FINA3351B Assignment 1
(Due by the end of March 7, 2025, Friday)

(*100 points in total*) Download Excel file "HW1.xlsm" and finish all the questions in Excel. Submit your Excel file to the "Assignment1" session on Moodle.

Q1. (*25 points*) Corporate finance.

Present value (PV) is the value of all future cash flows (positive and negative) over the entire life of an investment discounted to the present.

$$PV = \sum_{t=1}^{N} \frac{CF_t}{(1+r)^t}$$

CF_t is the future cash flow of the investment at time t, r is the discount rate.

Net Present value (NPV) is the present value minus the cost of acquiring the asset (cash flow at time zero):

$$NPV = CF_0 + \sum_{t=1}^{N} \frac{CF_t}{(1+r)^t} = \sum_{t=0}^{N} \frac{CF_t}{(1+r)^t}$$

Please note that Excel built-in function `NPV` is misleading. It differs somewhat from the standard finance nomenclature. Because it calculates the present value of future cash flows, which is actually PV. You must deduct the initial cash flow to get correct NPV of the investment.

Worksheet "Q1.NPV1" provides the discount rate and cash flows of an investment from Year 0 to Year 5.

(a) (*8 points*) Follow the instruction and record a macro.

1. Start recording and name the macro as `Q1_record`.
2. In cell C7, type formula "=B7/(1+$B$4)^A7". This calculates the present value of CF_0.
3. Copy C7 and paste special to C8:C12 by selecting "Formulas and number formats".
4. Change number format of cells in C8:C12 to "Number" with 2 decimal places.
5. In cell C15, type formula "=SUM(C7:C12)". This calculates NPV of the investment.
6. In cell C16, type formula "=NPV(B4,B8:B12)". This calculates the present value of the investment from Year 1 to Year 5.
7. In cell C17, type formula "=NPV(B4,B8:B12)+B7". This is the correct way to calculate NPV of the investment.
8. Change number format of cells in C15:C17 to "Number" with 2 decimal places, then change to Bold font.
9. Stop recording the macro.

The results after the calculation and formatting should look like this:

	A	B	C	D	E
1	**Corporate finance: Net Present Value**				
2					
3	**COMPUTING THE PRESENT VALUE OF CASH FLOWS**				
4	**Discount rate**	10%			
5					
6	**Year**	**Cash flow**	**Present value**		
7	0	-600	-600	<-- =B7/(1+$B$4)^A7	
8	1	100	90.91	<-- =B8/(1+$B$4)^A8	
9	2	120	99.17		
10	3	140	105.18		
11	4	160	109.28		
12	5	180	111.77		
13					
14	**Net present value:**				
15	Summing cells C7:C12		**-83.69**	<-- =SUM(C7:C12)	
16	Using NPV function		**516.31**	<-- =NPV(B4,B8:B12)	
17	NPV function + Initial cash flow		**-83.69**	<-- =NPV(B4,B8:B12)+B7	
18					

Place the recorded sub procedure `Q1_record` in Module `Q1`.

(b) (*7 points*) Worksheet "Q1.NPV2" contains the data with same format.

Write a sub procedure named `Q1_mod` that modifies the recorded macro. The sub procedure should do the followings:

1. Change `Value` property of cell C7 so that it calculates "=B7/(1+$B$4)^A7".
2. Copy C7 using `Copy` method, then use `PastSpecial` method to paste "Formulas and number formats" to C8:C12.
3. Also change `Value` property of cell C15, C16, C17 and insert the formula "=SUM(C7:C12)", "=NPV(B4,B8:B12)", "=NPV(B4,B8:B12)+B7", respectively.
4. Change `NumberFormat` of cells in C8:C12, C15:C17 to "Number" with 2 decimal places.
5. Change font of cells in C15:C17 to Bold.

Apply this sub procedure to worksheet "Q1.NPV2".

(c) (*10 points*) In Module `Q1`, write a function procedure called `NPVNEW` that requires two arguments `rate` and `Cashflow`. The function should return the net present value of a cash flow based on Discount Cash Flow (DCF) model:

$$NPV = \sum_{t=0}^{N} \frac{CF_t}{(1+r)^t}$$

The first argument `rate` is the discount rate. The second argument is the cash flow of the project from **period 0** to the last period of the investment, with `Range` data type.

Same as Excel built-in function `NPV`, `NPVNEW`'s argument `Cashflow` can accept a row vector, a column vector, and a 2-dimensional range in the spreadsheet.

