Day 10: Inheritance in Python

Topics Covered:

- *Introduction to Inheritance*
- *Types of Inheritance*
- *Inheritance in Python*
- *Benefits of Inheritance*
- *Python Implementation of Inheritance*

1. Introduction to Inheritance:

Inheritance is one of the core concepts of object-oriented programming (OOP). It allows one class (child class) to inherit the properties and methods from another class (parent class). This enables code reuse and creates a hierarchical relationship between classes.

2. Types of Inheritance:

There are several types of inheritance in Python, including:

- *Single Inheritance*
- *Multiple Inheritance*
- *Multilevel Inheritance*
- *Hierarchical Inheritance*
- *Hybrid Inheritance*

3. Single Inheritance:

In Single Inheritance, a child class inherits the properties and methods of a single parent class.

Example:

```
class Animal:
    def speak(self):
        print("Animal speaks")

class Dog(Animal):
    def bark(self):
        print("Dog barks")

dog = Dog()
dog.speak()  # Inherited method
dog.bark()  # Child class method
```

4. Multiple Inheritance:

In Multiple Inheritance, a child class inherits properties and methods from more than one parent class.

Example:

```
class Animal:
    def speak(self):
        print("Animal speaks")

class Dog:
    def bark(self):
        print("Dog barks")

class Puppy(Dog, Animal):
    def play(self):
        print("Puppy plays")

puppy = Puppy()
puppy.speak()  # Inherited method from Animal
puppy.bark()  # Inherited method from Dog
puppy.play()  # Method of Puppy
```

5. Multilevel Inheritance:

In Multilevel Inheritance, a child class inherits from another child class, forming a chain of inheritance.

Example:

```
class Animal:
    def speak(self):
        print("Animal speaks")

class Dog(Animal):
    def bark(self):
        print("Dog barks")

class Puppy(Dog):
    def play(self):
        print("Puppy plays")

puppy = Puppy()
puppy.speak()  # Inherited method from Animal
puppy.bark()  # Inherited method from Dog
puppy.play()  # Method of Puppy
```

6. Benefits of Inheritance:

Inheritance provides several advantages, such as:

- *Code Reusability: You can reuse methods and properties from parent classes in child classes.*

- *Method Overriding: Child classes can override methods from parent classes.*
- *Code Organization: It helps in organizing and maintaining a clean, modular code structure.*

Summary of Day 10:

Today, we explored the concept of Inheritance in Python. We learned how inheritance helps in reusing code and organizing classes into a hierarchical structure. We examined different types of inheritance, including Single Inheritance, Multiple Inheritance, and Multilevel Inheritance, with practical examples for each type.