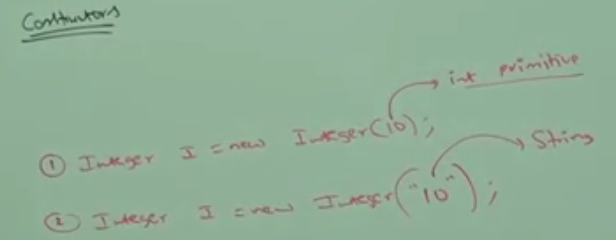
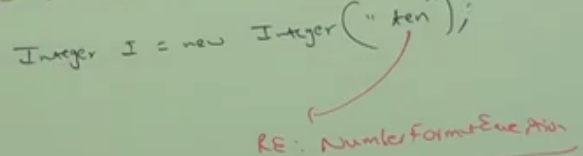
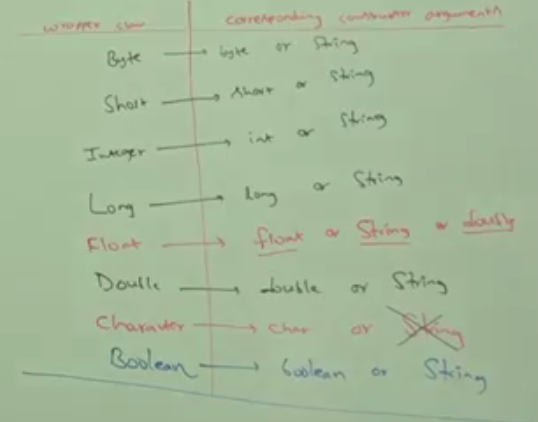
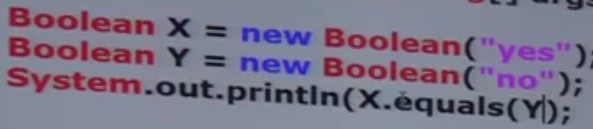
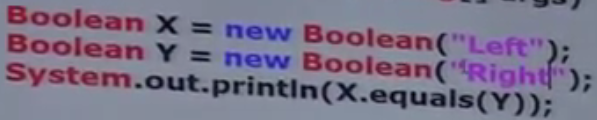
Wrapper Classes

1. What is wrapper?
   1. Cover.
   2. Purpose:
      1. To wrap primitive value so that we can introduce that value to object market.
      2. Scenario: Till 1.4v, We couldn’t add primitive values to collection as collection accept only objects. Then how to add those primitive values into the collection.
      3. The main objective of wrapper classes are
         1. To wrap primitive into object form so that we can handle primitive also just like objects.
         2. To define several utility methods which are required for the primitives such as
            1. to convert primitive to string, double, long etc.
            2. example: Integer.toString(10) : String

wrapperObj.longValue(): long

1. **Constructors**:
   1.   
        
      if a string argument not representing a number, we will get run time exception saying “NumberFormatException”

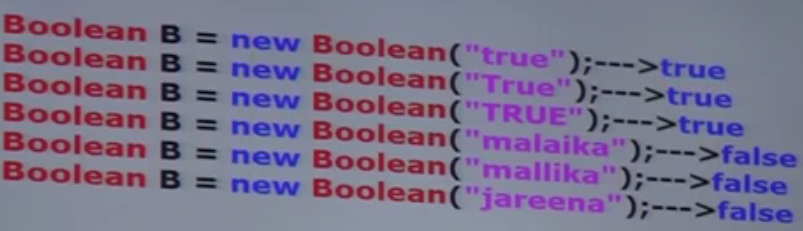
Almost all wrapper classes contain 2 constructors  
one taking corresponding primitive value  
the other one taking corresponding string value as arg

1.   
   NOTE: When passing “true”, “True”, “TRUE” to Boolean(String) is ok ☺  
    🡪 output: true  
    🡪 output: true

Wrapper classes

Constructor arguments

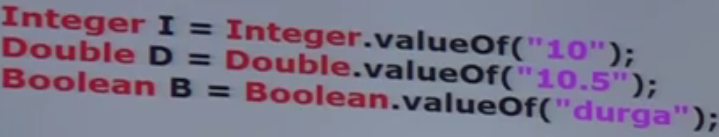
How it is working?

**BIG NOTE:** If we are passing string argument to Boolean(String) then case and content are not important. If content is string “true” ignoring case, then it’s treated as true otherwise false.   


1. NOTE: In all wrapper classes, toString() is overridden to return content directly.
2. NOTE: In all wrapper classes, equals() is overridden for content comparison.
3. **Utility Methods**:
   1. valueOf()
   2. xxxValue()
   3. parseXXX()
   4. toString()

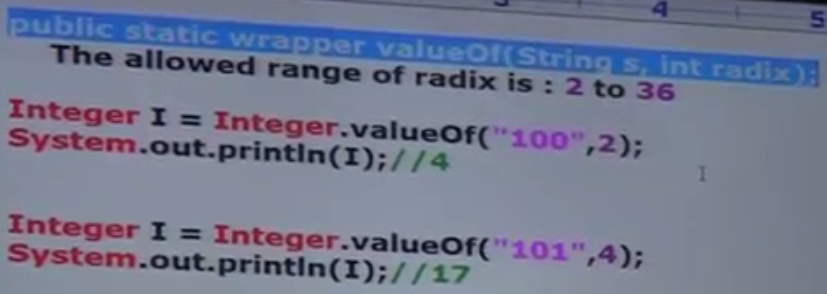
# Form 1 of valueOf()

