DEFINITIONS

- JavaScript is a **dynamic, weakly typed** programming language which is **compiled at runtime**. It can be executed as part of a webpage in a browser or directly on any machine (**"host environment"**).
- JavaScript was created **to make webpages more dynamic** (e.g. change content on a page directly from inside the browser). Originally, it was called LiveScript, but due to the popularity of Java, it was renamed to JavaScript.
- **JavaScript** is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions.
- **JavaScript** (**JS**) is a lightweight, interpreted, or just-in-time compiled programming language with first-class functions. While it is most well-known as the scripting language for Web pages, many non-browser environments also use it, such as Node.js, Apache CouchDB and Adobe Acrobat. JavaScript is a prototype-based, multi-paradigm, single-threaded, dynamic language, supporting object-oriented, imperative, and declarative (e.g. functional programming) styles.

JAVASCRIPT IS . . .

- **JavaScript** (**JS**) is the most popular, lightweight, high-level, weakly-typed, dynamic , interpreted, or just-in-time compiled, single-threaded, prototype-based, multi-paradigm, multi-platform language, supporting object-oriented, imperative, and declarative (e.g. functional programming) styles.

ONE OF THE MOST POPULAR

- Top 1 in GitHub ratings;
- Top 3 in PYPL ratings;
- Top 1 by stack-overflow;

LIGHTWEIGHT

- Newbies friendly;
- High-level (abstraction layers);
- Weakly-typed;
- Fast enough (2–7 times slower than corresponding C++) since V8 released (2008);

HIGH-LEVEL

- Abstraction layers;
- Usability;
- Syntax sugar;

WEAKLY-TYPED OR UNTYPED OR DYNAMIC TYPED

- No type declaration;
- Data types are assumed automatically;
- Data types can be changed;
- Implicit conversion;
- + Save time learning;
- - Potential problems;

DYNAMIC, INTERPRETED

- Not pre-compiled, instead parsed and compiled “on the fly”;
- Code evaluated and executed at runtime;
- Code can change at runtime (e.g. type of a variable);

SINGLE-THREADED, NON-BLOCKING, ASYNCHRONOUS

- Javascript code is executed in a **single thread** but Javascript **runtime** is **not** run in single thread. Thread pool exists in JS runtime but we don’t have to worry about it as Runtime takes care of it. **Event loop;**
- **Asynchronous** implemented by WebAPIs and callbacks;
- Threads are not blocking by errors, but you can block event loop;

PROTOTYPE-BASED

- Inherit parents props by prototypes;
- In javascript everything is act like object;

MULTI-PARADIGM

- Imperative;
- Declarative;
- OOP;
- Functional Programming;
- Reactive Programming;
- Structured Programming;
- Procedural Programming;

BRIEF HISTORY

- 1995 - Netscape introduces “LiveScript” / “JavaScript”;
- 1996 - Microsoft releases its own version for IE called “JScript”;
- 1996 - JavaScript submitted to ECMA International to start standardisation;
- 1997 - 2005 - Standardisation efforts, Microsoft didn’t really join and support the standardised JS version though;
- 2008 - Google Chrome was released with V8 engine;
- 2009 - Ryan Dahl created Node.js;
- 2012 - TypeScript released by Microsoft;
- 2015 - ES2015 released;