

Частное учреждение профессионального образования «Высшая школа предпринимательства (колледж)» (ЧУПО «ВШП»)

Кафедра информационных технологий

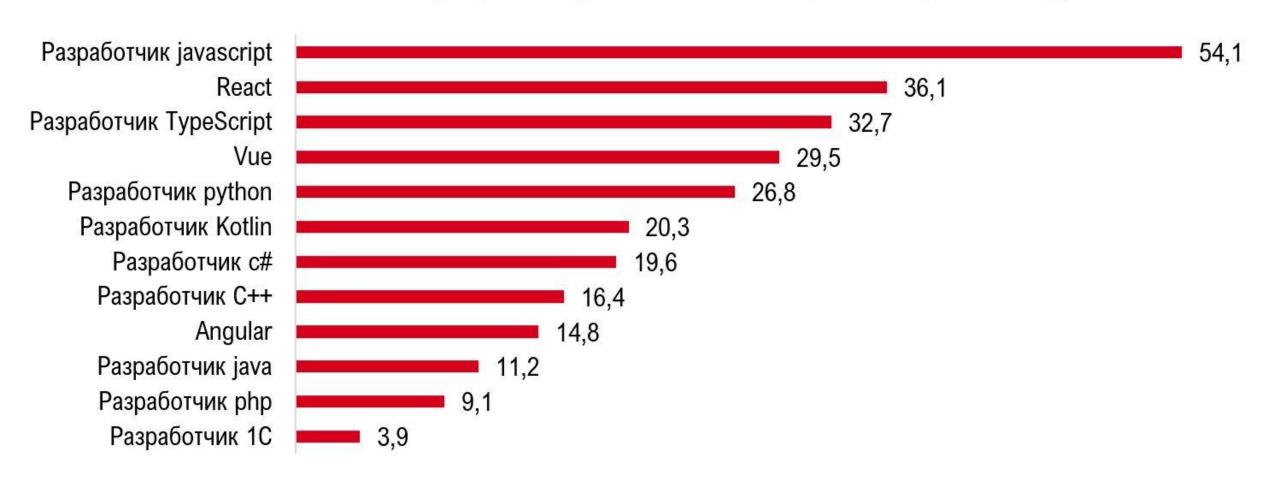
# Инструментальные средства разработки программного обеспечения

#### Уровень предлагаемых зарплат в области разработки ПО Россия, 2023 год



Регионы	Без опыта	От 1 года до 3 лет	От 3 до 6 лет	Более 6 лет
Россия	50 000 ₽	71 500 ₽	93 580 ₽	100 500 ₽
Москва	67 360 ₽	81 590 ₽	99 700 ₽	121 680 ₽
Петербург	59 000 ₽	72 700 ₽	89 300 ₽	100 000 ₽
Северо-Западный ФО	54 860 ₽	71 000 ₽	87 500 ₽	99 000 ₽
Сибирский ФО	46 100 ₽	72 800 ₽	103 000 ₽	110 000 ₽
Центральный ФО	52 700 ₽	71 570 ₽	91 770 ₽	114 370 ₽
Уральский ФО	50 630 ₽	74 820 ₽	97 110 ₽	100 000 ₽
Южный ФО	46 950 ₽	60 250 ₽	75 650 ₽	80 000 ₽
Приволжский ФО	47 840 ₽	66 760 ₽	83 630 ₽	89 100 ₽
Северо-Кавказский ФО	47 550 ₽	50 050 ₽	69 590 ₽	85 000 ₽
Дальневосточный ФО	59 970 ₽	103 700 ₽	120 540 ₽	130 000 ₽

## Уровень конкуренции (соотношение числа резюме и вакансий) по языкам программирования/стекам, Россия, 2023 год



## Повторим основные команды Git

git status git add git commit

git branch git checkout git clone git push git pull

## Типичная работа с Гитхабом

- 1. Форк оригинального удаленного репозитория
- 2.Клонируем локально форк
- 3.Создаем свою ветку для работы
- 4. Работаем, коммитим
- 5.Пушим изменения в форк
- 6.Делаем пуллреквест из форка в оригинал
- 7.\* Подтягиваем изменения из оригинального удаленного репозитория в локальный репозиторий

