**Interface in java**

* An interface is the blueprint of class.
* It has static constants and abstract methods .
* The interface in java is a mechanism to achieve the abstraction .
* There can be only abstract methods in the java interface not method body.
* It is used to achieve abstraction and multiple inheritance in java .
* Java interface also represented IS – A relationship .

**Introduction**

* “Abstraction” is a process of hiding the implementation details and showing only functionality to the user .
* It shows only essential things to the user and hides the internal details .
* There are two ways to achieve abstraction in java

1. Abstract class (0 to 100 %)
2. Interface(100 %)

**Abstract class and method**

* An abstract class is a class that is declared “abstract ” it may or may not include abstract methods .
* A method which is declared as “abstract” and does not have implementation is known as abstract method .

**Syntax :**

**modifier abstract class className {**

**abstract datatype methodName();**

**}**

**modifier class childClass {**

**datatType methodName(){}**

**}**

**POINTS TO REMEMBER :**

* An abstract class must be declared with an abstract method .
* It can have abstract and non- abstract methods .
* It can not be instantiated .
* It can have constructors and static methods also .
* It can have final methods which will force the subclass not to change body of the method
* Any class which contains an abstract method must also be abstract .

**Example :**

Abstract class animal { //Abstract class declaration

Public abstract void sound(); //Abstract method declaration

}

Public class Dog extends animal{ //Dog extends from animal   
 System.out.println(“woof”);

}

public static void main(String args[]){

animal obj = new Dog();

obj.sound();

}

}

Output : Woof