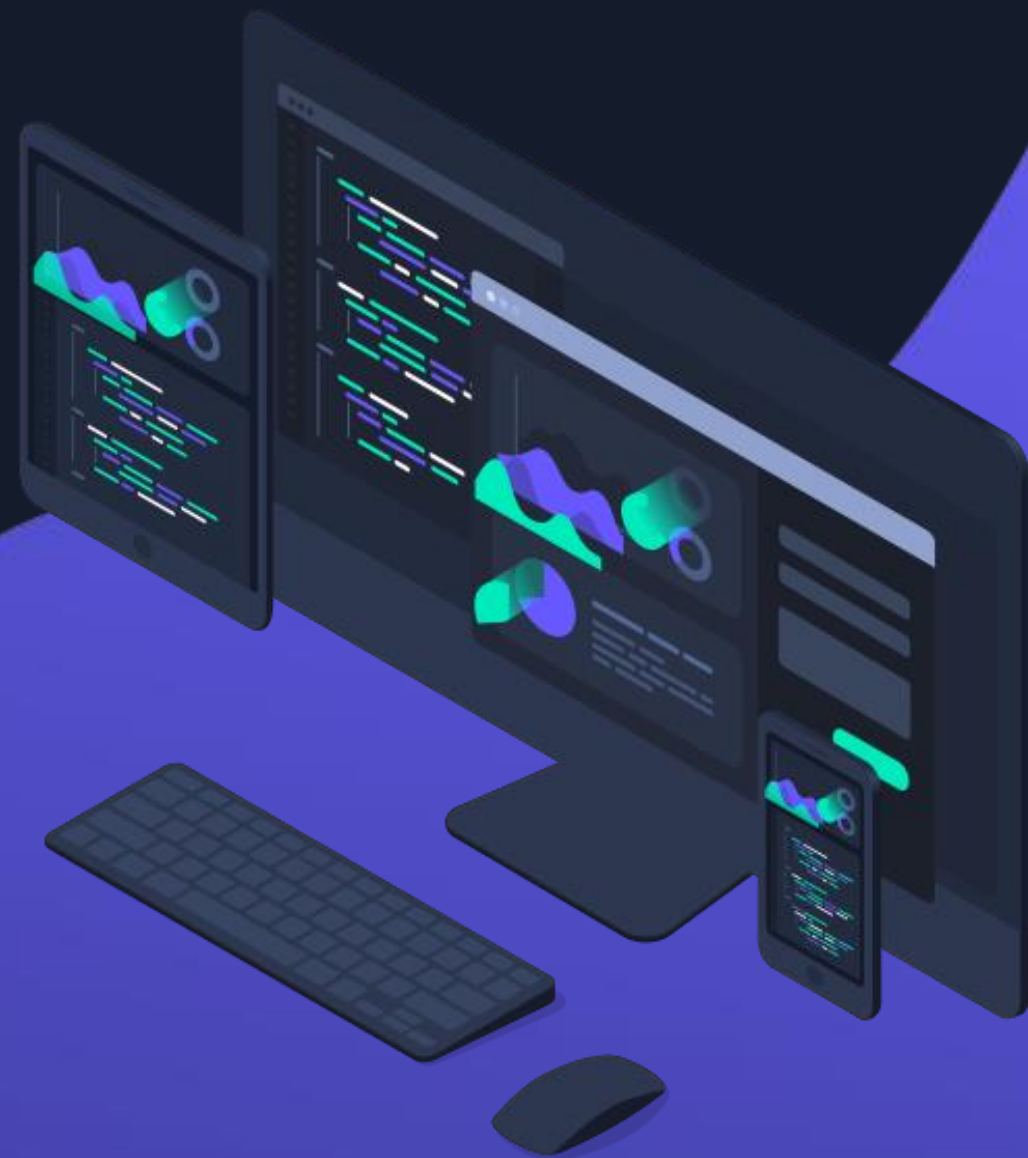


I T E A  
O N L I N E



# На сегодняшнем занятии:

1

Подготовка макетов к верстке

2

Модульная система верстки

3

8 ошибок новичков

4

Понятие метатега

5

Управляющие команды для браузера

6

Информация о странице и ее авторе

7

Команды для поисковых систем

8

Доменное имя

9

Domain Name System (DNS)

10

Хостинг

11

HTML 5

# Подготовка макетов к верстке

## Сбор информации

Даже для самого скромного задания нужно собрать базовую информацию о цели разработки и сроках выполнения.

