Basics of JS 1

Prepared by Alex Sobolewski

Hello!

My name is Alex Sobolewski

I have been working in IT as a programmer since 2013. My specialization is JavaScript and Java. I am strong adherent of Clean Code and good design solutions. Currently i am working as Full Stack Engineer. In this course i will guide you through the basics of JS.

1.
Introduction into basic conceptions

Introduction into basic conceptions

What is a programming language?

Programming language - is artificial, high level tool, that allows us to interact with hardware.

In a modern world there are plenty of programming languages, which handle a lot of aspects in a device. Starting from toothbrush to NASA Mars exploration program robots .

Introduction into basic conceptions

What is frameworks and libs and how they relate to programming language?

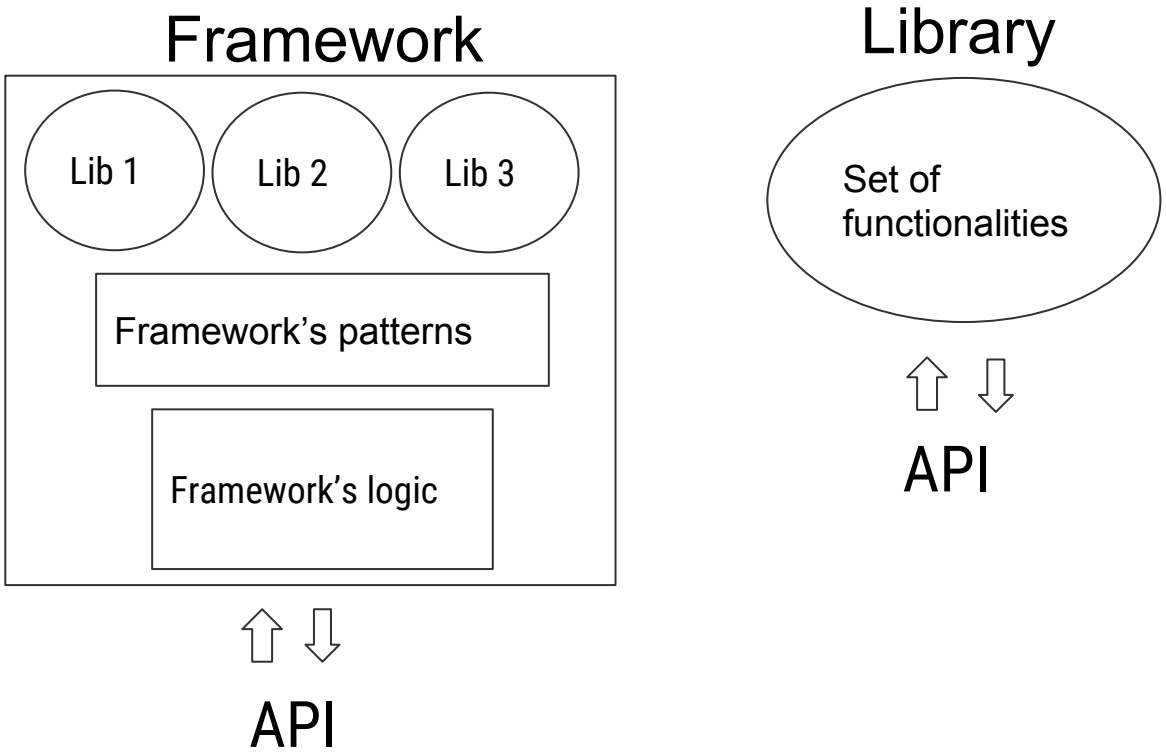
Framework - it is set of design patterns and libraries, that is created to solve strictly defined group of problems(Bootstrap).

Library - it is a standalone functionality created to solve minor problem or group of minor problems(JQuery).

API - a way to communicate with framework, library of another type of functionality.

Introduction into basic conceptions

What are frameworks and libs?



Introduction into basic conceptions

Summary

- Programming language - it is w way how to tell a computer/phone what to do.
- Framework - set of tested and prepared logic, which handles one or more aspects of business requirements.
- Library - prepared and tested functionality, which handles one or a few technical problems.
- Why does it matter?

2.
Introduction into JS

Introduction into JS

What is Js?

JS(EcmaScript) - high level, interpreted language, that is mainly used in Web.

Is it worth learning? Definitely.

- Web - as for applications and web pages
- Web - as NodeJS environment
- DB - as non SQL DB
- Mobile - as pseudo language for Cordova/React

Introduction into JS

Interpreted and compiled languages

An interpreted language(JS) - without previous compilation.

An interpreted language in order to be interpreted requires **interpreter**(JS engine).

A compiled language(C++) - with previously compiled files

A compiled language requires **compilator**.

Introduction into JS

JS engine and environment

Engine:
V8 is the engine for JS. It is the most popular and open source, developed by Google, written in C++.

Environment:
Web browser/NodeJs

An environment already has the engine.

Introduction into JS
History of JS

Creator of the language - Brendan Eich.

