Day -1
Introduction to Browser & web

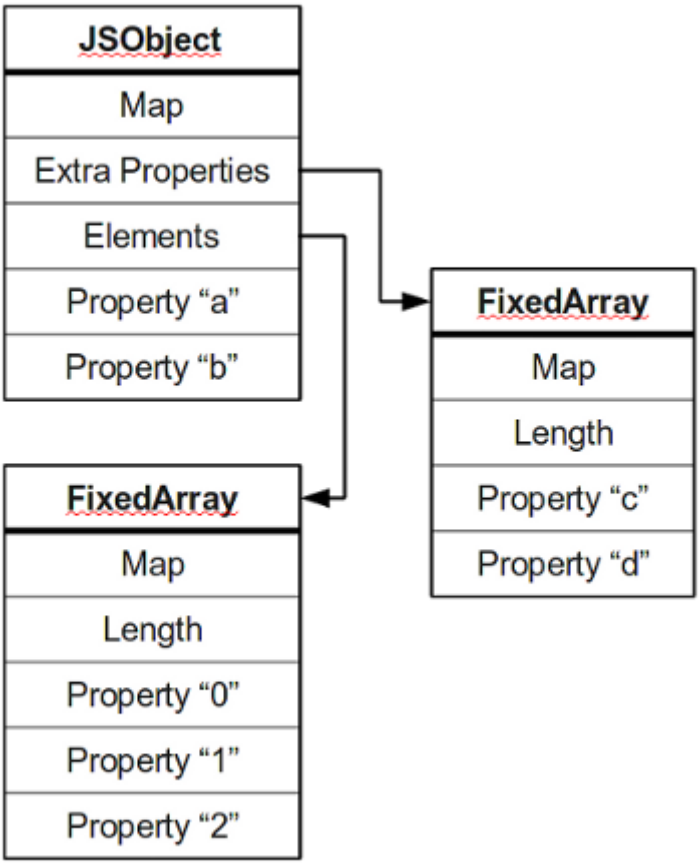
Date	24 June 2023
Batch ID	B49 WD3 Tamil
Name	Puvanesan .A

Objects of JavaScript :

- Booleans can be objects
- Numbers can be objects
- Strings can be objects
- Dates are always objects
- Maths are always objects
- Regular expressions are always objects
- Arrays are always objects
- Functions are always objects
- Objects are always objects

JavaScript's Internal representation of Objects :

A simple diagram is probably the best way to give a quick overview of the object representation in Javascript.



Most objects contain all their properties in a single block of memory (“a”, and “b”). All blocks of memory have a pointer to a map, which describes their structure.

Named properties that don’t fit in an object are usually stored in an overflow array (“c”, and “d”). Numbered properties are stored separately, usually in a contiguous array.

The JavaScript standard allows developers to define objects in a very flexible way, and it is hard to come up with an efficient representation that works for everything. An object is essentially a collection of *properties*: basically key-value pairs. You can access properties using two different kinds of expressions:

- obj. prop
- obj[“prop”]

According to the spec, property names are always strings. If you use a name which is not a string, it is implicitly converted to a string. This may be a little surprising: if you use a number as a property name, it gets converted to a string as well (at least according to the spec). Because of this, you can store values at negative or fractional array indices. So a JavaScript object is basically a map from strings to values.