

# **FIT5032 - Internet Applications Development**

Fundamental Client Side JavaScript

Prepared by - Jian Liew

Last Updated - 30th May 2018

Monash University (Caulfield Campus)

# Outline

- A Reintroduction to JavaScript
- What is JavaScript?
- What is ECMAScript?
- Programming Paradigm
- First Class Functions
- Closures
- Data Structures & Types
- JavaScript Libraries

If you are an advance student, you probably heard of things like ReactJS, AngularJS, VueJS, or similar frameworks.

These will not be covered until later in the semester.

What we aim to cover here, is more towards traditional JavaScript development, which is still widely used.

Besides that, it is a good idea to first gain a solid understanding of what JavaScript before we venture into the newer technologies.

# Amount of programming needed

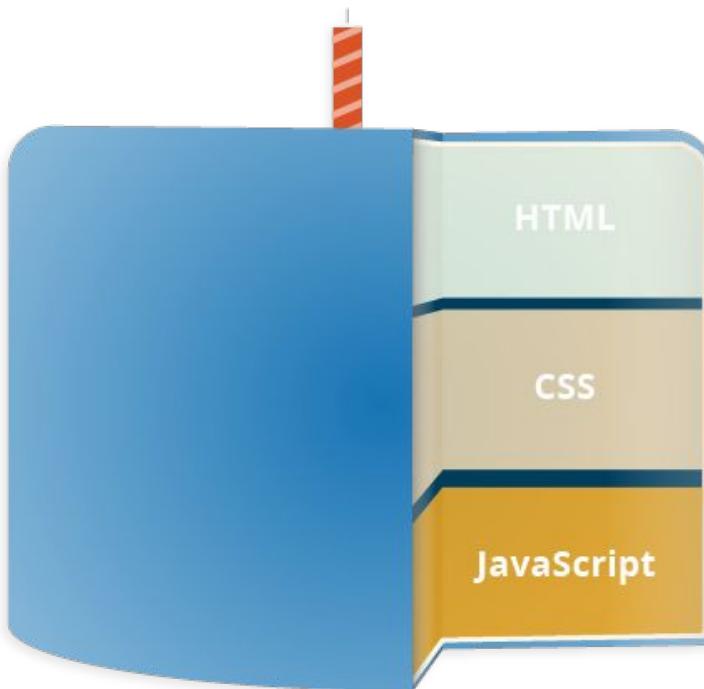
- The amount of programming needed for this subject is actually **considerably smaller in comparison to other subjects.**
- In this subject, we will be using **tools which are ready made** for us to simplify the development process, but it is still important for us to understand how to properly utilise the tools.
- In other words, we will **try “not reinvent the wheel”.**
- However, at the end of the day **everything depends on the context**. This subject is designed so that we try to avoid “reinventing the wheel”. In real life, however often times there are times when reinventing the wheel is needed.



# Before we venture into JavaScript...

It is assumed that you have the basic background of

- A general understanding of the Internet and the World Wide Web
- Good working knowledge of HyperText Markup Language (Background topics & Self Learning)
- Programming Experience. (Completed a foundation programming unit)



- HTML is the markup language that we use to structure and give meaning to our web content, for example defining paragraphs, headings, and data tables, or embedding images and videos in the page.
- CSS is a language of style rules that we use to apply styling to our HTML content, for example setting background colors and fonts, and laying out our content in multiple columns.
- JavaScript is a scripting language that enables you to create dynamically updating content, control multimedia, animate images, and pretty much everything else. (Traditionally)

# JavaScript Style Guide

A large yellow square containing the letters "JS" in a bold, black, sans-serif font.

A style guide is a **set of standards** that outline how code should be written and organized. As you read through these guides, you can get an idea for how code is written at the respective companies.

The main reason we need style guides is because everyone codes differently and it is important to have consistency amongst a group of developers.

There are several JavaScript style guides.

- AirBnB JavaScript Style Guide - <https://github.com/airbnb/javascript>
- Google JavaScript Style Guide - <https://google.github.io/styleguide/jsguide.html>
- JavaScript Standard Style Guide - <https://github.com/standard/standard>

In FIT5032, our Style Guide of choice is the AirBnB Style Guide. However, it does not matter which Style Guide we use as, it is more important for you to understand its **importance of a style guide.**

For those interested, here is a story <https://github.com/airbnb/javascript/issues/102>

# Wait..... JavaScript?

# Internet software war flares up

By Evan Ramstad

AP business writer

NEW YORK — A fight to shape the direction of the software industry sharpened Monday when Sun Microsystems Inc. and Netscape Communications Corp. rolled out a new product, hoping it will beat out those upcoming from Microsoft.

The new product, called Java Script, is designed to let everyday computer users, not just skilled programmers, make software that works easily on electronic networks, particularly the Internet.

Microsoft is due to promote its own software tool for such a purpose on Thursday.

It will likely be months or years before consumers and most businesses benefit. But the companies are now fighting to win the attention and respect of corporate programmers who decide what products become standard.

"The people they're trying to convince are the developers,"

## Sun, Netscape release JavaScript program

said Jerry Michalski, managing editor of Release 1.0, an industry newsletter.

With the evolution of faster communications lines and new kinds of computer devices, experts believe big opportunities are ahead for software that can be downloaded through a network from a different computer and used when needed.

Sun and Netscape hope to take charge of this opportunity before Microsoft, which has dominated the software industry since its operating program became the standard on IBM personal computers 15 years ago.

Sun earlier this year announced a programming language called Java to write software for network distribution. On Monday, the company revealed a simpler version, JavaScript, was ready for mass market tests.

A test version of JavaScript

'JavaScript is for someone who is at best an occasional programmer or when you want to do something that is lightweight.'

**Eric Schmidt**

chief technology officer at  
Sun

was made available with the test version of Netscape's new Navigator program, which people use to navigate the World Wide Web portion of the Internet.

"JavaScript is for someone who is at best an occasional programmer or when you want to do something that is lightweight," said Eric Schmidt, chief technology officer at Sun.

Mike Homer, vice president of

marketing for Netscape, said people who can create a specialized function in spreadsheets or word processors will be able to use JavaScript.

Sun and Netscape also lined up 28 companies to help promote and use JavaScript.

"For the Web to go big time, we need tools that make it easy for not just programming geniuses to create things," said John McCrea, a manager of Web software at Silicon Graphics, which endorsed the product.

On Thursday, Microsoft is expected to promote its Visual Basic language, now used to make Windows-related programs, for network applications.

Microsoft is also expected to announce several other network-related programs to go on sale next year, including one to use PCs instead of high-powered workstations as the device that serves information on the Web.

