**TASK 2**

**QUESTION 2:**

**Blog about objects and its internal representation in javascript.**

* 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.
* 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
* Like all JavaScript variables, both the object name (which could be a normal variable) and property name are case sensitive. You can define a property by assigning it a value. For example, let’s create an object named myCar and give it properties named make, model, and year as follows:
* var myCar = new Object();  
  myCar.make = 'Ford';  
  myCar.model = 'Mustang';  
  myCar.year = 1969;
* Unassigned properties of an object are [undefined](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/undefined) (and not [null](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/null)).
* myCar.color; // undefined
* myCar['make'] = 'Ford';  
  myCar['model'] = 'Mustang';  
  myCar['year'] = 1969;