INHERITANCE

COURSE TEACHER : MS. LAMEYA ISLAM

COURSE TITLE : OBJECT ORIENTED PROGRAMMING

COURSE CODE : CSE-214

SECTION : 61_X

DEPARTMENT OF CSE

WELCOME TO OUR

PRESENTATION

TEAM MEMBERS

AMIT KUMAR GHOSH

ID : 221-15-4650

MEHEDI HASAN

ID : 221-15-4680

TABLE OF CONTENT

- Introduction
- Types of inheritance
- Subclass
- Override
- Super keyword

In this presentation, we will explore the basics of inheritance in Java, including how to create subclasses, override superclass methods, and use inheritance to create more efficient and flexible code.

INTRODUCTION

Inheritance in one of the Four important pilar of Object Orientated Programming. which allows a class to inherit the properties and behaviors of another class. Inheritance promotes code reuse, make code more modular, extensible, and maintainable.

TYPES OF INHERITANCE

1. **Single Inheritance:**

In single inheritance, a subclass inherits from only one superclass.

2. **Multi-level Inheritance:**

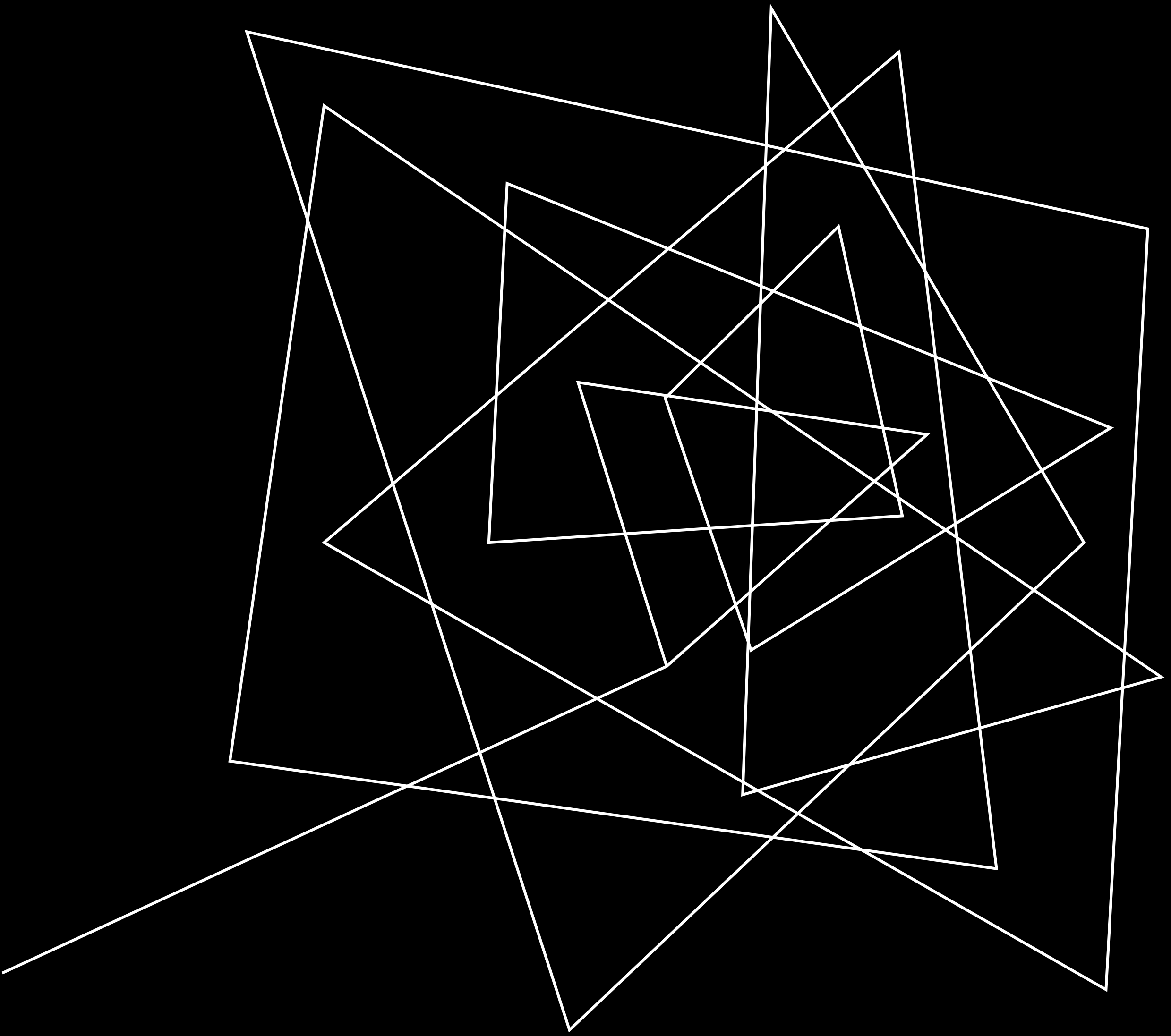
In multilevel inheritance, a subclass inherits from a superclass, which in turn inherits from another superclass, and so on.