Asked about Java during a luncheon Monday in New York, Microsoft chairman Bill Gates said he wasn't worried about the competition.

"Java is there to overthrow what we've done," he said. "We feel very confident the things we're doing will get out even ahead of the good things that are provided with Java."

But Sun, Netscape and others advocate the ability to use network-delivered software on any kind of machine, not just Windows-equipped personal computers.

"Having the market be dominated by one company, stagnated, is what we do not want to have happen to the Web," said Marc Andreessen, vice president of technology at Netscape.

Sun announced the Java language this spring. Though there are relatively few Java programs now available, it has become closely associated with the Internet.

# A reintroduction to JavaScript

A yellow and blue decorative bar is positioned in the top right corner. It features a large, bold, black "JS" logo. The background of the slide has a diagonal gradient from white to blue.

JS

- JavaScript is considered to be the **most misunderstood programming language**.
- A common misconception is that JavaScript is **just another language**, but in actual fact, it is probably one of the **most important language** to learn these days.
- It is often derided as being a toy, but beneath its layer of deceptive simplicity, **powerful language feature awaits**.
- In an ill-fated marketing decision, it was named JavaScript to capitalize on the popularity of Java. In actual fact, it has **nothing** to do with Java.
- JavaScript language has **no concept** of input and output.
- It is designed to **run as a scripting language in a host environment**, and it is up to the host environment to provide mechanisms for communicating with the outside world.
- The most common host environment is the **browser**.

# What is JavaScript?

JS

- often shortened to **(JS)**
- is a **lightweight, interpreted or JIT** compiled programming language with **first class function**.
- the standard for JavaScript is **ECMAScript**
- is the most well known as the **scripting language** for Web pages.
- is a **prototype-based, multi-paradigm, dynamic scripting language, supporting object-oriented, imperative and declarative styles**.
- Please do **NOT** confuse JavaScript with the Java programming language. They are two different programming language with **different syntax, semantics and uses**.
- These day it is important to understand that, there are
  - **Client Side** JavaScript (JavaScript Code that is ran on the client, browser)
  - **Server Side** JavaScript (Runs on the server, for example **node.js**)
- These slides aims to cover **Client Side JavaScript**, we will cover server side in later weeks.
- Client side JavaScript **can change the way the web pages look**.

# What is ECMAScript?

- is the scripting language that forms the **basis of JavaScript**
- is currently at **version 7** (however most browsers have full support for version 5)
- JavaScript is **standardized at ECMA International**. (European association for standardizing information and communication systems)
- The ECMAScript standard is documented in the ECMA-262 specification.
- The ECMAScript specification does not describe the Document Object Model (DOM), which is standardized by the World Wide Web Consortium (W3C) and/or WHATWG (Web Hypertext Application Technology Working Group).
- ECMA Standard is based on several originating technologies, the most well known being JavaScript (Netscape) and JScript (Microsoft).
- So, **JavaScript is the most popular implementation** of the EMCA Script standard.

# JavaScript is a multi paradigm language

JS

- **Programming paradigms are a way to classify programming languages based on their features.** Languages can be classified into multiple paradigms.
- For example, in object-oriented programming (OOP), code is organised into objects that contain state that is only modified by the code that is part of the object. For example Java.
- A multi-paradigm programming language is a programming language that **supports more than one programming paradigm**.
- JavaScript is a **multi paradigm** language.
- JavaScript supports
  - Declarative & Imperative Programming
  - Prototype based supporting object oriented

# Declarative vs Imperative Programming

JS

Characteristic	Declarative	Imperative
Example	HTML, XML, CSS, SQL, Prolog, Haskell, F# and Lisp. Functional programming is a form of declarative programming.	Object oriented programming belongs under imperative paradigm. Examples are like C# and Java
Explanation	The code focuses on building logic of software without actually describing its flow.	The code focuses on creating statements that change program states by creating algorithm that tell the computer how to do things.
State Changes	Non-existent.	Important.
Order of execution	Low importance	Important
Primary Flow Control	Function calls, including recursion.	Loops, conditionals, and function (method) calls.
Primary manipulation unit	Functions as first-class objects and data collections.	Instances of structures or classes.
Example	<pre>const sum = a =&gt; b =&gt; a + b; console.log (add (5) (3)); // 8</pre> <p>This example here uses something known as JavaScript arrow functions.</p>	<pre>class Number {   constructor (number = 0) {     this.number = number;   } }</pre>

# Dynamic Scripting Language

JS

- JavaScript is a loosely typed dynamic language.
- Variables in JavaScript are not associated to any particular type.
- Any variable can be **assigned (and re-assigned)** values of all types:

