Abstract method in Java with examples

BY CHAITANYA SINGH | FILED UNDER: OOPS CONCEPT

A method without body (no implementation) is known as abstract method. A method must always be declared in an abstract class, or in other words you can say that if a class has an abstract method, it should be declared abstract as well. In the last tutorial we discussed Abstract class, if you have not yet checked it out read it here: Abstract class in Java, before reading this guide. This is how an abstract method looks in java:

```
public abstract int myMethod(int n1, int n2);
```

As you see this has no body.

Rules of Abstract Method

1. Abstract methods don't have body, they just have method signature as shown above.
2. If a class has an abstract method it should be declared abstract, the vice versa is not true, which means an abstract class doesn't need to have an abstract method compulsory.
3. If a regular class extends an abstract class, then the class must have to implement all the abstract methods of abstract parent class or it has to be declared abstract as well.

Example 1: abstract method in an abstract class

```
//abstract class
abstract class Sum{
   /* These two are abstract methods, the child class
    * must implement these methods
    */
   public abstract int sumOfTwo(int n1, int n2);
   public abstract int sumOfThree(int n1, int n2, int n3);

   //Regular method
   public void disp(){
        System.out.println("Method of class Sum");
   }
}
//Regular class extends abstract class
class Demo extends Sum{

   /* If I don't provide the implementation of these two methods, the
    * program will throw compilation error.
    */
   public int sumOfTwo(int num1, int num2){
        return num1+num2;
   }
```

```
   public int sumOfThree(int num1, int num2, int num3){
        return num1+num2+num3;
   }
   public static void main(String args[]){
        Sum obj = new Demo();
        System.out.println(obj.sumOfTwo(3, 7));
        System.out.println(obj.sumOfThree(4, 3, 19));
        obj.disp();
   }
}
```

Output:

```
10
26
Method of class Sum
```

Example 2: abstract method in interface

All the methods of an interface are public abstract by default. You cannot have concrete (regular methods with body) methods in an interface.

```
//Interface
interface Multiply{
   //abstract methods
   public abstract int multiplyTwo(int n1, int n2);

   /* We need not to mention public and abstract in interface
    * as all the methods in interface are
    * public and abstract by default so the compiler will
    * treat this as
    * public abstract multiplyThree(int n1, int n2, int n3);
    */
   int multiplyThree(int n1, int n2, int n3);

   /* Regular (or concrete) methods are not allowed in an interface
    * so if I uncomment this method, you will get compilation error
    * public void disp(){
    *    System.out.println("I will give error if u uncomment me");
    * }
    */
}

class Demo implements Multiply{
   public int multiplyTwo(int num1, int num2){
      return num1*num2;
   }
   public int multiplyThree(int num1, int num2, int num3){
      return num1*num2*num3;
   }
   public static void main(String args[]){
      Multiply obj = new Demo();
      System.out.println(obj.multiplyTwo(3, 7));
      System.out.println(obj.multiplyThree(1, 9, 0));
   }
}
```

Output:

```
21
0
```