- Dark time before 2009(ES 5)(all business logic is placed in BE)
- Better in 2011(ES 5.1)(business logic is still placed in BE)
- New era after 2015(ES 6)(SPA, moving business logic in FE)

Current version - ES 9/2018.
https://stateofjs.com/

Introduction into JS

Summary

- **JS(EcmaScript)** - high level, interpreted language, that is mainly used in Web.
- JS is worth learning
- In order to be executed JS requires an engine and an environment.
- The rise of JS started in 2015.

3.
First steps in JS

First steps in JS
Working with DevTools and first hello world

DevTools(F12) is our best friend in the world of JS.

Most interesting parts of DevTools:

- Network
- Console
- Elements

We can execute JS code right in console window.

Task 1

Open your console and write alert(‘Hello World’);

First steps in JS
Attaching JS

In order to attach JS code we need to place it in separate .js file or put it just in the HTML markup between script tags. We can do it in head tag or in body tag.

```
<script src="js/plugins.js"></script>
<script src="js/main.js"></script>

<!-- Google Analytics: change UA-XXXXX-Y to be your site's ID. -->
<script>
  window.ga = function () { ga.q.push(arguments) }; ga.q = []; ga.l = +new Date;
  ga('create', 'UA-XXXXX-Y', 'auto'); ga('send', 'pageview')
</script>
```

Task 2

Put alert(‘Hello World’); in script tags on your page or attach it via external js file.

First steps in JS
Types of variables in JS

Variable - it is named place for some value or reference.

Each variable has own type. Always.

- String
- Number
- Boolean
- Null
- Undefined

All other types are essentially objects, including arrays and functions.
Primitive variables are passed by value and objects by reference.

First steps in JS
Literals

- String literals - ‘i am a string’, “i am a string too”
- Number literals - 123, 1233, 0
- Null literals - null
- Array literals - [1,2,3,4], [‘1’, ‘2’, 3, ‘4’, 5]
- Boolean literals - true, false
- Object literals - {a: ‘1’, b: 2}
- Undefined literal - undefined

We always can verify type of variable by **typeof.**

Task 3

Create string and number literal and use typeof operator in order to check its type

First steps in JS
Variables in JS

There are 3 ways of declaring variables:

- var x - old way
- let x - new way for temp variables
- const x - new way for constant variables

```
let x; // declaration - marking certain amount of memory
let x = 10; // declaration and (definition - initialization, first assignment)
x = x + 10; // assignment
```

First steps in JS

How to name variables?

- Use camelCase
- The name should be self describing
- The name can contain underscore, dollar sign, digits and letters. Don’t use special symbols.
- Reserved names can’t be used as names
- Variables are case sensitive
- Do not re-declare variables
- Try to declare variables at the beginning of the scope

First steps in JS

What is console object? How to look into variable?

Console object - it is an object that provides access to the browser's debugging console.

let name = ‘console object’;

- console.log(name)
- console.warn(name)
- console.error(name)
- console.table([1,2,3,4])\console.log([1,2,3,4])

First steps in JS
Truthy and falsy variables

In JavaScript, a truthy value is a value that is considered true when evaluated in a boolean (true/false) context.

All values are truthy unless they are defined as falsy.

First steps in JS

Truthy and falsy variables

Falsy variables are:

false,
0,
"",
null,
undefined,
NaN.

Task 4

Create let variable and log it.
Create const variable and try to reassign it.

4.
Grammar

First steps in JS
Reserved keywords

https://www.w3schools.com/js/js_reserved.asp

We **can't** override reserved words!

Grammar and keywords

Grammar

White space doesn't matters!

Lines above do the same:

```
result = a + b // However, it is a good style
result=a+b
result = a          +          b
```

Grammar and keywords

Grammar

Lines should end by semicolon, but JavaScript has automatic semicolon insertion (ASI)!

Semicolons can be omitted! // However, it is good style to use them

Be careful about line breaks!

Grammar and keywords
Grammar

Variables are case sensitive!

result = 10
RESULT = 11

Lines above are different variables!

Grammar and keywords
Comments

One line:
```
// lofo - it is unique ID, should be declared manually
let lofo = 123123;
```

Multi-line:
```
/*
comment
comment
comment
*/
```

Grammar and keywords
Comments

Code should be self-describing as much as we can write it.
Comment is the last resort if everything else fails.

Do not use comments until it is really necessary

5.
Basic math operations

Basic math operations

+ and -

```
var a = 1 + 1

var b = 2 - 3

var c = a + b
```

Basic math operations
+ and -

WARNING!

\+ is also concatenation operator!

Try:
var result = ‘ala’ + ‘ ’ + ‘ma kota’

It can make problems!

Basic math operations
* and /

```
var a = 2 / 2

var b = a * 2

var c = a * b
```

Basic math operations
Modulo %

```
var a = 2 % 2 // result is 0

var b = 4 % 3 // result is 1

var c = 2 % 4 // result is 2
```

Basic math operations
+= and -= operators

```
var a = 2
var b = 4

a += a // result is 4
b -= b // result is 0
```

Basic math operations

Increment ++ and decrement -- operators

```
var a = 2
var b = 4
```

Checkout on console:

```
a--         b++
a           b
--a         ++b
a           b
```

Basic math operations

Unary plus (+)

```
+3      // 3
+'3'    // 3
+true   // 1
+false  // 0
+null   // 0
+{}     // NaN
```