Object-Oriented Programming (CS F213)

Module III: Inheritance and Polymorphism in Java

CS F213 RL 11.2: Wrapper Classes in Java

BITS Pilani

Dr. Pankaj Vyas
Department of Computer Science, BITS-Pilani, Pilani Campus

CS F213 RL 11.2 : Topics

- Wrapper Classes in Java

Wrapper Classes in Java

innovate achieve lead

1. Primitive types aren't Objects in Java.
2. We Can wrap a primitive type value in a wrapper type object value
3. Many Methods in Java's Collections Framework Requires a class type values not primitive types
4. Wrapper class for each type are:

Integer	Short	Long	Byte
Character	Float	Double	Boolean

5. Wrapper Classes are Immutable.
6. Two Concepts Associated with Wrappers : **Auto-boxing and Auto-unboxing**

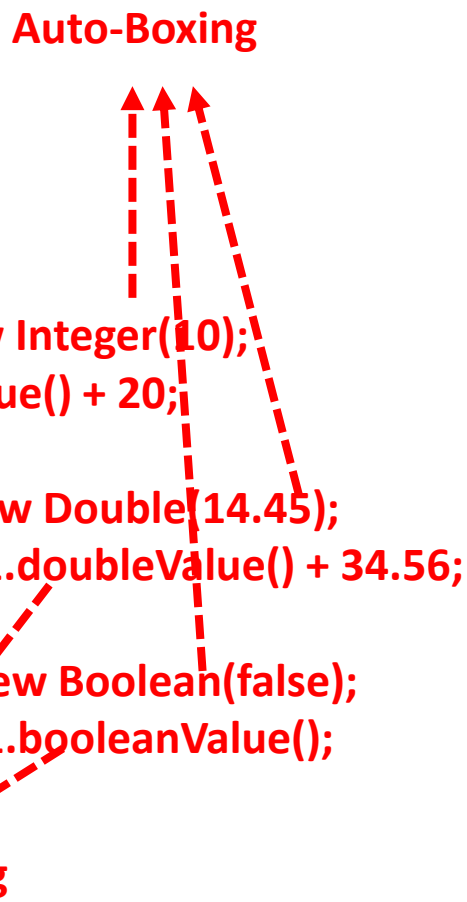
- **Auto-Boxing** : Automatic Conversion of a primitive value to its corresponding Wrapper type Object
- **Auto-Unboxing** : Automatic Retrieving of primitive type value of a Wrapper Type Object

Auto boxing / Auto Unboxing Example 1


```
// File name: Wrapper.java
import java.util.*;
class WrapTest
{
        public    static    void      main(String[] args)
        {
            Integer   a  = 10;          // Integer a = new Integer(10);
            int       x  =  a + 20;     // int  x = a.intValue() + 20;

            Double    b1  = 14.45;      // Double b1 = new Double(14.45);
            double    b2  = b1 + 34.56; // double  b2 = b1.doubleValue() + 34.56;

            Boolean   c1  = false;      // Boolean c1 = new Boolean(false);
            boolean   c2   = c1;        // boolean c2 = c1.booleanValue();
        }// End of Method
}// End of class
```



Auto boxing / Auto Unboxing Example 2


```
import java.util.*;
class Wraptest
{
      public static void main(String args[])
      {
            Vector<Integer> ints = new Vector<Integer>();
            ints.add(10); // ints.add(new Integer(10));
            ints.add(20); // ints.add(new Integer(10));
            ints.add(30); // ints.add(new Integer(10));
            ints.add(40); // ints.add(new Integer(10));
            for(int i=0;i<ints.size();i++)
            System.out.println(ints.get(i));
      }// End of Method
}// End of Class
```

Auto-Boxing

ints.get(i).intValue() → Auto-UnBoxing

Auto boxing / Auto Unboxing Example 3


```
import java.util.*;
class WrapTest
{
        public    static     void main(String args[])
        {
                Integer a = new Integer("10");       // NO ERROR
                Integer b = new Integer("20");       // NO ERROR

                Boolean b10 = new Boolean("true");          // NO ERROR
                Double  d10 = new Double("10.56");          // NO ERROR

                Vector<Integer> vecs = new Vector<Integer>();
                vecs.add(10);      // NO ERROR (Auto-Boxing Works Here)
                vecs.add("20");   // COMPILE-TIME ERROR (No Auto-Boxing here)
                vecs.add(10.45); // COMPILE-TIME ERROR (Wrong Value)
        } End of Method
}// End of class
```

Wrapper Classes: Example 4


```
import java.util.*;
class WrapTest
{
        public    static    void main(String args[])
        {
                int       a = 10;
                Integer   b = 10;
                if(a == b) // if( a == b.intValue())
                          System.out.println("Hello");
                else
                          System.out.println("Hi");
                int       a1 = 400;
                Integer   b1 = 400;
                if(a1 == b1)          // if( a == b.intValue())
                          System.out.println("Hello");
                else
                          System.out.println("Hi");
        } //End of Method
}// End of class
```

Wrapper Classes: Example 4


```
import java.util.*;
class WrapTest
{
        public    static    void main(String args[])
        {
                int       a = 40;
                Integer   b1 = 40;
                Integer   b2 = 40;
                if(a == b1 && a == b2)
                        System.out.println("Hello");
                else
                        System.out.println("Hi");
                if( b1== b2)
                        System.out.println("Hello");
                else
                        System.out.println("Hi");
        } //End of Method
}// End of class
```

Wrapper Classes: Example 4


```
import java.util.*;
class WrapTest
{
        public    static    void main(String args[])
        {
                int       a = 128;
                Integer   b1 = 128
                Integer   b2 = 128;
                if(a == b1 && a == b2)
                        System.out.println("Hello");
                else
                        System.out.println("Hi");
                if( b1== b2)
                        System.out.println("Hello");
                else
                        System.out.println("Hi");
        } //End of Method
}// End of class
```

Thank You