Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Experiment No. 7
Implement a program on single inheritance.
Date of Performance:
Date of Submission:

Aim: To implement the concept of single inheritance.

Objective: Ability to design a base and child class relationship to increase reusability.

Theory:

Single inheritance can be defined as a derived class to inherit the basic methods (data members and variables) and behaviour from a superclass. It's a basic is-a relationship concept exists here. Basically, java only uses a single inheritance as a subclass cannot extend more superclass.

Inheritance is the basic properties of object-oriented programming. Inheritance tends to make use of the properties of a class object into another object. Java uses inheritance for the purpose of code-reusability to reduce time by then enhancing reliability and to achieve run time polymorphism. As the codes are reused it makes less development cost and maintenance. Java has different types of inheritance namely single inheritance, multilevel, multiple, hybrid. In this article, we shall go through on basic understanding of single inheritance concept briefly in java with a programming example. Here we shall have a complete implementation in java.

Syntax:

The general syntax for this is given below. The inheritance concepts use the keyword 'extend' to inherit a specific class. Here you will learn how to make use of extending keyword to derive a class. An extend keyword is declared after the class name followed by another class name. Syntax is,

```
class base class
{…. methods
}
class derived class name extends base class
{
methods … along with this additional feature
}
```

Java uses a keyword 'extends' to make a new class that is derived from the existing class. The inherited class is termed as a base class or superclass, and the newly created class is called derived or subclass. The class which gives data members and methods known as the base class and the class which takes the methods is known as child class.

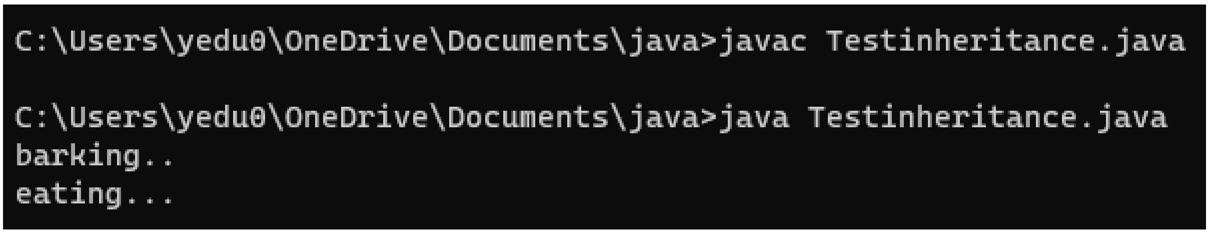
Code:

```
class Singleinheritance {
   public static void main(String args[]) {
      Dog d = new Dog();
      d.bark();
      d.eat();
   }
}

class Animal {
   void eat() {
      System.out.println("eating");
   }
}

class Dog extends Animal {
   void bark() {
      System.out.println("barking");
   }
}
```

OUTPUT :



```
C:\Users\yedu0\OneDrive\Documents\java>javac Testinheritance.java

C:\Users\yedu0\OneDrive\Documents\java>java Testinheritance.java
barking..
eating...
```

Conclusion:

Comment on the Single inheritance.

In Java, single inheritance refers to the concept where a class can inherit properties and behavior from only one superclass. This means that a Java class can have only one direct superclass, also known as a single parent class. Single inheritance is a fundamental aspect of Java's object-oriented programming model, and it helps in maintaining a simple and straightforward class hierarchy. Here are some key points regarding single inheritance in Java:

Class Hierarchy: With single inheritance, a class can have only one immediate superclass. However, this superclass can have its own superclass, creating a hierarchical tree-like structure.

Superclass-Subclass Relationship: Single inheritance establishes a relationship where a subclass inherits the properties and behavior of its immediate superclass. This allows the subclass to extend the functionality of the superclass while maintaining a hierarchical relationship.

Java Interface Implementation: Java supports multiple interface implementation, allowing a class to implement multiple interfaces. This feature helps in achieving a form of multiple inheritance through interfaces, where a class can inherit the capabilities of multiple interfaces.