

Модуль - Верстка

8/10 занятие

ФОРМЫ


ФОРМЫ

- Форма собирает информацию от пользователя и передает ее в *файл-обработчик* на сервере.
- *Обработчик - скрипт*, который например, может сформировать письмо, включив в него сообщение, полученное из формы и отправить на email администратора.
- *Обработчик* может быть написан на php, javascript, java или другом языке программирования на стороне Back-end
- В современном фронтенде данные из формы собираются javascript-ом
- *Атрибут action - расположение обработчика на сервере*

```
<form action="phpmailer/mail.php" method="POST">
```

ФОРМЫ

 Формы обрабатываются в тег `<form>` и содержат поля `<input>`, подписи к полям `<label>` и кнопки `<button>`

 `<label>` можно связать с полем ввода конструкцией из атрибутов (`for`, `id`)
При этом `<label>` будет передавать фокус курсора мышь на связанное поле

```
<form>
```

```
<label for="age">Возраст</label>
```

```
<input type="text" id="age">
```

```
</form>
```

Задание 1

Сверстать форму в модальном окне, используя теги `<form>`, `<input>`, `<label>`

Untitled.

)))

OLIVIA RHYE 06/24

1234 1234 1234 1234

Update payment method

Update your card details.

Name on card

Expiry

Olivia Rhye

06 / 2024

Card number

CVV

1234 1234 1234 1234

...

Cancel

Update

ПОЛЯ ФОРМ

Существует несколько тегов для получения информации от пользователя

- `<input>` - Тип получаемых данных задается атрибутом `type`
- `<textarea>...</textarea>` - Многострочный текст
- `<select>` - Поле с возможностью выбора из списка
 - `<option>...</option>` - Элемент списка `<select>`
- `</select>`
- `<button>...</button>` - Кнопка для отправки данных или иных действий

INPUT



Атрибут **type** этого универсального поля, указывает, какой тип данных пользователь должен вводить

text

checkbox ☒

submit

password

hidden

button

file

radio ☐

reset

color

number

time

date

range

url

datetime

search

week

email

tel

month

АТТРИБУТЫ ПОЛЕЙ

placeholder - текстовый лейбл внутри поля ввода (скрывается при введении символов)

required - браузер блокирует отправку данных, пока поле не заполнено

```
<input placeholder="Введите имя" required >
```



checked - поле изначально отмечено, как **true**

```
<input type="checkbox" checked >
```



ФУНКЦИОНИРОВАНИЕ

Все данные из полей внутри тега `<form>` передаются в виде **POST** или **GET** запроса на указанный в **action** ресурс в формате "ключ = значение"

Примерно так: "имя_пользователя" = "Иван"

Для связывания используются атрибуты **name** и **value**

```
<form action="/mail.php" method="POST" >
  <input type="text" name="username" value="">

  <input type="checkbox" name="contact" value="phone">
  <input type="checkbox" name="contact" value="email">
  <input type="checkbox" name="contact" value="pigeon_post">

  <select name="year">
    <option value="2018">2018 год</option>
    <option value="2019">2019 год</option>
    <option value="2020">2020 год</option>
  </select>
</form>
```

```
//example of DATA
username=%значение%
contact=email
year=2019
```

Задание 2

Создать файл form.html и подключить form.css, где сверстать по фигме

1. Чекбоксы с различными состояниями

<input type="checkbox"/> Remember me	<input checked="" type="checkbox"/> Remember me
<input type="checkbox"/> Remember me	<input checked="" type="checkbox"/> Remember me
<input type="checkbox"/> Remember me	<input checked="" type="checkbox"/> Remember me
<input type="checkbox"/> Remember me	<input checked="" type="checkbox"/> Remember me

2. Селектор выбора(select)

