# Objects And Its Internal Representation

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.
* Loosely speaking, **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.

Object :

let school = {

name : "Vivekananda School",

location : "Delhi",

established : "1971"

}

In the above example ***“****name****”, “****location****”, “****established****”****are all****“****keys****”****and****“****Vivekananda**School****”, “****Delhi****”*** *and**1971*are values of these keys respectively.

Each of these keys is referred to as properties of the object.

In case of array, An array can store a list of items that may have different data types with a key-value pair of sequential numbers to value. Only values are necessary in this case or we don't consider the keys but just values. The irony is we can even store objects and functions as an item in an array.

var person = I name: "John", age: "22"

Example of Array be like :

var emptyArray - []; // Empty Array

var person = ["Clark", "Kent", 36]; // Array of person

var car = ["BMW X3", "white", 5]; // Array of car

Here, the keys will be sequential numbers starting from '0' and they are called as index numbers. Objects are called by the variable name and key : Car.model / car["modal"] Arrays are called by the variable and index number : Car[0] Car[1] and goes on...

In the case of functions, the values are passed to it and the function returns some value if needed. Functions are first-class objects because they can have properties and methods just like any other object