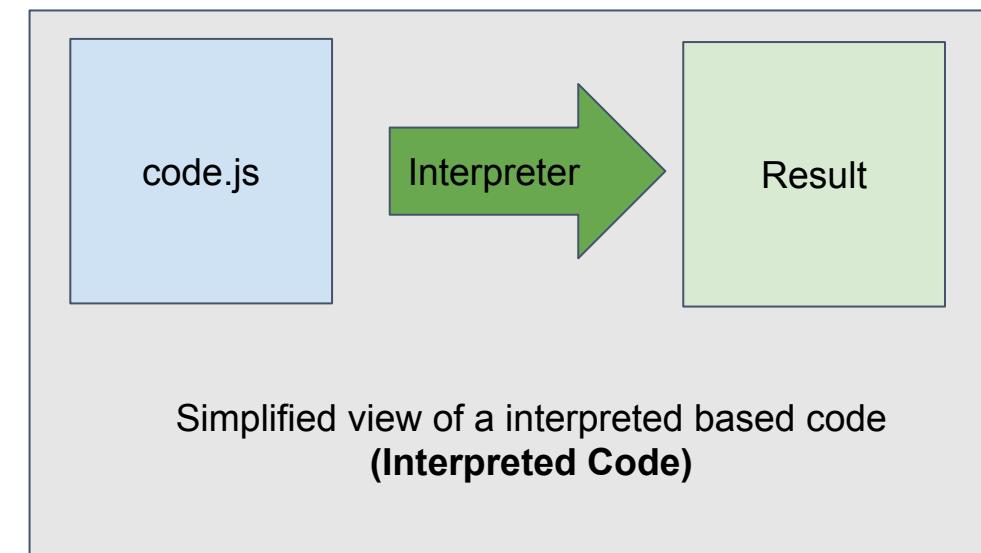
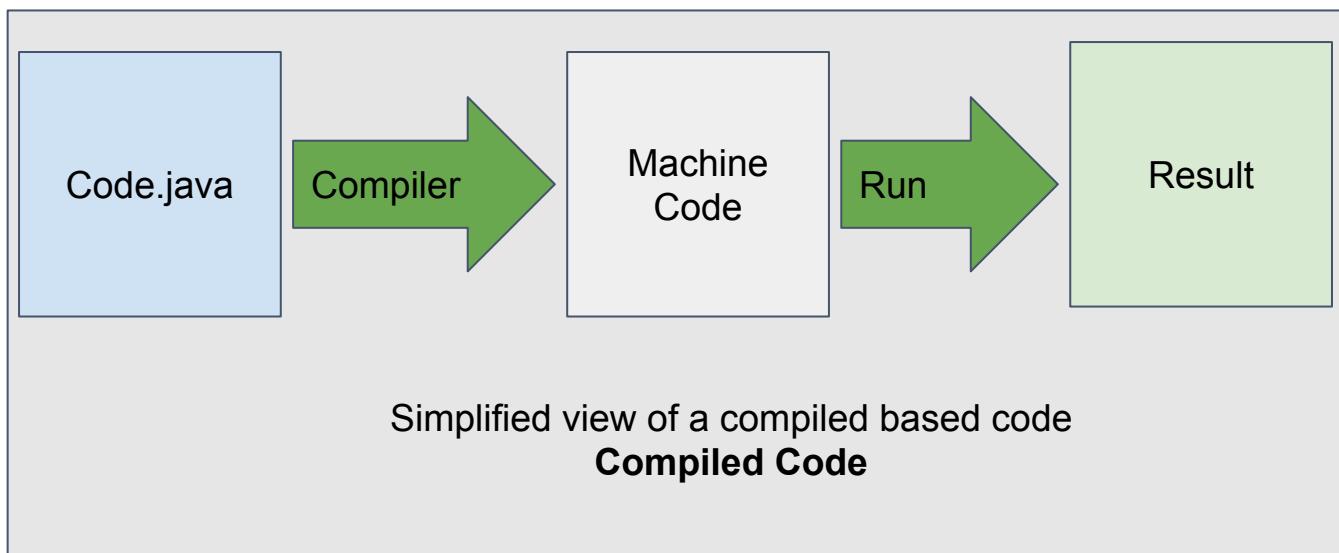
```
var foo = 42; // foo is now a Number
```

```
var foo = 'bar'; // foo is now a String
```

```
var foo = true; // foo is now a Boolean
```

# Interpreted vs Compiled Code

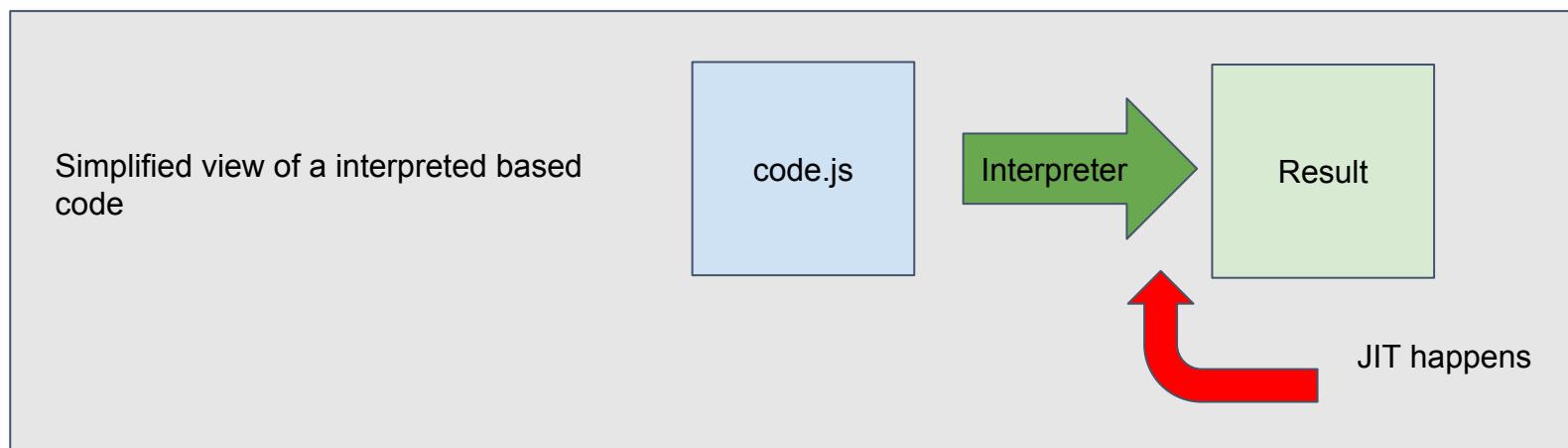
- JavaScript is an **interpreted programming** language.
- The code is run from **top to bottom** and the result of the running code is immediately returned.
- You **do not have to transform** the code into a different form before the browser runs it (interpreted)
- For example a language like **Java** on the other hand are transformed (compiled) into another form before they are run by the computer.
- Remember that a language is **never bound to either interpreted or compiled**, but the implementation of the language is. This is true for both Java (FIT9131) & Python (FIT9133) \*.



