In JavaScript, an object is a collection of key-value pairs that represent properties of an entity. The keys are strings or Symbols, and the values can be any data type, including other objects. The internal representation of an object in JavaScript is called its "object structure" or "object layout."

This includes information about the object's properties, such as their names, values, and types, as well as information about the object's prototype and any other internal data.

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 the property name value can be anything.

Every object has some property associated with some value. These values can be accessed using these properties:

var myCar = new Object();

myCar.make = 'Suzuki';

myCar.model = 'Altros';

myCar.year = 1978;

myCar.wheels = 2;

After creating myCar object, the value inside the object can be accessed using keys:

*myCar.year*

Output: 1978

These values can be accessed using brackets notation also.

*myCar[year]*

Output: 1978

The syntax for adding a property to an object is :

**ObjectName.ObjectProperty = propertyValue;**

The syntax for deleting a property from an object is:

**delete ObjectName.ObjectProperty;**

The syntax to access a property from an object is:

**objectName.property**

**//or**

**objectName["property”]**