# Что такое GitHub Pages и для чего он нужен?

## Как выложить сайт на GitHub Pages?

## Как выложить сайт на GitHub Pages

- Шаг 1. Создаём новый репозиторий либо делаем форк
- Шаг 2. Настраиваем сайт
- Шаг 3. Редактируем страницу
- Шаг 4. Привязываем домен

## Вопросы по ДЗ-2?

## Домашнее задание 2

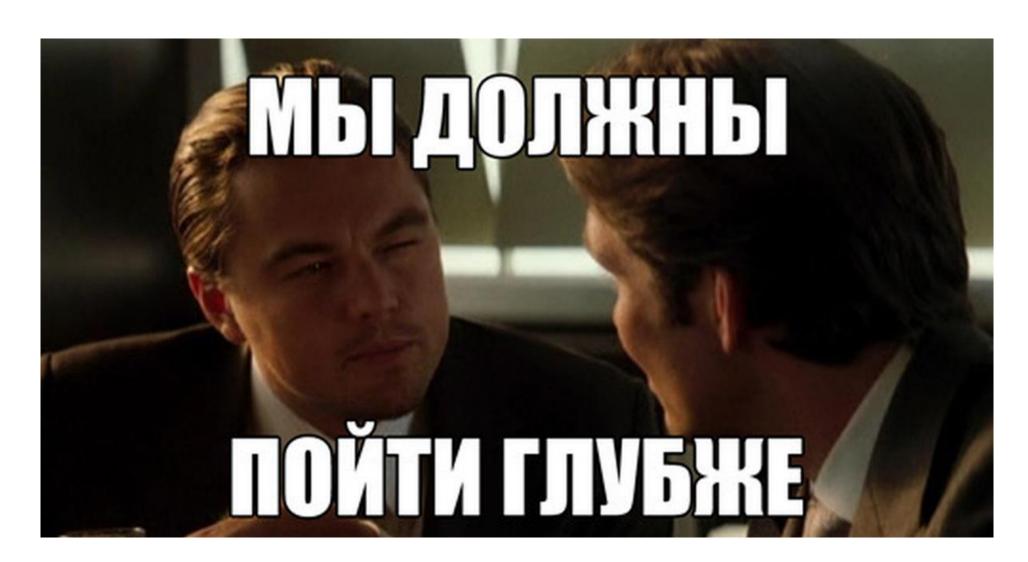
- 1. Форк оригинального удаленного репозитория
- 2. Клонируем локально форк
- 3. Создаем свою ветку для работы
- 4.Создать файл task2.md
- 5. Заполнить его контентом (хобби, увлечение, информация):
- Заголовки (разных уровней)
- Выделение текста (жирный, курсив)
- Списки (нумерованные и простые)
- Ссылки
- Картинки
- 6.Опубликовать в Github Pages
- 7.Сделать пулл-реквест с ссылкой на опубликованную страничку

## Дедлайн

24.11.2023 23:59:59

форков - 4 / реквестов - 3

## Подробнее про HTML



## **DOCTYPE**

Если вы хоть раз работали с HTML-файлом, то наверняка видели в начале страницы ключевое слово DOCTYPE. Если разобрать его по частям, то получается DOC (сокращение от слова document) + TYPE (с английского «тип»).

Ключевое слово DOCTYPE в файле HTML — это декларация типа документа. Так браузер узнает, на каком языке и с какими технологиями он был создан.

## **DOCTYPE**

Если вы хоть раз работали с HTML-файлом, то наверняка видели в начале страницы ключевое слово DOCTYPE. Если разобрать его по частям, то получается DOC (сокращение от слова document) + TYPE (с английского «тип»).

Ключевое слово DOCTYPE в файле HTML — это декларация типа документа. Так браузер узнает, на каком языке и с какими технологиями он был создан.

# Что будет, если указать DOCTYPE неправильно или не указывать его вообще?

Браузеры покажут страницу по совместимости со своими старыми версиями.

Формально регистр тут значения не имеет. Но корректнее всё же писать заглавными буквами. Потому что в XHTML5 есть разница между именами элементов, написанными в нижнем регистре и ключевыми словами, которые пишут в верхнем регистре.

