

## TITLE: TO ILLUSTRATE THE CONCEPT OF INHERITANCE

### OBJECTIVES:-

- Familiarization with the inheritance with the different keywords
- Understanding the usage of super keywords in different forms.

### Introduction:

Inheritance is one of the key features of OOP that allows us to create a new class from an existing class.

Inheritance is a concept that explains how one class gains the attributes of another, including the inherited class's methods and variables.

### Terms:

#### 1. Classes:

- a. Base class/Superclass/Parent class-class from where subclass inherits the features.
- b. Derived class/Sub class/Extended class/Child class-class which inherits the other class.

#### 2. Super keyword:It refers to an immediate parent class's object.

- a. If you create an instance of the child class, then the super keyword implicitly refers to the parent class instance.
- b. It allows to invoke immediate parent class method.
- c. It allows to invoke parent class constructor.

### Syntax of inheritance:

**Class Subclassname extends Superclassname**

```
{  
  
//methods and fields  
  
}
```

**Extends:** It indicates you are making a new class that derives from an existing class.

### ***WORKING OF INHERITANCE***

Inheritance works by taking existing code and building on top of it, and programmers can define a class to “inherit” existing code from another class by using the keyword “extends” in the class header

1.Create a class named Animal, and make a class for Birds. Since Birds and Animals share many of the same base attributes, you could extend the Animal class and add any additional parts needed to the Bird class.Also, use parameterized constructor in main class Mammal and display the attributes.

```
class Animal {  
  
    String noise;  
  
    int age;  
  
    String name;
```

```
public Animal(String noise, int age, String name){  
    this.noise = noise;  
    this.age = age;  
    this.name = name;  
}
```

```
public String getName() {  
    return name;  
}
```

```
public void speak(){  
    System.out.println("My name is " + name);  
    System.out.println("I am " + age + " years old");  
    System.out.println("I say " + noise);  
}  
}
```

```
class Bird extends Animal {  
    public double height;  
    public String color;
```

```
    public Bird(String noise, int age, String name, double height, String
color){

        super(noise, age, name);

        this.height = height;

        this.color = color;

    }
```

```
    public void printBird(){

        System.out.printf("Hello! I am %s and I am a %s Bird! %s!",
this.getName(), this.color, this.noise);

    }
```

```
    public void speak(){

        super.speak();

        System.out.println("I am " + color);

        System.out.println("I am " + height + " inches tall");

    }

}
```

```
public class Mammal

{
```

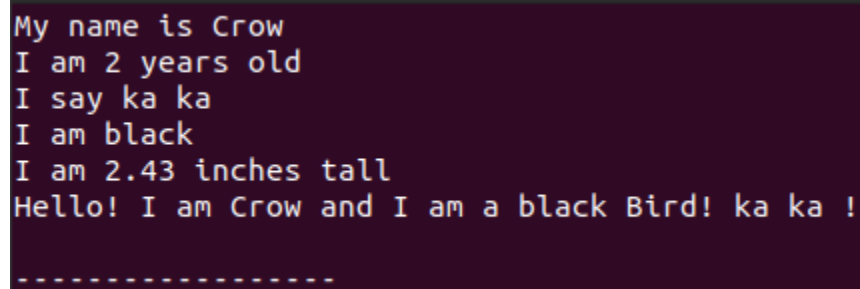
```
public static void main(String[] args)
{
    Bird b=new Bird("ka ka ", 2,"Crow", 2.43,"black");

    b.printBird();

    b.speak();

}
}
```

Output:

A screenshot of a terminal window with a dark purple background. The output of the Java program is displayed in a light-colored monospace font. The text consists of six lines: 'My name is Crow', 'I am 2 years old', 'I say ka ka', 'I am black', 'I am 2.43 inches tall', and 'Hello! I am Crow and I am a black Bird! ka ka !'. A dashed line is visible at the bottom of the output.

```
My name is Crow
I am 2 years old
I say ka ka
I am black
I am 2.43 inches tall
Hello! I am Crow and I am a black Bird! ka ka !
-----
```

2.Create a class Circle and using inheritance concept derive another class Cylinder from it. Then, using main class Check, access the member variables.

```
import java.lang.Math.*;
```

```
class Circle
```

```

{
    int radius;

    double area(int r) {
        return Math.PI*r*r;
    }
}

class Cylinder extends Circle {
    Cylinder() {

    }

    int height;

    double volume(int r, int h)
    {
        return (Math.PI*r*r*h);
    }
}

public class Check {
    public static void main(String[] args)
    {
        Cylinder c=new Cylinder();
        Circle cr= new Circle();
    }
}

```

```
System.out.println("circle area is:" +cr.area(5));
```

```
System.out.println("cylinder volume is:" + c.volume(2,3));
```

```
}
```

```
}
```

3. Create a class Rectangle and using inheritance concept, create another class cuboid . Make one main class named MainShape and access the variables through that class.

**Usage of super keyword:**

4. Write a program to show base class and derived class and access the variables and methods through the derived class.

```
class Parent
```

```
{
```

```
    int a=10;
```

```
    void show()
```

```
    {
```

```
        System.out.println("Parent class");
```

```
    }
```

```
}
```

```
class Child extends Parent
```

```
{
```

```
    int b=20;
```

```
    void show()
```

```
    {
```

```
        System.out.println("Derived class");
```

```
        System.out.println(super.a);
```

```
        super.show();
```

```
    }
```

```
}
```

```
public class Inherit
```

```
{
```

```
    public static void main(String[] args)
```

```
    {
```

```
        Child c=new Child();
```

```
        System.out.println(c.b);
```

```
        c.show();
```

```
        System.out.println(c.a);
```

```
    }
```



}

Output:

```
20
Derived class
10
Parent class
10
```

5. Create a class Teacher with attributes as name, schoolname , age . Derive a class ComputerTeacher with unique attributes as id and access the school name from the superclass. Also, display the attributes of superclass through sub class .

6. WAP to illustrate the concept of super () to access the superclass constructor (class )

