Department: BCA
Faculty Name: Er. Ashish Kumar
Subject: JAVA (5th sem.)
Topic: Interface in Java **(part I)**

Interface in Java

- An **interface in Java** is a blueprint of a class.
- It has static constants and abstract methods.
- The interface in Java is *a mechanism to achieve 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.
- In other words, we can say that interfaces can have abstract methods and variables. It cannot have a method body.
- Java Interface also **represents the IS-A relationship**.
- It cannot be instantiated just like the abstract class.
- Since Java 8, we can have **default and static methods** in an interface.
- Since Java 9, we can have **private methods** in an interface.

Why use Java interface?

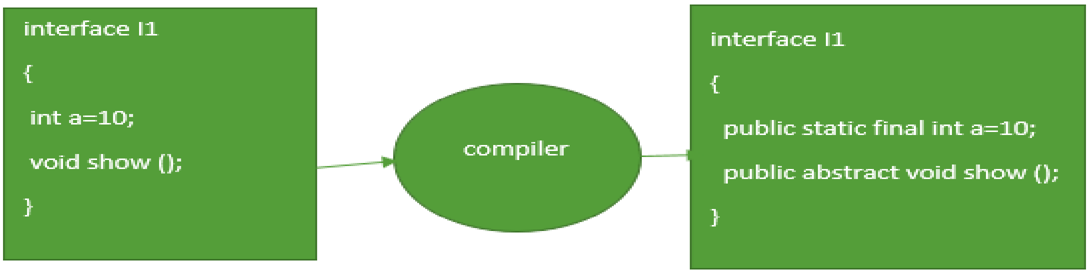
There are mainly three reasons to use interface:

- It is used to achieve abstraction.
- By interface, we can support the functionality of multiple inheritance.
- It can be used to achieve loose coupling.

Syntax:

```
Interface <interface_name>
   {
    // declare constant fields
    // declare methods that abstract
   // by default.

   }
```

Example:



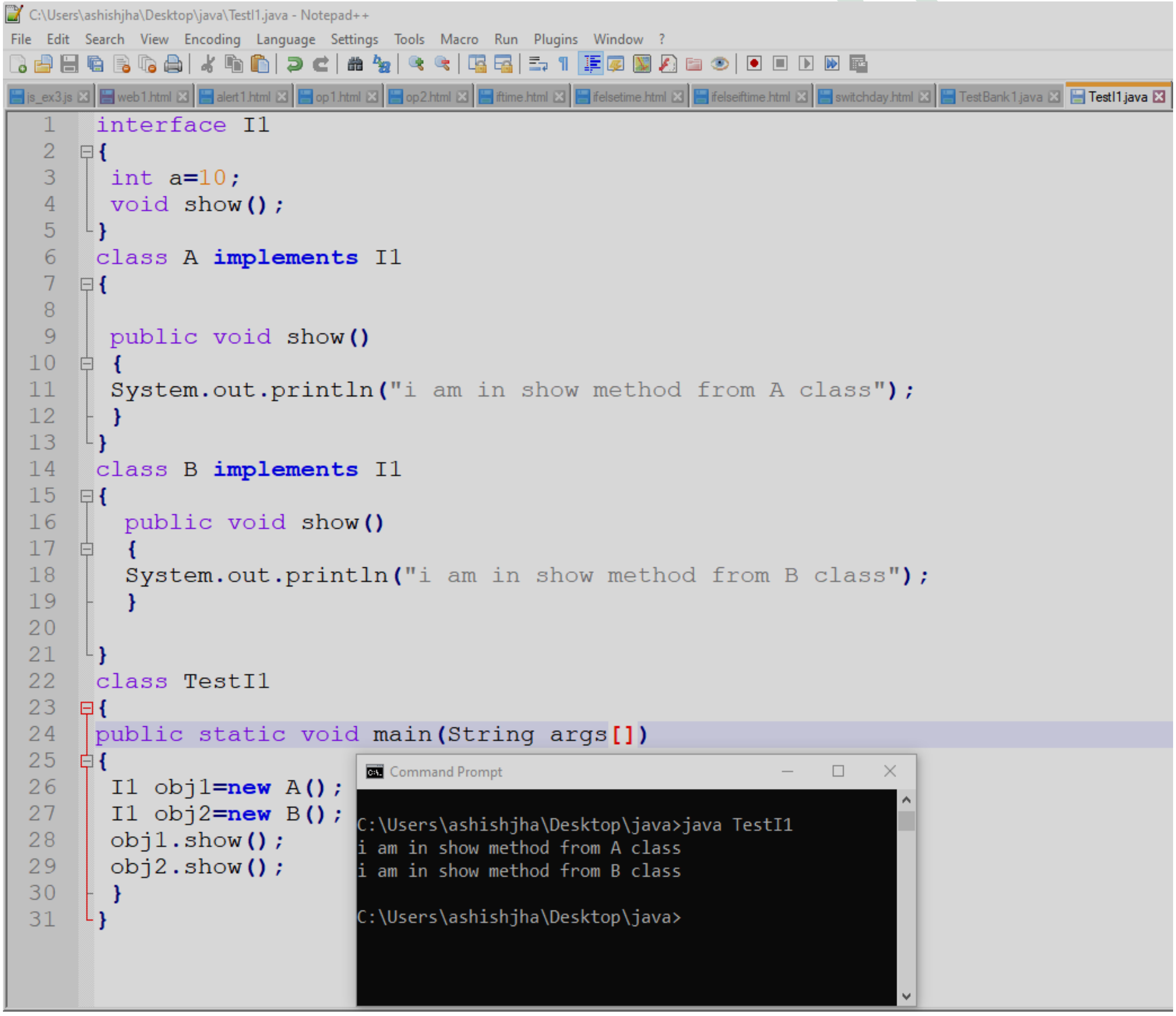
```
interface I1
{
 int a=10;
 void show ();
}
```

```
interface I1
{
 public static final int a=10;
 public abstract void show ();
}
```