<!DOCTYPE html> — запомнить просто

## Элемент <html>

Ter <html> является корневым элементом веб-страницы. Все содержимое документа, включая <head> и <body>, находится внутри этого тега. Тег <html> сообщает браузеру, что документ является HTML-документом.

<html> — обязательный элемент структуры HTMLдокумента.

Добавьте атрибут lang в тег <html>, чтобы указать язык содержимого, например: <html lang="ru"> для русскоязычного контента.

## Элемент <head>

Ter <head> используется для определения заголовочной части HTML-документа.

<head> не отображается на веб-странице, но содержит информацию, которая может быть использована браузером и другими приложениями. Этот тег содержит мета-информацию о документе, такую как заголовок, ключевые слова, ссылки на стили, скрипты и другие данные, которые необходимы для правильного отображения и интерпретации веб-страницы.

## Элемент <head>

#### Заголовок документа

```
<head>
<title>Moй сайт</title>
</head>
```

#### Установка кодировки

```
<head>
<meta charset="UTF-8">
</head>
```

#### Подключение CSS-файла

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

## Элемент <body>

Ter <body> используется:

- 1. Для отображения содержимого на веб-странице.
- 2. Для форматирования текста и другого содержимого с помощью HTML-тегов.
- 3. Для размещения ссылок на другие веб-страницы или ресурсы.
- 4. Для вставки мультимедийного содержимого, такого как изображения и видео.
- 5. Для стилизации содержимого с помощью CSS.
- 6. Для определения структуры веб-страницы с помощью HTMLтегов.

## Элемент <body>

#### Ограничения

Ter <body> должен использоваться только один раз в HTMLдокументе. Он не может быть вложен внутрь другого тега <body> или любого другого элемента.

#### Нюансы

Если в документе нет тега <body>, браузер всё равно попытается отобразить страницу. Но полученная страница может быть плохо сформирована или в ней будет отсутствовать важное содержимое.

## Заголовки <h1> - <h6>

Элемент <h1> можно использовать только один раз на каждой странице. Текст такого заголовка должен быть крупнее, чем у следующего по уровню заголовка — <h2>

Ter <h2> используется для создания заголовка второго уровня. Это более мелкий заголовок, чем <h1>, он подходит для названия раздела или подраздела на странице.

Тег <h2> должен располагаться только после тега <h1>, который создаёт главный заголовок страницы. Также не стоит использовать элемент только для стилизации текста — для этого больше подходят соответствующие теги

## <b>

Тег <b> используется для выделения текста без дополнительного акцента или значимости. В результате содержимое этого тега обычно отображается жирным шрифтом. Несмотря на это, этот тег не должен использоваться для стилизации; для таких задач лучше использовать CSS

В отличие от тегов, таких как <strong>, который указывает на важность текста, тег <b> не имеет семантической значимости. Это просто стилистическое выделение.

## <strong>

Ter <strong> используется для выделения важного текста на странице — например, чтобы привлечь внимание к тексту или выделить подзаголовок внутри абзаца. Обычно такой текст выделяется жирным шрифтом.

He используйте <strong> для того, чтобы просто сделать текст жирным.

## <i>>

Ter <i> не предназначен для использования в семантических целях, поскольку не передаёт никакого конкретного значения. Он только выделяет текст курсивом.

Чрезмерное использование <i> может сделать документ трудным для чтения и понимания, поэтому используйте тег только по мере необходимости.

В большинстве случаев тег можно заменить тегом <em>. А чтобы специально указать, что текст должен быть выделен курсивом, используйте CSS.

#### **<U>**

Тег <u> использовался для подчёркивания текста, например, выделения ключевых слов, названий, имени автора или чегонибудь в этом роде.

Но если вам нужно просто подчеркнуть текст, лучше используйте CSS.

## >

#### Тег <р> используется:

- 1. для разделения блоков текста на абзацы;
- 2. для добавления текстовых описаний к изображениям;
- 3. для добавления текстовых описаний к видео;
- 4. для написания содержания статей;
- 5. для написания постов для блога;
- 6. для описания продуктов.

