

Assignment 2: Future Value

Due date: 8 October 2021 (Fri)

Full mark: 100
Expected normal time spent: 6 hours

Aims:

1. To implement future value calculation for growing asset by writing a Java console application.
2. To practise the use of variables/expression, looping/branching statements and basic Math functions.

Background:

Future value (FV) is the value of an asset at a specific date. It measures the future sum of money at a specified time in the future assuming a certain interest rate; it is the present value, or *principal*, multiplied by the accumulation function. Compound interest is commonly applied where the interest shall be added to the principal sum of deposit for accumulation, or in other words, interest on interest. For example, given the principal amount of money P , annual interest rate r , the FV being accumulated after T years is given by:

$$FV = P \times (1 + r)^T$$

In this assignment, we are going to deal with principal between **\$10,000.00** to **\$109,700.00**, fixed annual interest rate from **1.0%** to **10.0%**, timespan of **2** to **10** years, while compounding period is assumed to be either **bi-monthly**, **quarterly** or **half-yearly** where it could be accounted like this:

$$FV = P \times \left(1 + \frac{r}{m}\right)^{Tm}$$

where m is the compounding frequency per year

In the output, we shall print out the first and the last future value (see examples shown later).

Another common inquiry related to future value is that given the same annual interest rate and compounding period, how long it shall take to double the asset? [Hint: To determine the required duration, one could take a natural logarithm to both sides and obtain the result by re-arranging terms.]. For example, in order to double an asset, we can start with the following:

$$2P = P \times \left(1 + \frac{r}{m}\right)^{Tm}$$

$$\log_e 2 = T \times m \times \log_e \left(1 + \frac{r}{m}\right)$$

$$T = \log_e 2 / (m \times \log_e (1 + \frac{r}{m}))$$

In our assignment, we are going to determine the time, in years and months, to double a given asset. You need to round up the time to nearest compounding period. Say if the compounding period is 3 months, 7.02 years would become 7 years 3 months; while 6.99 years would become 7 years.

Problem Definition:

In this assignment, you are required to write a program to perform future value calculation. You are required to obtain the input from the console standard input using *Scanner* and *System.in*. A sample run of the program is shown below (user input are **bold** and blue in color):

```
Input Principal [$10000.00 - $109700.00]: 109700.0
Input Annual Interest Rate [1.0% - 10.0%]: 10.0
Input Timespan [2 - 10 years]: 2
Input Compounding Period [2, 3 or 6 months]: 3
Future Value after 1 year: $121088.27
Future Value after 2 years: $133658.80
Time to double asset: 7 years 3 months
```

You are **NOT** required to validate the input number types; meaning that you can assume all input numbers are of valid number type. **However**, you are **REQUIRED** to validate the input range as indicated above, e.g. timespan should be in-between 2 to 10, warning “Invalid Timespan” should be given to user on invalid input and request input again until the entered value is within the range:

```
Input Timespan [2 - 10 years]: 12
Invalid Timespan, please enter again.
Input Timespan [2 - 10 years]: 8
Input Compounding Period [2, 3 or 6 months]: 6
```

Similar warnings should be given for **Invalid Principal**, **Invalid Annual Interest Rate** and **Invalid Compounding Period**.

The number of printed future value should depend on timespan in years. For 2 years only, the output should be:

```
Input Principal [$10000.00 - $109700.00]: 109700.0
Input Annual Interest Rate [1.0% - 10.0%]: 10.0
Input Timespan [2 - 10 years]: 2
Input Compounding Period [2, 3 or 6 months]: 6
Future Value after 1 year: $120944.25
Future Value after 2 years: $133341.04
Time to double asset: 7 years 6 months
```

For 3 years, we shall print the future values in this way:

```
Input Principal [$10000.00 - $109700.00]: 109700.0
Input Annual Interest Rate [1.0% - 10.0%]: 10.0
Input Timespan [2 - 10 years]: 3
Input Compounding Period [2, 3 or 6 months]: 3
Future Value after 1 year: $121088.27
Future Value after 2 years: $133658.80
Future Value after 3 years: $147534.30
Time to double asset: 7 years 3 months
```

For more than 3 years, we shall replace some of the items with “...”:

```
Input Principal [$10000.00 - $109700.00]: 109700.0
Input Annual Interest Rate [1.0% - 10.0%]: 10.0
Input Timespan [2 - 10 years]: 10
Input Compounding Period [2, 3 or 6 months]: 2
Future Value after 1 year: $121137.37
Future Value after 2 years: $133767.20
...
Future Value after 10 years: $295747.92
Time to double asset: 7 years
```

You shall define one package **futurevalue** and one class called **FutureValue** in a new NetBeans project named **FutureValue**. The class shall contain a main method that performs all the operations including input, conversion and output. Optionally, you may define more than one method in addition to the main method to carry out the calculation.

Procedure:

1. Create a new NetBeans project with a source file named **FutureValue.java** that contains the class **FutureValue**, under package **futurevalue**.
2. Define the main method, together with optionally other fields and methods (if any), to perform I/O as well as future value calculation in the **FutureValue** class.
3. If you have completed writing the **FutureValue** class, try build the project (press the function key **[F11]** on the keyboard). If there are errors, don't panic. Double-click on the first error message in the Output window. Check the error, correct it and re-compile. Feel tired? Take a rest.
4. If you have many opened projects, close others or click menu **[Run] [Set Main Project]**.
5. You may insert **println()** statements in your work to inspect variables and intermediate results.
6. When you finish and there is no more error, you are ready to try out the program by pressing the function key **[F6]** on the keyboard. Then you can type the input in the standard input. Enjoy!

Submission:

1. **Locate** your NetBeans project folder, e.g. **H:\FutureValue**.
2. ZIP the project folder **FutureValue** and Submit the file **FutureValue.zip** via our Online Assignment Collection Box on Blackboard <https://blackboard.cuhk.edu.hk>

Marking Scheme and Notes:

1. Name your project, package and file correctly.
2. Corresponding student name, student ID and date should be filled into the specified position in the annotation.
3. The indentation of the code in the .java file should make the code easy to read. The .java file should contain comments to indicate the computational logic.
4. The program should be free of any compilation errors such as missing semicolon, type mismatch, etc. If your code suffers from only one or two minor and correctable error, marker will try to apply adequate correction and attempt grading the functionalities of the work, i.e., checking output format and applying the test cases.
5. The program should be runnable.
6. Input and output format will be checked, for examples, spelling mistakes, user input handling, spacing, etc.
7. To evaluate the accuracy of the program, the program has to pass different test cases. Such test cases shall not be released in advance, so test your work using different sets of inputs.

Remember to do your submission before 6:00pm of the due date. **NO late submission would be accepted.**

If you submit multiple times, **ONLY** the content and timestamp of the **latest** one would be counted. You may delete (i.e. take back) your attached file and re-submit. We shall take into account the last submission **ONLY**.

University Guideline for Plagiarism:

Attention is drawn to University policy and regulations on honesty in academic work, and to the disciplinary guidelines and procedures applicable to breaches of such policy and regulations. Details may be found at <http://www.cuhk.edu.hk/policy/academichonesty/>.

With each assignment, students are required to submit a statement that they are aware of these policies, regulations, guidelines and procedures, in a header comment block.

Faculty of Engineering Guidelines to Academic Honesty:

MUST read: <https://www.erg.cuhk.edu.hk/erg/AcademicHonesty>
(you may need to access via CUHK campus network/ CUHK1x/ CUHK VPN)