

Exercise 3: Expressions and operators

Task 0 – Setting up IntelliJ

Install the IntelliJ development environment.

<https://www.jetbrains.com/idea/download/#section=windows>

Please note:

- Use the latest version of the **Community Edition**.
- Make sure that a correct path for the Java Development Kit is defined under “Project SDK”.

Preliminary work

- Create a new package in your Programming Basics project called `Exercise03`.
- Create a new Java class inside the project with the name `ScannerExample`.
- In previous exercises, you have always explicitly initialised variables in the source code; to read values from the keyboard, you can use the `Scanner` class from the Java library. To do this, import `java.util.Scanner` right at the beginning of your class, and use the methods of the type `DataType.ParseDataType`

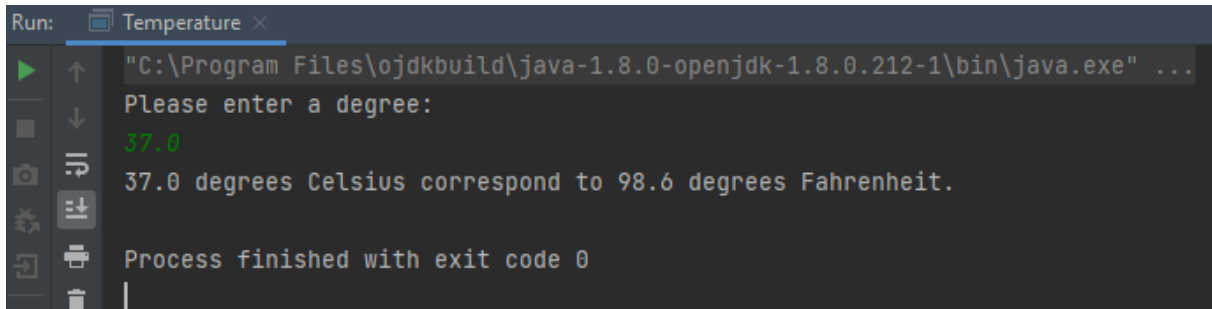
```
import java.util.Scanner;
public class ScannerExample {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        byte byteValue = Byte.parseByte(scanner.nextLine());
        short shortValue = Short.parseShort(scanner.nextLine());
        int intValue = Integer.parseInt(scanner.nextLine());
        long longValue = Long.parseLong(scanner.nextLine());
        float floatValue = Float.parseFloat(scanner.nextLine());
        double doubleValue = Double.parseDouble(scanner.nextLine());
        boolean booleanValue =
        Boolean.parseBoolean(scanner.nextLine());
    }
}
```

To enter a value, first start the programme, then click in the "Console" window (to set the input focus) and type the value followed by `Enter`. The method `Scanner.nextLine()` reads all characters up to the end of the line.

Expand the programme to make it easier to the user to follow the requested inputs by adding `System.out.println` statements where appropriate.

Task 1 - Temperature conversion

Create a new `Temperature` class and add a `main` method to the class. Then ensure that the class provides the following functionality: read a floating-point number from the keyboard (e.g. `Double.parseDouble()`), which is to be converted to degrees Fahrenheit and output:



```
Run: Temperature x
"C:\Program Files\jdkbuild\java-1.8.0-openjdk-1.8.0.212-1\bin\java.exe" ...
Please enter a degree:
37.0
37.0 degrees Celsius correspond to 98.6 degrees Fahrenheit.
Process finished with exit code 0
```

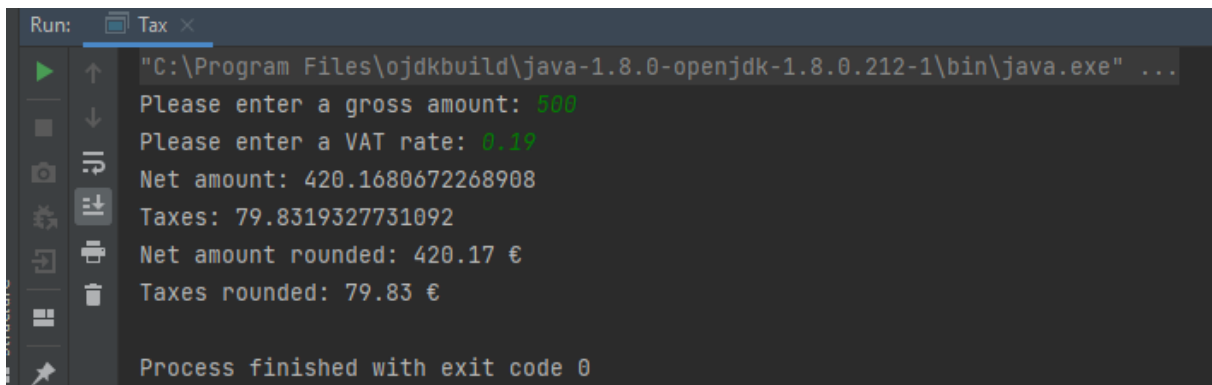
The following ratio applies for converting Celsius to Fahrenheit:

$$\text{Fahrenheit} = 9/5 * \text{Celsius} + 32$$

Keep in mind that 9 and 5 are interpreted as integers, and thus integer division is also performed. What are the different ways to solve this problem?

