3) Write a blog about objects and its internal representation in JavaScript?

Objects – Objects in JavaScript can be compared with real life objects(for example – A car is an object). Objects are collection of properties which has association with the key and the value. Based upon this association objects are classified into three types namely- simple objects, array of objects, objects of objects(nested objects). JavaScript objects have properties which describes their properties. Objects are defined as an unordered collection of related data in the form of “key: value” pairs. These values can be updated using methods. Objects are variables too but can contain many values. Objects can be created in several methods in JavaScript namely,

* Object literals - In this method objects are created in between {…} which is the most simplest way of creating it. Object literals are expressions and each of them results in a new object whenever it is executed. The inputs can be anything. For example)

Var obj ={

Input1: value1,

Input2: value2,

};

* Object constructor - With the help of the constructor function an object is created in this method which defines the characteristics of the object. Once we create a constructor function we can create numerous objects. For example)

Creating a constructor:

function info(name, age, d.o.b){

studentname=name,

studentage=age,

studentd.o.b=d.o.b,

}

(now we can create as much object as we can since we have created the object constructor)

1. Varstudent1= new info (‘ram’,’18’,’12/3/2002’)
2. Varstudent2= new info (‘ashok’,’18’,’2/6/2002’)

* Prototypes - With the help of a fixed object prototype created several new objects can be created by calling the prototype. These objects are created by calling object.create(). For example)

Let students={

Assemblyrole:”singing”

}

Let student1= object.create(students);

Console.log(student1:assemblyrole);

Thus objects are represented in JavaScript and these object members are accessed using dot notations, bracket notations. There is delete operator in order to remove the unwanted property of an object.