# Assignment 11: Personal Finances

## Preparation

You are expected to be familiar with the following lecture *before* beginning the assignment. You are required to read the assignment in advance. You are required to submit external documentation with your code. Your internal documentation is *not* required to be in the Javadoc format.

### Required programs

- `GenerateMonthlySpending.class` from D2L

### Most relevant lectures

- L18

### Exercises

1. Run the `GenerateMonthlySpending` program from the command line. Look in your current directory and check for a new directory called Finances. Open the directory and look at the files inside. Pay attention to their filenames. Open some of the files and look at the contents.

2. Describe the possible filenames generated by `GenerateMonthlySpending`. What is the meaning of the filename?

3. Describe the format of the files generated by `GenerateMonthlySpending`. Explain what each row must look like. What is the meaning of the + and - symbols?

4. Read through the assignment.

## Introduction

You may work in pairs. You will create a file processing program. The program will create a summary file for each year. Each summary file will list the year, and the net financial gain (or loss) for each month in that year.

## Specifications

- Write a program called `SummarizeFinances`.

- Your program should go through all the files in the Finances directory, and generate a summary of the activity for each year.

- You need to generate a separate text file for each year.

    – The name of each file generated should be `yyyy_summary.txt`, where `yyyy` is replaced by the actual year.

- Each generated file must have two columns.

    – The columns should be aligned nicely, as a table.

- The first column will contain the names of all the months that had associated data.
- The second column will contain the net earnings for that month (or the net spending, if the amount is negative).
- See sample output below.

## Documentation

You are required to:

1. Write internal documentation as usual.

2. Write external documentation for the `SummarizeFinances` program, similar to the external documentation example posted to D2L, but also including a top-down design diagram. Do not forget to include examples of the files your program will read, and examples of output files. Submit a `.pdf` file. You do not need to include algorithms for each of your methods. Just include a general algorithm for your `main` or top-level methods.

## Submission

Recall that submission instructions are in the **Lab Guide**. Your group is required to submit **one** `.zip` folder (in one person's D2L dropbox) containing:

- the internally documented and properly styled source code file, `SummarizeFinances`

- external documentation for `SummarizeFinances`, including top-down design diagram, in a `.pdf`

The submitted files should contain both partners' names in the headers. *Make sure both partners save copies of the finished code to their personal H: drive.*

## Sample Output

One of the generated files could be `2013_summary.txt`, and may contain the following:

```
Month        Net Earning/Spending
===================================
January      +$145.34
March        +$203.41
June         -$79.12
September    -$12.45
October      +$84.58
```