In worksheet "Q1.NPV2" cell C20, use function `NPVNEW` to calculate the NPV of this investment.

Q2. (*20 points*) In Module `Q2`, write 2 sub procedures that calculate total revenue of a product under various scenarios.

First sub is named `Project1` that uses `If…Then` statements as decision-making statements.

Second sub is named `Project2` that uses `Select Case` as decision-making statements.

(1) Variable `Scenario` is the state of the market.

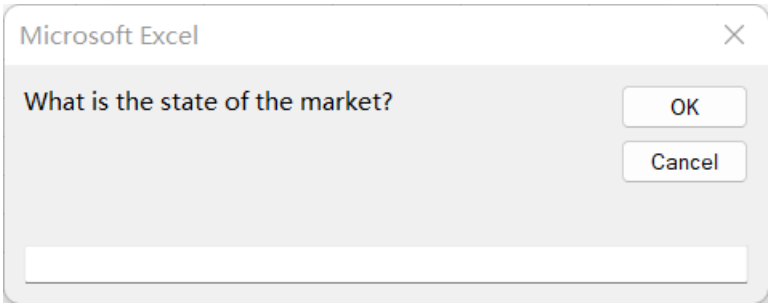
Variable `Price` is the price of the product.

Variable `Unit` is the number of products sold.

Variable `Revenue = Price * Unit` is the total revenue of the firm's project.

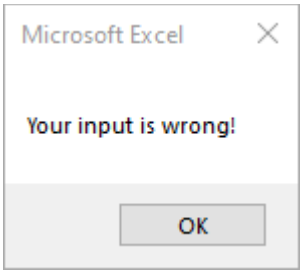
You may decide the data type of four variables when you declare them.

`Price` and `Unit` under each scenario of the market is given in sheet "Q2". Use an input box as below to assign a value to `Scenario` variable.



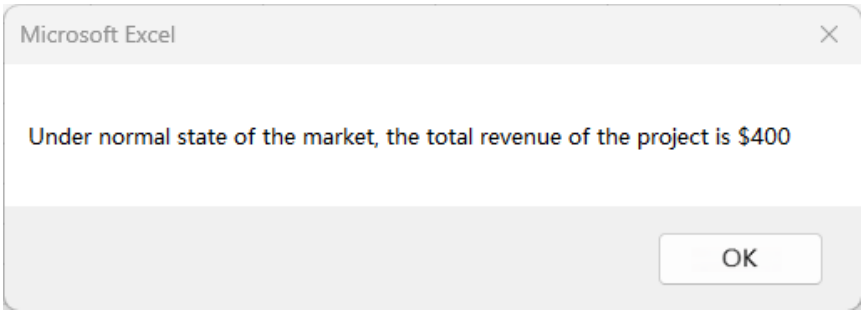
The sub is terminated immediately, if the user clicks Cancel or Close (X) button.

`Scenario` variable only accepts three values: `normal`, `best`, and `worst`. If the user enters wrong value of `Scenario`, a dialog box as below



should be displayed and the sub procedure is terminated immediately. Note that input values are case sensitive, e.g., "Normal" and "NORMAL" are both invalid inputs.

If the user enters an acceptable value of `Scenario`, use a message box as below to display the calculated `Revenue`. Note that in the message box, "normal" is the value of `Scenario` and "400" is the result of `Revenue`.



Q3. (*20 points*) In Worksheet "Q3", column B contains the 4-week Treasury Bill rates from October 31, 2023 to October 31, 2024. The interest rate is reported as a percentage per annum. The link to the data source is given in the spreadsheet.

Column C contains the daily adjusted close price of Google stock (Ticker: GOOG) in the same period.

Write a sub procedure called `DataAnalysis` in Module `Q3`.

Create three variables `Dates`, `annualRate`, and `stkPrice`. Then read "Date" data in column A into `Dates` variable, read "Interest Rate (Annual, %)" data in column B into `annualRate` variable, read "GOOG Adj. Close Price" data in column C into `stkPrice` variable.

In the sub procedure, calculate the log excess return of GOOG stock starting from 2023-11-01 (2nd date in the sample period) to 2024-10-31 using the equation

$$r_t^e = \ln(P_t) - \ln(P_{t-1}) - r_{f,t-1}$$

where r_t^e is daily log excess return on date t, P_t is stock price on date t, and $r_{f,t}$ is daily interest rate on date t. The daily interest rate is equal to the `annualRate/252/100`. (We assume 252 trading days in a year.) Write log excess return results into Column D.

