

Blog about objects and its internal representation in Javascript

Objects are important data types in javascript. Objects are different than primitive datatypes (i.e. number, string, boolean, etc.). Primitive data types contain one value but Objects can hold many values in form of Key: value pair. These keys can be variables or functions and are called properties and methods, respectively, in the context of an object.

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.

Objects and properties

A JavaScript object has properties associated with it. A property of an object can be explained as a variable that is attached to the object.

Object properties are basically the same as ordinary JavaScript variables, except for the attachment to objects. The properties of an

object define the characteristics of the object. You access the properties of an object with a simple dot-notation:

```
objectName.propertyName
```

Create JavaScript Object with Object Literal

One of easiest way to create a javascript object is object literal, simply define the property and values inside curly braces as shown below

```
let bike = {name: 'SuperSport', maker:'BMW s1000rr', engine:'999cc'};
```

Understanding Objects in JavaScript

In JavaScript, objects are collections of key-value pairs, where keys are strings (or symbols) and values can be of any data type, including other objects. Objects are used to represent real-world entities, data structures, and more complex data types.

Internal Representation of Objects

Internally, JavaScript engines use various data structures to represent objects efficiently. One common approach is using a hash table or a similar structure to store the object's properties and their corresponding values. This allows for fast access and manipulation of properties.

```
Internal Representation:
{
  name: "Sathishkumar M",
  age: 28,
  email: "sathishmechon.111@gmail.com"
}
```