**Q1: Define Object Oriented Programming Language?**

Object-oriented programming (OOP) is a programming language model in which programs are organized around data, or [objects](https://searchmicroservices.techtarget.com/definition/object), rather than functions and logic. An object can be defined as a data field that has unique attributes and behavior.

**Q2: List down the Benefits of OOP?**

1. **Data Re-usability:**

“Write a once and use multiple time” we can achieve this by using class. Once We Write a class we can bus it number of time by creating the object for class.

1. **Data Redundancy:**

Inheritance is the good feature for data redundancy if you need a same functionality in multiple class you can write a common class for the same functionality and inherit that class to sub class.

1. **Easy Maintenance:**

It is easy to maintain and modify existing code as new objects can be created with small differences to existing ones.

1. **Data hiding:**

Implementation details are hidden from other modules and other modules has a clearly defined interface.

1. **Security**:

Using data hiding and abstraction, we are providing necessary data only it mean we are maintaining security.

**Q3: Differentiate between function and method?**

**Function:** A function is a piece of code that is called by name. It can be passed data to operate on (i.e. the parameters) and can optionally return data (the return value). All data that is passed to a function is explicitly passed.

**Method:**

A method is a piece of code that is called by a name that is associated with an object. In most respects it is identical to a function except for two key differences:

A method is implicitly passed the object on which it was called.

A method is able to operate on data that is contained within the class (remembering that an object is an instance of a class - the class is the definition, the object is an instance of that data).

**Q4: Define the following terms:**

1. Class

2. Object

3. Attribute

4. Behavior

**Class:**

A class is an entity that determines how an object will behave and what the object will contain. In other words, it is a blueprint or a set of instruction to build a specific type of object.

**Object:**

An object is nothing but a self-contained component, which consists of methods and properties to make a particular type of data useful. Object determines the behavior of the class. When you send a message to an object, you are asking the object to invoke or execute one of its methods.

From a programming point of view, an object can be a data structure, a variable or a function. It has a memory location allocated. The object is designed as class hierarchies.

**Attribute:**

In Object-oriented programming (OOP), classes and objects have attributes. Attributes are data stored inside a class or instance and represent the state or quality of the class or instance. In short, attributes store information about the instance. In addition, attributes should not be confused with class functions also known as methods.

**Behavior:**

The behavior of an object is defined by its methods, which are the functions and subroutines defined within the object class. Without class methods, a class would simply be a structure.