Week 5 Day 1 29/05/23

More on Objects, Methods, and Intro to OOP.

# Datatypes

Describes the different types or kinds of data that you will be working with and storing in variables.

The main datatypes in JavaScript include:

* String type
* Number type
* Boolean type
* Objects
* Arrays

# Objects

Objects are an unordered collections of key/value pairs, where the

* Keys are strings
* Values can be any type, even other objects.

Objects are defined by the list of pairs key: value, comma-separated and enclosed by curly braces.

A picture containing text, font, screenshot, graphics

Description automatically generated

# Dot Operator



The dot notation can be used to access the property of an object. For example, to access the firstName property of the person object, you use the following expression:



# Square bracket notation – []

The square brackets property accessor has the following syntax.



To access the value of an object’s property via the array like/square bracket notation, we use



Adding/Removing an object property

A JavaScript object is a collection of unordered properties.

Properties are the values associated with a JavaScript object.

You can add new properties to an existing object by simply giving it a value.



The delete keyword deletes a property from an object.

The delete keyword deletes both the value of the property and the property itself.



Iterating over properties of an object.

The for...in statement iterates over the properties of an object.

A picture containing text, screenshot, font

Description automatically generated

# A picture containing text, font, screenshot, graphics Description automatically generatedFunctions

A function is a block of organized, reusable lines of code that is used to perform a single, related action.

A function definition (also called a function declaration, or function statement) consists of the function keyword , followed by•

The name of the function.

A list of parameters to the function, enclosed in parentheses and separated by commas.

The JavaScript statements that define the function, enclosed in curly brackets {…}.

What we’ll look at today

Execution Context & Call Stack

More on Methods

The this keyword

The new keyword

Object oriented programming

Classes & Objects

Constructors

# Execution contexts

Execution context is the environment in which JavaScript code is

evaluated and executed

All code in JavaScript runs inside an execution context

Ultimately; what happens inside the execution context is the parsing of code

line by line and the storing of variables and functions into memory

There are two types of context in JavaScript:

Global context

Function/Local context

Global Execution Context

•

It is the first thing that is createdA picture containing text, screenshot, display, software

Description automatically generated when you write JavaScript code.

It is the default context.

When the JS engine starts reading

your code, it creates the global

execution context.

It starts parsing line by line and adds

your variables to memory also known

as global variable environment

A picture containing text, screenshot, font

Description automatically generatedA picture containing text, screenshot, font

Description automatically generated

# Local Execution Context

While the JavaScript engine is parsing, if it needs to

execute a function, a new local execution context is created.

In that execution context, parsing takes place and the number variable is added to local memory, and then parsing continues.

After this, the engine returns to the previous execution context.

Exiting the local execution context and continuing parsing in the previous execution context is achieved with the return

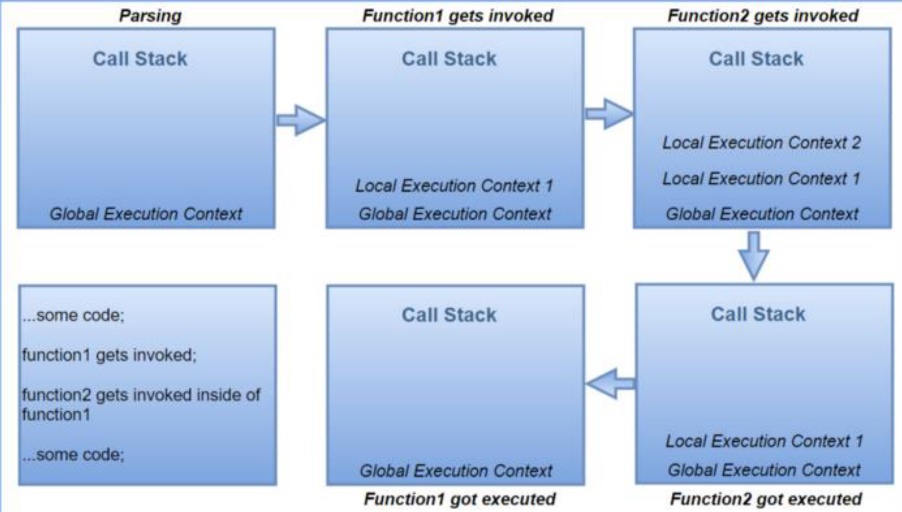
keyword.

Every time a function gets called, this happens again.

Every function call results in a different local execution context.

The Call Stack

•

The Call Stack is a

mechanism for the

JavaScript Engine to

keep track of

execution contexts ,

which to enter,

which to exit or

which to return to.

•

At the bottom of the

stack is the global

execution context