

Основы JavaScript

Занятие 1

План занятия

- Введение
- IDE, консоль, online редакторы
- Основы JS
- Структура кода
- Переменные
- Типы данных
- Справочники и литература



Введение

LiveScript

JavaScript

ECMAScript

Компіляція

Інтерпретація

Что может JS в браузере

- Создавать, удалять HTML-теги, прятать, показывать элементы, изменять стили элементов.
- Реагировать на действия пользователя, обрабатывать нажатия на клавиатуру, клики мыши, перемещение курсора, скролл и т.п.
- Отправлять запросы на сервер и загружать данные без перезагрузки страницы (AJAX).
- Получать и устанавливать cookie, запрашивать данные у пользователя, выводить сообщения...
- И много чего еще!

Чего не может JS в браузере

- JavaScript не может читать/записывать произвольные файлы на жесткий диск, копировать их или вызывать программы. Он не имеет прямого доступа к операционной системе.
- Современные браузеры могут работать с файлами, но эта возможность ограничена специально выделенной для них директорией — sandbox. Возможности по доступу к устройствам также прорабатываются в современных стандартах и частично доступны в некоторых браузерах.

Чего не может JS в браузере

- JS, работающий в одной вкладке, не может общаться с другими вкладками и окнами, кроме случая, когда он сам открыл эти окна из одного источника (одинаковый домен, порт, протокол).
- Из JavaScript можно легко посылать запросы на сервер, с которого пришла страница. Запрос на другой домен тоже возможен, но менее удобен, т.к. и здесь есть ограничения безопасности.

Браузерные технологии

Java Applet

JavaScript

Flash

HTML5 + JavaScript

- Чтение/запись файлов на диск (в sandboxe).
- Встроенная в браузер база данных, которая позволяет хранить данные на компьютере пользователя.
- Многозадачность с одновременным использованием нескольких ядер процессора.
- Проигрывание видео/аудио, без Flash.
- 2d и 3d-рисование с аппаратной поддержкой, как в современных играх.

Надстройки над JS

TypeScript

CoffeScript

Dart

IDE, online editors, console

IDE

- WebStorm
- Visual Studio
- Aptana,
- Komodo
- Netbeans
- SublimeText
- SciTe (free)
- Notepad++ (free)
- Vim (super free)
- Emacs

Online editors

- codepen.io
- dabblet.com
- jsfiddle.net

Console

- Chrome
- FireFox
- Safari
- IE

Lets Start!



Hello world!

```
<!doctype html>
<html>

<head>
  <meta charset="utf-8"/>
</head>

<body>
  <script>
    alert( 'Hello world!' );
  </script>
</body>

</html>
```

Extra Hard Task

Сделайте страницу, которая выводит «I love JavaScript!» во всплывающем диалоговом окне.

External/Internal scripts

```
<script src="/folder/script.js"></script>
```

```
<script src="https://yastatic.net/jquery/1.10.1/jquery.min.js"></script>
```

```
<script src="/js/script1.js"></script>
```

```
<script src="/js/script2.js"></script>
```

```
<script src="script.js">
```

```
  alert(1); //ignored
```

```
</script>
```

Deffer/Async

```
<!doctype html>
<html>
<body>
  <p>Start</p>
  <script>
    alert( 'First!' );
    alert( 'Second!' );
    alert( 'Third!' );
  </script>
  <p>Finish!</p>
</body>
</html>
```


Deffer/Async

```
<script src="1.js" async></script>  
<script src="2.js" async></script>
```

```
<script src="1.js" defer></script>  
<script src="2.js" defer></script>
```

Extra Hard Task 2

Скрипт из предыдущего задания
вынести в отдельный файл и
подключить на страницу.

Code structure

Semicolumn

```
alert('Hello'); alert('world!');
```

```
alert('Hello');  
alert('world!');
```

```
alert('Hello')  
alert('world!')
```

```
alert(3 +  
1  
+ 2);
```

```
//For example  
[1, 2].forEach(alert)
```

```
alert("Error")  
[1, 2].forEach(alert)
```

```
alert("Error");  
[1, 2].forEach(alert)
```


use strict / ES-shim

```
"use strict";  
'use strict';
```

[es-shim](#)

Переменные

Variables

```
var message;
```

Variables

```
var message;  
message = 'Hello';
```

Variables

```
var message;  
message = 'Hello';  
alert(message);
```

```
var newMessage = 'Hi!';  
alert(newMessage);
```

```
var user = 'Sergey', age = 30, message = 'Hi, GOIT';
```

