

Assignment # 6

Question 1:

Define Object Oriented Programming Language?

Answer 01:

Object-oriented programming (OOP) refers to a type of computer programming (software design) in which programmers define the data type of a data structure, and also the types of operations (functions) that can be applied to the data structure.

In this way, the data structure becomes an object that includes both data and functions. In addition, programmers can create relationships between one object and another. For example, objects can inherit characteristics from other objects.

The Basic OOP Concepts:

1-Abstraction: The process of picking out (abstracting) common features of objects and procedures.

2-Class: A category of objects. The class defines all the common properties of the different objects that belong to it.

3-Encapsulation: The process of combining elements to create a new entity. A procedure is a type of encapsulation because it combines a series of computer instructions.

4-Information hiding: The process of hiding details of an object or function. Information hiding is a powerful programming technique because it reduces complexity.

5-Inheritance: a feature that represents the "is a" relationship between different classes.

6-Interface: the languages and codes that the applications use to communicate with each other and with the hardware.

7-Messaging: Message passing is a form of communication used in parallel programming and object-oriented programming.

8-Object: a self-contained entity that consists of both data and procedures to manipulate the data.

9-Polymorphism: A programming language's ability to process objects differently depending on their data type or class.

10-Procedure: a section of a program that performs a specific task.

Question 2:

List down the Benefits of OOP?

Answer 02:

- Modularity for easier troubleshooting. Something has gone wrong, and you have no idea where to look
- Reuse of code through inheritance.
- Flexibility through polymorphism.
- Effective problem solving.

Question 3:

Differentiate between function and method?

Answer 03:

S.No	Functions	Methods
1	Functions do not have any reference variables	Methods are called by reference variables
2	All data that is passed to a function is explicitly passed	It is implicitly passed the object for which it was called
3	It does not have access controlling i.e.,Function(other than static functions) declares and defines anywhere in the code	It has access controlling i.e.,Method should declare and define in the class only
4	Function applies to both object oriented and non-object oriented language(procedural language.eg. C, Scripting language eg; JavaScript etc)	Method is only applicable to object oriented programming language like C++, C#, Java etc

Question 4:

Define the following terms:

1. Class
2. Object
3. Attribute
4. Behavior

Answer 04:

Class: A category of objects. The class defines all the common properties of the different objects that belong to it.

Object: a self-contained entity that consists of both data and procedures to manipulate the data.

Attribute: belong to the class itself they will be shared by all the instances. Such attributes are defined in the class body parts usually at the top, for legibility.

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. For example, to go back to the theoretical Motorcycle class, here are some behaviors that the Motorcycle class might have: Start the engine Stop the engine Speed up Change gear Stall

Question 5:

Write a code in python in which create a class named it Car which have 5 attributes such like (model, color and name etc.) and 3 methods. And create 5 object instance from that class.

Answer 05:

```
class car(object):  
    def __init__(self,model,color,company,speed_limit,name):  
        self.color=color  
        self.model=model  
        self.company=company  
        self.speed_limit=speed_limit  
        self.name=name  
    def start(self):  
        print("Started")  
    def stop(self):  
        print("stopped")  
    def accelerate(self):  
        print("accelarating")  
car_b= car("Civic","Blue","Honda",220,"AudiQ5",)
```