I1.java

I1.class

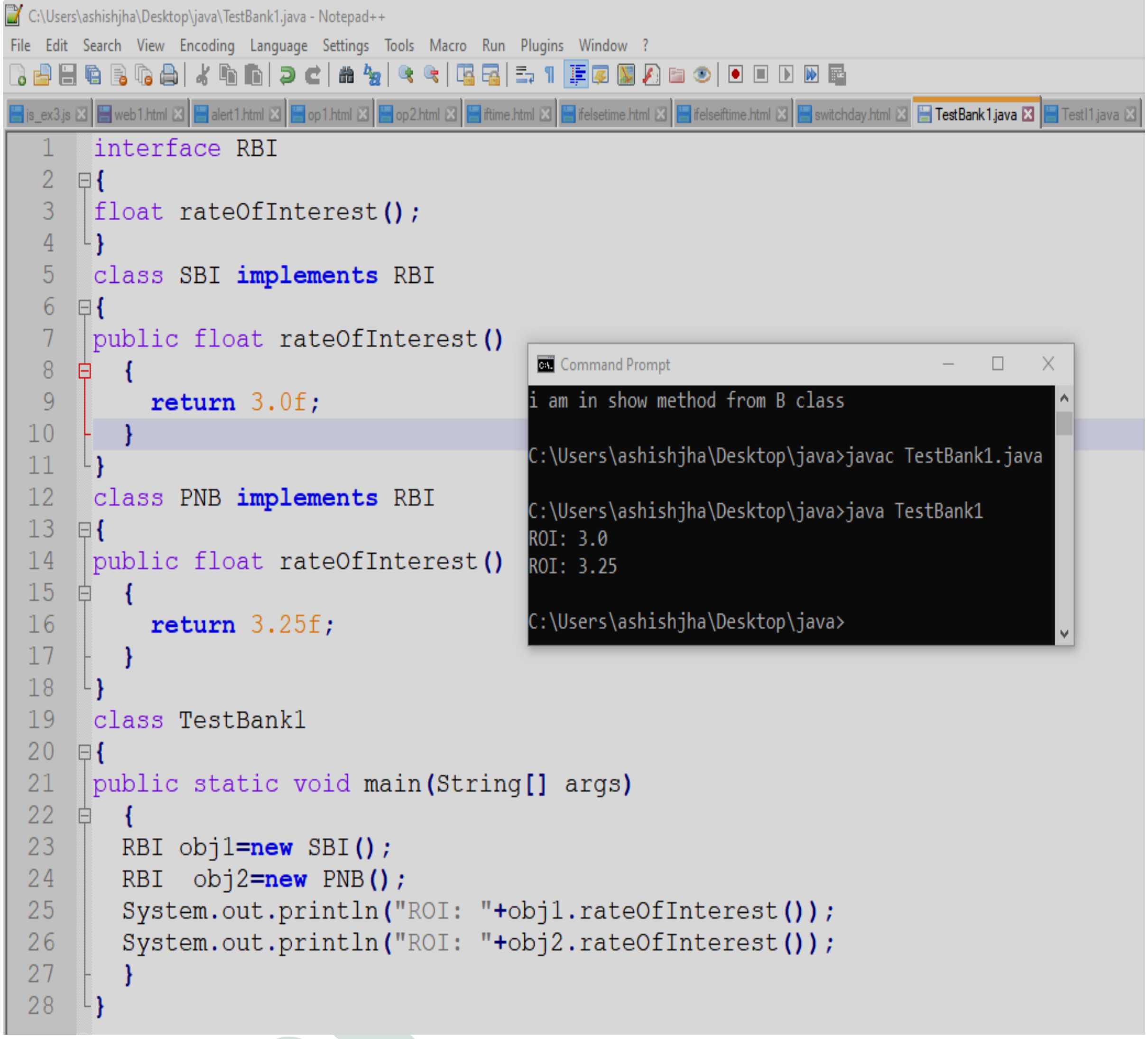
Note: The Java compiler adds public and abstract keywords before the interface method. Moreover, it adds public, static and final keywords before data members.



```
interface I1
{
 int a=10;
 void show();
}
class A implements I1
{

 public void show()
 {
 System.out.println("i am in show method from A class");
 }
}
class B implements I1
{
  public void show()
  {
  System.out.println("i am in show method from B class");
  }

}
class TestI1
{
public static void main(String args[])
{
 I1 obj1=new A();
 I1 obj2=new B();
 obj1.show();
 obj2.show();
 }
}
```

```
C:\Users\ashishjha\Desktop\java>java TestI1
i am in show method from A class
i am in show method from B class

C:\Users\ashishjha\Desktop\java>
```

Example:



```java
interface RBI
{
float rateOfInterest();
}
class SBI implements RBI
{
public float rateOfInterest()
  {
    return 3.0f;
  }
}
class PNB implements RBI
{
public float rateOfInterest()
  {
    return 3.25f;
  }
}
class TestBank1
{
public static void main(String[] args)
  {
  RBI obj1=new SBI();
  RBI  obj2=new PNB();
  System.out.println("ROI: "+obj1.rateOfInterest());
  System.out.println("ROI: "+obj2.rateOfInterest());
  }
}
```

```
i am in show method from B class

C:\Users\ashishjha\Desktop\java>javac TestBank1.java

C:\Users\ashishjha\Desktop\java>java TestBank1
ROI: 3.0
ROI: 3.25

C:\Users\ashishjha\Desktop\java>
```

notes