INHERITANCE IN JAVA

Dr. Nancy Arya
Faculty of Engineering and Technology
SGT University

CONTENT

- Introduction to Inheritance
- Types of inheritance
- Define sub class
- Single Inheritance
- Super keyword
- Multilevel Inheritance
- Method Overriding
- Abstract class
- Final keyword

INTRODUCTION TO INHERITANCE

- Reusability is yet another factor of OOP's. Java classes are used in several ways. This is basically by creating new classes ,reusing the properties existing one.
- The deriving the new class from an old one is called inheritance.
- The old class is known as base class or super class.
- The class who inherit the properties of the base class is called derived class or sub class.

TYPES OF INHERITANCE

- Single inheritance(one super class)
- Multiple inheritance(several super classes)
- Multilevel inheritance(
- Hierarchical inheritance(one super class many sub class)

DEFINE SUBCLASS

```
class sub-classname extends super-classname
{
   variable declaration;// sub class variables
   method declaration;// sub class method
   }
```

The keyword extends signifies that the properties of the super class are extended to the sub class.

EXAMPLE OF SINGLE INHERITANCE

```
class sum                    // Super class
{
    int a=10;
    int b=20;
    void show1()
    {
            System.out.println("value of a :- " +a);
            System.out.println("value of b :- " +b);
    }
}
class sum2 extends sum              // base class
{
    public static void main(String args[])
    {
            sum2 obj = new sum2();
            obj.show1();
}
}
```

SUB CLASS CONSTRUCTOR

- A subclass constructor is used to construct the instance variable of both the subclass and super class.
- The subclass constructor uses the keyword super to invoke the constructor method of the super class.

CONDITION FOR SUPER KEYWORD

- Super may only be used within a subclass constructor.
- The call to super class constructor must appear as the first statement with in the subclass constructor.
- The parameter in the super call must match the order and type of the instance variable declared in the super class.

USES OF SUPER KEYWORD

- It calls the super class constructor.

SYNTAX:- super(parameter list);

- Access the member of the super class.

SYNTAX:- super. member variable;

USING SUPER TO CALL SUPER CLASS CONSTRUCTOR

```
class demo
{    int x,y;
     public demo()
{    x=10;
     y=20;}
public demo(int i,int j)
{    x=i;
     y=j;       }}
class demo1 extends demo{
int j;
public demo1(){
super(10,15);
j=30;}
void show(){
System.out.println(x+ " " +y + " "  +j);}
public static void main(String args[]){
demo1 d = new demo1();
d.show();}}
```

MULTILEVEL INHERITANCE

When a subclass derived from a super class and a subclass is further derived from that subclass or when a subclass acts as a super class for some other subclass, it creates a multilevel hierarchy.

METHOD OVERRIDING

- In a class hierarchy , when a method in a sub class has the same name and type signature as a method in its super class, then the method is said to override the method in the super class.
- When an overridden method is called from within a sub class, it will always refer to the version of that method defined by the subclass. The version of the method defined by the super class will be hidden.

EXAMPLE OF METHOD OVERRIDDING

```
class sum                       // Super class
{
      int a=10;
      int b=20;
      void show()
      {
                  System.out.println("value of a :- " +a);
                  System.out.println("value of b :- " +b);
      }}
class sum2 extends sum    // base class
{     int i=30,j=40;
      void show()
      {           System.out.println(i+ " " +j);// only this value will be print
      }
      public static void main(String args[])
      {
                  sum2 obj = new sum2();
                  obj.show();}}
```

THANK YOU