**Q. JavaScript ES 6 Object Oriented Concept…**

Object Orientation is a software development paradigm that follows real-world modeling. Object Orientation, considers a program as a collection of objects that communicates with each other via mechanism called methods.

* **JS classes**

Classes are one of the features introduced in the **ES6** version of JavaScript.

A class is a blueprint for the object. You can create an object from the class.

* **JS Prototypes**

The prototype is an object that is associated with every functions and objects by default in JavaScript, where function's prototype property is accessible and modifiable and object's prototype property (aka attribute) is not visible.

* **Constructors**

A constructor is a special function that creates and initializes an object instance of a class. In JavaScript, a constructor gets called when an object is created using the new keyword. The purpose of a constructor is to create a new object and set values for any existing object properties.

* **Static Methods**

A static method in JavaScript is a method that has a **static** keyword prepended to itself. Such methods cannot be accessed through instantiated objects but could be accessed through the class name.

This is because static methods belong to the class directly. Inheritance even applies to static methods. Also, these methods can be invoked by non-static methods and even constructors.

Static methods are used to create utility functions and create objects that contain default information.

* **OOP Encapsulation**

Encapsulation, one of the principles of Object-Oriented Programming is the key to achieve such goals. By definition Encapsulation in JavaScript is a process of binding the data with the functions which act upon the data. Encapsulation allows us to control and validate the data. In JavaScript, variables resemble data.

* Inheritance

Inheritance enables you to define a class that takes all the functionality from a parent class and allows you to add more.

Using class inheritance, a class can inherit all the methods and properties of another class.

Inheritance is a useful feature that allows code reusability.

* **Polymorphism**

The polymorphism is a core concept of an object-oriented paradigm that provides a way to perform a single action in different forms. It provides an ability to call the same method on different JavaScript objects. As JavaScript is not a type-safe language, we can pass any type of data members with the methods.

* **Abstraction**

In JavaScript, most of the elements are based on “objects,” and it utilizes “Object-Oriented Programming” (OOP) in its own ways.

The concept of Abstraction in JavaScript is to hide the implementation details and highlight an object’s essential features to the users. That’s how embedding Abstraction in a JavaScript program can enhance the readability of the code and avoid duplication.

By providing only important details to the users, it also improves the security of an application.