## >

#### Ограничения:

если внутри тега нет содержимого, тег все равно создаст пустое пространство на веб-странице;

если содержимое внутри тега содержит только пробелы или переносы строк, тег создаст пустое место на веб-странице.

### <br>

Ter <br >br> — это элемент переноса строки, он используется для создания новой строки или начала нового абзаца в блоке текста.

#### Добавление пробела между двумя элементами:

<h1> Добро пожаловать на мой сайт!</h1>

<br/>br>

<Спасибо за визит! </p>

#### Разбиение длинного предложения на несколько строк:

Посетите мой сайт: <a href="https://www.example.com"> https://www.example.com</a><br/>
для получения дополнительной информации.

## <l

Ter ul> используется для создания неупорядоченного списка элементов.

#### Простой неупорядоченный список:

```
Пункт 1Пункт 2Пункт 3
```

## <l

#### Ограничения

- Ter 
   Ter 
   Ter 
   не следует использовать для списков, требующих определенного порядка. В этом случае вместо него следует использовать тег 
   тел 
   <l>тел 
   тел 
   тел 
   тел 
   тел 
   тел 
   тел 
   тел 
   тел <ul
- Ter 
   cледует использовать только для списка связанных элементов. Если элементы списка не связаны между собой, их следует разделить на разные списки или отобразить в другом формате.

#### Нюансы

- Если элемент в списке требует нескольких абзацев или подзаголовков, рекомендуется использовать вложенные элементы, такие как и ии
- Если элементы списка содержат изображения или видео, рекомендуется использовать теги <figure> и <figcaption> для предоставления альтернативного текста и подписей.
- У тега должен быть хотя бы один вложенный элемент !...

## <0l>

Ter используется для создания упорядоченного списка элементов, то есть списка, в котором каждый элемент имеет порядковый номер. Такие списки полезны при создании пошаговых инструкций или других списков, где важен порядок действий.

#### Простой упорядоченный список:

```
    Ol>
    I > Пункт 1
    I > Пункт 2
    I > Пункт 3
```



Тег используется для создания элементов списка. Тег может быть вложен в элемент маркированного(неупорядоченного) списка — ul>, нумерованного списка — или меню — <menu>.

#### Простой упорядоченный список:

```
    Ol>
    Ol<
    <li>Ol<
    </li>
    Ol<
    <li>Ol<
    <li>Ol<
    <li>Ol<
    <li>Ol<
    <li>Ol<
    <li>Ol<
    </li>
    Ol<
    </li>
    Ol<
    <li>Ol<
    </li>
    Ol<
    </li
```

#### <a>>

Тег <a> нужен для создания гиперссылок, которые связывают страницу с другой странице или с определённым местом на той же веб-странице.

Чаще всего ссылка выглядит как-то так:

<a href="url">Текст ссылки</a>

При использовании тега <a> следует помнить: если связанная страница будет удалена или перемещена, ссылка перестанет работать.

#### <a>>

href — URL страницы или ресурса для ссылки.

target — указывает, где открыть страницу или ресурс, на который дана ссылка.

download — указывает, что цель будет загружена, когда пользователь нажмёт на гиперссылку.

hreflang — язык связанного документа.

**type** — тип медиафайла связанного документа.

## <img>

Элемент <img> используется для вставки изображений на вебстраницы.

У <img> нет закрывающего тега. Он поддерживает изображения в различных форматах — например, JPEG, PNG, WebP, GIF, SVG.

Существуют определенные правила или стандарты использования тега <img>, которые рекомендуются для большинства случаев в вебразработке. Несмотря на то что конкретные требования могут варьироваться в зависимости от проекта, ниже приведены атрибуты и практики, которые часто рекомендуются к использованию.

# Что такое семантическая вёрстка и зачем она нужна?

Семантическая вёрстка — подход к разметке, который опирается не на содержание сайта, а на смысловое предназначение каждого блока и логическую структуру документа. Даже в этой статье есть заголовки разных уровней — это помогает читателю выстроить в голове структуру документа. Так и на странице сайта — только читатели будут немного другими.

