

Course Name - Object Oriented Programming using Java

Lecture 24 – Multiple inheritance by using Interface

Presented By
Dr. Sudipta Sahana
Associate Professor
Dept. of CSE
UEM - Kolkata

Topic of Interest

- ▶ Concept of Multiple Inheritance
- ▶ Incapability of Java to Support Multiple Inheritance
- ▶ Multiple Inheritance Achieved using Interface
- ▶ Example

Concept of Multiple Inheritance

- ❑ Multiple Inheritance is a feature of object oriented concept, where a class can inherit properties of more than one parent class.
- ❑ The problem occurs when there exist methods with same signature in both the super classes and subclass.
- ❑ On calling the method, the compiler cannot determine which class method to be called and even on calling which class method gets the priority.

Incapability of Java to Support Multiple Inheritance

Below is an example to show the problem:

```
class Parent1 {  
    void fun() {  
        System.out.println("Parent1");  
    }  
}  
class Parent2 {  
    void fun()  
    {  
        System.out.println("Parent2");  
    }  
}  
class Test extends Parent1, Parent2 {  
    public static void main(String args[]) {  
        Test t = new Test();  
        t.fun();  
    }  
}
```

Output:

Compiler Error

Incapability of Java to Support Multiple Inheritance

The Diamond Problem: The below Java program throws compiler error when run. Multiple inheritance causes diamond problem when allowed in other languages like C++.

```
class GrandParent {  
    void fun() {  
        System.out.println("Grandparent");  
    }  
}  
class Parent1 extends GrandParent {  
    void fun() {  
        System.out.println("Parent1");  
    }  
}  
class Parent2 extends GrandParent {  
    void fun() {  
        System.out.println("Parent2");  
    }  
}  
class Test extends Parent1, Parent2 {  
    public static void main(String args[]) {  
        Test t = new Test();  
        t.fun();  
    }  
}
```

Output: Error : prog.java:31: error: '{'
expected class Test extends Parent1, Parent2
^
1 error

Incapability of Java to Support Multiple Inheritance

- Simplicity – Multiple inheritance is not supported by Java using classes , handling the complexity that causes due to multiple inheritance is very complex.
- It creates problem during various operations like casting, constructor chaining etc and the above reason is that there are very few scenarios on which we actually need multiple inheritance, so better to omit it for keeping the things simple and straightforward.

Multiple Inertance Achieved using Interface

- With the help of interface we can solve this problem.
- Java supports default methods where interfaces can provide default implementation of methods. And a class can implement two or more interfaces.
- In case both the implemented interfaces contain default methods with same method signature, the implementing class should explicitly specify which default method is to be used or it should override the default method.

Example

Code:

```
interface Printdocument {  
    void print();  
}  
interface Showdocument {  
    void print();  
}  
class Test implements Printdocument, Showdocument  
{  
    public void print() {  
        System.out.println("Hello");  
    }  
    public static void main(String args[]) {  
        Test obj = new Test();  
        obj.print();  
    }  
}
```

Output: Hello

Extending Multiple Interfaces

A Java class can only extend one parent class. Multiple inheritance is not allowed. Interfaces are not classes, however, and an interface can extend more than one parent interface.

The extends keyword is used once, and the parent interfaces are declared in a comma-separated list.

For example, if the Hockey interface extended both Sports and Event, it would be declared as:

```
public interface Hockey extends Sports, Event
```

*Thank
You*