# JIT Compiled Programming Language

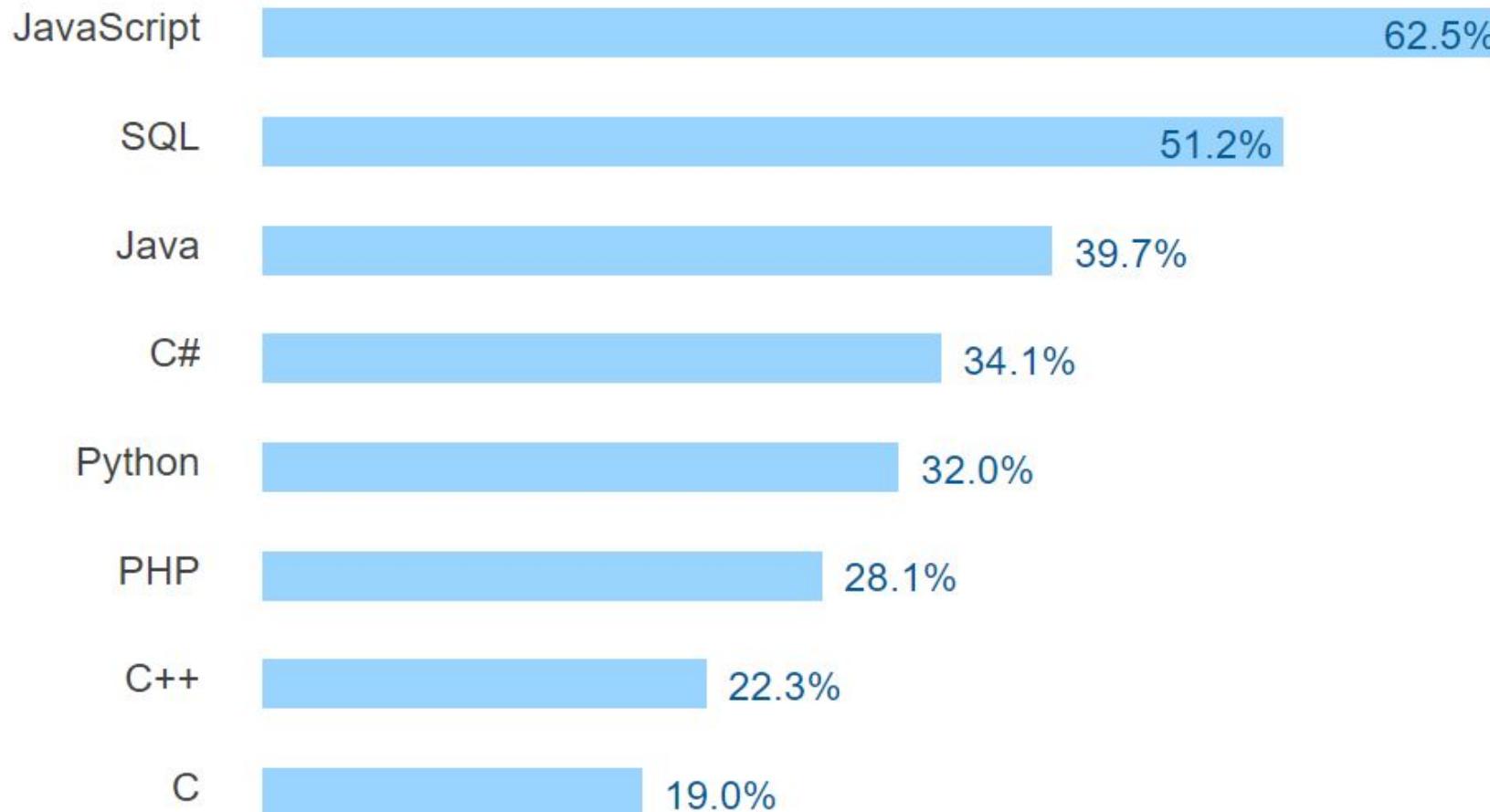
JS

- JavaScript has “**no**” compilation “step”. It has an **interpreter**.
- An interpreter in the browser reads over the JavaScript code.
- It interprets each line and runs it.
- Modern browsers use a technology known as **Just-in-Time compilation**, which compiles JavaScript into executable bytecode **just about it is to run**.
- At the end of the day, programming languages are created for humans. **It must be translated into machine language**. So in a way, there must be a “compilation” step but it happens **Just In Time**.
- JIT is not specific to JavaScript. Languages like Java also do have these mechanisms to compile the code just before the execution.
- The most important concept when describing an interpreted language is, the compiled language takes a longer time to prepare itself to start executing while an interpreted language like JS starts executing in **no time**.



JIT happens  
microseconds before  
code is executed. This  
ensures the fastest  
performance.

# Most popular technologies



Based on a Stack Overflow Developer Survey (2017), JavaScript is currently the **most popular technology**.

Java is the 3rd most popular technology. While C# is the 4th most popular. (This might be due to Unity)

# First Class Function?

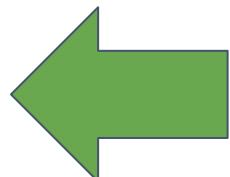
JS

A programming language is said to have first class functions when **functions are treated like any other variable**.

A function can be **passed as an argument to other functions, can also be returned by another function, can be assigned as a value to a variable or stored into a data structure**.

For example, in JavaScript, you can do something like

```
function sayHello() {  
  return function() {  
    console.log("Hello!");  
  }  
}
```



In this example, we **returned a function from another function**. The reason we can do this is because we treat function in JavaScript as a value. This is a very important concept in JavaScript.

# Higher Order Function (callback function)

In mathematics & computer science, a higher-order function is a function that **does at least one of the following**

- takes or more functions as arguments
- returns a function as a result

Both of these requirements rely on JavaScript functions being first class objects in a language.

Because of this characteristic, it is well suited for **functional programming**.

# Closures

JS

- A closure is the combination of a **function and the lexical environment** within which that function was declared.
- A closure is one way of supporting first-class functions; it is an expression that can reference variables within its scope, be assigned to a variable, be passed as an argument to a function or be returned as a function result.
- **Functions in JavaScript form closures.**

```
function makeAdder(x) {  
  return function(y) {  
    return x + y;  
  }  
}  
  
var add5 = makeAdder(5);  
var add10 = makeAdder(10);  
  
console.log(add5(2)); // 7  
  
console.log(add10(2)); //12
```

## The example explained....

- 1) makeAdder is a function which takes a single argument x and returns a new function. The function it returns takes a single argument y, and returns the sum of x and y.
- 2) In a way, makeAdder is a function factory. It creates functions which can add a specific value to their argument.
- 3) add5 and add10 are both closures. They **share the same function body definition but store different lexical environments.**
- 4) add5's lexical environment, x is 5 while add10, x = 10

# Server-side versus client-side code

JS

- Client-side code is code that is ran on the user's computer - when the page is viewed, the client side code is downloaded and then run and displayed by the browser. This is what we call client-side JavaScript.
- Server-side code is ran on the server. Then the results are downloaded and displayed on the browser. Examples of popular server side web languages include, PHP, Python, Ruby and ASP.NET.
- JavaScript can also be used as server-side. Recently, there is a huge surge in the popularity of using JavaScript as server side code in the Node.js environment.

# Dynamic versus static code

- The word dynamic is used to describe both client-side JavaScript and server-side languages.
- It refers to the ability to update the display of a web page to show different things in different circumstances. In a way, generating new contents when required.
- Server side code dynamically generates new contents on the server, whereas client side JavaScript dynamically generates new contents inside the browser on the client. For example, creating a new HTML table, filing it with data requested from the server, then displaying the table in a web page shown to the user.
- For FIT5032, we will use ASP.NET for the server side codes. So, in combination these two will work together.
- A web page with no dynamically updating content is referred to as static. It just shows the same content all the time.

In short.....

JS

JavaScript is a **prototype-based, multi-paradigm, dynamic scripting language**, supporting **object-oriented, imperative** and **declarative** styles.



