Wrapper classes

Wrapper classes is used for perform the conversion between or type casting between primitive data type to object or reference data type and reference data type to primitive data type.

Before Learn the Wrapper classes we need to what is the Type casting or Type Conversion

Type casting means if we change the one type of value in to the type for single line of code called as type casting.

Basically there are two types of casting in java

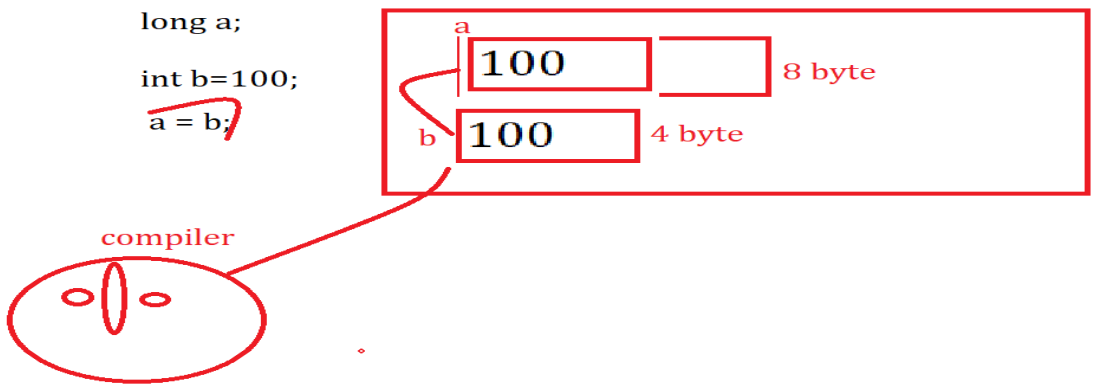
1) Primitive conversion: primitive conversion means if we perform casting between primitive data type called as primitive conversion.

Example: convert in to float or float to int

There Are Two Types of Primitive Conversion

a) Implicit conversion: implicit conversion means when we put the larger value at left hand side and smaller type value at right hand side then compiler is able to perform conversion automatically called as implicit conversion.

Example



Program for Implicit Conversion

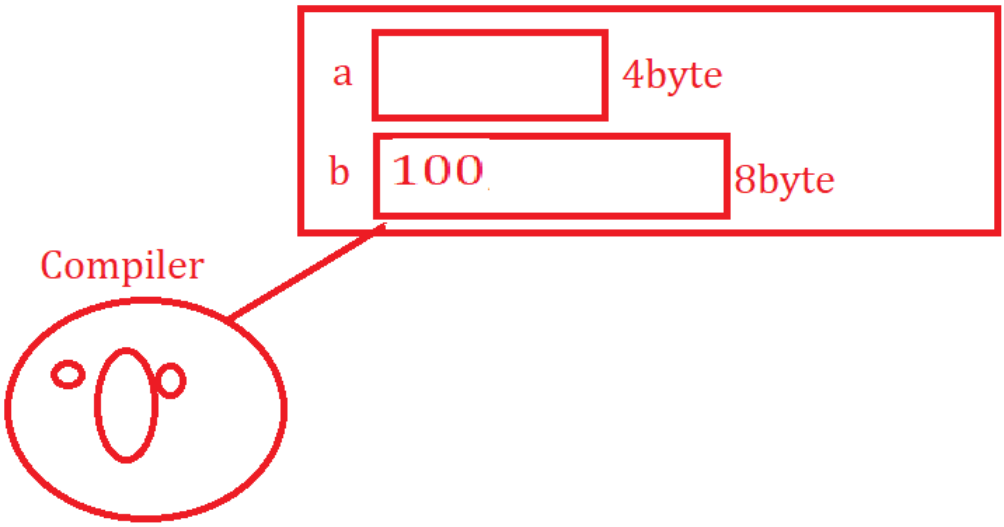
Example

```
package org.techhub;
public class TestConversionApp {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		long a;
		int b = 100;
		a = b; // implicit conversion
		System.out.println("A is " + a);
	}
}
```

b) Explicit conversion : if we put the smaller type value at left hand side and larger type value at right hand side then compiler is unable to perform conversion automatically then programmer need to perform conversion manually called as explicit conversion.