#### Почему семантика важна:

- 1. Чтобы сделать сайт доступным
- 2. Чтобы сайт был выше в поисковиках.
- 3. Семантика прописана в стандартах.

```
<!DOCTYPE html>
<html lang="ru">
 <head>
  <meta charset="utf-8">
  <title>Заголовок страницы</title>
 </head>
 <body>
  <header class="main-header">
   <!— Шапка сайта —>
  </header>
  <main>
   <!— Основное содержимое страницы —>
  </main>
  <footer class="main-footer">
   <!— Подвал сайта —>
  </footer>
 </body>
</html>
```

#### <header>

**Значение:** вводная часть смыслового раздела или всего сайта, обычно содержит подсказки и навигацию. Чаще всего повторяется на всех страницах сайта.

Особенности: этих элементов может быть несколько на странице.

Типовые ошибки: использовать только как шапку сайта.

#### <main>

Значение: основное, не повторяющееся на других страницах, содержание страницы.

**Особенности:** должен быть один на странице, исходя из определения. **Типовые ошибки:** включать в этот тег то, что повторяется на других страницах (навигацию, копирайты и так далее).

#### <footer>

**Значение:** заключительная часть смыслового раздела или всего сайта, обычно содержит информацию об авторах, список литературы, копирайт и так далее. Чаще всего повторяется на всех страницах сайта.

**Особенности:** этих элементов может быть несколько на странице. Тег <footer> не обязан находиться в конце раздела.

Типовые ошибки: использовать только как подвал сайта.

#### <article>

**Значение:** независимая, отделяемая смысловая единица, например комментарий, твит, статья, виджет ВК и так далее.

Особенности: желателен заголовок внутри.

Типовые ошибки: путают с тегами <section> и <div>.

#### <nav>

Значение: навигационный раздел со ссылками на другие страницы или другие части страниц.

Особенности: используется для основной навигации, а не для всех групп ссылок. Основной является навигация или нет — на усмотрение верстальщика. Например, меню в подвале сайта можно не оборачивать в <nav>. В подвале обычно появляется краткий список ссылок (например, ссылка на главную, копирайт и условия) — это не является основной навигацией, семантически для такой информации предназначен <footer> сам по себе.

**Типовые ошибки:** многие считают, что в <nav> может быть только список навигационных ссылок, но согласно спецификации там может быть навигация в любой форме.

#### <section>

**Значение:** смысловой раздел документа. Неотделяемый, в отличие от <article>.

Особенности: желателен заголовок внутри.

Типовые ошибки: путают с тегами <article> и <div>.

#### <aside>

Значение: побочный, косвенный для страницы контент.

Особенности: может иметь свой заголовок. Может встречаться несколько раз на странице.

**Типовые ошибки:** считать <aside> тегом для «боковой панели» и размечать этим тегом основной контент, который связан с окружающими его элементами.

## Как разметить страницу с точки зрения семантики?

- 1. Крупные смысловые блоки на каждой странице сайта. Теги: <header>, <main>, <footer>.
- 2. Крупные смысловые разделы в блоках. Теги: <nav>, <section>, <article>, <aside>.
- 3. Заголовок всего документа и заголовки смысловых разделов. Теги: <h1>-<h6>.
- 4. Мелкие элементы в смысловых разделах. Списки, таблицы, демоматериалы, параграфы и переносы, формы, цитаты, контактная информация и прогресс.
- 5. Фразовые элементы. Изображения, ссылки, кнопки, видео, время и мелкие текстовые элементы.

## Когда какие теги использовать?

#### Есть простые правила для выбора нужных тегов.

- 1. Получилось найти самый подходящий смысловой тег использовать его.
- 2. Для потоковых контейнеров <div>.
- 3. Для мелких фразовых элементов (слово или фраза) <span>.

#### Правило для определения <article>, <section> и <div>:

- 1. Можете дать имя разделу и вынести этот раздел на другой сайт? <article>
- 2. Можете дать имя разделу, но вынести на другой сайт не можете? <section>
- 3. Не можете дать имя? Получается что-то наподобие «новости и фотогалерея» или «правая колонка»? <div>

## Последний раз напоминаю про Д3-1

12/38

не копите плохие долги

## Дополнительный материал

docs.github.com/ru

ru.stackoverflow.com

habr.com

ya.ru