## Мозговой штурм

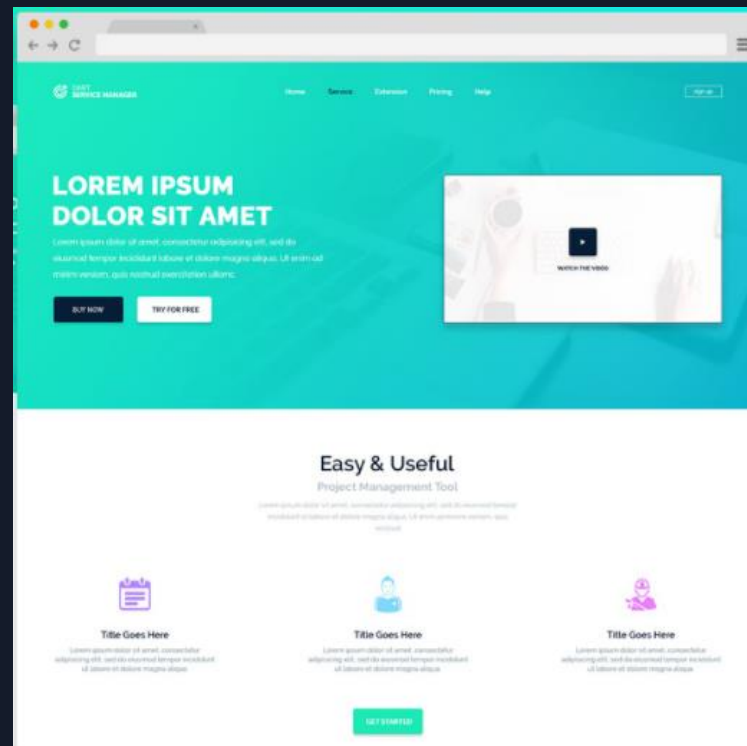
Оперативный метод решения проблемы на основе стимулирования творческой активности

## Наброски

Лучший способ быстро изучить возможности дизайна — это делать эскизы, небольшие наброски макета, которые можно быстро нарисовать и быстро отвергнуть.

## Эскизы (макеты, прототипы)

В веб-пространстве эквивалентом макета является бета-версия сайта, пока недоступная пользователям.





