

Введение в JavaScript

Чистяков Денис

Сегодня

Что такое JavaScript?

История JavaScript

Как выбрать текстовый редактор?

Основные типы данных

Зачем нужна система контроля версий?

Что нужно знать, чтобы сдать домашку?

На каких языках вы уже
программировали?

JavaScript

Мультипарадигменный

Объектно-ориентированный

Императивный

Функциональный

Утиная типизация

Краткая история JavaScript

1995 – Первая реализация



There was an error with your submission:

- Enter Job title.
- Email must be a valid e-mail.
- Please select your state.
- Enter Industry.

Please retry.

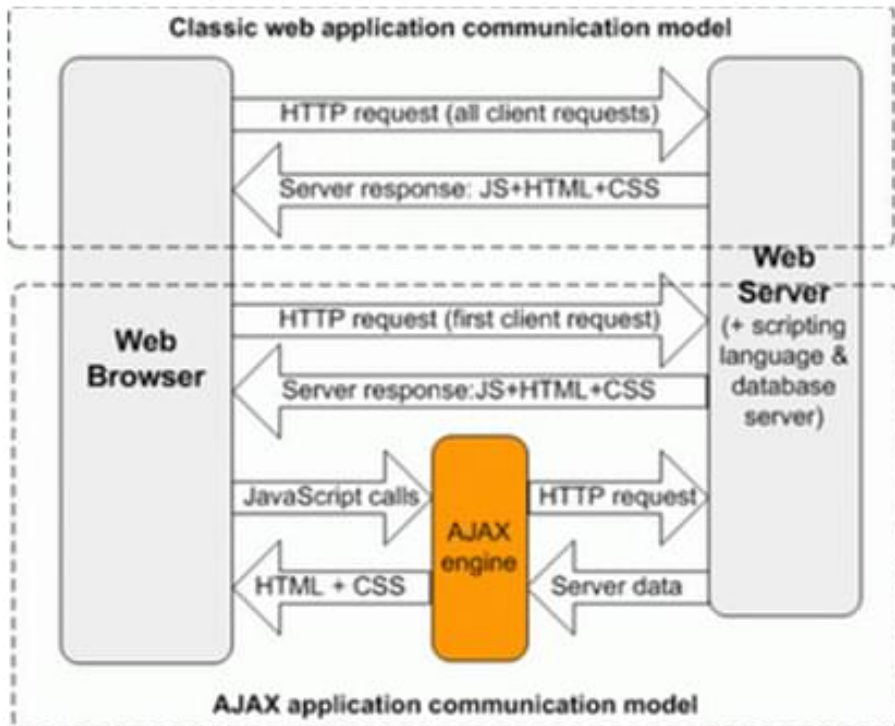
First Name: *	<input type="text" value="Kieran"/>
Last Name: *	<input type="text" value="Chapman"/>
Job Title: *	<input type="text"/>
Company: *	<input type="text" value="Kieran Chapman Design"/>
Email: *	<input type="text" value="kieranchapman"/>
Phone:	<input type="text"/>
Country: *	<input type="text" value="United States"/>
State/Province: *	<input type="text" value="Please select..."/>
Industry: *	<input type="text" value="Please Select..."/>
Comments:	<div></div>

Submit

Краткая история JavaScript

1995 – Первая реализация

1998 – Ajax



Краткая история JavaScript

1995 – Первая реализация

1998 – Ajax

2006 – jQuery

JavaScript

```
document.getElementsByTagName('header')[0]
```

jQuery

```
$('header')
```

Краткая история JavaScript

1995 – JavaScript

1998 – Ajax

2006 – jQuery

2008 – Chromium + V8



Chromium + V8

Компиляция

Эффективное управление памятью

Скрытые классы и встроенные кэши

Краткая история JavaScript

1995 – JavaScript

1998 – Ajax

2006 – jQuery

2008 – Chromium + V8

2009 – Node.js



Текстовый редактор

В нём вы проводите большую часть времени. Поэтому отнеситесь к выбору ответственно.

Варианты

Visual Studio Code

WebStorm

Atom

Sublime Text

Visual Studio Code

Мощная поддержка языка

Инструменты для работы с git

Встроенный терминал

Декларация переменных

Вы не планируете переприсваивать значение
переменной

```
const year = 1999;
```

Декларация переменных

```
const year = 1999;  
year = 2000; // TypeError: Assignment to constant variable.
```

Декларация переменных

Значение переменной может быть переприсвоено

```
let year;  
year = 1999;
```

Именованние переменных

```
const year = 1999;      // OK
const $year = 1999;     // OK
const _year = 1999;     // OK
const year1999 = 1999;  // OK
const 1year = 1999;     // Not OK!
```

★ В качестве первого символа имени переменной можно использовать буквы, символ доллара (\$) и подчёркивание (_). Последующие символы могут содержать всё вышеперечисленное, а также цифры.

