IT22007

Abstraction

```
interface Sound{
    void makeSound();
}
abstract class Animal{
    String name;
    public Animal(String name){
        this.name = name;
    }
    abstract void move();
    public void showName(){
      System.out.println("Animal Name: "+name);
    }
}

class Dog extends Animal implements Sound{
   public Dog(String name){
      super(name);
   }
   @Override
   public void makeSound(){
     System.out.println("Dog says: Woof!");
   }
```

```
  @Override
  void move(){
    System.out.println("Dog runs.");
  }
}
class Cat extends Animal implements Sound{
    public cat(String name){
      super(name);
    }
    @Override
    public void makeSound(){
      System.out.println("Cat says: Meow!");
    }
    @Override
    void move(){
      System.out.println("Cat walks gracefully.");
    }
}
public class Abstruction{
  public static void main(String[] args){
    Animal dog = new Dog("Buggy");
    dog.showName();
    ((Sound)dog).makeSound();
    dog.move();
```

```
        System.out.println();
        Animal cat = new Cat("Whiskers");
        cat.showName();
        ((Sound) cat).makeSound();
        cat.move();
    }
}
```