O N L I N E

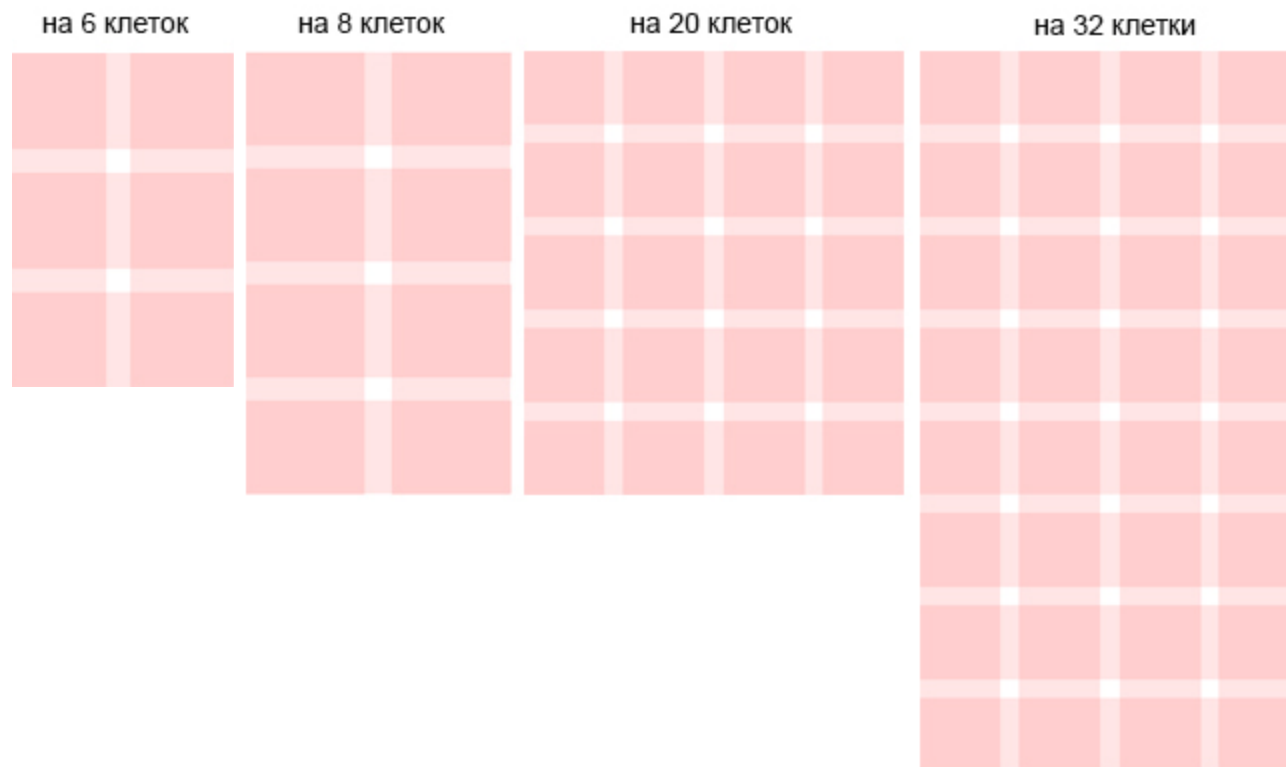
# Модульная система верстки

Модульная система вёрстки — система вёрстки, при которой основой становится модульная сетка с определенным шагом (модулем), одинаковым или разным по горизонтали и вертикали.

Задачи сетки:

- Ускорение работы: не тратится лишнее время на поиск геометрического места элемента в макете.
- Сбалансированность и пропорциональность: элементы в макете соизмеримы и пропорциональны между собой.
- Ускорение и шаблонирование, единообразие элементов: применяя сетку, создается основа для решений на будущее.

Модульные сетки



## 8 ошибок новичков

1. Все, что мигает без перерыва. Сведите моргание анимированных картинок к минимуму.
2. Деформированные фото. Пропорции фотографии должны соответствовать макету.
3. Открытые фото. Используйте волосяные линии для обводки фотографий с нечеткими границами.
4. Объекты во всех четырех углах. Нагромождать — плохо. Группировать — хорошо.
5. Перегруженный фон.
6. Издевательства над шрифтом. Дважды подумайте, прежде чем делать выворотку, набирать капсом или использовать подчеркивание и обводку.
7. Неудачные буллиты. Используйте подходящие символы и задавайте правильные отступы, выравнивая списки.
8. Коридоры. Обходитесь без коридоров в колонках, образующихся при выравнивании текста по ширине.





O N L I N E

# Понятие метатега

Метатеги — это теги языка HTML, содержащие различную служебную информацию. Метатеги размещаются в заголовке страницы (между тегами `<head>` и `</head>`).

Условно метатеги делятся на 3 группы:

- содержащие управляющие команды для браузера;
- содержащие управляющие команды для поисковых систем;
- содержащие информацию о странице и ее авторе.

```
<meta http-equiv="Content-Language" content="ru" />  
<meta name="site-created" content="01.27.2012" />  
<meta name="description" content="HTML примеры" />
```



# Управляющие команды для браузера.

На сегодняшний день наиболее важными считаются метатеги для управления браузером, которые задаются с помощью атрибута `http-equiv`.

Атрибут может принимать следующие параметры:

1. **Content-type** — указывает браузеру кодировку страницы.
2. **Content-Language** — сообщает браузеру, на каком языке написана страница.
3. **Pragma** — запрещает браузеру кэшировать страницу.
4. **Refresh** — через N секунд после загрузки текущей страницы в нее будет загружена другая URL-страница.



# Информация о странице и ее авторе

- **Generator** — сообщает, с помощью какой программы был сгенерирован код страницы.
- **site-created** — указывает дату создания страницы.
- **expires** — этот метатег указывает, когда страница будет удалена.
- **Author** — содержит имя автора страницы.
- **Copyright** — указывает владельца авторских прав.
- **Reply-to** — указывает способ связи с автором страницы.
- **Owner** — указывает собственника страницы.
- **Address** — содержит адрес автора страницы.





# Команды для поисковых систем

Когда-то эти метатеги были очень важны. Правильно используя их, можно было достаточно легко вывести свой сайт на хорошие позиции в поисковиках.

1. **description** — содержит в себе основное описание страницы, является наиболее важным из метатегов.
2. **keywords** — содержит ключевые слова, по которым будет индексироваться страница.
3. **robots** — этот метатег управляет индексированием страниц.
  - **index/noindex** — индексировать / не индексировать страницу;
  - **follow/ nofollow** — идти / не идти по ссылкам с этой страницы;
  - **all** — эквивалентно index, follow
  - **none** — эквивалентно noindex, nofollow
4. **revisit, revisit-after** — дает команду поисковой системе индексировать сайт с нужной периодичностью.





