Programming 2

Tutorial 3

Activity 1 (required)

Given below is a program named Ex1 that uses a java parser tool, named JavaParser to parse the program text of the class Hello and produce the standard Java source code text. The in-line comments give information about how to use JavaParser.

a) Run this program to produce the source code that is listed immediately below the program.
   Note: You need to:
   - add to the class path the library file named javaparser-core-3.0.1.jar (provided in the resources/lib folder)
   - use import statements to import the necessary classes into your program.

b) Change the program text (i.e. value of the variable progText) by removing the semi-colon (';') at the end of the print statement. Rerun the program and record what happens.

c) Change the program text by removing the word "main". Rerun the program and record what happens.

```
1 import com.github.javaparser.JavaParser;
2 import com.github.javaparser.ast.CompilationUnit;
3 public class Ex1 {
4 public static void main(String[] args) {
5 // program text
6 String progText = "class Hello { "
7     + "public static void main(String[] args) { "
8     + " System.out.println(\"Hello world!\"); "
9     + "} "
10    + "}";
11 // parse the program text
12 CompilationUnit codeUnit = JavaParser.parse(progText);
13 // obtain the generated source code
14 System.out.println(codeUnit);
15 }
16 }
```

Source code output:

```
class Hello {
public static void main(String[] args) {
System.out.println("Hello world!");
}
}
```

Activity 2 (required)

Create a class named "Product" with three attributes: **`name, price`**, and **`discount`**. The class should also include two methods: calculate import tax (10% of the product price) and display information on the screen. The information displayed on the screen should include:

Product name
Unit price
Discount
Import tax

Product
+ name: String **+ price: double** **+ discount: double**
+ getImportTax():double **+ display(): void** **+ input(): void**

Activity 3 (required)

Write a program use class in Activity 2 to create 2 products, with information entered from the keyboard, and then call the display method to output the information of the 2 created **`Product`** objects.
Create a class containing the **`main()`** method:
In the **`main()`** method, create 2 objects, product1 and product2, from the Product class.
Call the **`input()`** method of the product1 and product2 objects to input data from the keyboard.
Call the **`display()`** method of the product1 and product2 objects to output the information of each object to the screen.

Activity 4 (required)

Upgrade the **`Product`** class in Activity 3 by adding **`'public'`** access modifiers for the **`display`**() method and **`'private'`** for the **`getIncomeTax`**() method. Additionally, add two constructors. The first constructor takes three parameters**`: name, price`**, and **`discount`**. The second constructor takes two parameters: **`name`** and **`price`** (implicitly assuming no discount). Write a program to create two products, one with a discount and one without, then display the information of the two products on the screen.

Activity 5 (optional)

A company has *N* employees, called Employee *1* through *N*.
There are two work orders, called Work A and B, which must be completed.
Employee *i* can complete Work A in *Ai* minutes and Work B in *Bi* minutes.

Each work will be assigned to one employee.
You can assign both works to the same employee, in which case the time it takes for him/her to complete them is the **sum** of the times it takes for him/her to do them individually.
If you assign the works to different employees, the time it takes to complete them is the **longer** of the times it takes for them to do their respective works.
Find the shortest possible time needed to complete the works.

Constraints
$2 \leq N \leq 1000$
$1 \leq Ai \leq 100$
$1 \leq Bi \leq 100$
All values in input are integers.

Input: data.txt

```
N
A1 B1
A2 B2
A3 B3
...
AN BN
```

E.g.
data.txt

```
3
6 5
4 2
6 6
```

Output: 5

If you assign Work A to Employee 2 and Work B to Employee 1, they will complete them in 4 and 5 minutes, respectively. Since you assigned the works to different employees, it will take max(4,5)=5 minutes for the two works to be finished. It is impossible to finish them earlier.

data.txt

```
3
6 5
2 2
4 6
```

Output: 4

It is optimal to assign both works to Employee 2.

Note that if you assign both works to the same employee, the time it takes for him/her to complete them is the **sum** of the times it takes for him/her to do them individually.

Submission

Submit a **zip** file containing all Java programs to this tutorial's submission box in the course website on FIT Portal.