# JavaScript Basics

JS

- JavaScript borrows most of its syntax from Java, but is also influenced by **Awk, Perl and Python**.
- JavaScript is case-sensitive and uses the Unicode character set. For example, the word Früh (which means "early" in German) could be used as a variable name.
- But, the variable früh is not the same as Früh because JavaScript is case sensitive.
- In JavaScript, instructions are called statements and are separated by a semicolon (;).
- A semicolon is **not necessary after every statement** if we are writing a command from a new line.
- If we want to write more than one statement in one line, then they should be separated by a semicolon.
- The source text of JavaScript script gets scanned from left to right and is converted into a sequence of input elements which are tokens, control characters, line terminators, comments or whitespace.
- ECMAScript also defines certain keywords and literals and has **rules for automatic insertion of semicolons (ASI)** to end statements.
- It is recommended to always add semicolons to end your statements; it will avoid side effects

# Data Structures and Types

JS

The latest ECMAScript standard defines seven data types:

1. Boolean → true and false.
2. null → A special keyword denoting a null value. Because JavaScript is case-sensitive, null is not the same as Null, NULL, or any other variant.
3. undefined → A top-level property whose value is not defined.
4. Number → An integer or floating point number. For example: 42 or 3.14159.
5. String → A sequence of characters that represent a text value. For example: "Howdy"
6. Symbol (new in ECMAScript 2015) → A data type whose instances are unique and immutable.
7. Object

# $0.1 + 0.2$

The screenshot shows the browser developer tools open to the 'Console' tab. The tabs at the top are Elements, Console (which is selected), Sources, Network, Performance, and Memory. Below the tabs, there are icons for back, forward, and search, followed by the text 'top'. A dropdown menu is open next to 'top', and a 'Filter' input field is present. To the right of the filter are buttons for 'Default levels' and 'Group similar', and a gear icon for settings. The main area displays the following text:  
Console was cleared  
< undefined  
>  $0.1 + 0.2$   
< 0.3000000000000004  
> |

Floating number can't store properly all decimal numbers, because they store stuff in binary.

So, when  $0.1 + 0.2$  happens the value given is inaccurate, this is due to the nature of computers store these numbers.

# JavaScript Libraries

- Pre Written JavaScript which allows for easier development of JavaScript applications.
- Examples of JavaScript libraries are like
  - jQuery
  - jQueryUI
  - Google Maps Platform
  - Leaflet.js
- Please also note there is a difference between a JavaScript library and a JavaScript framework. A framework defines would define the entire application design.
- Examples of a framework would be like **(Shown on Week 10)**
  - AngularJS
  - VueJs
  - React

# What is jQuery?

- jQuery is a fast, small and feature-rich JavaScript **library**.
- It makes things like **HTML document traversal and manipulation, even handling, animation and AJAX much simpler with easy-to-use API**.
- jQuery works across a multitude of browsers.
- As of 30th April 2018, jQuery is at version 3.3.1 with versions 1 and 2 no longer being patched.
- These days, there are less and less people using jQuery as there are alot more tools which can do a better job in comparison to jQuery. [Link](#)
- However, millions of website still uses jQuery and it is currently the number 1 choice of developers for small and medium web applications. It is the most widely deployed JavaScript library by a large margin.
- It is needed for every web developer to at least understand the capabilities of jQuery due to its usage.

# What is jQuery UI?

- jQuery UI is a collection of GUI widgets, animated visual effects, and themes implemented with jQuery.
- One of the more common reasons to use jQuery UI is for their widgets.

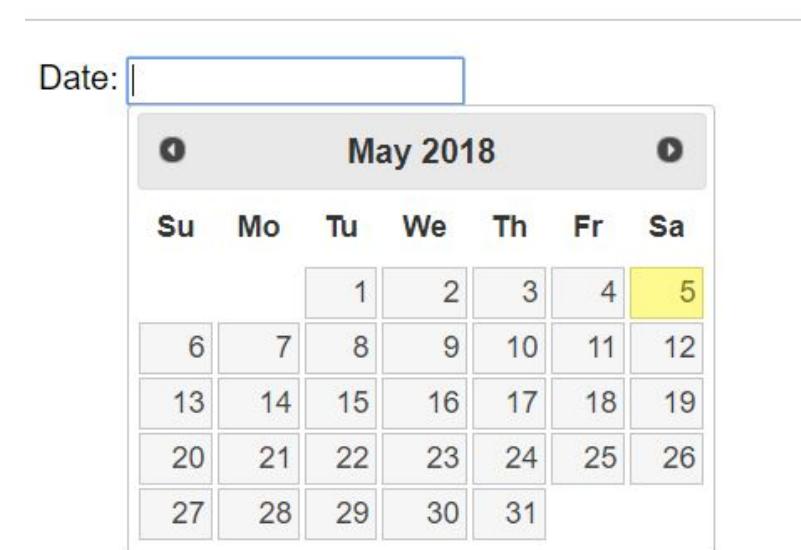
▼ Section 1

Mauris mauris ante, blandit et, ultrices a, suscipit eget, quam. Integer ut neque. Vivamus nisi metus, molestie vel, gravida in, condimentum sit amet, nunc. Nam a nibh. Donec suscipit eros. Nam mi. Proin viverra leo ut odio. Curabitur malesuada. Vestibulum a velit eu ante scelerisque vulputate.

