

## Язык JavaScript

- Как написать и запустить простейший скрипт (Hello, world)?
- Какие есть версии языка, чем они отличаются?
- Как работать с переменными?
- Как объявлять и вызывать подпрограммы?
- Как организовать пользовательский интерфейс?
- Можно ли разделять код на модули и как с ними работать?

# Как написать и запустить простейший скрипт (Hello, world)?

---

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <h1>Первый заголовок</h1>

  <script>
    console.log('Hello, world!');
  </script>

  <script src="1.js"> </script>

</body>
</html>
```

# Какие версии языка поддерживаются?

EcmaScript-262 – это **спецификация** языка  
JavaScript – это **реализация** спецификации

Современные редакции стандарта:

2009	ES5
2015	ES6 (ES-2015)
2016	ES7 (ES-2016)
2017	ES8 (ES-2017)

## Какие версии языка поддерживаются?

---

Движки JavaScript:

- **V8** – Google Chrome и Node.js
- **SpiderMonkey** – Mozilla Firefox
- **Nitro** – Apple Safari

<https://caniuse.com> – проверка реализации возможностей языка

## Как работать с переменными?

- Слабая динамическая типизация
- Объявление: **var**, **let** (ES6) и **const** (ES6)

## Как работать с подпрограммами?

- Функции в JS - объекты первого рода
- Объявление: Function Declaration, Function Expression
- Анонимные функции
- Стрелочные функции (ES6)
- Вложенные функции, замыкания
- Область видимости, лексическое окружение
- Строки, числа, логические значения передаются в функцию по значению. Объекты и массивы передаются по ссылке.

## Как работать с UI?

- JavaScript может манипулировать элементами и текстовыми узлами DOM
- Можно задавать обработчики событий, происходящих на узлах DOM-дерева

# Модульность

Возможны два варианта:

1. Код разделяется на модули только во время разработки. На страницу подключается единственный файл - результат сборки модулей.
2. В ES6 можно разделять JS-код на модули непосредственно в браузере.



## Сборка проекта

Трансформация и оптимизация кода:

- Недостижимый код удаляется.
- Неиспользуемые экспорты удаляются («tree-shaking»).
- Специфические операторы для разработки (console и debugger) удаляются.
- Современный синтаксис JavaScript может быть трансформирован в предыдущий стандарт, с похожей функциональностью (Babel).
- Полученный файл минимизируется (удалить пробелы, заменить названия переменных на более короткие и т.д.).