INTERFACES

➢ What are Interfaces?

- Inheritance is used for borrowing methods.
- Abstract is used for achieving **polymorphism** as well as **Inheritance.**
- Inheritance is completely used for achieving **Polymorphism.**
- Interface can be call as Abstract Class with all abstract methods.
- All the methods are by default abstract.
- Classes are extended but Interfaces are implemented.
- In Interface we can have reference of interface and the object of the class which is implemented.
- In java a class can extend from one class only but if a class is implementing an interface then it can implement from multiple interfaces.

Example Program

```
interface test1

{
  void meth1();
  void meth2();
}
class test2 implements test1
{
  public void meth1()
  public void meth2()
}
class test
{
 public static void main(0
 {
  test1 t=new test2
 }
}
```

➢ Do's and Don'ts of Interfaces

- By default, methods are Public and Abstract.

- As methods are to be implemented by the classes, they can't be made private.
- Identifiers can be used in interfaces but the identifiers must be given in Upper cases.
- Identifiers are by default final and static.
- Method inside an interface cannot have body but the method can have body if the method is static.
- Static members can be accessed in main method by using interface name and dot operator.
- An interface can be extended from another interface.

➢ Interface VS Multiple Inheritance

- In C++ one class can inherit from multiple classes.
- Multiple Inheritance in java is achieved using Interfaces.
- Interfaces are perfect than using Multiple Inheritance.
- Way of thinking in java is more perfect than C++.