▶ Section 2

▶ Section 3

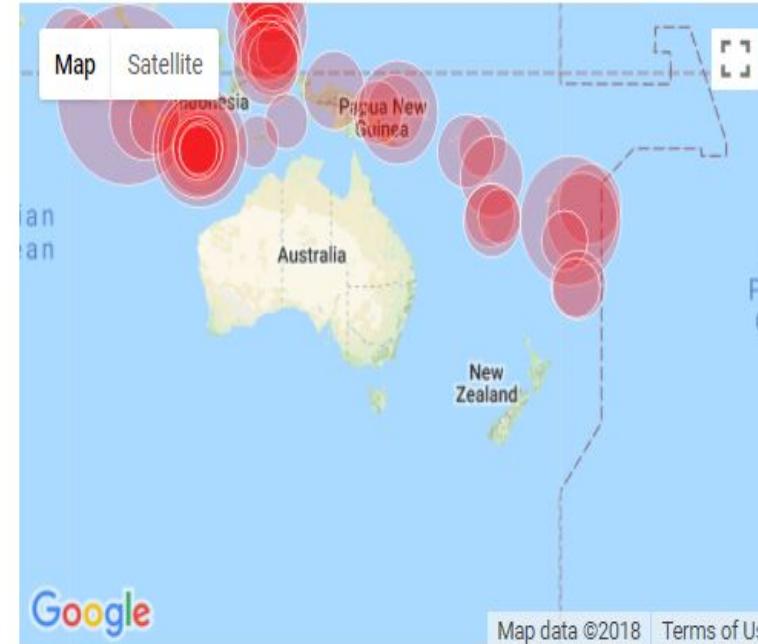
▶ Section 4



A Datepicker

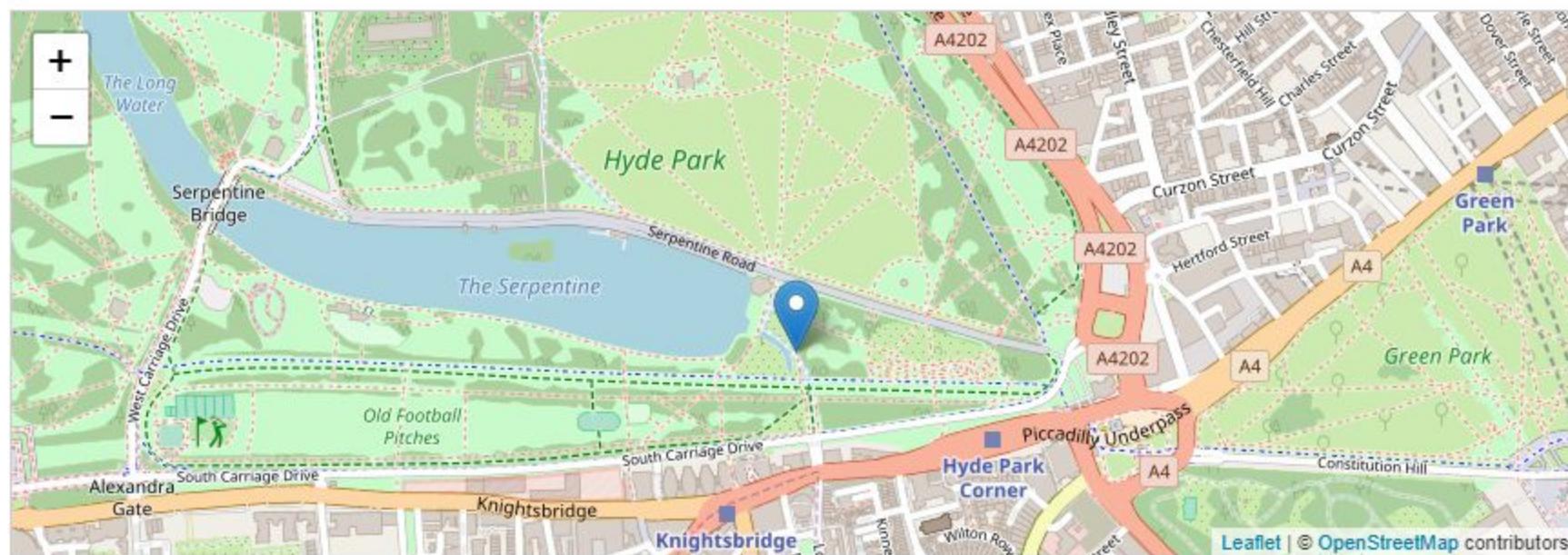
# Google Maps JavaScript API

- Google Maps Platform
- The Google Maps JavaScript API lets you customize maps with your own content and imagery for display on web pages and mobile devices.
- Using maps is a very common use case.



# Leaflet.js

- An open source JavaScript library for mobile-friendly interactive maps.
- It is the leading open-source JavaScript library for mobile-friendly interactive maps. (With just 38 KB of JS).
- It is designed with simplicity, performance and usability in mind.



# d3.js

d3.js is a JavaScript library for **manipulating documents based on data**. D3 helps you bring data to life using HTML, SVG, and CSS. This is considered to be basic visualisation technique.



This is an example of a [word cloud](#) generated from the about page of the FIT website. d3 is capable of doing much more!

You can view the showcase at  
<https://github.com/d3/d3/wiki/Gallery>

# DataTables

DataTables is a plug-in for the jQuery Javascript library. It is a highly flexible tool, build upon the foundations of progressive enhancement, that adds all of these advanced features to any HTML table.

- Pagination
- Instant Search
- Multi Column Ordering
- Use almost any data source
- Mobile Friendly
- Fully internationalisable

Show 10 entries						Search:
Name	Position	Office	Age	Start date		
Airi Satou	Accountant	Tokyo	33	2008/11/28		
Angelica Ramos	Chief Executive Officer (CEO)	London	47	2009/10/09		
Ashton Cox	Junior Technical Author	San Francisco	66	2009/01/12		
Bradley Greer	Software Engineer	London	41	2012/10/13		
Brenden Wagner	Software Engineer	San Francisco	28	2011/06/07		
Brielle Williamson	Integration Specialist	New York	61	2012/12/02		
Bruno Nash	Software Engineer	London	38	2011/05/03		
Caesar Vance	Pre-Sales Support	New York	21	2011/12/12		
Cara Stevens	Sales Assistant	New York	46	2011/12/06		
Cedric Kelly	Senior Javascript Developer	Edinburgh	22	2012/03/29		