O N L I N E

# Доменное имя

У каждого web-ресурса есть свой IP Адрес — уникальный номер, который присваивается для его идентификации в Сети. Это число трудно запоминать, поэтому используются доменные имена.

**Доменное Имя** — уникальный идентификатор, который присваивается определенному IP-адресу (двух одинаковых быть не может).

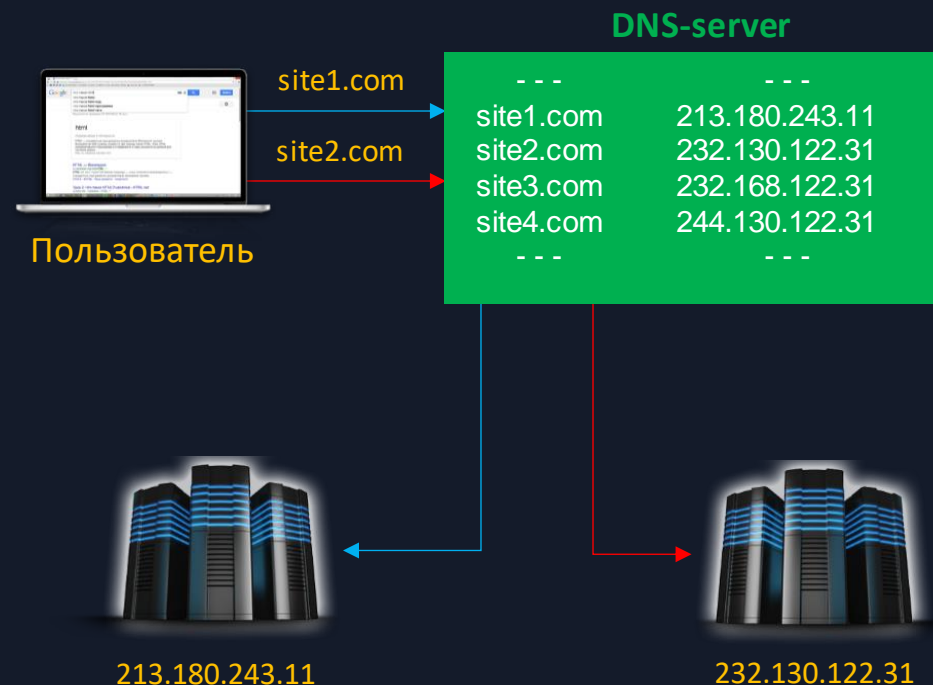


# Domain Name System

Доменные Имена обслуживаются и централизованно администрируются набором серверов доменных имен **DNS (Domain Name System)**.

Вся информация о Доменных Именах хранится в центральной базе данных DNS.

В базе данных хранится информация о дате регистрации, о физическом или юридическом владельце Доменного Имени, а также путь к так называемому серверу имен — **NAMESERVER (NS)**, где содержится информация, на которую указывает Доменное Имя.





O N L I N E

# ХОСТИНГ

**Хостинг** — услуга по предоставлению дискового пространства для физического размещения файлов сайта на сервере, постоянно находящегося в сети (обычно Интернет).

**Хостинг и домен** — это понятия, которые определяют место жительства сайта в Интернете. Хостинг обеспечивает необходимое пространство для хранения вашего сайта в интернете, а домен отвечает за его онлайн адрес.





# HTML 5

Новые семантические элементы

`<header>`, `<footer>`, `<article>`  
и `<section>`.

Новые атрибуты элементов  
формы, такие как `number`, `date`,  
`time`, `calendar`, and `range`.

Новые графические элементы:  
`<svg>` и `<canvas>`.

Новые мультимедийные  
элементы: `<audio>` и `<video>`.



## Советы при верстке

### Используйте имена атрибутов в нижнем регистре

HTML5 позволяет смешивать прописные и строчные буквы в именах атрибутов.

### Пробелы и знаки равенства

HTML5 позволяет оставлять пробелы вокруг знаков равенства. Но код без пробелов легче читать и лучше объединять в группы.

### Избегайте длинных строк кода

При использовании редактора HTML неудобно прокручивать вправо и влево, чтобы прочитать код HTML.

Старайтесь избегать строк кода длиннее 80 символов.

### Используйте кавычки для значений атрибутов

HTML5 позволяет устанавливать значения атрибутов без кавычек.

### Используйте имена тегов в нижнем регистре

HTML5 позволяет смешивать прописные и строчные буквы в именах тегов.

### Закрывайте пустые теги HTML

В HTML5 необязательно закрывать пустые теги.



---

# Q&A

I T E A  
O N L I N E

