

Министерство образования Республики Беларусь  
Учреждение образования  
Гомельский государственный университет им. Ф. Скорины

Физический факультет

«Компьютерные сети и Web-дизайн»

# Лекция - Основы адаптивного веб-дизайна

Лектор – ст. преподаватель Грищенко В.В.

# Основы HTML 5

**<header>** - Определяет «шапку» сайта или раздела.

**<article>** - Задаёт содержание сайта вроде новости, статьи, записи блога, форума или др.

**<aside>** - Определяет блок, который не относится к основному контенту, для размещения рубрик, ссылок на архив, меток и другой информации. Такой блок, если он располагается сбоку, называется, как правило, «сайдбар» или «боковая панель».

**<figure>** - Используется для группирования любых элементов, например, изображений и подписей к ним

**<footer>** - Задаёт «подвал» сайта или раздела, в нем обычно располагается имя автора, дата документа, контактная и правовая информация

**<nav>** - Задаёт навигацию по сайту. Запрещается вкладывать **<nav>** внутрь **<address>**.

**<section>** - Определяет раздел документа, который может включать в себя заголовки, шапку, подвал и текст. Допускается вкладывать один тег **<section>** внутрь другого.

**<time>** - Помечает текст внутри тега **<time>** как дата, время или одновременно дата и время. Может указываться непосредственно внутри контейнера **<time>**, либо задаваться через атрибут `datetime`

**<hgroup>** - блок заголовков, когда например, заголовок состоит из главного заголовка и подзаголовка.

**<mark>** - выделение какого-то слова, например, совпадающего с поисковым запросом.

**<output>** - вывод программы или результаты вычислений.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Document</title>
</head>
<body>

</body>
</html>
```

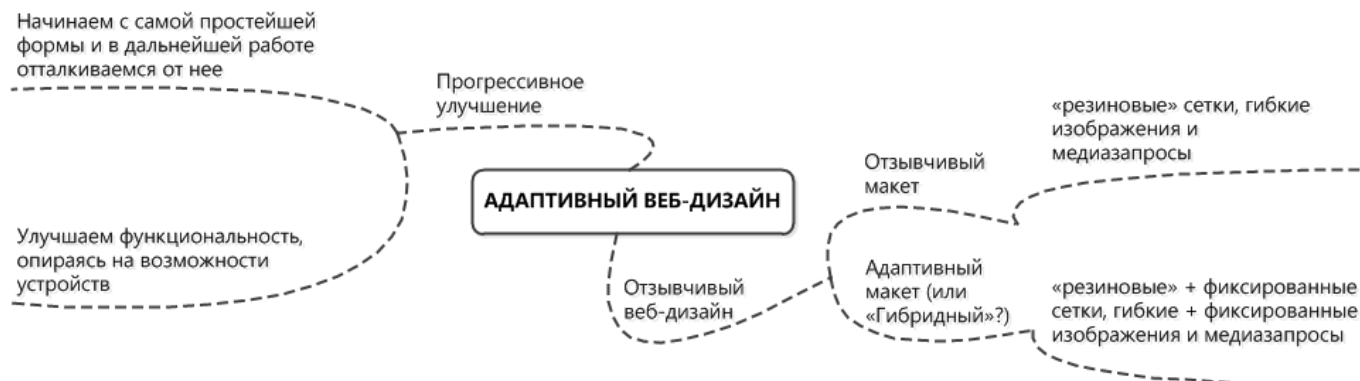
**Адаптивный веб-дизайн** (англ. Adaptive Web Design) — дизайн веб-страниц, обеспечивающий корректное отображение сайта на различных устройствах, подключённых к интернету и динамически подстраивающийся под заданные размеры окна браузера.

**Целью** адаптивного веб-дизайна является универсальность веб-сайта для различных устройств с различным разрешением и форматом, при которой не нужно создавать отдельные версии веб-сайта для отдельных видов устройств.

Исходя из этого один и тот же сайт может работать на смартфоне, планшете, ноутбуке и телевизоре с выходом в интернет, то есть на всем спектре устройств.

### Основные принципы адаптивного дизайна:

1. Гибкий макет на основе сетки (flexible, grid-based layout).
2. Гибкие изображения (flexible images).
3. Медиазапросы (media queries), модуль спецификации CSS3.
4. Применение постепенного улучшения
5. Проектирование для мобильных устройств с самых ранних этапов



### Алгоритм расчета параметров адаптивности

$$\text{(Цель/Контекст)} * 100\% = \text{Результат}$$

$$\text{(650/960)} * 100 = \text{Результат}$$

## Гибкий макет на основе сетки

Гибкие (сеточные) макеты имеют контейнеры на страницах, каждая из которых определена как ширина в процентах, т.е они полностью основаны на окне браузера пользовательского устройства, а не начальном внешнем блоке. Созданный на основании такой сетки макет подстраивается под различные размеры окна браузера.

## Пример гибкой сетки Bootstrap

.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1
.col-md-8								.col-md-4			
.col-md-4				.col-md-4				.col-md-4			
.col-md-6						.col-md-6					

	Очень маленькие устройства Телефоны (<768px)	Малые устройства Планшеты (≥768px)	Средние устройства Настольные (≥992px)	Большие устройства Настольные (≥1200px)
<b>Поведение разметки</b>	Горизонтальный все время	Терпеть неудачу при начатии, горизонтальный выше контрольной точки		
<b>Ширина контейнера</b>	Нет (автом.)	750px	970px	1170px
<b>Класс префикса</b>	.col-xs-	.col-sm-	.col-md-	.col-lg-
<b># колонок</b>	12			
<b>Ширина колонки</b>	Авто	60px	78px	95px
<b>Промежуточная ширина</b>	30px(15px) на каждой стороне колонки			
<b>Вкладка</b>	Да			
<b>Отступ</b>	Да			
<b>Выравнивание колонки</b>	Да			

**Медиа запросы (Media Queries)** допускают для выборочной таблицы стилей для HTML документа в зависимости от особенностей пользовательского агента, такие параметры как размер окна браузера, ориентация страницы, разрешение экрана и т.д.

Современные мобильные устройства, изготовленные после выхода CSS3 (например, iPad и Android устройства) поддерживают медиа функции, в том числе максимальную ширину, ширину устройства, ориентацию и цвета. Так, называемый медиа запрос с использованием функций CSS3 для разрешения этих устройств работает просто отлично, и оно игнорируется, как доступ к старому браузеру, который не поддерживает CSS3.

Пределы для размера экрана и разрешения исчезают при использовании медиа-запросов и легки в разработке. Это позволяет нам пойти дальше и приступить к созданию сайтов, гибко приспособляющихся к различным медиа.

## @media

**All** - Все типы устройств. [по-умолчанию]

**Aural** - Речевые браузеры и программы чтения вслух.

**Braille** - Брайлевские принтеры.

**Handheld** - Мобильные устройства, такие как смартфоны, планшеты и т.д.

**Print** - Принтеры.

**Projection** - Проекторы.

**Screen** - Экран монитора.

**Tv** - Телевизор.

## Синтаксис записи @media