Phoenix Baker

Olivia Rhye ✓

Lana Steiner

Demi Wilkinson

Candice Wu

Natali Craig

Drew Cano

перерыв

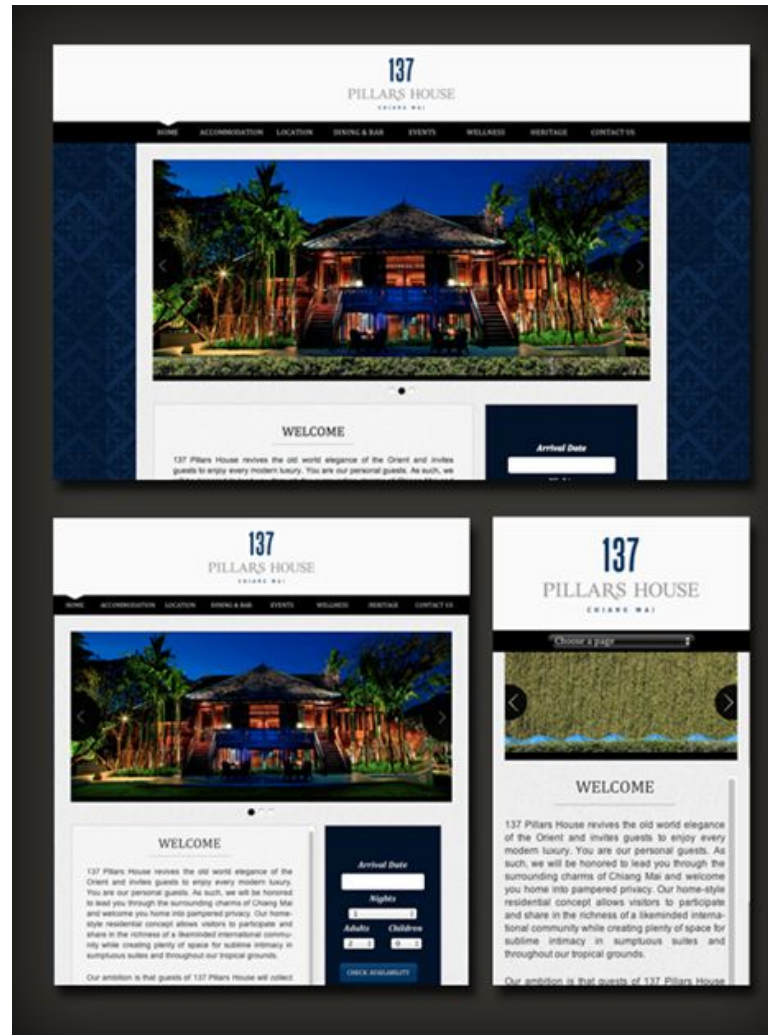
АДАПТИВНОСТЬ

Адаптивность(responsive) - Способность сайта или приложения корректно отображать контент на различных устройствах, таких как:

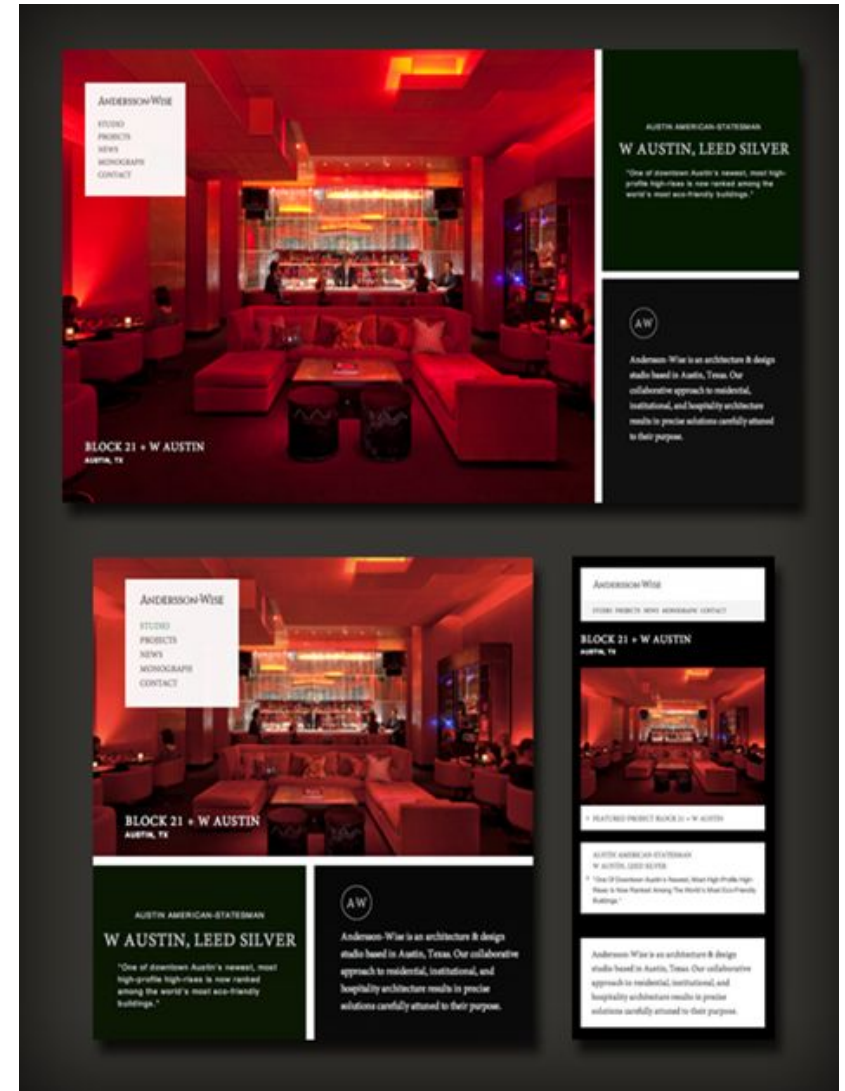
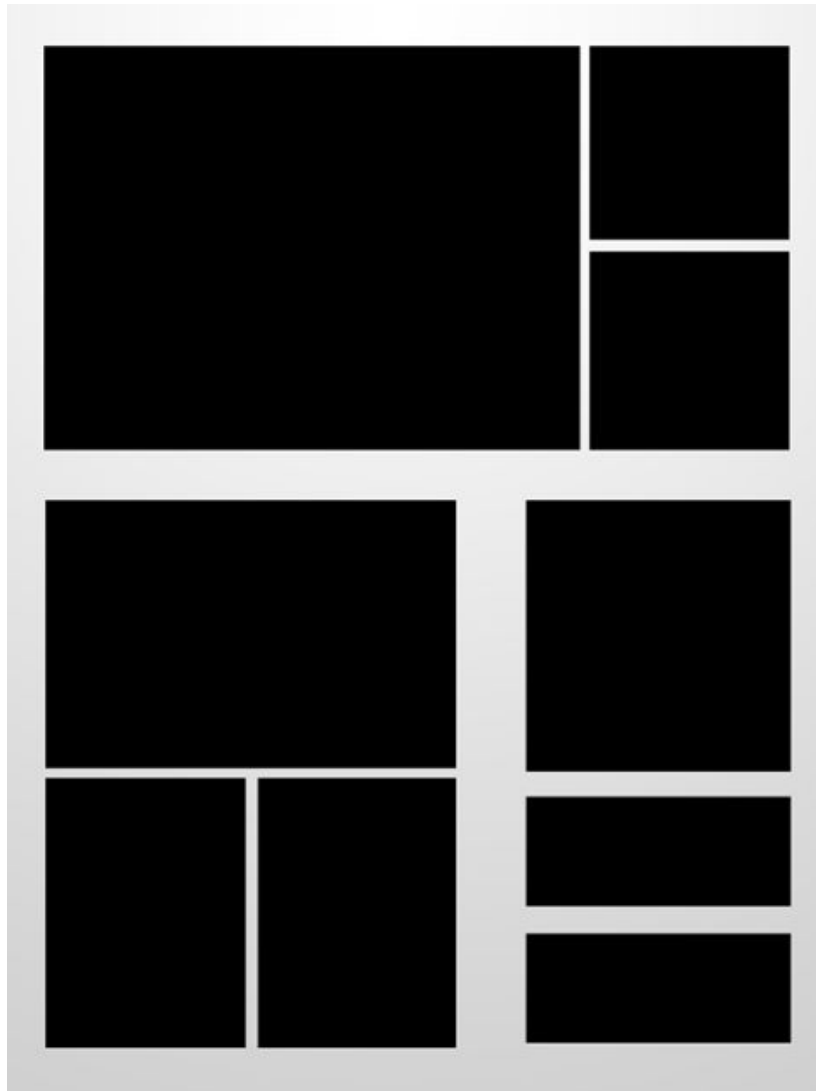
- Мониторы.
- Телевизоры;
- Планшеты;
- Смартфоны;

