Q#1: Define Object Oriented Programming Language?

Ans:

Object-oriented programming (OOP) is a programming language model in which programs are organized around data, or objects, rather than functions and logic. An object can be defined as a data field that has unique attributes and behavior. Examples of an object can range from physical entities, such as a human being that is described by properties like name and address, down to small computer programs, such as widgets. This opposes the historical approach to programming where emphasis was placed on how the logic was written rather than how to define the data within the logic.

Q#2: List down the Benefits of OOP?

Ans:

1. Modularity for easier troubleshooting
2. Reuse of code through inheritance
3. Flexibility through polymorphism
4. Effective problem solving

Q#3: Differentiate between function and method?

Ans:

Method and a function are the same, with different terms. A method is a procedure or function in object-oriented programming.

A function is a group of reusable code which can be called anywhere in your program. This eliminates the need for writing the same code again and again. It helps programmers in writing modular codes.

Q#4:

Define the following terms:

1. Class

2. Object

3. Attribute

4. Behavior

Ans:

CLASS:

Classes (OOP) In object-oriented programming, a **class** is a blueprint for creating objects (a particular data structure), providing initial values for state (member variables or attributes), and implementations of behavior (member functions or methods). The user-defined objects are created using the class keyword

OBJECT:

objects are the things you think about first in designing a program and they are also the units of code that are eventually derived from the process. In between, each object is made into a generic class of object and even more generic classes are defined so that objects can share models and reuse the class definitions in their code. Each object is an instance of a particular class or subclass with the class's own methods or procedures and data variables.

ATTRIBUTE:

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. Also, attributes should not be confused with class functions also known as methods.

BEHAVIOR:

A class's behavior determines how an instance of that class operates; for example, how it will "react" if asked to do something by another class or object or if its internal state changes. Behavior is the only way objects can do anything to themselves or have anything done to them.