

## **АДАПТИВНЫЙ ДИЗАЙН С ПОМОЩЬЮ МЕДИА-ЗАПРОСОВ В CSS**

**Андросова Татьяна Евгеньевна**

студент, Самарский университет, РФ, г. Самара

**Курочкин Владислав Михайлович**

студент, Самарский университет, РФ, г. Самара

**Болдырев Артем Сергеевич**

студент, Самарский университет, РФ, г. Самара

**Чернов Роман Вячеславович**

студент, Самарский университет, РФ, г. Самара

### **Использование media queries**

Media queries были добавлены в CSS3. Это больше не свойства, а правила, которые можно применить в определённых условиях. В частности, вы можете сказать: «Если разрешения экрана пользователя меньше некоторого значения, тогда будут применены следующие CSS свойства». Это позволяет изменять внешний вид сайта при определенных условиях: вы можете увеличить размер текста, изменить цвет фона, изменять отображение вашего меню в зависимости от разрешений экрана и т.д.

Более того, media queries позволяют изменять внешний вид сайта не только по критерию разрешения экрана, но и в зависимости от типа экрана (смартфон, телевизор, проектор), от количества цветов, поддерживаемых экраном, от ориентации экрана (книжная, альбомная) и т.д. Критериев очень много!

### **Применение media query**

Медиазапросы – это правила, которые указывают, когда должны применяться те или иные свойства CSS. Есть два способа использовать их:

- Загружая другой файл со стилями .cssна основе правила (например, «Если ширина экрана меньше 1280px, то загрузи файл petite\_resolution.css»);
- Написав правила прямо в действующий файл .css(например, «Если ширина экрана меньше 1280px, то используй следующие свойства»).

### **Изменение загрузки файла .css**

Стандартный способ подключения .css (рисунок 1):

```
1 <link rel="stylesheet" href="style.css" />
```

*Рисунок 1. Стандартный способ подключения .css*

Можно добавить атрибут `media`, в котором прописать правило, которое будет определять, какой файл будет загружен в зависимости от условия так называемый «медиа-запрос» (рисунок 2):

```
1 <link rel="stylesheet" media="screen and (max-width: 1280px)"
  href="petite_resolution.css" />
```

*Рисунок 2. Добавление атрибута `media`*

Тем самым HTML будет иметь возможность загрузить несколько файлов CSS: один по умолчанию (который будет загружен в любом случае) или другие, которые будут загружены в отдельных случаях в зависимости выполнения условия или нет (рисунок 3).

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <link rel="stylesheet" href="style.css" /> <!-- Pour tout le monde -->
6     <link rel="stylesheet" media="screen and (max-width: 1280px)"
  href="petite_resolution.css" /> <!-- Pour ceux qui ont une résolution inférieure à
  1280px -->
7     <title>Media queries</title>
8   </head>
```

*Рисунок 3. Загрузка нескольких файлов*

### Изменения непосредственно в файле

Другой метод, который я лично предпочитаю, по практическим причинам, чтобы написать эти правила в одном CSS файл, как обычно (рисунок 4):

```

1 @media screen and (max-width: 1280px)
2 {
3     /* Rédigez vos propriétés CSS ici */
4 }

```

### *Рисунок 4. Изменение в файле*

В этом случае мы пишем свойство непосредственно в файл .css, как здесь:

#### **Существующие правила**

Существует много CSS-правил, с помощью которых можно сделать медиа-запрос. Приведу несколько основных:

- color: цвет (биты/пиксели).
- height: высота (окна).
- width: ширина (окна).
- device-height: высота экрана.
- device-width: ширина экрана.
- orientation: ориентация экрана (альбомная, книжная).
- media: тип экрана. Возможные значения:
- screen: обычный;
- handheld: мобильный;
- print: печатный;
- tv: телевизионный;
- projection: проектор;
- all: все типы.

Можно приписать префиксы min-или max-перед большинством свойств. Так, например, min-width обозначает «минимальная ширина», max-height «максимальная высота», и т.д.

Разница между width и device-width воспринимается больше на смартфонах и мобильных устройствах, но об этом чуть позже.

Правила можно комбинировать с помощью:

- only: «только»;
- and: «и»;
- not: «нет».

Несколько примеров использования правил (рисунок 5).

```

1 /* Sur les écrans, quand la largeur de la fenêtre fait au maximum 1280px */
2 @media screen and (max-width: 1280px)
3
4 /* Sur tous types d'écran, quand la largeur de la fenêtre est comprise entre 1024px
   et 1280px */
5 @media all and (min-width: 1024px) and (max-width: 1280px)
6
7 /* Sur les téléviseurs */
8 @media tv
9
10 /* Sur tous types d'écrans orientés verticalement */
11 @media all and (orientation: portrait)

```

*Рисунок 5. Пример встраивания в файл*

## Проверка работы

Проведём простой тест: давайте изменим цвет и размер текста, если окно больше или меньше, чем 1024 пикселей в ширину.

Для этого теста я буду использовать второй метод, чтобы написать правило, непосредственно в том же .css, что и обычно (рисунок 6):

```

1 /* Paragraphes en bleu par défaut */
2 p
3 {
4     color: blue;
5 }
6
7 /* Nouvelles règles si la fenêtre fait au plus 1024px de large */
8 @media screen and (max-width: 1024px)
9 {
10     p
11     {
12         color: red;
13         background-color: black;
14         font-size: 1.2em;
15     }
16 }

```

*Рисунок 6. Пример теста*

В CSS, мы сначала установили, что текст параграфов написан синим цветом (пока ничего нового). Тем не менее, мы добавили медиа-запрос, который относится ко всем экранам, ширина которых не превышает 1024px. Внутри мы применили правила CSS для параграфов, а

именно жирный красный текст.

Результат: Страница имеет различный вид в зависимости от размера окна. Внешний вид текст изменяется в зависимости от размера окна.

### **Использование медиа-запросов на практике**

Важность media query проявляется при изменении внешнего вида сайта в соответствии с разрешением. Для этого примера, я предлагаю вам обратиться к вёрстке для веб-сайта Zozog (ссылка будет доступна в списке литературы).

Сайт подходит для большинства разрешений экрана, но не очень удобен для просмотра на маленьком экране.

Необходимо использовать медиа-запросы, чтобы изменить внешний вид сайта на разрешениях менее 1024 точек в ширину. Мы сделаем следующие изменения:

- меню навигации в правом верхнем углу будет растягиваться в высоту, а не в ширину, также ссылки будут написаны меньшим шрифтом;
- баннер с мостом Сан-Франциско (Golden Gate) будет удален, поскольку он занимает много места и не несёт особую информацию;
- Блок <aside> «Об авторе» будет находиться под статьей (а не рядом), и его содержимое будет изменено (фото из Zozog будет позиционироваться, как плавающий элемент)

Можно, конечно, сделать много других изменений: изменить цвет, макет колонтитул, и т.д. Но это уже можно будет сделать в качестве хорошей тренировкой по работе с медиа-запросами.

### **Список литературы:**

1. OpenClassrooms – [Электронный ресурс] / <https://openclassrooms.com>.