Homework #JSIV

1. Explanations of terms as explained to a 12-year-old –

* **Objects** are a type of container in computer programming that holds a lot of information about a particular thing. An object is defined with its own properties and methods. The objects properties define the characteristics of the object and its methods are actions that the object, and only that object can perform. An object is created by giving it a variable name and defining its properties and methods within curly brackets. Here is an example object definition that defines a particular driver’s car.

const car = {

make: ‘Ford’,

model: ‘Focus’,

year: ‘2011’,

color: ‘blue’,

condition: ‘excellent’,

regularMaintenance: true,

};

* **Properties** are all the values that are defined within an object that are not functions. The properties are given a name of the property, known as the key, and a value. These are known as key value pairs. The properties are descriptive of that particular object and can include all different data types, such as strings, numbers, Booleans, or they can be null, or undefined for the particular object they describe. Any object can call any of its properties with its associated key to alter it, or perform other operations, without affecting any other objects key:value pairs.
* **Methods** are functions that are defined within a particular object. Methods are defined as key:value pairs the same as the object’s properties, with a variable name for the key and the value is the function definition. The methods can be called the same as any of the objects properties to perform the function’s task, but they can only be called by the object in which they are defined.
* A **for in loop** is set up to loop over each of the items defined in an object. It does this by looping over each of the key:value pairs. You can loop over all of the object’s key:value pairs without knowing how many key:value pairs the object has. This is handy for getting all of the values that are stored in a particular object.
* **Dot notation vs. bracket notation** is very simply the difference between calling an objects properties and methods using either a dot between the name of the object and the property name, or putting the property name in brackets after the object’s name. Either of these methods works for calling the object’s methods, as well, by using the method name in place of the property name. It is best to use the dot notation method in almost all circumstance, accept for when property or method name is given a separate variable name as a reference. If you are calling it with the variable name instead of the actual property name, then you will want to use bracket notation instead. An example of dot notation to call a method from the car object created above is:

car.model;

If, however you have defined a variable to contain the property name, like as below:

const carType = ‘model’;

Then you would want to use the variable name within brackets to call the objects parameter:

car[carType];