением запишите, что должно измениться; после обновления сравните ожидание с результатом.

Образец для ссылки - это две разные позиции в HTML

```
<!-- Внутри nav, рядом с двумя ссылками -->
<a href="#prepare">Подготовка</a>

<!-- Вместо прежнего заголовка в разделе details -->
<h3 id="prepare">Что подготовить</h3>
```

Сохраните файлы. Проверьте новую ссылку мышью и клавиатурой. Не создавайте второй заголовок «Что подготовить»: добавьте id к существующему.

Если закончили раньше

Измените количество широких колонок с трёх на две и объясните, что произойдёт с третьей карточкой. После проверки верните три колонки. Это дополнительный опыт, он не заменяет обязательные изменения.

В отчёте покажите одну свою строку HTML и одно своё правило CSS и объясните их назначение.

15

Шаг 11. Проверьте страницу и исправьте ошибки

10 минут. Сначала сохраните файлы. Для каждого теста выполните действие и отметьте «пройдено» или «исправлено». После исправления повторите тот же тест.

Тест	Действие	Ожидаемый результат
1. Файлы	Откройте index.html из папки проекта.	Страница открывается, CSS применяется; второй файл рядом.
2. Содержание	Сверьте дату, время, аудиторию.	15.11.2026, 12:00, аудитория 305; нет подмены фактов.
3. Структура	Найдите h1, h2 и h3 в коде.	Один главный h1 в нашем макете; разделы h2, карточки h3; списки через ul/li.
4. Переходы	Нажмите ссылки меню и ссылку в главном блоке.	Переход к нужному разделу; Подготовка ведёт к соответствующему заголовку.
5. Клавиатура	Нажимайте Tab и Shift+Tab, на ссылке Enter.	Фокус виден; к ссылкам можно дойти и открыть их без мыши.
6. Ширина	Проверьте 1280 и 320 CSS-пикселей.	Три колонки на широком окне, одна на узком; текст не требует прокрутки вправо.
7. Увеличение	Через меню браузера задайте масштаб 200%.	Текст и ссылки остаются доступны; нет перекрытий и обрезки. Затем верните 100%.
8. Свои правки	Сверьте четыре задания со стр. 14.	Тема и автор изменены, карточка и пункт списка свои, ссылка добавлена, padding объяснён.

Дополнительная автоматическая проверка, если доступен интернет

Откройте Nu HTML Checker. В списке **Check by** выберите **text input**, вставьте содержимое своего index.html и нажмите **Check**. Исправьте ошибки разметки и запустите проверку ещё раз. Валидатор не подтверждает правильность даты и удобство страницы.

Для сдачи сделайте два снимка: общий вид широкого окна и узкое окно с карточками. Ширину укажите в отчёте.

16 Если не получилось: найдите конкретную причину

Что наблюдаете	Что проверить и исправить
Браузер показывает код или открывается Блокнот	Проверьте имя index.html без .txt. В Проводнике: Открыть с помощью → браузер. Редактировать продолжайте в текстовом редакторе.
CSS не применяется	Оба файла в одной папке? В HTML есть link rel="stylesheet" href="styles.css" внутри head? Проверьте точное имя, сохраните оба файла и нажмите Ctrl+F5.
Изменилась не та страница	Сверьте путь к файлу в адресной строке браузера и папку в редакторе. Возможно, открыты две копии проекта.
Один стиль не работает	Проверьте точку у селектора класса, двоеточие, значение и фигурные скобки. В HTML class="card", в CSS .card. Следующее правило не должно оказаться внутри предыдущего.
Нажатие на ссылку не даёт нужного перехода	Сопоставьте href="#prepare" и id="prepare". Убедитесь, что такой id один. Символ # нужен в href, но не в значении id.
Карточки не стоят в ряд	У родителя есть class="cards", у CSS - .cards с display: grid. На ширине до 44rem одна колонка задана намеренно; проверьте широкое окно.
Карточки не перестраиваются	Проверьте @media в конце CSS, условие max-width и обе пары скобок. Убедитесь, что в head сохранён meta viewport.
Текст уходит за правый край	Уберите случайно добавленную фиксированную ширину. Сохраните width: 100%, max-width и box-sizing: border-box. Сверьте grid-template-columns с образцом.
Русские буквы испорчены	Пересохраните оба файла в UTF-8; в head должна быть строка meta charset="utf-8".
Пропала часть текста после вставки	Проверьте место вставки, совпадение открывающих и закрывающих тегов. Все видимые разделы внутри body; section внутри main; footer после main.