Example

```
int a;

long b=100;

a=b; //here compiler will generate the error
```



So if we want to solve the error mention in above diagram we have to perform conversion manually called as explicit conversion shown in following diagram.

```
int a;

long b=100;

a=(int)b; // explicit conversion

System.out.printf("A is  %d\n",a);
```

Source Code example

```
package org.techhub;
public class TestConversionApp {
	public static void main(String[] args) {
		// TODO Auto-generated method stub
		long a;
		int b = 100;
		a = b; // implicit conversion
		System.out.println("A is " + a);

		int c;
		long d=1000;
		c=(int)d; //expicit conversion
		System.out.printf("C is %d",c);
	}
}
```

Implicit conversion and explicit conversion get failed if we try to convert the primitive type value to object type and object type value to primitive type

Example

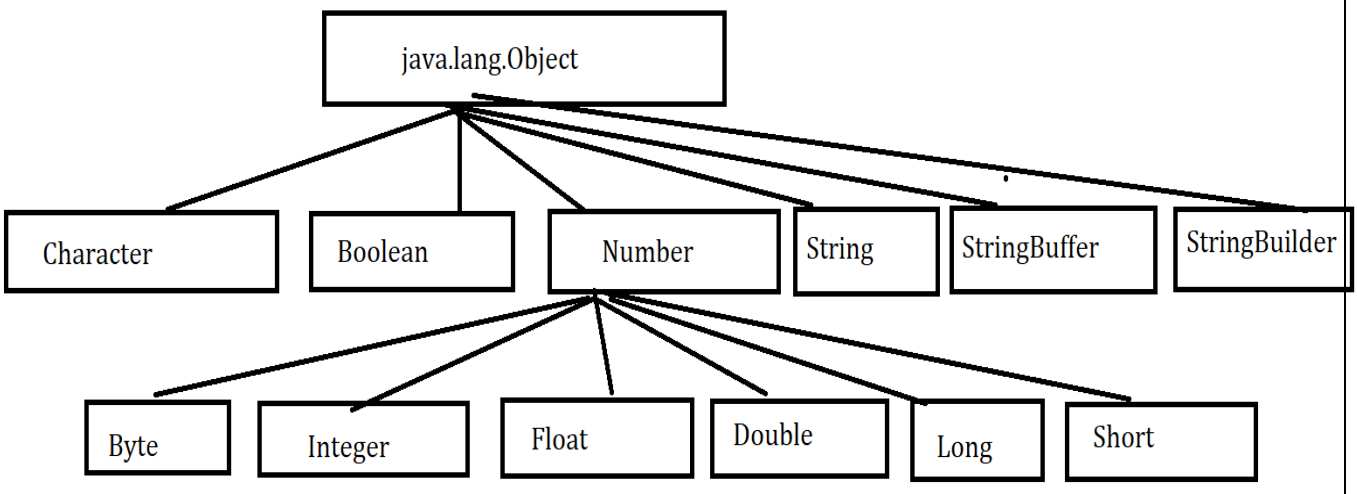
```
public class TestConversionApp {

    public static void main(String[] args) {
        String str="12345";
        int a=(int)str;
        System.out.printf("A is %d\n",a);
    }
}
```

1 it will generate the error to us

2

here String is class means str is reference of of String class
and int a is primitive data type
so we cannot convert the reference value to
primitive value so we cannot perform conversion
by implicit way or explicit way

If we want to solve this type of problem in java.Java Provide the special type of classes to us called as wrapper classes.
If we want to work with wrapper classes in java we have to know the following class hierarchy in java



2) Object conversion or referential conversion: here we perform conversion between two different objects

There Are Two Types of conversion in Wrapper classes

Autoboxing : auto boxing means if we convert any primitive value to object value called as auto boxing.
Autounboxing : auto unboxing means if we convert any object to primitive value directly called as auto unboxing.