Концепция заключается в использовании “*фильтров*” для css правил, тем самым, задействуя или игнорируя их, в зависимости от того, на каком устройстве запущен браузер

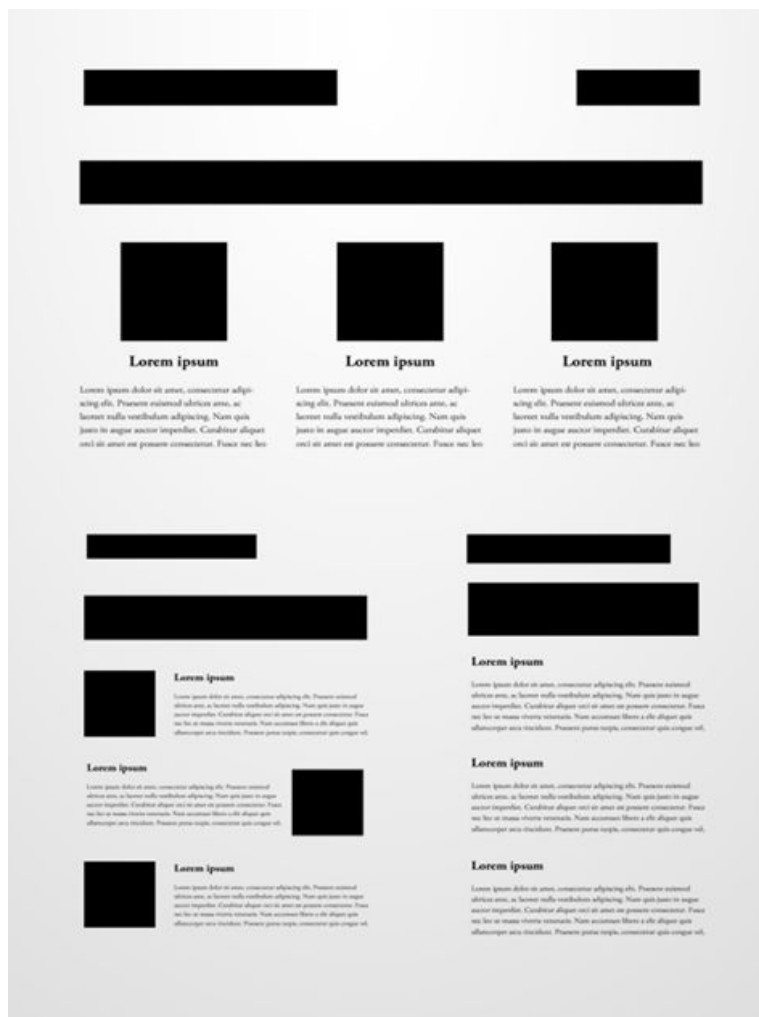
ПРОЯВЛЕНИЕ АДАПТИВНОСТИ



ОТЗЫВЧИВАЯ КОМПОЗИЦИЯ



ИЗМЕНЕНИЕ ВИДИМОСТИ И СТИЛЕЙ



МАСШТАБ ОБЛАСТИ ПРОСМОТРА

Мета-тег **viewport** определяет то, как веб-страница отображается на мобильном устройстве.

Если он не указан, ширина страницы считается равной стандартному значению для ПК, и она уменьшается, чтобы поместиться на экране.

```
<head>
  ...
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  ...
</head>
```


ФИЛЬТРАЦИЯ СТИЛЕЙ

Для определения того, какие правила будут задействованы, а какие нет, используются **медиазапросы**(**media-queries**)

Они позволяют *изменять стили* на основании *характеристик устройства*, связанных с отображением контента, включая *тип, ширину, высоту, ориентацию* и даже *разрешение экрана*.