* valueOf(): To create wrapper object alternative to constructor for the given primitive or string.
  + new Integer.valueOf(10)
  + new Integer.valueOf(“10”);
  + Every wrapper class, except Character class contains a static valueOf(String) to create wrapper object for the given string.  
    public static wrapperClass(String)

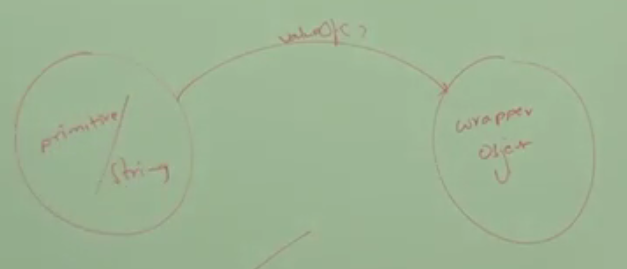


# Form 2:

* valueOf(): Every integral type wrapper class (Byte, Short, Integer, Long) contains the following valueOf() methods to create wrapper object for the given specified radix string

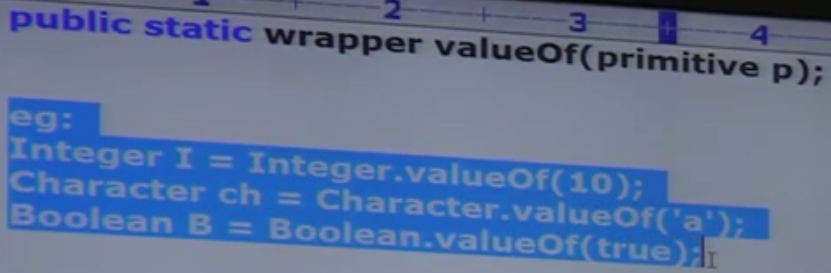


from 2 to 36 because we have 0-9 and a-z altogether 36 symbols   
NOTE: radix says that the 1st arg passed is of this radix such as   
new Integer(“100”, 2); the passed “100” is of base 2 🡪 4

* + 

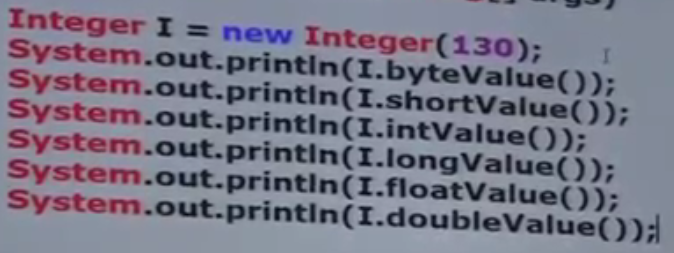
valueOf()

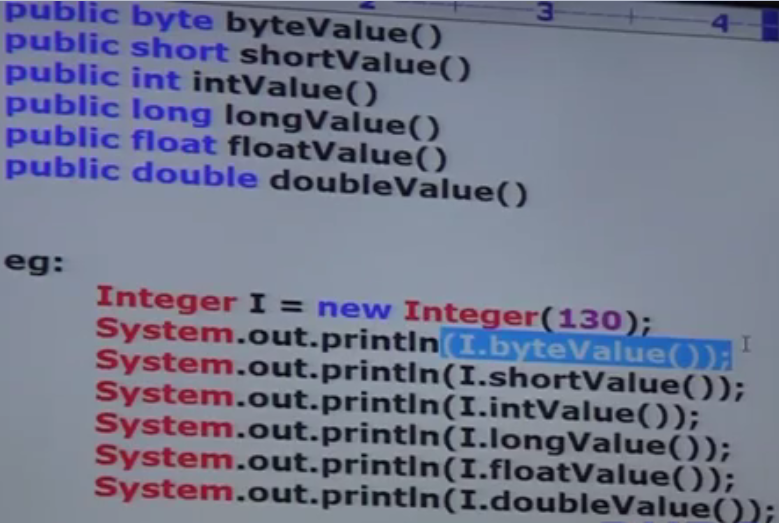
# form 3

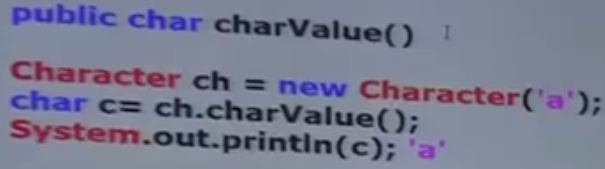
* All wrapper classes including character class contains a static valueOf(primitive) to create wrapper object for the given primitive.
* public static WrapperObject valueOf(primitive p)  
  

1. k

xxxValue()

1. We can use xxxValue() to get primitive for the given wrapper object  
   
2. Every number type wrapper class (Byte, Short, Integer, Long, Float Double) contains the following 6 methods to get the primitive for the given wrapper object



1. charValue(): Character class contains charValue() to get char primitive from the Character Wrapper object.  
   
2. booleanValue(): Boolean class contains booleanValue() to get Boolean primitive for the given Boolean Object.  
   