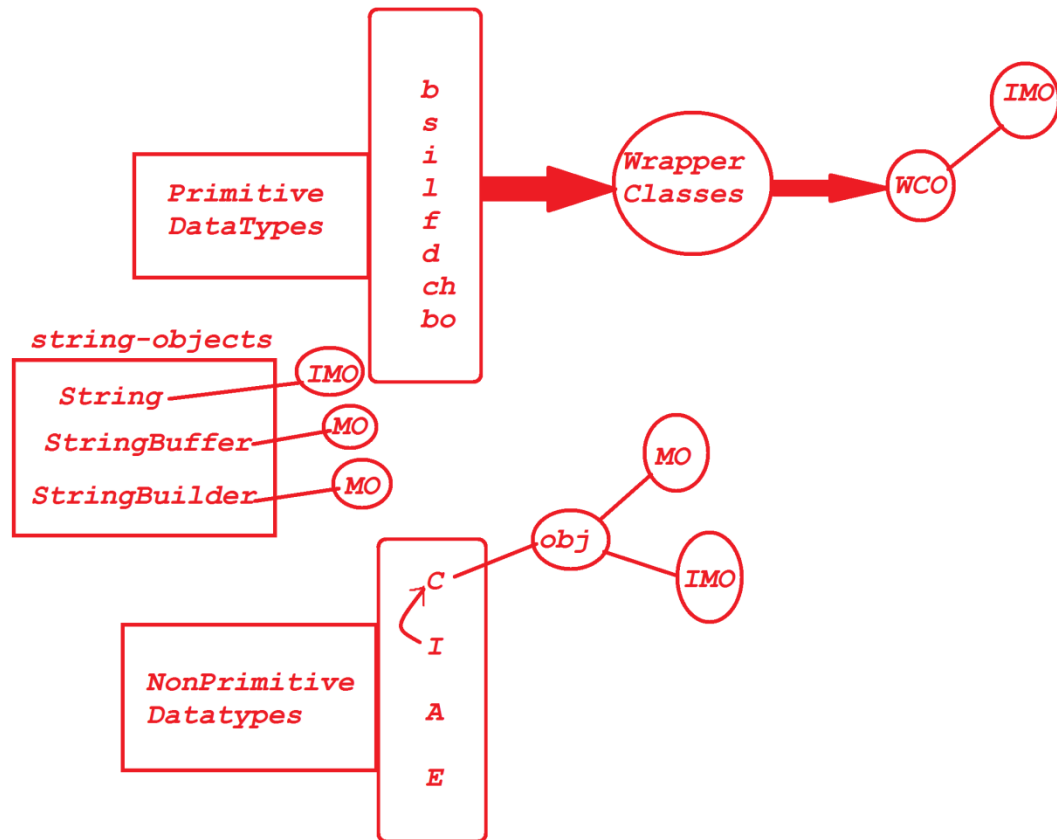


Dt : 28/9/2023



**imp*

Wrapper Classes in Java:

=>The pre-defined classes from java.lang package which are used to make primitive datatypes available in the form of objects are known as WrapperClasses.

=>Every primitive datatype will have its own WrapperClass and there are

eight WrapperClasses.

=>The following are the list of WrapperClasses:

