**Java – Constructor in Interface?**

This is a most frequently asked java interview question. The answer is No, interface cannot have constructors. In this post we will discuss why [**constructors**](http://beginnersbook.com/2013/03/constructors-in-java/) are not allowed in [**interface**](http://beginnersbook.com/2013/05/java-interface/)?

As we know that all the methods in interface are public abstract by default which means the method implementation cannot be provided in the interface itself. It has to be provided by the implementing class. Consider the below piece of code:

For e.g.

public interface SumInterface{

public int mymethod(int num1, int num2);

}

public class SumClass implements SumInterface{

public int mymethod(int num1, int num2){

int op= num1+num2;

return op;

}

public static void main(Sring args[])

{

SumClass obj= new SumClass();

System.out.println(obj.mymethod(2, 3));

}

As you can see we have defined the method body in the class which is implementing our interface. Also, we can call our method using class object we don’t need interface object (object of interface is not allowed).

Lets come to the point now: All the methods of interface doesn’t have body so there is no need to call the methods in the interface itself. In order to call any method we need an object since there is no need to have object of interface, there is no need of having constructor in interface (Constructor is being called during creation of object).