Objects and its internal Representation in js

Objects, in JavaScript, is it's most important data-type and forms the building blocks for modern JavaScript. These objects are quite different from JavaScript's primitive data-types(Number, String, Boolean, null, undefined and symbol) in the sense that while these primitive data-types all store a single value each (depending on their types).

Objects are more complex and each object may contain any combination of these primitive data-types as well as reference data-types.

An object, is a reference data type. Variables that are assigned a reference value are given a reference or a pointer to that value. That reference or pointer points to the location in memory where the object is stored. The variables don't actually store the value.

Objects in JavaScript may be defined as an unordered collection of related data, of primitive or reference types, in the form of "key: value" pairs. These keys can be variables or functions and are called properties and methods, respectively, in the context of an object.

An object can be created with figure brackets {…} with an optional list of properties. A property is a "key: value" pair, where a key is a string (also called a "property name"), and value can be anything.

We can create objects in many ways in JavaScript:

1.Object literal (Direct way) — Object literals are a comma-separated list of key-value pairs wrapped in curly braces. Object literal property values can be of any data type, including array literals, functions, nested object literals or primitive data type.

2. Object.create()— the method creates a new object with the specified prototype and properties of the old object.

3. Object Instance— The use of Objectconstructor in conjunction with the "new" keyword allows us to initialize new objects.

4. Object Constructor — Constructors can be useful when we need a way to create an object "type" that can be used multiple times without having to redefine the object every time and this could be achieved using the Object Constructor function.

5. Object.assign() —this is another method to create a new object from other objects. It copies the values of all enumerable own properties from one or more source objects to a target object. It will return the target object.

6. Object.fromEntries() — method transforms a list of key-value pairs into an object.