Variables

```
var message;
```

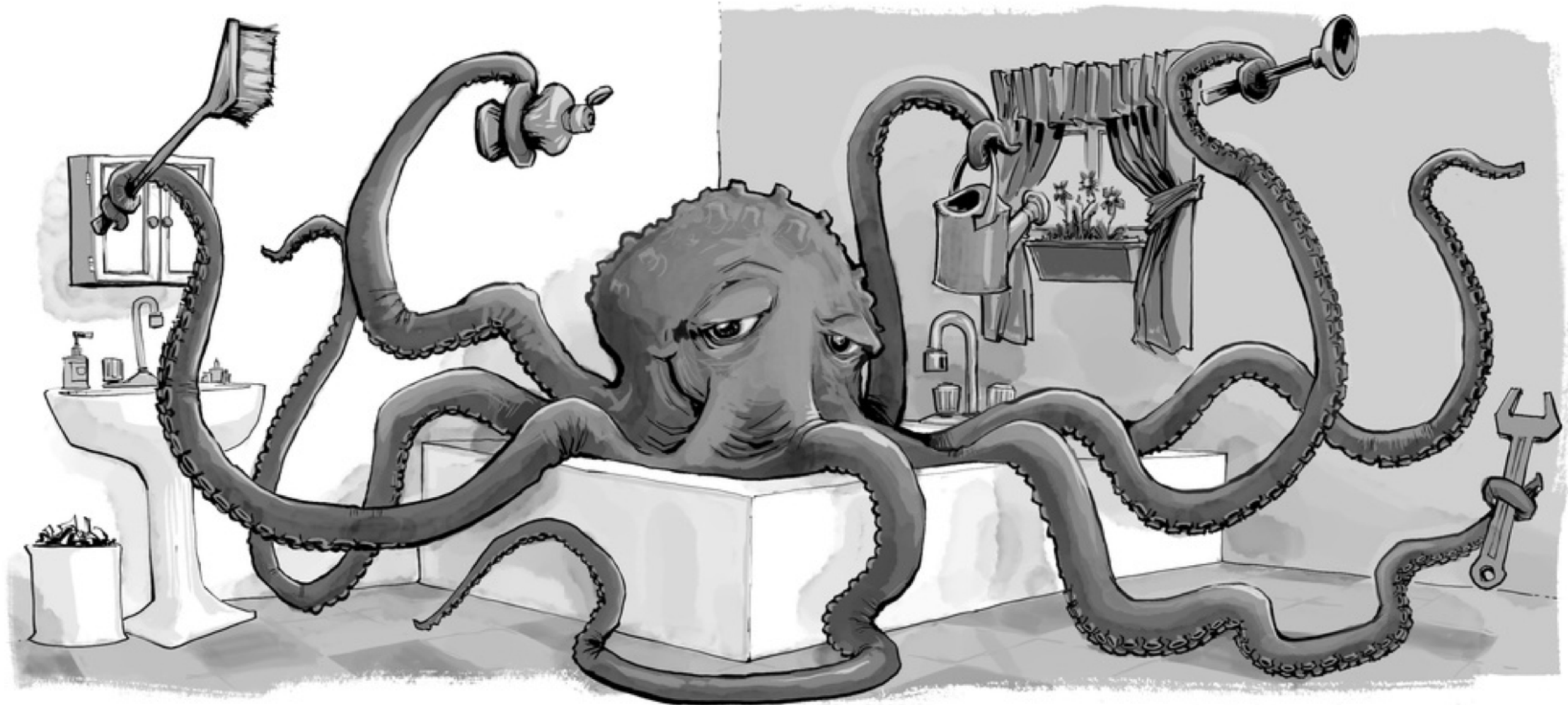
```
message = 'Hello!';
```

```
message = 'World!'; //update
```

```
alert( message );
```



Variables



Variables

```
var one = 1, two = 2;  
console.log(one + two);  
// → 3
```

```
var vasyaDebt = 140;  
vasyaDebt = vasyaDebt - 35;  
console.log(vasyaDebt);  
// → 105
```

Variables

- Имя может состоять из: букв, цифр, символов \$ и _
- Первый символ не должен быть цифрой.

```
var $ = 1;
```

```
var _ = 2;
```

```
console.log($ + _);
```

```
// → 3
```

Зарезервированные имена

`break case catch continue debugger default
delete do else false finally
for function if implements in instanceof
interface let new null package
private protected public return static
switch throw true try typeof var
void while with yield this`

var

```
a = 5;
```

```
"use strict";
```

```
a = 5; // error: a is not defined
```

```
var a = 5;
```

var

```
var COLOR_RED = "#F00";  
var COLOR_GREEN = "#0F0";  
var COLOR_BLUE = "#00F";  
var COLOR_ORANGE = "#FF7F00";
```

```
var color = COLOR_ORANGE;  
alert( color ); // #FF7F00
```

**Какая самая сложная задача
в программировании?**

**Придумать хорошее название
для переменной!**

Имена переменных

- Без транслита
- Переменные `i`, `j`, `k` ... только локальные
- camelCase, or dashed_notation
- Название определяет что в переменной

Типы данных

Типы данных

- Number (1, 1.345, Infinity)
- String ('Hello', "World")
- Boolean (true, false)
- Undefined
- Null
- Object {name: 'Sergey'}

typeof

```
typeof undefined // "undefined"
```

```
typeof 0 // "number"
```

```
typeof true // "boolean"
```

```
typeof "foo" // "string"
```

```
typeof {} // "object"
```

```
typeof null // "object"
```

```
typeof function(){} // "function"
```

Взаимодействие с пользователем

Взаимодействие с пользователем

- `alert('hello');`
- `prompt('Pls, enter your name!', default);`
- `confirm('A you sure?');`
- `console.log('Ku-ku',2);`

Extra Hard Task 3

При открытии страницы спрашиваем имя пользователя после чего выводим его на экран.

Extra Hard Task 3

При открытии страницы спрашиваем имя пользователя после чего выводим его на экран.

Переспрашиваем его правда ли это его имя? И выводим true/false на экран.



Справочники и литература



[ECMAScript](#)

[ECMAScript RU](#)

Справочники

- [Mozilla Developer Network](#)
- [MSDN](#)
- [Safari Developer Library](#)

- [help.dottoro.com](#)
- [javascript.ru/manual](#)
- [www.quirksmode.org](#)
- [caniuse.com](#)

Книги

- Фленеган
- Изящный JS
- Шаблоны
- Good Parts

Итого

- Введение
- IDE, консоль, online редакторы
- Основы JS
- Структура кода
- Переменные
- Типы данных
- Справочники и литература



2015