Interface

Interface looks like class but it is not a class. An interface can have methods and variables just like the class but the methods declared in interface are by default abstract (only method signature, no body). Also, the variables declared in an interface are public, static & final by default.

What is the use of interfaces?

As mentioned above they are used for abstraction. Since methods in interfaces do not have body, they have to be implemented by the class before you can access them. The class that implements interface must implement all the methods of that interface. Also, java programming language does not support multiple inheritance, using interfaces we can achieve this as a class can implement more than one interfaces, however it cannot extend more than one classes

Declaration
Interfaces are declared by specifying a keyword "interface". E.g.:

```
interface MyInterface
{
  /* All the methods are public abstract by default
   * Note down that these methods are not having body
   */
  public void method1();
  public void method2();
}
```

Benefits of having interfaces:

Following are the advantages of using interfaces:

1. Without bothering about the implementation part, we can achieve the security of implementation
2. In java, **multiple inheritance** is not allowed, However by using interfaces you can achieve the same . A class can extend only one class but can implement any number of interfaces. It saves you from Deadly Diamond of Death(DDD) problem.

Example program

```
package basics;

//Area Of Rectangle and Triangle using Interface

interface Area
{
```

```
float compute(float x, float y);
}

class Rectang implements Area
{
public float compute(float x, float y)
{
return(x * y);
}
}

class Triang implements Area
{
public float compute(float x,float y)
{
return(x * y/2);
}
}

class Multinheritance
{
public static void main(String args[])
{
Rectang rect = new Rectang();
Triang tri = new Triang();
Area area;
area = rect;
System.out.println("Area Of Rectangle = "+
area.compute(1,2));

area = tri;
System.out.println("Area Of Triangle = "+
area.compute(10,2));
}
}
/* * OUTPUT **

Area Of Rectangle = 2.0
Area Of Triangle = 10.0
*/
```