Чаще всего используется фильтр по максимальной(**max-width: 991px**) или минимальной(**min-width: 768px**) ширине или диапазон ширин.

320px

768px

1024px

1920px

4096px



Как использовать

1. Вращеры внутри CSS: **@media <условие> { ... css-правила ... }**
2. Внутри тега **<link>**

Условие может содержать:

- типы носителей (all, screen, print, ...);
- логические операторы (and, not, only);
- медиа-функции(включающие свойства css).

CSS:

```
@media only screen and (min-width: 320px) and (max-width: 768px) {  
  .col {  
    width: 50%;  
  }  
}
```

HTML:

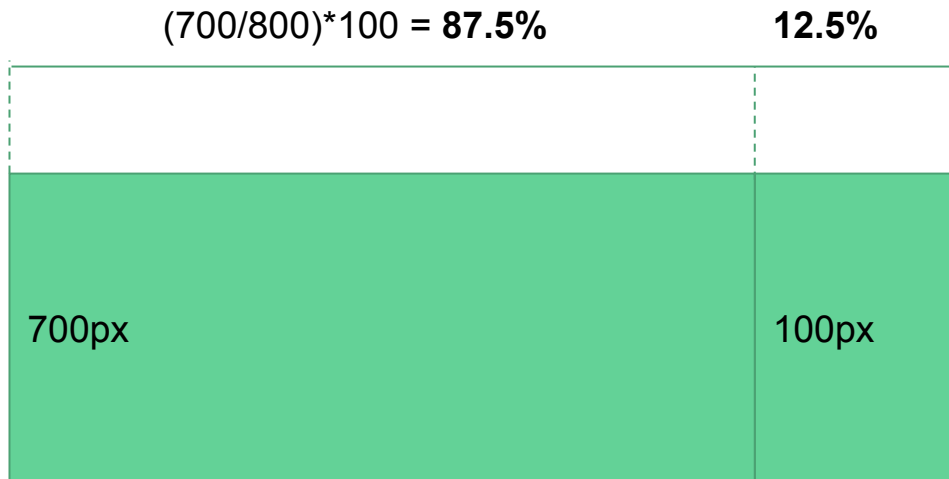
```
<link href="mobile.css" rel="stylesheet" media="screen and (max-width: 600px)" />
```

ВЫЧИСЛЕНИЕ ПРОПОРЦИЙ

$\% = (\text{element-size} / \text{parent-size}) * 100$, где

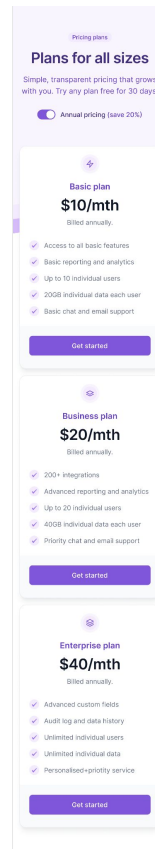
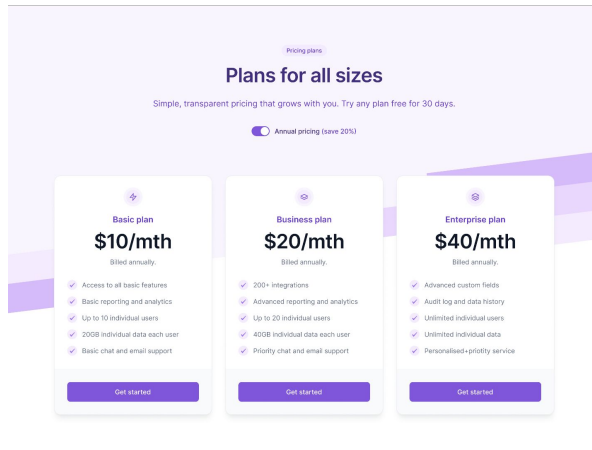
- parent-size – размер родительского контейнера в макете
- element-size – размер элемента

Единицы измерения parent-size и element-size должны совпадать!



Задание 3

В Landing-ехартпле скрыть все секции, кроме указанных на скриншоте и добавить адаптивные стили через @media-запросы в файл mobile.css



Конец занятия