Числа

```
const duration = 136;
```

Числа

```
const duration = 136;  
console.log(duration); // 136
```


Числа

```
const duration = 136;  
const hours = duration / 60;    // 2.2666666666666666
```

Числа

```
const duration = 136;  
const hours = Math.trunc(duration / 60);    // 2
```

Числа

```
const duration = 136;  
const hours = Math.trunc(duration / 60);    // 2  
const minutes = duration - (hours * 60);    // 16
```

Числа

```
const duration = 136;  
const hours = Math.trunc(duration / 60);    // 2  
const minutes = duration % 60;             // 16
```

Числа

```
const duration = 136;  
const hours = Math.trunc(duration / 60);    // 2  
const minutes = duration % 60;             // 16
```

«Магические числа»

```
const minutesInHour = 60;
```

```
const duration = 136;
```

```
const hours = Math.trunc(duration / minutesInHour); // 2
```

```
const minutes = duration % minutesInHour; // 16
```

Условные операторы

```
const minutesInHour = 60;
const duration = 25;
const hours = Math.trunc(duration / minutesInHour); // 0
const minutes = duration % minutesInHour; // 25

if (hours > 0) {
  console.log('Hours:', hours);
  console.log('Minutes:', minutes);
}
```

Условные операторы

```
const minutesInHour = 60;
const duration = 25;
const hours = Math.trunc(duration / minutesInHour); // 0
const minutes = duration % minutesInHour; // 25

if (hours > 0) {
    console.log('Hours:', hours);
    console.log('Minutes:', minutes);
} else {
    console.log('Minutes:', minutes);
}
```


Условные операторы

```
const minutesInHour = 60;
const duration = 60;
const hours = Math.trunc(duration / minutesInHour); // 1
const minutes = duration % minutesInHour; // 0

if (hours > 0 && minutes > 0) {
    console.log('Hours:', hours);
    console.log('Minutes:', minutes);
} else if (hours > 0) {
    console.log('Hours:', hours);
} else {
    console.log('Minutes:', minutes);
}
```

Boolean

```
const minutesInHour = 60;  
const duration = 60;  
const hours = Math.trunc(duration / minutesInHour); // 1  
const minutes = duration % minutesInHour; // 0  
  
const isHoursPositive = hours > 0; // true  
const isHoursEqZero = hours === 0; // false  
const isHoursAndMinutesPositive = hours > 0 && minutes > 0;  
// true
```

Boolean

```
const minutesInHour = 60;
const duration = 60;
const hours = Math.trunc(duration / minutesInHour); // 1
const minutes = duration % minutesInHour; // 0

const isHoursPositive = hours > 0;
const isHoursAndMinutesPositive = hours > 0 && minutes > 0;

if (isHoursAndMinutesPositive) {
  console.log('Hours:', hours);
  console.log('Minutes:', minutes);
} else if (isHoursPositive) {
  console.log('Hours:', hours);
} else {
```

Строки

```
const movie = 'The Matrix';
```

Строки

```
const movie = "The Matrix";
```

Строки

```
const movie = 'Операция «Ы»';
```

Строки

```
const movie = 'La vita è bella';
```

Строки

```
const movie = 'ハウルの動く城';
```


Строки

```
const movie = 'The Matrix';  
const description = 'It\'s action movie';
```

Строки

```
const movie = 'The Matrix';  
const description = 'It\'s action movie';  
console.log(description.length);    // 17
```

Строки

```
const movie = 'The Matrix';  
const description = 'It\'s action movie';  
  
console.log(movie + '. ' + description);  
// 'The Matrix. It's action movie'
```

Строки

```
const movie = 'The Matrix';  
movie[0];    // 'T'
```

Строки

```
const movie = 'The Matrix';  
movie[0] = 'D';  
console.log(movie);      // 'The Matrix'
```

Цикл

```
const movie = 'The Matrix';
```

```
for (let i = 0;;) {}
```

Цикл

```
const movie = 'The Matrix';
```

```
for (let i = 0; i < movie.length;) {}
```

Цикл

```
const movie = 'The Matrix';  
  
for (let i = 0; i < movie.length; i++) {  
    const char = movie[i]; // 'Т'  
}
```


Цикл

```
const movie = 'The Matrix';  
  
for (let i = 0; i < movie.length; i++) {  
    const char = movie[i]; // 'h'  
}
```

Цикл

```
const movie = 'The Matrix';  
  
for (let i = 0; i < movie.length; i++) {  
    const char = movie[i]; // 'e'  
}
```

Цикл

```
const movie = 'The Matrix';
```

```
for (let i = 0; i < movie.length; i++) {  
    const char = movie[i]; // 'x'  
}
```

```
console.log(i); // ReferenceError: char is not defined  
console.log(char); // ReferenceError: char is not defined
```

Массивы

```
const genres = ['action', 'fantasy', 'sci-fi'];
```

Массивы

```
const genres = ['action', 'fantasy', 'sci-fi'];  
console.log(genres.length); // 3
```

Массивы

```
const genres = ['action', 'fantasy', 'sci-fi'];  
genres[0]; // 'action'
```