```
@media [тип устройства] {  
    блок правил применяемый для  
    этого устройства  
}
```

```
@media screen {  
    header{...}  
    .sidebar {...}  
}
```

# Свойства устройств используемых @media

Orientation:



portrait



landscape



max-width  
min-width

```
@media and (orientation: landscape){ ... }  
@media and (orientation: portrait){ ... }
```

```
@media all and (max-width:400px){ ... }  
@media handheld and (min-width:300px){ ... }
```

## Комбинирование типов устройств и свойств

```
@media screen, handheld { ... }
```

Использование 'and'

```
@media handheld and (orientation: landscape){ ... }
```

```
@media handheld and (min-width: 320px) and (max-  
width: 660px) { ... }
```

**Гибкие изображения (flexible images)** одним из важных вопросов при работе с гибкой конструкцией, при чем пропорциональное изменение размера изображения может быть достигнуто различными способами.

**Масштабирование** - это жесткий способ привести изображение к мобильному устройству.

**Использование CSS max-width** - самый простой и самый распространенный способ на данный момент.

**Обрезка изображения с CSS overflow** - способ позволяющий обрезать изображения динамически, и контейнеры вокруг них сдвигаются, чтобы соответствовать новым условиям показа.

Для кроссбраузерного использования свойств **min-width**, **max-width** необходимо сообщить браузеру что ширина области просмотра, равняется ширине экрана данного устройства. Для этого используется **meta** тег:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

**width = device-width** – этим правилом мы сообщаем браузеру что ширина области просмотра, равняется ширине экрана данного устройства

**initial-scale=1.0** – не масштабировать

# Инструменты для создания адаптивных сайтов

**Html5 Shiv** — плагин языка программирования JavaScript, позволяющий обеспечивать базовые стили HTML5 для Internet Explorer 6-9, Safari 4.x, Firefox 3.x. и iPhone 3.x.

<https://github.com/aFarkas/html5shiv>



**Pie** — это набор CSS-функций, которые вы можете прикрепить к любому файлу CSS, учитывая определенные правила, за счет которых реализуется поддержка функций в Internet Explorer 6, 7 или 8. Поддерживаются: **border-radius**; **box-shadow**; **border-image**; множественные фоновые изображения; **linear-gradient** в качестве фонового изображения.

<http://css3pie.com>

```
<!DOCTYPE html>
<html>
  <head>
    <!--[if lt IE 9]>
      <script src="//cdnjs.cloudflare.com/ajax/libs/html5shiv/r29/html5.min.js"></script>
    <![endif]-->
  </head>
  <body>
  </body>
</html>
```

**jQuery** — библиотека JavaScript, фокусирующаяся на взаимодействии JavaScript и HTML. Библиотека jQuery помогает легко получать доступ к любому элементу DOM, обращаться к атрибутам и содержимому элементов DOM, манипулировать ими.

**Twitter Bootstrap** - свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML и CSS шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейсов, включая JavaScript расширения.