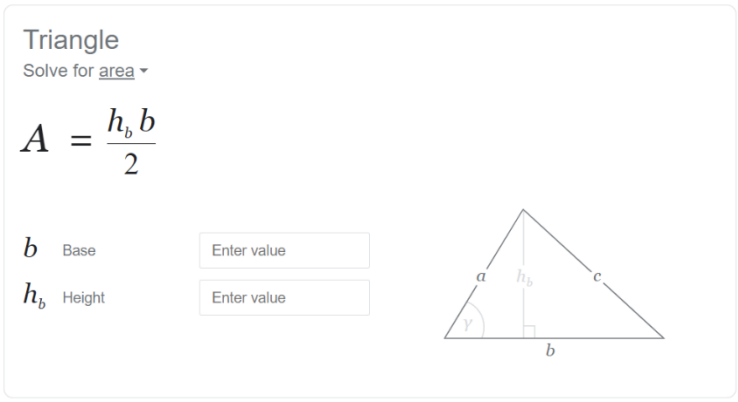
Programming 1

Tutorial 2

Activity 1

The area of a triangle can be calculated from its base b and height h_b. Write a simple Java program to calculate and display the area of a triangle from its base and height value. The program needn't use variables. Run the program using both CMD and IDE.



Sample program output

```
The triangle's base is 3 (cm) and height is 1.5 (cm).
Its area (cm2) is:
2.25
```

Deliverable

TriangleArea.java

Activity 2

Given three integers: `a`, `b` and `c`, print out the largest number. You should initialize values for `a`, `b` and `c` in the code. That means, you don't have to get these values from user (no need to use `Scanner` in this program). Name your program `MaxOfThree`.

Sample program output

```
Among 3, 6 and 2, the largest is 6.
```

Deliverable

MaxOfThree.java

Activity 3

(optional)

Given the following rates of exchange:

1 quan = 100 dong

1 dong = 10 hao

1 hao = 10 xu

You have 483,274 xu. Write a program named `CoinConverter` to convert this amount of xu into quan, dong, hao and xu.

Expected program output:

```
483274 xu converts into:
48 quan, 32 dong, 7 hao, 4 xu
```

Hints:

Refer to the exercises in a recent lecture for solutions to a similar problem.

Deliverable

CoinConverter.java

Activity 4

(optional)

Write a Java program that declares a floating-point number and prints “zero” if the number is zero. Otherwise, print “positive” or “negative”. Add “small” if the absolute value of the number is less than 1, or “large” if it exceeds 1,000.

1	`A = 0`	Zero
2	`A >= 1 && A <= 1,000`	Positive
3	`A <= -1 && A >= -1,000`	Negative
4	`A > 0 && A < 1`	Small positive

5	`A < 0 && A > -1`	Small negative
6	`A > 1,000`	Large positive
7	`A < -1,000`	Large negative

In short, the program should generate a sentence of the following pattern:

`A + (small|large|) + (negative|positive) + number.`

The `|` sign means "or" so `(negative|positive)` means "either *negative* or *positive*" and `(small|large|)` means "either *small* or *large* or *nothing*". The `+` sign here just serves as the space charater.

Hint

Use `if-else` to handle the `Zero` case. If the number is not zero, you should start with an empty string and gradually add the parts of the output sentence to it. For the parts that are variable (e.g. `(negative|positive)`), the `if-else` statement should be used.

Expected program output (these are the results of multiple runs)

```
(0.0) The number is zero.
(0.9) A small positive number.
(1.0) A positive number.
(1000000.0) A positive number.
(1000000.1) A large positive number.
(-0.1) A small negative number.
(-1.0) A negative number.
(-1000000.0) A negative number.
(-1000000.1) A large negative number.
```

Deliverable

ReadFloatNumer.java

Extra Info

Real numbers in Java

By default, a real number values such as `5.0` or `60.2` are treated as `double` type by Java. Therefore, the following statement will cause an "incompartible type" error:

```
float a = 1.65;
```

The way Java sees it, you are trying to assign a `double` value to a `float` variable, and a `float` variable cannot hold a `double` value.

Question: *So how do I write float values in Java??*

Answer: You can mark a value as `float` by adding an `f` right after it.

```
float a = 1.65f;
```

Question: *So should I use float or double?*

Answer: It depends, but in most cases it doesn't hurt to use `double` type for real numbers. Consider using `float` if there are compatibility reasons or memory requirement is crucial.

The `System.out.print()` statement

Do you know that you can display a text or value without moving onto the next line in Java? Try out the `System.out.print()` statement, like so:

```
int n = 30;
System.out.print("There are ");
System.out.print(n);
System.out.println(" students in my class.")
```

Submission

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