Порядок исправления

1. Назовите наблюдаемую ошибку.
2. Найдите соответствующую строку в коде.
3. Измените одну причину.
4. Сохраните и обновите.
5. Повторите исходный тест.

Если пробный код скопировался с заголовком PDF или номером страницы, удалите лишний обычный текст. В HTML не заменяйте прямые кавычки на типографские «ёлочки».

17 Шаг 12. Подготовьте результат к сдаче

5 минут. В папке Web_Фамилия оставьте index.html и styles.css. Создайте короткий отчёт в доступном формате: TXT, DOCX или PDF. Добавьте два снимка отдельно либо в отчёт. Перед передачей откройте HTML ещё раз из этой папки.

Шаблон короткого отчёта

Автор, группа: ...
Тема страницы: ...
Webkit, уроки 1 и 4: результаты проверок ... / 100; ... / 100.
Что перенёс из учебника в свою страницу: ...
Своя строка HTML и назначение частей: ...
Изменённое правило CSS: ...
Ожидал → получил после изменения padding: ...
Тесты 1-8: пройдено / исправлено, что именно.
Одна найденная ошибка и повторная проверка: ...
Снимки: широкое окно 1280, узкое 320 CSS-пикселей.

Оценивание: максимум 10 баллов

Критерий	За что даются 2 балла
HTML и содержание	Верные факты, понятная структура, корректные теги и уровни заголовков.
CSS	Внешний файл подключён, оформление работает; отступы и сетка заданы осмысленно.
Переходы и адаптация	Работают ссылки и видимый фокус; узкое окно не обрезает текст.
Самостоятельные изменения	Выполнены все четыре задания со стр. 14.
Проверка и объяснение	Есть два снимка, тесты, результаты Webkit и объяснение переноса HTML/CSS в свою страницу.

По каждой строке: 2 - выполнено и подтверждено; 1 - выполнено частично или не объяснено; 0 - результат отсутствует. Баллы не заменяются декоративными дополнениями. Перевод в отметку - по шкале преподавателя.

Ответьте перед сдачей

Почему class в HTML пишется без точки? Как ссылка находит раздел по id? Чем padding отличается от margin? Какое правило перестраивает карточки в одну колонку?

Работа завершена, когда страница открывается из вашей папки, проходит проверки, а вы можете объяснить хотя бы одну свою строку HTML и одно правило CSS.

18 Справочные источники и быстрый поиск по шагам

Основной учебник: https://kait20.tech/webkit, курс «Webkit - создание веб-сайтов». Конкретные занятия встроены в шаги и маршрут на стр. 2. Ссылки и содержание проверены 05.11.2026.

1. HTML: элементы, атрибуты и вложенность
https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Basic_HTML_syntax

2. CSS: подключение и запись правил
https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Styling_basics/Getting_started

3. CSS Grid: колонки и интервалы
https://developer.mozilla.org/en-US/docs/Web/CSS/Guides/Grid_layout/Basic_concepts

4. Медиазапросы @media
https://developer.mozilla.org/en-US/docs/Web/CSS/Reference/At-rules/@media

5. Заголовки и структура
https://www.w3.org/WAI/tutorials/page-structure/headings/

6. WCAG: перекомпоновка содержимого
https://www.w3.org/WAI/WCAG22/Understanding/reflow.html

7. WCAG: контраст текста
https://www.w3.org/WAI/WCAG22/Understanding/contrast-minimum.html

Навигация по работе

Нужное действие	Страница
Маршрут Webkit / термины / файлы	2 / 3 / 4
Каркас и содержимое HTML	5-8
Базовый CSS и оформление блоков	9-11
Адаптивная версия / вид результата	12 / 13
Самостоятельные изменения	14
Проверка / исправление / сдача	15 / 16 / 17

Пример использует обычные HTML и CSS, без JavaScript и внешних библиотек. Показанные размеры, цвета и точка перестроения выбраны для учебного макета. Ручные проверки охватывают отдельные требования, а не доказывают полное соответствие WCAG.