**HTML. Урок 11: Основы CSS для верстальщика**

**📌 Цели урока**

* Понять роль CSS в современном вебе и его базовые принципы.
* Научиться подключать CSS к HTML, использовать селекторы и базовые свойства.
* Освоить способы задания цвета, шрифтов, отступов, рамок, фонов и размеров.
* Научиться оформлять кнопки, ссылки, списки, формы.
* Получить представление о структуре CSS-файлов и стилевых методологиях.

**🧠 Теория**

**1. Что такое CSS и зачем он нужен?**

**CSS** (Cascading Style Sheets — каскадные таблицы стилей) — язык, который “раскрашивает” HTML-страницу, делает её красивой, удобной и современной.  
HTML отвечает за структуру, а CSS — за оформление, внешний вид, анимацию и адаптацию под разные устройства.

**2. Способы подключения CSS**

* **Внешний файл** (рекомендуется):

html

Копировать код

<link rel="stylesheet" href="styles.css">

* **Внутренний стиль** (только для небольшой страницы):

html

Копировать код

<style>

body { background: #fafafa; }

</style>

* **Инлайн-стиль** (не рекомендуется):

html

Копировать код

<h1 style="color: green; font-size: 32px;">Заголовок</h1>

**3. Селекторы CSS**

* По тегу:

css

Копировать код

p { color: gray; }

* По классу:

css

Копировать код

.important { font-weight: bold; }

* По id:

css

Копировать код

#main-header { background: #444; color: white; }

* Комбинированные, вложенные, псевдоклассы (hover, focus), псевдоэлементы (::before, ::after).

**4. Базовые свойства CSS**

* **Цвет**:  
  color, background-color  
  (hex: #ff9900, rgb: rgb(255,153,0), слова: orange)
* **Шрифты**:  
  font-family, font-size, font-weight, font-style
* **Отступы**:  
  margin (внешний), padding (внутренний)
* **Размеры**:  
  width, height, единицы px, %, em, rem, vw/vh
* **Рамки**:  
  border: 2px solid #777;
* **Фоны**:  
  background-image, background-repeat, background-position
* **Выравнивание**:  
  text-align, vertical-align

**5. Пример оформления сайта**

css

Копировать код

body {

font-family: Arial, sans-serif;

background: #f9f9f9;

color: #222;

margin: 0;

}

header {

background: #337ab7;

color: white;

padding: 24px 0;

text-align: center;

}

h1, h2 {

margin: 0 0 16px 0;

}

nav ul {

list-style: none;

padding: 0;

display: flex;

gap: 24px;

justify-content: center;

}

a {

color: #337ab7;

text-decoration: none;

transition: color .2s;

}

a:hover {

color: #23527c;

text-decoration: underline;

}

button {

background: #337ab7;

color: white;

border: none;

padding: 12px 22px;

border-radius: 5px;

font-size: 16px;

cursor: pointer;

}

button:hover {

background: #23527c;

}

**6. Работа со списками, таблицами, формами**

* Убираем маркеры и отступы у списков:

css

Копировать код

ul { list-style: none; padding: 0; }

* Оформляем таблицу:

css

Копировать код

table { border-collapse: collapse; width: 100%; }

th, td { border: 1px solid #bbb; padding: 8px 10px; }

th { background: #f5f5f5; }

* Оформляем input:

css

Копировать код

input, textarea, select {

border: 1px solid #aaa;

border-radius: 3px;

padding: 8px;

margin-bottom: 10px;

width: 100%;

box-sizing: border-box;

}

input:focus { outline: 2px solid #337ab7; }

**7. Комментарии и структура CSS-файла**

* Комментарии: /\* Комментарий \*/
* Группируй стили по блокам: “основа”, “шапка”, “контент”, “подвал”, “адаптация”.

**8. Современные возможности CSS**

* Псевдоклассы: :hover, :focus, :active
* Псевдоэлементы: ::before, ::after
* Медиазапросы (адаптивность):

css

Копировать код

@media (max-width: 600px) {

nav ul { flex-direction: column; gap: 8px; }

header { padding: 14px 0; }

}

* Переменные (custom properties):

css

Копировать код

:root { --main-color: #337ab7; }

header { background: var(--main-color); }

**9. Типичные ошибки и советы**

* Не используешь внешние файлы — сложно поддерживать код.
* Классы называешь “как попало” — сложно переиспользовать.
* Используешь только px, не пробуешь %, vw/vh, rem.
* Нет адаптации под мобильные (не используешь media).
* Дублируешь стили для одних и тех же элементов.
* Не сбрасываешь базовые стили браузера (normalize.css или reset.css).

**📚 Глоссарий**

| **Понятие** | **Значение** |
| --- | --- |
| CSS | Язык описания внешнего вида сайта |
| Селектор | Правило, определяющее, к каким элементам применить стиль |
| Класс | Множественное назначение стиля через class="..." |
| ID | Уникальный идентификатор |
| Свойство | Конкретная настройка (color, margin, border, и др.) |
| Каскадность | Последовательность применения стилей (важен порядок) |
| Специфичность | “Сила” селектора (id > класс > тег) |
| Медиазапрос | Условие для разных устройств (адаптивность) |
| Псевдокласс | Состояние элемента (:hover, :focus) |
| Псевдоэлемент | Виртуальная часть (::before, ::after) |

**🛠 Практика**

1. Оформи шапку сайта: фон, отступы, цвет, центрирование.
2. Стилизуй кнопки и ссылки: цвета, наведение, скругления, плавность.
3. Сделай адаптивное меню: на мобильном — вертикальное, на desktop — горизонтальное.
4. Оформи таблицу с зебра-линиями (чередование цвета строк).
5. Сверстай и оформи форму регистрации: отступы, цвета, стили для ошибок.

**🎯 Мини-проекты**

**Проект 1:**  
Сделай страницу-визитку с красивым оформлением: своя палитра, разные шрифты, адаптация для мобильных.

**Проект 2:**  
Сделай макет интернет-магазина: шапка, карточки товара, кнопки “Купить”, таблица корзины — всё стилизуй и адаптируй.

**💡 Лайфхаки**

* Используй сервис coolors.co для подбора цветовых схем.
* Не пугайся длинных CSS-файлов — главное, чтобы код был структурирован.
* Смотри на сайт через мобильный эмулятор и планшет — сразу увидишь недочёты.
* Подсматривай стиль популярных сайтов через “Инспектор” в браузере.
* Используй normalize.css для сброса базовых стилей.

**❓ Вопросы для самопроверки**

* Какие есть способы подключения CSS к HTML?
* Чем отличается класс от id?
* Как сделать ссылку синей, а при наведении — оранжевой?
* Как задать общий шрифт для всего сайта?
* Как реализовать адаптацию меню под мобильные?

**📑 Итоги**

Теперь ты умеешь:

* Подключать CSS и использовать его для оформления сайта
* Применять базовые и продвинутые стили к разным элементам
* Делать сайт красивым и удобным для всех устройств
* Быстро прототипировать и верстать собственные проекты