Programming 1

Tutorial 2

# Activity 1

Given an amount of money in VND, such as 1,200,000 VND, and the exchange rate of 207 VND to 1 JPY. Write a simple Java program to calculate and display the amount of equivalent JPY for a given amount of VND.

Let vnd be the amount of VND and jpy be the equivalent amount of JPY. Then we have:

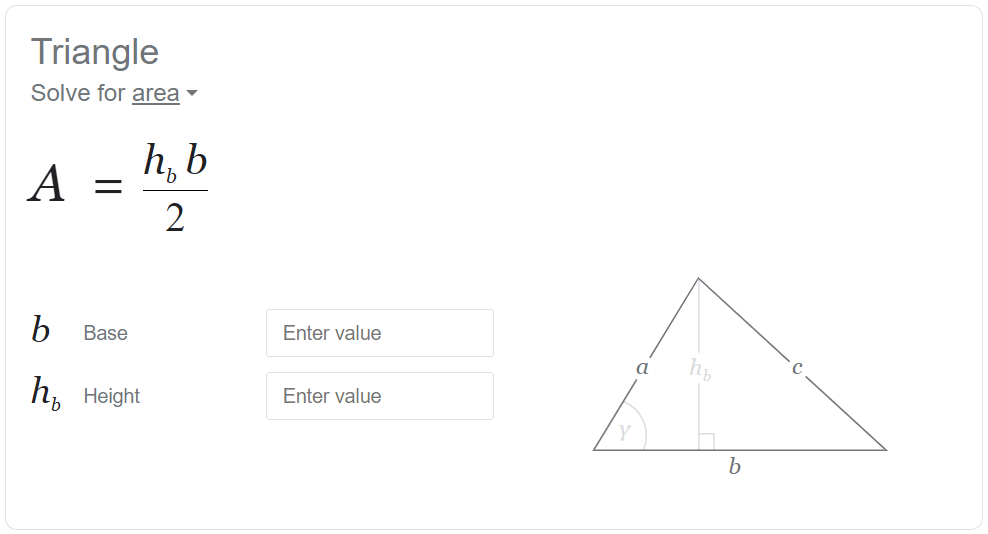
*jpy* = *vnd* / 207

## Sample result

1200000 VND is roughly 5797 JPY.

# Activity 2

The area of a triangle can be calculated from its base *b* and height *hb*. 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 result

Base length: 1.5 (cm)

Height: 2.0 (cm)

The triangle's area: 1.5 (cm2)

# Activity 3

Given two non-negative integers, tell if they have the same last digit, such as with 27 and 57.

## Hint

The mod (%) operator can be used to extract the last digit. It is the remaining of the division of a number by 10.

## Sample results

Two numbers 27 and 57 have the same last digit.

Two numbers 6 and 17 have different last digit.

Two numbers 3 and 113 have the same last digit.

# Activity 4

Given three integers: a, b and c, print out the largest number.

## Sample result

Among 3, 6 and 2, 6.

# Activity 5

## Task

1 quan = 100 dong

1 dong = 10 hao

1 hao = 10 xu

You have 483274 xu. Convert them into quan, dong, hao and xu.

## Expected result:

483274 xu is:

48 quan, 32 dong, 7 hao, 4 xu

## Instructions:

Refer to a recent lecture for solutions to a similar problem.

# 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?