3. **Hierarchical Inheritance:**

In hierarchical inheritance, multiple subclasses inherit from a single superclass.

4. **Multiple Inheritance:**

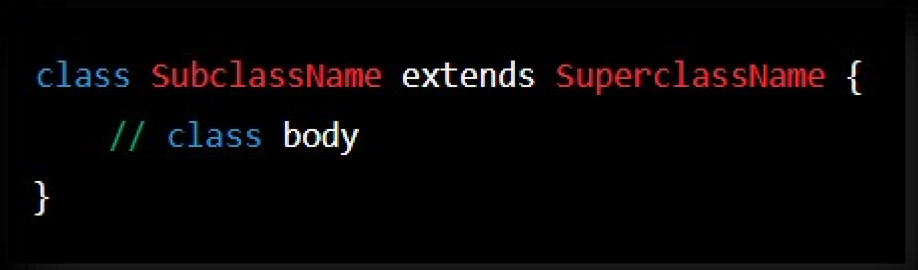
Multiple inheritance allows a subclass to inherit from multiple superclasses. However, Java does not support multiple inheritance of classes, only interfaces.



SUBCLASS

In Java, a Class inherit properties and behavior of parent class when we use ‘**extends’** keyword, followed by the name of the superclass.

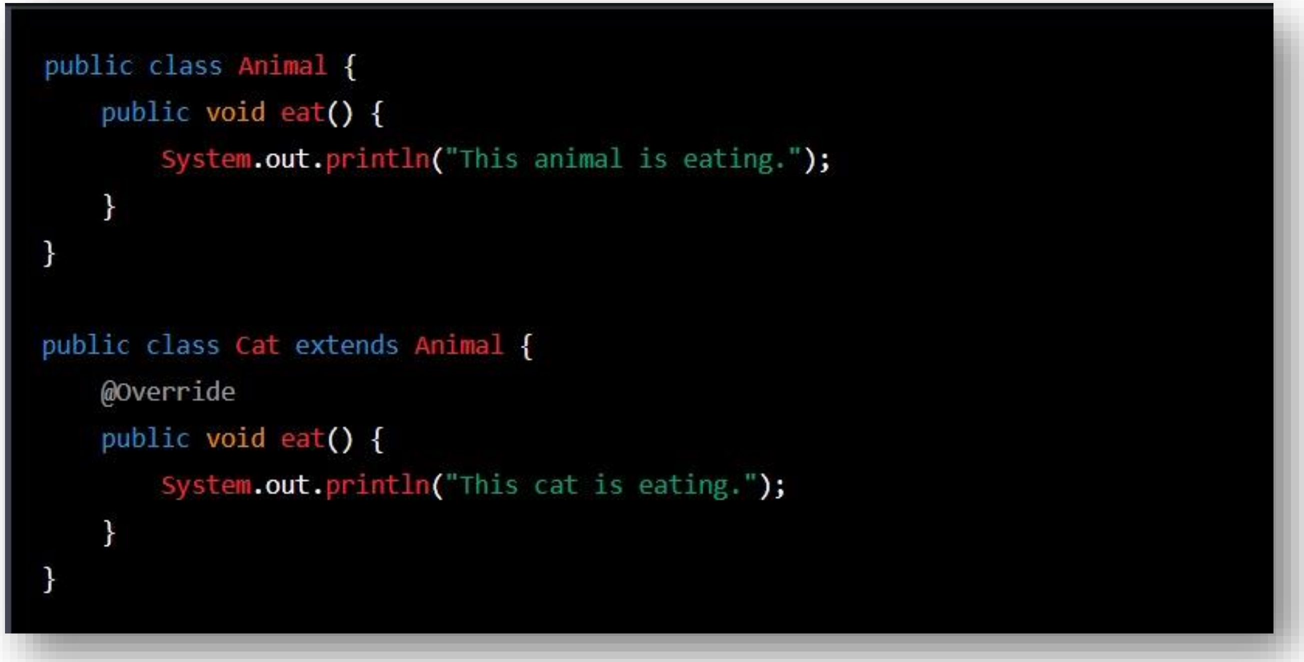
The syntax for creating a subclass is as follows:



```
class SubclassName extends SuperclassName {
    // class body
}
```

