**objects and its internal representation**

Objects are quite different from JavaScript’s primitive data-types(Number, String, Boolean, null, undefined and symbol.

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 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.

o understand this rather abstract definition, let us look at an example of a JavaScript Object :

let bio = {  
name : “durgesh”,  
location : “chennai”,  
birth: “1997”  
}  
In the above example “name”, “location”, “birth” are all “keys” and “durgesh”, “chennai” and “1997 “are values of these keys respectively.

Each of these keys is referred to as properties of the object. An object in JavaScript may also have a function as a member, in which case it will be known as a method of that object.