Following Examples the autoboxing and autounboxing

```
public class BoxingApplication {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
Integer a; //Integer is class and a is reference here
int b=100; //int is data type so it is primitive type data

	a=b; //autoboxing.
	System.out.printf("A is %d\n",a);

	int c;
	Integer d = 600;
	c=d; //auto unboxing
	System.out.printf("C is %d\n",c);
	 }
}
```

The auto boxing and auto unboxing get failed if we try to convert the different type of primitive with a different type of reference.

```
public class BoxingApplication {

	public static void main(String[] args) {

		String str="123";
		int b=str;
	}

}
```

If we think about code we try to convert the string reference to integer reference using auto unboxing technique.
In this case auto boxing and auto unboxing get failed
So if we perform this type of conversion so java provides the special type of classes to us called as wrapper classes.

Number class

Number class is used for convert the any numeric value to primitive type value and for that Number class provide the xxxValue() method to us for convert the Numeric object to primitive object.

int intValue() : this is used for convert any numeric object to the primitive type.

Example of intValue() method

```
public class BoxingApplication {
```

```
    public static void main(String[] args) {
   Float  a=5.4f;

int b= a.intValue();//convert float object to integer primitive .

            System.out.printf("B is %d\n", b);
    }

}
```

float floatValue(): this is used for convert the any numeric object to floating primitive type.

Example

```
public class BoxingApplication {

    public static void main(String[] args) {

            Long l=100L;
    float b= l.floatValue();//convert Long  object to float primitive .

            System.out.printf("B is %d\n", b);
    }

}
```

double doubleValue(): this is used for convert the any numeric object to double type.

```
public class BoxingApplication {
```

```
	public static void main(String[] args) {

		Long l=100L;

		double b= l.doubleValue();//convert Long  object to
double primitive .

		System.out.printf("B is %d\n", b);
	}

}
```

long longValue(): this is used for convert the any numeric object to long type.

```
public class BoxingApplication {

	public static void main(String[] args) {

		Float f=56.5f;
		long b=f.longValue();//convert Float object to long
primitive .
		 System.out.printf("B is %d\n", b);
	}

}
```

byte byteValue(): this is used for convert the any numeric object to byte type.

etc

```
public class BoxingApplication {

	public static void main(String[] args) {
```

```
        Integer a=5;
    byte b=a.byteValue();//convert Float object to long primitive .
    System.out.println("B is "+ b);
    }

}
```

parseXXX(): this is used for convert the string object to primitive type and it is static method present in every wrapper class.

Example

---

int variable=Integer.parseInt(string): convert string to integer
float variable=Float.parseFloat(String): convert string to float
long variable=Long.parseLong(String): convert string to long
double variable=Double.parseDouble(String):convert string to double.
etc

Example

```
public class BoxingApplication {
  public static void main(String[] args) {
            String str="12345";
            int val =Integer.parseInt (str);
            System.out.printf ("Value is "+val);
           String str1="5.4";
            float value1=Float.parseFloat (str1);
            System.out.printf ("\nValue 1 is %f\n", value1);
    }
}
```

valueOf(): valueOf() method is used for convert the primitive type value to object value and it is also static method present in every wrapper class.

Example

```
package org.techhub;
public class BoxingApplication
{
 public static void main (String[] args) {
int a=12345;
String str=String.valueOf (a); //convert integer primitive to string
object
System.out.printf ("String is %s\n",str);
 float b=5.4f;
 Float d=Float.valueOf (b); //convert float primitive to float object
  System.out.printf ("D is %f\n", d);
 }

}
```

String toString(): this method is used for convert the any object to string.

```
public class BoxingApplication
{
  public static void main(String[] args) {

                Integer a=100;
                String str=a.toString();
                System.out.println("Str is  "+str);
        }
}
```

Note: toString () we will discuss in depth when we learn the Object class.