

## Lab Exercise-3

### Mobile and Application Development Lab

- 1. Problem Statement:** You are required to create an interface named Animal that contains the following method signatures:

- void makeSound();
- String getType();

Next, implement this interface in two classes: Dog and Cat. Each class should provide its own implementation of the makeSound() method. The Dog class should output "Woof!" when makeSound() is called, while the Cat class should output "Meow!". The getType() method should return the type of animal as a string ("Dog" for the Dog class and "Cat" for the Cat class).

Finally, write a main method that creates an array of Animal references, populates it with instances of both Dog and Cat, and iterates through the array to call each animal's methods, printing their sounds and types.

#### Requirements:

- a) Define the Animal interface.
  - b) Implement the Dog and Cat classes.
  - c) In the main method:
    - Create an array of type Animal.
    - Add instances of Dog and Cat to the array.
    - Loop through the array and invoke both methods (makeSound() and getType()) for each animal.
- 2. Problem Statement:** Java does not support multiple inheritance with classes due to the potential for ambiguity, such as the diamond problem. However, you can achieve similar functionality using interfaces. Your task is to implement multiple inheritance in Java using interfaces. Follow these steps:

**Define Interfaces:** Create two interfaces, Flyable and Swimmable. Each interface should declare one method:

- Flyable: void fly();
- Swimmable: void swim();

**Implement Interfaces in a Class:** Create a class named Duck that implements both Flyable and Swimmable. Provide implementations for both methods:

The fly() method should print "Duck is flying!".

The swim() method should print "Duck is swimming!".

**Demonstrate Functionality:** In the main method, create an instance of the Duck class and call both methods to demonstrate that the duck can both fly and swim.

**Requirements:**

- Define the interfaces with their respective methods.
- Implement the methods in the Duck class.
- Write a main method to create an object of Duck and call its methods.