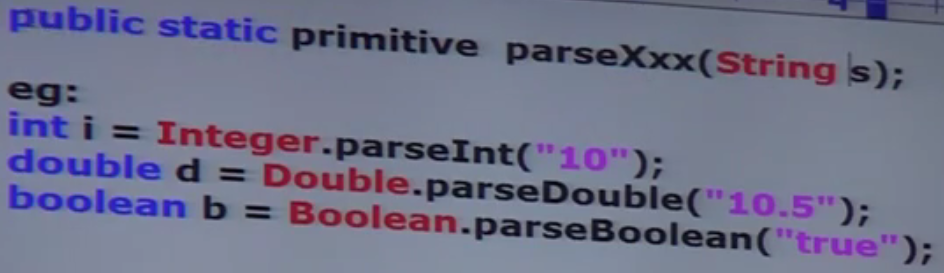
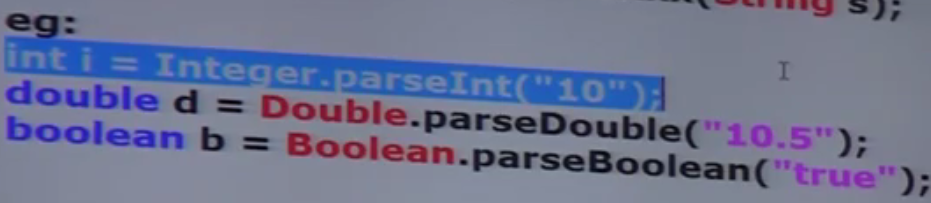


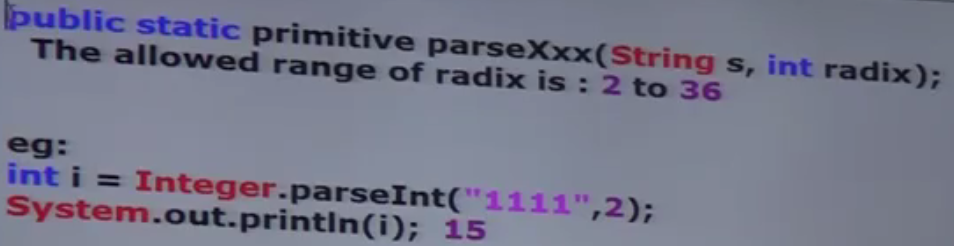
parseXXX()

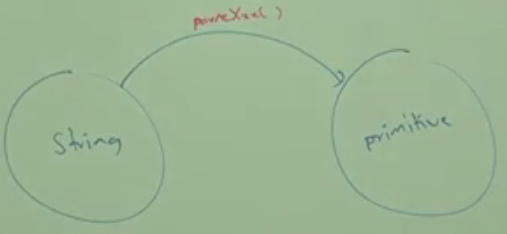
1. We can use parseXXX() to convert **String to primitive**.
2. There are two forms of parseXXX(). Every wrapper class except Character class contains this facility.
   1. Form 01:





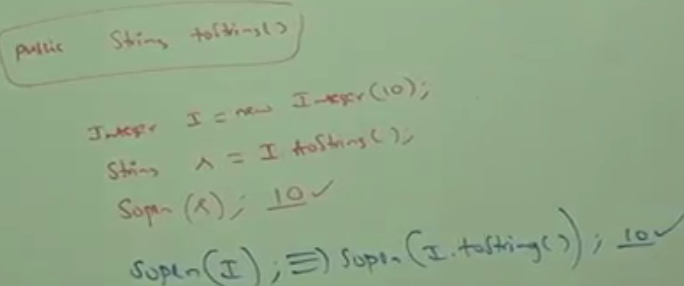
* 2. Form 02:
     1. Every integral type wrapper class (Byte, Short, Integer, Long) contains the following parseXXX() to convert the specified radix string to primitive.

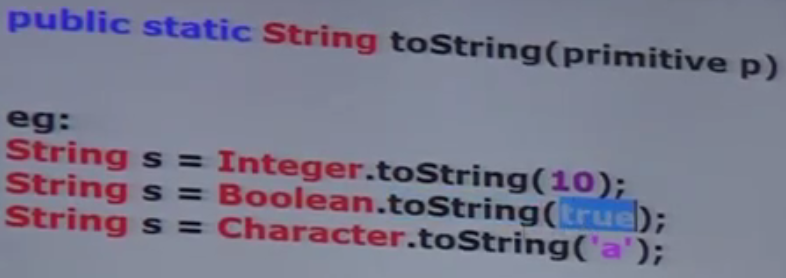


* + 1. **purpose**:  
       

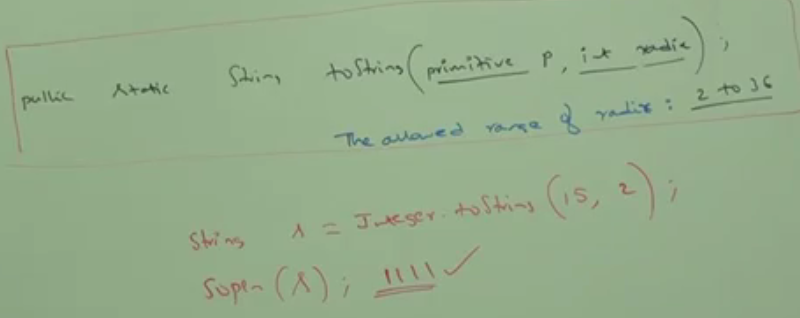
1. d

toString()