OVERRIDE

To override a method in the superclass, the subclass needs to provide a method with the same name, return type, and parameters, and use the ‘**@Override** ‘ annotation to indicate that it is intended to override the superclass method.



```
public class Animal {
    public void eat() {
        System.out.println("This animal is eating.");
    }
}

public class Cat extends Animal {
    @Override
    public void eat() {
        System.out.println("This cat is eating.");
    }
}
```

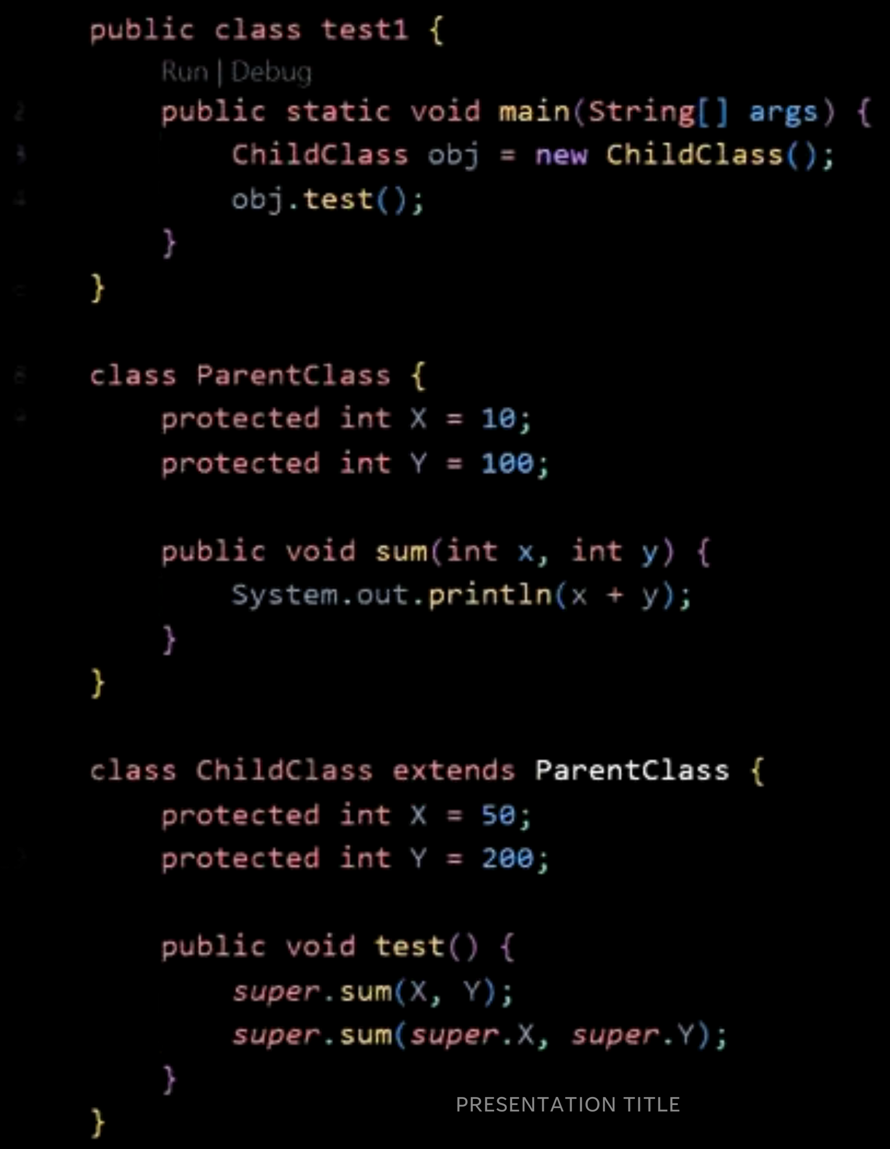
SUPER KEYWORD

As the name suggest super keyword used to access the member of parent class.

It is used for two purposes in java

1. The first use of super keyword is to access the hidden data variables of parent class from child class.
2. The second use of super keyword is to call parent class constructor or method.

EXAMPLE OF SUPER KEYWORD



```
public class test1 {
    Run | Debug
    public static void main(String[] args) {
        ChildClass obj = new ChildClass();
        obj.test();
    }
}

class ParentClass {
    protected int X = 10;
    protected int Y = 100;

    public void sum(int x, int y) {
        System.out.println(x + y);
    }
}

class ChildClass extends ParentClass {
    protected int X = 50;
    protected int Y = 200;

    public void test() {
        super.sum(X, Y);
        super.sum(super.X, super.Y);
    }
}
```

Output:

```
250
110
```

THANK YOU

Reference :

- https://www.slideshare.net/Tech_MX/inheritance-in-java
- https://www.educba.com/object-oriented-programming-in-java/