

Assignment -

(1)

Explain Inheritance and polymorphism from OOPS concept with example.

Inheritance and polymorphism are two fundamental concepts in Object Oriented programming (OOPS). Both concepts facilitate code reuse and enhance the flexibility and extensibility of software. To understand these concepts from a loops perspective, let's use the analogy to explain each one:

* Inheritance :

Inheritance is a concept of OOP where one class can inherit properties and behaviour from another class. The subclass can extend the functionality of the superclass by adding new features (or) overriding existing ones.

* Loop Analogy:

Thinking of inheritance as a looping mechanism where you start with a basic loop and then use it as

a template to create additional specialized loops.

Example:

The "shape" class is the basic class and the "Rectangle" and "Circle" classes are derived classes. The "shape" classes provide a generic method "area" and each derived class implements its own version of the "area" method.

Example:

The "shape" class, The "rectangle" and "circle" classes reuse the common behaviour (The "area" method) while adding their specific implements.

* Polymorphism:

Polymorphism is the ability of objects take on multiple forms

of the content of OOP, it allows different classes to have methods with same name, but the

behaviour can vary depending on the actual object type at runtime.

Loop Analogy:
Think of polymorphism as a loop mechanism that iterates over a list of different objects, treating each other object uniformly even though they belong to different classes.

The "print-area" function takes a "shape" object as an argument. It doesn't know about the specific type of shape, it only knows that all shapes have "area" method. This is polymorphism in action - treating objects.