TP03

Java Datatypes

# Remark

1. In Java, there are 2 types of basic datatypes:
   1. Primitive Types
   2. Class Types

Sometimes, we need to use Class Type instead of Primitive Type. In Java, each Primitive Type has corresponding wrapper Class Type as shown in table below:

|  |  |  |
| --- | --- | --- |
| **Primitive type** | **Wrapper class** | **Constructor Arguments** |
| byte | [Byte](https://docs.oracle.com/javase/8/docs/api/java/lang/Byte.html) | byte or String |
| short | [Short](https://docs.oracle.com/javase/8/docs/api/java/lang/Short.html) | short or String |
| int | [Integer](https://docs.oracle.com/javase/8/docs/api/java/lang/Integer.html) | int or String |
| long | [Long](https://docs.oracle.com/javase/8/docs/api/java/lang/Long.html) | long or String |
| float | [Float](https://docs.oracle.com/javase/8/docs/api/java/lang/Float.html) | float, double or String |
| double | [Double](https://docs.oracle.com/javase/8/docs/api/java/lang/Double.html) | double or String |
| char | [Character](https://docs.oracle.com/javase/8/docs/api/java/lang/Character.html) | char |
| boolean | [Boolean](https://docs.oracle.com/javase/8/docs/api/java/lang/Boolean.html) | boolean or String |

Example:

**public** **class** TestWrapper {

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

**byte** b = 8;

Byte B = Byte.valueOf(b);

System.***out***.println("b="+b);

System.***out***.println("B="+B);

b = 20;

System.***out***.println("b="+b);

System.***out***.println("B="+B);// won't change

}

}

Output:

b=8  
B=8  
b=20  
B=8

Example 02:

Short is range from -32768 to 32767 (2 bytes, -215 to 215-1):

**public** **class** TestShortWrapper {

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

**short** sh = 120;

Short aShort = sh;

Short aShort2 = **new** Short(sh);

Short aShort3 = **new** Short("34");

System.***out***.println("sh=" + sh);

System.***out***.println("aShort=" + aShort);

System.***out***.println("aShort2=" + aShort2);

System.***out***.println("aShort3=" + aShort3);

}

}

Output:

sh=120  
aShort=120  
aShort2=120  
aShort3=34

1. Datatype Conversion types:
   1. Widening conversions
      1. byte to short, int, long, float, or double
      2. short to int, long, float, or double
      3. char to int, long, float, or double
      4. int to long, float, or double
      5. long to float or double
      6. float to double
   2. Narrowing conversions
      1. Converting from bigger to smaller size, there must be lost something. Example double to int, there must be lost the floating point.
      2. Conversion must be done explicitly using casting  
         Example:

int doubleToInt = **(int)**aDoubleValue;  
byte doubleToByte = **(byte)**aDoubleValue;

# TP03.1. Seconds to Time

Implement an application Java that take a given number of seconds for converting into time with format Hours:Minutes:Seconds. Example

Input number of seconds: 1259  
Time corresponding to 1259seconds is 00:20:59.

# TP03.2. Time to Seconds

Write a program in Java to a Time (hours, minutes, seconds) to number of seconds. Hours, minutes and seconds are inputted from keyboard. Example:

Program for converting time to seconds.  
Please input hours: 20  
Please input minutes: 30  
Please input seconds: 50  
  
Number of seconds = 20x3600 + 30x60 + 50 = 73850

# TP03.3. Calling Cost

Write a program in Java to calculate cost of calling with given Time Start (hours, minutes, and seconds) and Time End (hours, minutes, and seconds). The cost of call per minute is 0.05$. The program will display total number of minutes called and total cost of the call. Example:

Program for calculating cost of a call.  
Please input start hours: 20  
Please input start minutes: 30  
Please input start seconds: 50

Please input end hours: 20  
Please input end minutes: 35  
Please input end seconds: 5

Total call duration: 0h 4mn 15s  
Total cost of this call: 0.2125$

# TP03.4. Riels to Dollar

Write a program in Java to money in Riels to Dollar. Suppose that conversion rate is 1$ = 4000៛. Example:

Program for converting money in Riels to Dollars.  
Conversion rate is: 1 USD = 4000 RIELS  
Please input money in Riels: 2200  
  
2200 RIELS = 0.55 USD

# TP03.5. Traveling Duration

Write a program in Java to calculate duration of travel from ITC to Airport. The distance is 7 km and the average speed is 30km/h. The traffic jam factor is given as percentages of the average speed. After the calculation, display duration in format hh:mm:ss. Example:

Program for calculating duration of travel from ITC to Airport.  
Please input traffic jam factor (in percentage [0-100]): 50  
Travelling Duration = 00:28:00