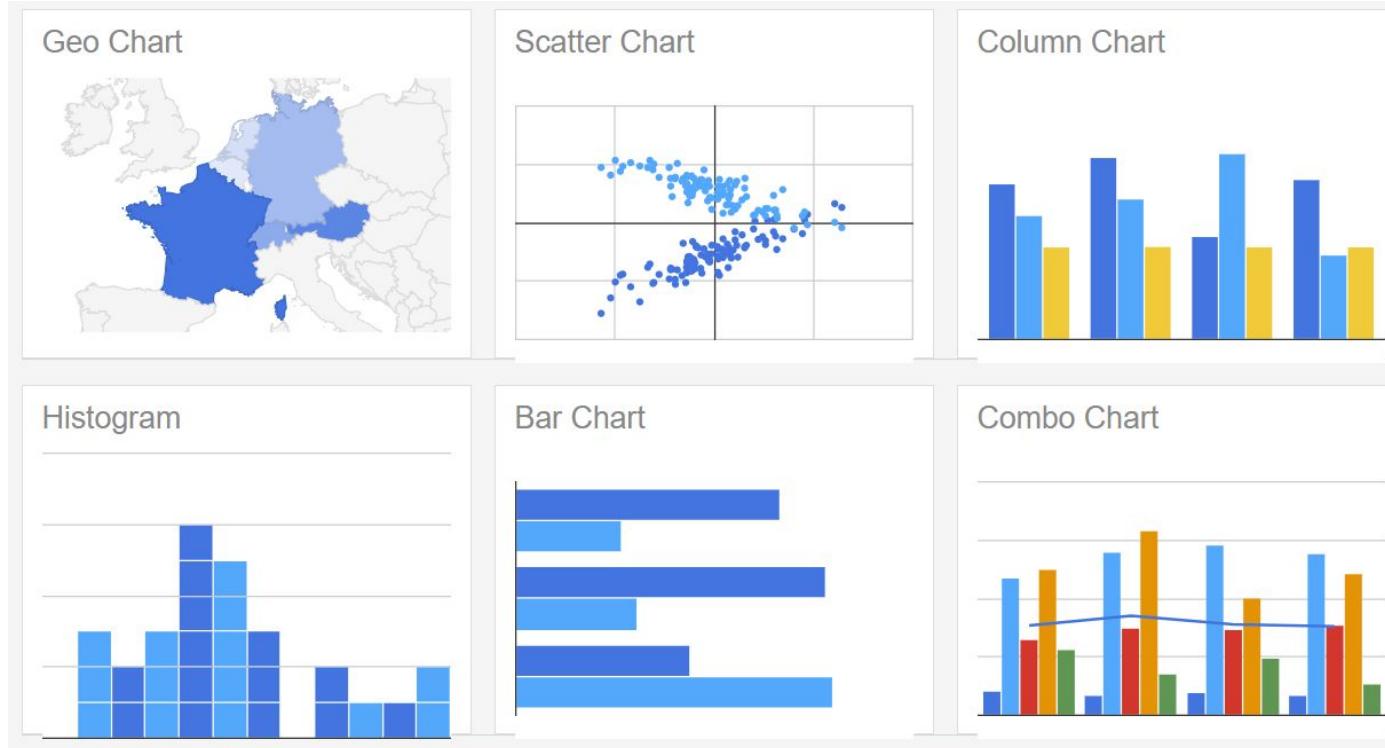
Name	Position	Office	Age	Start date

Showing 1 to 10 of 57 entries

Previous [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) Next

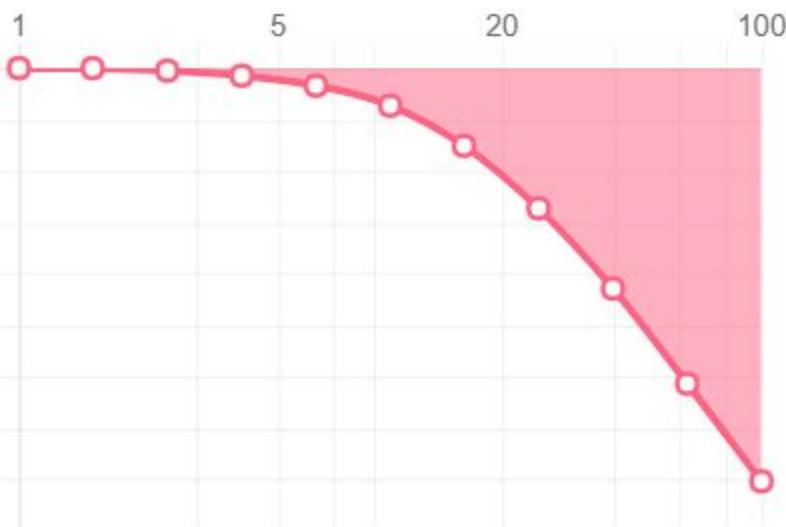
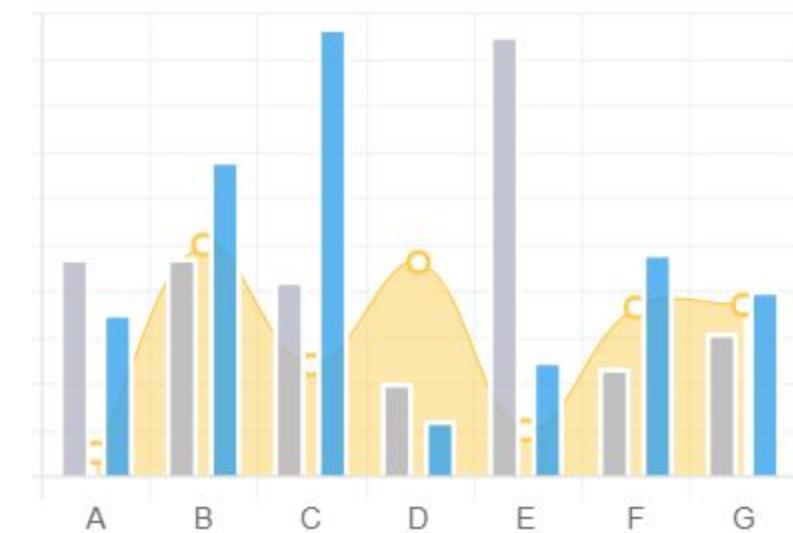
# Google Charts