Массивы

```
const genres = ['action', 'fantasy', 'sci-fi'];  
genres[0] = 'comedy';
```

```
console.log(genres); // ['comedy', 'fantasy', 'sci-fi']
```

Массивы

```
const genres = ['action', 'fantasy', 'sci-fi'];  
genres.push('comedy');  
  
console.log(genres);  
// ['action', 'fantasy', 'sci-fi', 'comedy']
```


Цикл

```
const genres = ['action', 'fantasy', 'sci-fi'];  
  
for (let i = 0; i < genres.length; i++) {  
    const genre = genres[i]; // 'action'  
}
```

Цикл

```
const genres = ['action', 'fantasy', 'sci-fi'];  
  
for (let i = 0; i < genres.length; i++) {  
    const genre = genres[i]; // 'fantasy'  
}
```

Цикл

```
const genres = ['action', 'fantasy', 'sci-fi'];  
  
for (let i = 0; i < genres.length; i++) {  
    const genre = genres[i]; // 'sci-fi'  
}
```

Undefined

```
const genres = ['action', 'fantasy', 'sci-fi'];  
genres[5]; // undefined
```

Undefined

```
let year;  
year; // undefined
```

```
year = 1999;  
year; // 1999
```

Объекты

```
const movie = {};
```

Объекты

```
const movie = {  
  name: 'The Matrix'  
};
```

Объекты

```
const movie = {  
  name: 'The Matrix',  
  year: 1999,  
  genres: ['action', 'fantasy', 'sci-fi']  
};
```


Объекты

```
const movie = {  
  name: 'The Matrix',  
  year: 1999,  
  genres: ['action', 'fantasy', 'sci-fi']  
};
```

```
console.log(movie.name); // 'The Matrix'
```

Объекты

```
const movie = {  
  name: 'The Matrix',  
  year: 1999,  
  genres: ['action', 'fantasy', 'sci-fi']  
};
```

```
const key = 'year';  
console.log(movie[key]); // 1999  
console.log(movie['na' + 'me']); // 'The Matrix'
```

Объекты

```
const movie = {  
  name: 'The Matrix',  
  year: 1999,  
  genres: ['action', 'fantasy', 'sci-fi']  
};  
  
movie.name = 'Gatsby';  
console.log(movie);  
// { name: 'Gatsby',  
//   year: 1999,  
//   genres: [ 'action', 'fantasy', 'sci-fi' ] }
```

Объекты

```
const movie = {  
  name: 'The Matrix',  
  year: 1999,  
  genres: ['action', 'fantasy', 'sci-fi']  
};
```

```
movie.name = 'Gatsby';  
movie.genres.push('comedy');  
movie.duration = 136;
```

Объекты

```
const movie = {  
  name: 'The Matrix',  
  year: 1999,  
  genres: ['action', 'fantasy', 'sci-fi']  
};  
  
delete movie.genres;  
console.log(movie); // { name: 'The Matrix', year: 1999 }
```

Объекты

```
const movie = {  
  name: 'The Matrix',  
  year: 1999,  
  genres: ['action', 'fantasy', 'sci-fi']  
};  
  
const keys = Object.keys(movie);  
console.log(keys); // ['name', 'year', 'genres']
```

Функции

```
const movie = {  
  name: 'The Matrix',  
  
  year: 1999  
};
```

Функции

```
function createMovie(name, year) {  
  return {  
    name: name,  
    year: year  
  };  
}
```


Функции

```
function createMovie(name, year) {  
  return {  
    name: name,  
    year: year  
  };  
}
```

Функции

```
function createMovie(name, year) {  
    return {  
        name,  
        year  
    };  
}
```

Функции

```
function createMovie(name, year) {  
    return {  
        name,  
        year  
    };  
}
```

```
console.log(createMovie('The Matrix', 1999));  
// { name: 'The Matrix', year: 1999 }
```

Функции

```
function createMovie(name, year) {  
    return {  
        name,  
        year  
    };  
}
```

```
console.log(  
    createMovie('The Matrix', 1999),  
    createMovie('The Green Mile', 1999),  
    createMovie('Bad Boys', 1995)  
)
```

git – система контроля версий.

В целом, нужна для двух вещей: версионирования кода
и совместной разработки.

git-scm.org

GitHub – графический интерфейс для git.

Есть и другая версия: это социальная сеть
для программистов.

github.com

Ссылки про git/GitHub

Введение. Основы языка. GitHub

Язык программирования Javascript 2016

git

Разработка веб-приложений 2016

Github (слайды, видео)

Разработка веб-приложений 2015

lab.github.com

learn.javascript.ru

Демо

Отправляем пулл-реквест с домашкой.

Задача «Сложить два числа»

Перед началом работы
клонировем репозиторий
и устанавливаем зависимости.

```
git clone https://github.com/urfu-...  
demo-task-1  
npm install
```

Перед отправкой домашки
запускаем линтер и тесты.

```
npm run lint  
npm run test
```

После прохождения проверок
создаём коммит в VS Code
и отправляем его на гитхаб.

```
git push
```

Спасибо!
Вопросы?