**Wrapper classes in java**

For each and every fundamental data type there exist a pre-defined class, Such predefined class is known as wrapper class. The purpose of wrapper class is to convert numeric string data into numerical or fundamental data.

**Why use wrapper classes ?**

We know that in java whenever we get input form user, it is in the form of string value so here we need to convert these string values in different different datatype (numerical or fundamental data), for this conversion we use wrapper classes.

**Example of wrapper class**

**class** WraperDemo

{

**public** **static** **void** main(String[] args)

{

String s[] = {"10", "20"};

System.**out**.println("Sum before:"+ s[0] + s[1]); // 1020

**int** x=Integer.parseInt(s[0]); // convert String to Integer

**int** y=Integer.parseInt(s[1]); // convert String to Integer

**int** z=x+y;

System.**out**.println("sum after: "+z); // 30

}

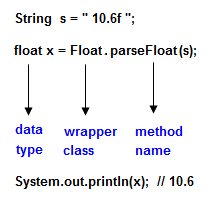
}

**Output**

Sum before: 1020

Sum after: 30

**Explanation:**In the above example 10 and 20 are store in String array and next we convert these values in Integer using "int x=Integer.parseInt(s[0]);" and "int y=Integer.parseInt(s[1]);" statement  
In "System.out.println("Sum before:"+ s[0] + s[1]);" Statement normally add two string and output is 1020 because these are String numeric type not number.



**Converting String data into fundamental or numerical**

We know that every command line argument of java program is available in the main() method in the form of array of object of string class on String data, one can not perform numerical operation. To perform numerical operation it is highly desirable to convert numeric String into fundamental numeric value.

**Example**

"10" --> numeric String --> only numeric string convert into numeric type or value.

"10x" --> alpha-numeric type --> this is not conversion.

"ABC" --> non-numeric String no conversion.

"A" --> char String no conversion.

Only 'A' is convert into ASCII value that is 65 but 'A' is not convert into numeric value because it is a String value.

**Fundamental data type and corresponding wrapper classes**

The following table gives fundamental data type corresponding wrapper class name and conversion method from numerical String into numerical values or fundamental value.

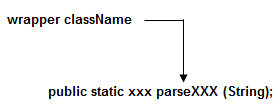
|  |  |  |
| --- | --- | --- |
| **Fundamental DataType** | **Wrapper CalssName** | **Conversion method from numeric string into fundamental or numeric value** |
| byte | Byte | public static byte parseByte(String) |
| short | Short | public static short parseShort(String) |
| int | Integer | public static integer parseInt(String) |
| long | Long | public static long parseLong(String) |
| float | Float | public static float parseFloat(String) |
| double | Double | public static double parseDouble(String) |
| char | Character |  |
| boolean | Boolean | public static boolean parseBoolean(String) |

**How to use wrapper class methods**

All the wrapper class methods are static in nature so we need to call these method using class.methodName().

* for Integer: int x=Integer.parseInt(String);
* for float: float x=Float.parseFloat(String);
* for double: double x=Double.parseDouble(String);

Each and every wrapper class contains the following generalized method for converting numeric String into fundamental values.



Here xxx represents any fundamental data type.