- provides a perfect way to visualize data on your website



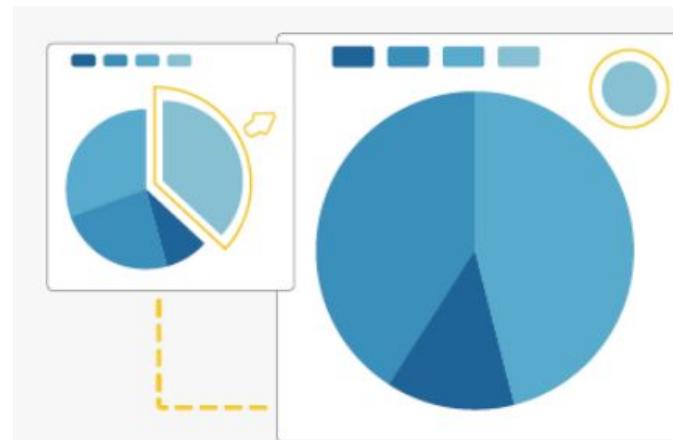
# Chart JS

Simple yet flexible JavaScript charting for designers & developers

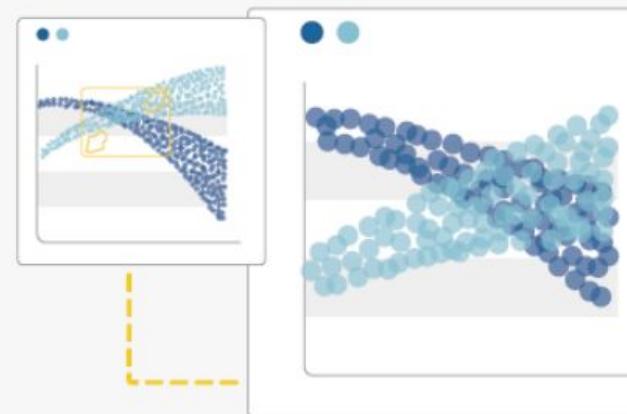


# Baidu eCharts

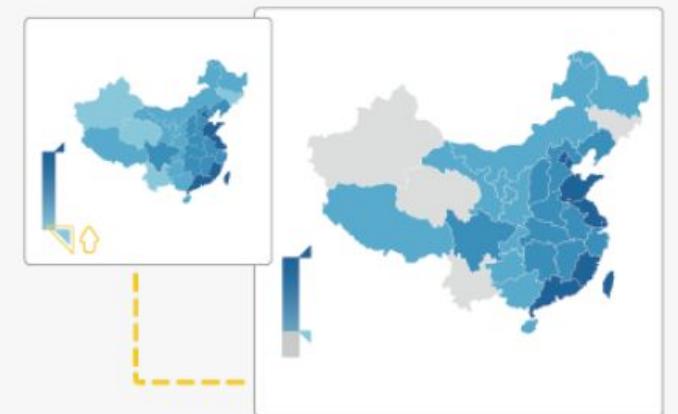
- ECharts (a contraction of Enterprise Charts) is a commercial charting solution originally intended to address the report need of the Company's various business systems, Baidu.



DRAG-RECALCULATE

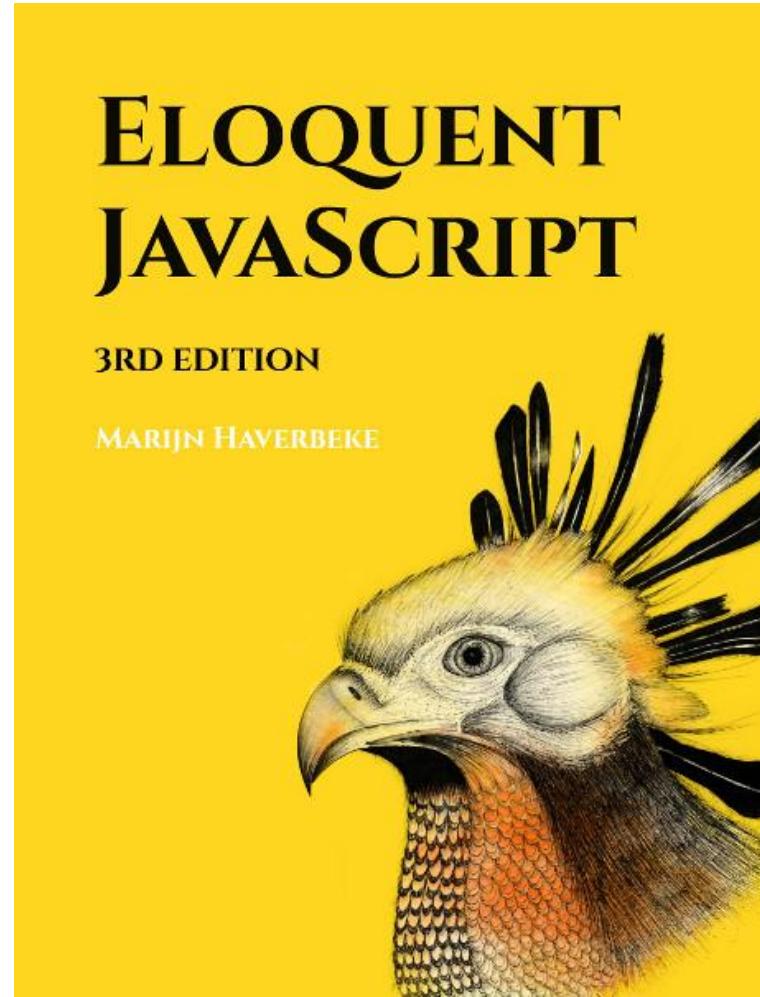
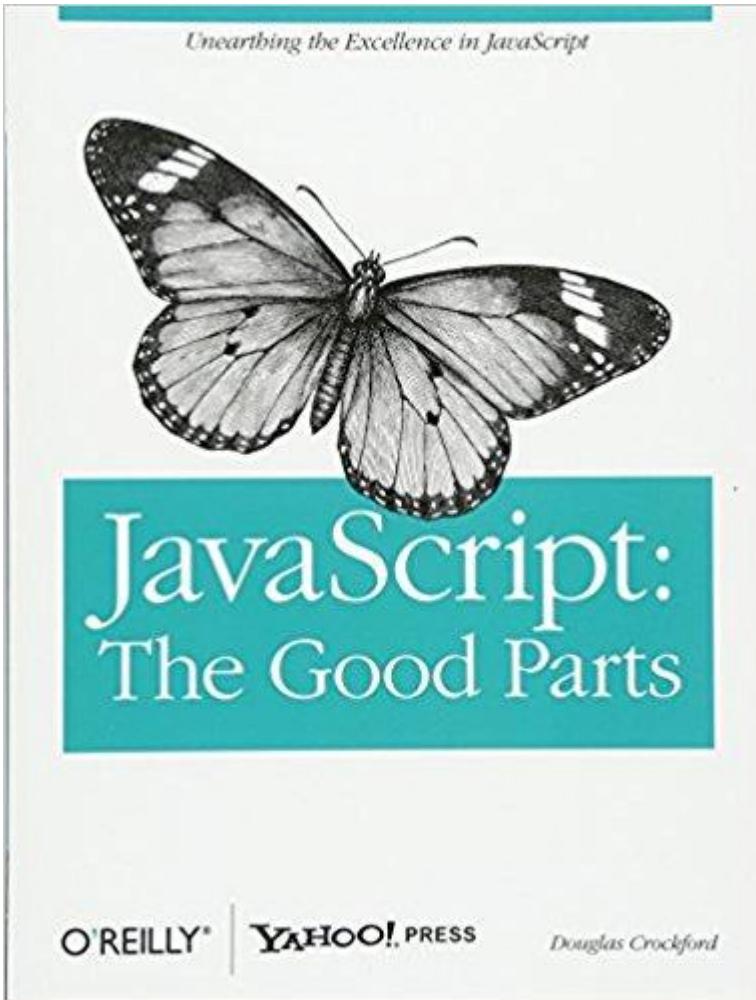


BIG DATA MODE



SCALE ROAMING

# Recommended readings for JavaScript



# In the labs....

- You will be writing simple JavaScript
- You will learn the importance of loading JavaScript in order
- You will learn how to use jQuery
- How to use Google Maps
- How to use DataTables