Task 2 - Value added tax calculation

Create a new `Tax` class and add a `main` method to the class. The class should read a gross amount and VAT rate from the keyboard, and then calculate the corresponding net amount and the VAT included. After the calculation, the results should be appropriately displayed on the screen, e.g.



```
Run: Tax x
"C:\Program Files\jdkbuild\java-1.8.0-openjdk-1.8.0.212-1\bin\java.exe" ...
Please enter a gross amount: 500
Please enter a VAT rate: 0.19
Net amount: 420.1680672268908
Taxes: 79.8319327731092
Net amount rounded: 420.17 €
Taxes rounded: 79.83 €
Process finished with exit code 0
```

Use the following calculation formula: $\text{net} = \text{gross} / (1 + \text{VATrate})$

Use appropriate data types and name your variables in a meaningful way. Then test your programme with different values.

Task 3 - Satellite time

A satellite transmits time periods to Earth as a "number of seconds". Write a Java programme `SatelliteTime`, which converts a number of seconds into a time specification in the format

`d h m s` and then outputs the calculated result to the console. The following applies:

`d` = number of days,

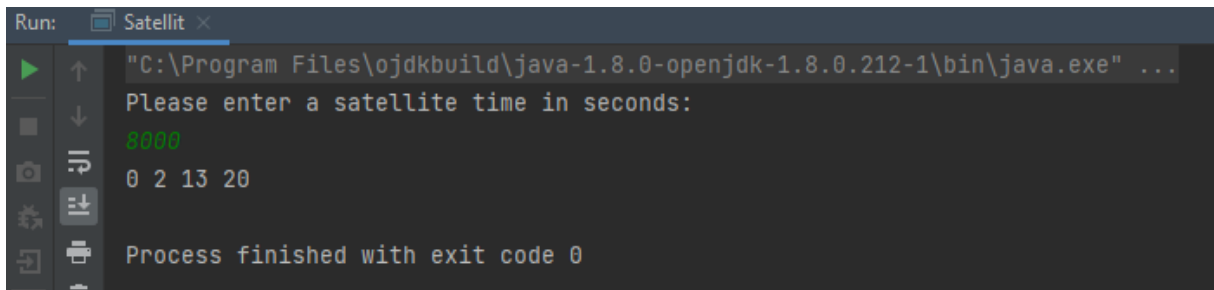
`h` = number of hours in the range from 0 to 23,

`m` = number of minutes in the range from 0 to 59,

`s` = number of seconds in the range from 0 to 59.

Implementation notes:

- Create a new class `SatelliteTime`.
- At the beginning of the programme, define a variable which the number of seconds to be converted is assigned to. **After a single assignment, the value should no longer be able to be changed.**
- Use the division and modulo (remainder) operators to calculate the individual values.
- Test your programme using 8000 as the number of seconds. As a result, the following sequence of numbers should be output to the console:

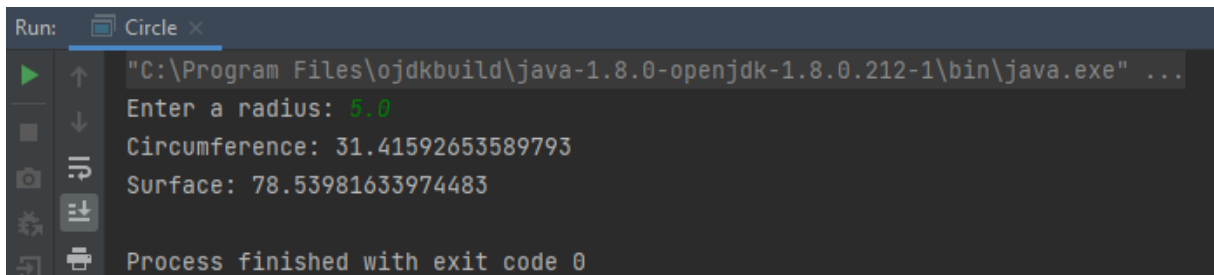


```
Run: SatelliteTime x
"C:\Program Files\ojdkbuild\java-1.8.0-openjdk-1.8.0.212-1\bin\java.exe" ...
Please enter a satellite time in seconds:
8000
0 2 13 20
Process finished with exit code 0
```

8000 seconds equals 0 days, 2 hours, 13 minutes and 20 seconds.

Task 4 - Circle calculation with console output

Create a new `Circle` class and add a `main` method to the class. The class should calculate both the circumference and the area of the circle, from a radius read from the keyboard. After the calculation, the two calculated values should be output to the console.



```
Run: Circle x
"C:\Program Files\ojdkbuild\java-1.8.0-openjdk-1.8.0.212-1\bin\java.exe" ...
Enter a radius: 5.0
Circumference: 31.41592653589793
Surface: 78.53981633974483
Process finished with exit code 0
```

Use the predefined `Math.PI` constant for the calculation, along with the following two calculation formulas:

- Circle circumference: $U = 2 * \pi * r$
- Circle surface: $A = \pi * r^2$

Then test your programme with different radii.