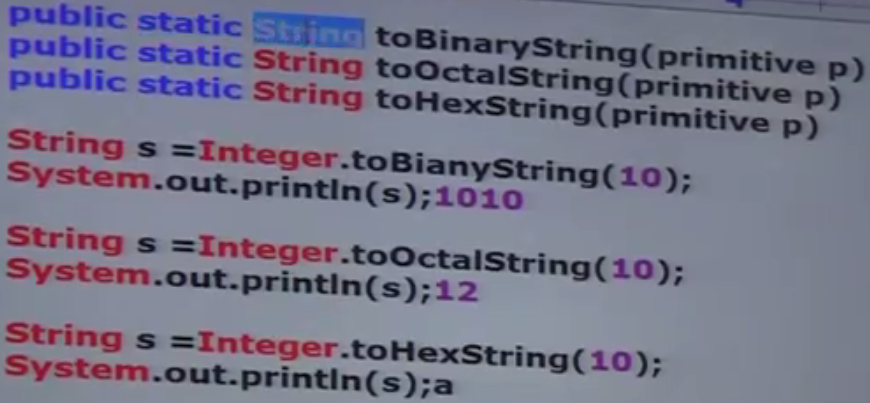
1. **Application**: To convert wrapper object or primitive to string form.
2. **Forms**
   1. **Form1**:   
      ****
   2. **form2**: Every wrapper class including Character class contains the following static method

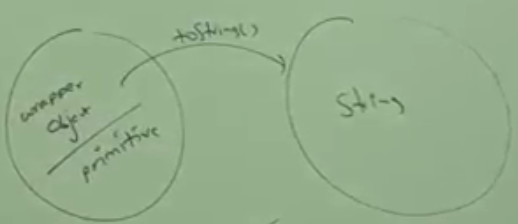


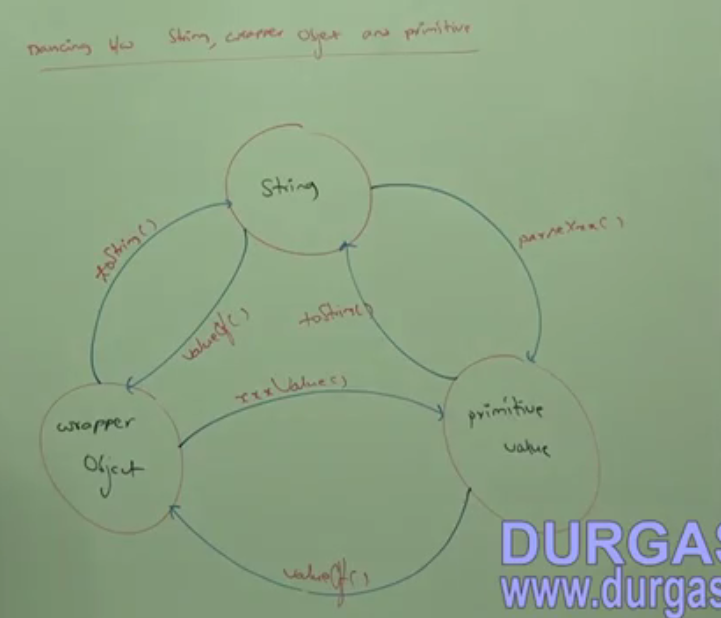
* 1. **form3:** Integer and Long classes contain the following toString() to convert primitive to specified radix string.  
     public static toString(primitive p, int radix);   
     The allowed radix 🡪 2 to 36



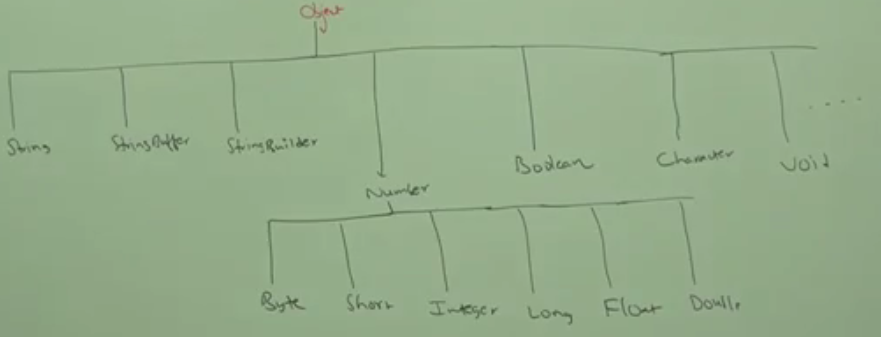
* 1. **Form4**:
     1. Integer and Long wrapper classes contains the following toXXXString()



* + 1. Purpose:   
       
  1. parseXXX(): To convert String to Primitive.
  2. valueOf(): To Convert primitive/String to Wrapper object
  3. xxxValue(): To convert wrapper object to primitive
  4. toString(): To convert wrapper object to string

1. 
2. d

Partial Hierarchy of java.lang package



1. The wrapper classes which are not child class of Number are 🡪 Boolean, Character ( See Above)
2. The wrapper classes which are not direct child class of Object🡪 See Above.
3. All Wrapper classes are final.
4. In addition to String object, All wrapper class objects also **immutable**.
5. Sometimes **Void class** is also considered as Wrapper class.

Void Class

1. final
2. Direct child of Object
3. Doesn’t contain any methods but only one variable
4. Application: In general, we can use Void class in reflection to check whether the method return type is void or not.