datatype WrapperClass

byte Byte

short Short

int Integer

long Long

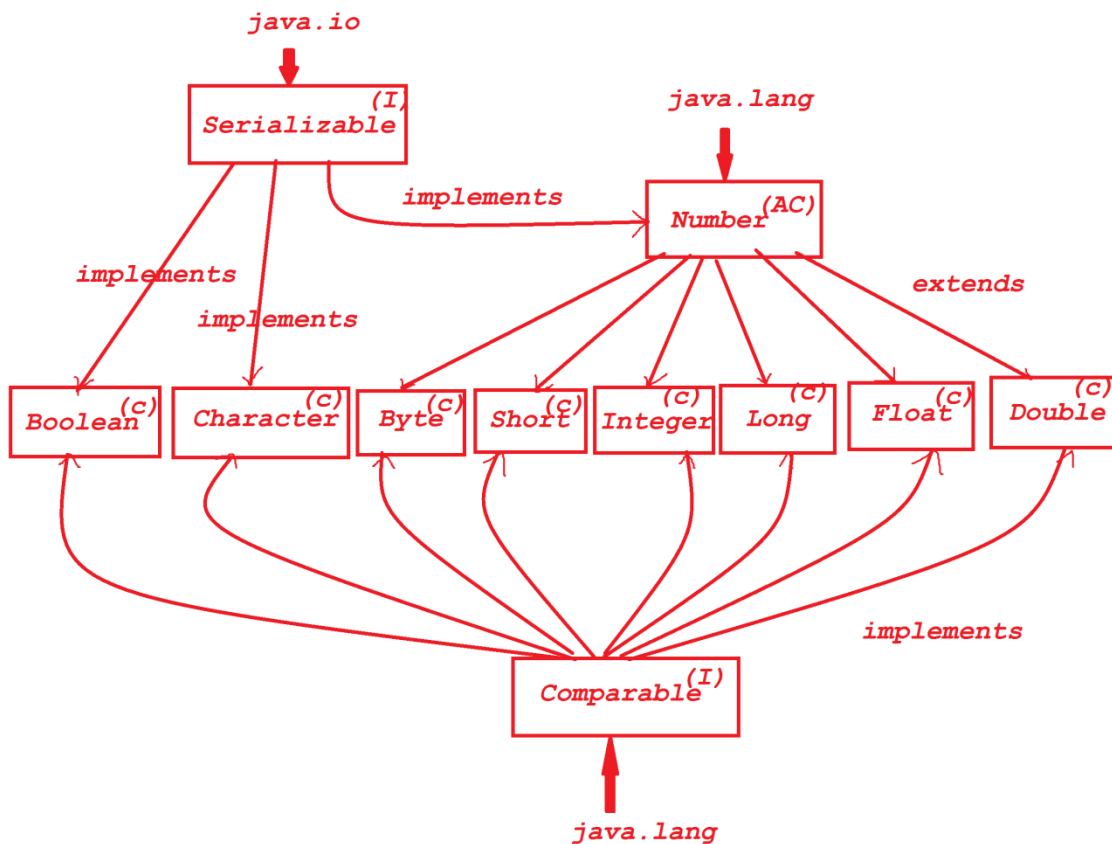
float Float

double Double

char Character

boolean Boolean

Hierarchy of WrapperClasses:



faq:

define Boxing process?

=>The process of making primitive datatypes available in the form of Objects using WrapperClasses is known as Boxing Process.

=>we use Constructors to perform Boxing process.

=>The following are the list of constructors from WrapperClasses:

WrapperClass ListOfConstructors

Byte **Byte(byte),Byte(String)**

Short **Short(short),Short(String)**

Integer **Integer(int),Integer(String)**

Long **Long(long),Long(String)**

Float **Float(float),Float(double),Float(String)**

Double **Double(double),Double(String)**

Character **Character(char)**

Boolean **Boolean(boolean),Boolean(String)**

Ex-program-1 : DemoWrapperClass1.java

```
package maccess;
public class DemoWrapperClass1 {
    @SuppressWarnings("removal")
    public static void main(String[] args) {
        //Boxing Process
        Integer ob1 = new Integer(12);
        Integer ob2 = new Integer("13");
        Float ob3 = new Float(23.45F);
        Float ob4 = new Float(1234.12);
        Float ob5 = new Float("34.12F");
        Character ob6 = new Character('A');
        Boolean ob7 = new Boolean(true);
        Boolean ob8 = new Boolean("false");
        System.out.println("****Display from
Objects****");
        System.out.println("Value in ob1 :
"+ob1.toString());
```

```
        System.out.println("Value in ob2 :  
"+ob2.toString());  
        System.out.println("Value in ob3 :  
"+ob3.toString());  
        System.out.println("Value in ob4 :  
"+ob4.toString());  
        System.out.println("Value in ob5 :  
"+ob5.toString());  
        System.out.println("Value in ob6 :  
"+ob6.toString());  
        System.out.println("Value in ob7 :  
"+ob7.toString());  
        System.out.println("Value in ob8 :  
"+ob8.toString());  
  
    }  
}
```

o/p:

******Display from Objects******

Value in ob1 : 12

Value in ob2 : 13

Value in ob3 : 23.45

Value in ob4 : 1234.12

Value in ob5 : 34.12

Value in ob6 : A

Value in ob7 : true

Value in ob8 : false

faq:

define UnBoxing Process?

=>The process of taking primitive datatype values out of WrapperClass

Objects is known as UnBoxing Process.

=>we use the following methods to perform UnBoxing Process:

public byte byteValue();

public short shortValue();

public int intValue();

public long longValue();

public float floatValue();

public double doubleValue();

public char charValue();

public boolean booleanValue();

Ex-program-2 : DemoWrapperClass2.java

```
package maccess;
public class DemoWrapperClass2 {
    @SuppressWarnings("removal")
    public static void main(String[] args) {
        //Boxing Process
        Integer ob1 = new Integer(12);
        Integer ob2 = new Integer("13");
        Float ob3 = new Float(23.45F);
        Float ob4 = new Float(1234.12);
        Float ob5 = new Float("34.12F");
        Character ob6 = new Character('A');
        Boolean ob7 = new Boolean(true);
    }
}
```

```

        Boolean ob8 = new Boolean("false");
        //UnBoxing process
        int i1 = ob1.intValue();
        int i2 = ob2.intValue();
        float f1 = ob3.floatValue();
        double d = ob4.doubleValue();
        float f2 = ob5.floatValue();
        char ch = ob6.charValue();
        boolean b1 = ob7.booleanValue();
        boolean b2 = ob8.booleanValue();
        System.out.println("====Display values====");
        System.out.println("int value : "+i1);
        System.out.println("int value : "+i2);
        System.out.println("float value : "+f1);
        System.out.println("double value : "+d);
        System.out.println("float value : "+f2);
        System.out.println("char value : "+ch);
        System.out.println("boolean value : "+b1);
        System.out.println("boolean value : "+b2);
    }
}

```

o/p:

====Display values====

int value : 12

int value : 13

float value : 23.45

double value : 1234.1199951171875

float value : 34.12

char value : A

boolean value : true

boolean value : false

faq:

define Auto Boxing?

=>The Boxing process which is performed automatically is known as Auto Boxing Process.

=>In Auto Boxing process,we assign values to NonPrimitive datatype variables

faq:

define Auto UnBoxing Process?

=>The UnBoxing Process which is performed automatically is known as Auto UnBoxing process.

=>In Auto UnBoxing process,the NonPrimitive datatype variables are assigned to Primitive datatype variables.

Ex-program-3 : DemoWrapperClass3.java

```
package maccess;  
public class DemoWrapperClass3 {  
    public static void main(String[] args) {  
        //Auto Boxing Process  
        Integer ob1 = 12;  
        Float ob2 = 23.45F;  
        Character ob3 = 'A';  
        Boolean ob4 = true;  
        //AutoUnBoxing process  
    }  
}
```



```

    int i1 = ob1;
    float f1 = ob2;
    char ch = ob3;
    boolean b1 = ob4;

    System.out.println("====Display values====");
    System.out.println("int value : "+i1);
    System.out.println("float value : "+f1);
    System.out.println("char value : "+ch);
    System.out.println("boolean value : "+b1);
}
}

```

o/p:

====Display values====

int value : 12

float value : 23.45

char value : A

boolean value : true

=====

faq:

why we we have to make primitive datatypes available in the form of Objects?

=>Java Frameworks and Java Tools,will work only with Objects because of this reason Primitive datatypes must be available in the form of Objects.

=====