VBA built-in functions `Month` and `Year` are used to return month and year from a given date. The online documents are listed below:

In the sub procedure, use these VBA functions to get year and month values from the date data in column A, and write the results in column E and F, respectively.

The outputs of the sub procedure should look like this:

	A	B	C	D	E	F	G
1	**Stock pricing data**						
2							
3	**Date**	**Interest Rate (Annual, %)**	**GOOG Adj. Close Price**	**Log Excess Return**	**Year**	**Month**	
4	2023-10-31	5.29	124.99				
5	2023-11-01	5.29	127.26	0.0177444	2023	11	
6	2023-11-02	5.29	128.27	0.0076762	2023	11	
7	2023-11-03	5.29	130.05	0.0136152	2023	11	

Q4. (*15 points*) In Module `Q4`, write two function procedures. The first one is `BSCall`. This function returns the Black-Scholes-Merton price of a European call written on a stock that pays continuous dividend yield. The expression of call price (*C*) is given by

$$d_1 = \frac{\ln(S/K) + (r - q + \sigma^2/2)T}{\sigma\sqrt{T}}$$

$$d_2 = d_1 - \sigma\sqrt{T}$$

$$C = Se^{-qT}N(d_1) - Ke^{-rT}N(d_2)$$

The second function is `BSPut`. This function returns the Black-Scholes-Merton price of a European put written on a stock that pays continuous dividend yield. The expression of put price (*P*) is given by

$$d_1 = \frac{\ln(S/K) + (r - q + \sigma^2/2)T}{\sigma\sqrt{T}}$$

$$d_2 = d_1 - \sigma\sqrt{T}$$

$$C = Ke^{-rT}N(-d_2) - Se^{-qT}N(-d_1)$$

The arguments of both functions are the same:

- the current stock price `S`
- strike price of the option `K`
- time-to-maturity `T`
- risk-free interest rate `r`
- continuous dividend yield q
- volatility of the stock returns `sig`

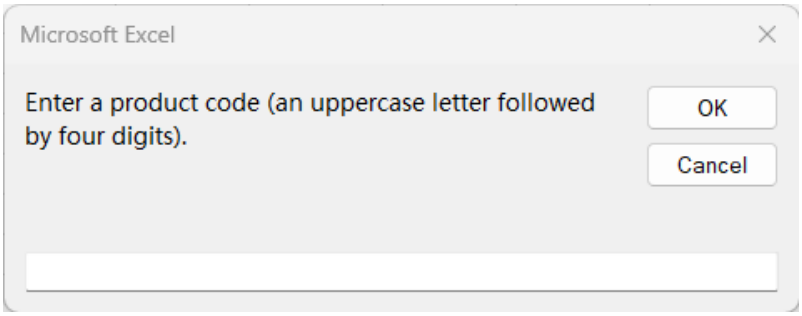
There is no requirement in the datatype of these arguments.

Note: In Excel, built-in function `NORMSDIST` is to return the cumulative distribution function (CDF) of a standard normal distribution, which is $N(\cdot)$ function. You can call this function in VBA to return $N(d_1)$ and $N(d_2)$. VBA built-in function `Exp` is to calculate exponential of a number.

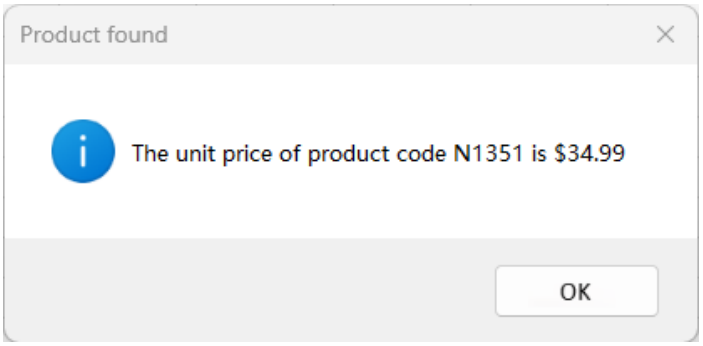
Worksheet "Q4" gives the inputs values of a European option. Please use the VBA functions to calculate the call price in cell B11 and put price in cell B12.

Q5. (*20 points*) Worksheet "Q5" contains information of product codes, and the unit prices for products. In Module `Q5`, write a sub procedure named `LookupPrice`.

1. Create a variable named `productCode` and read "Product Code" data into this variable. Create a variable named `unitPrice` and read "Unit price" data into this variable.
2. Use an input box to ask the user to enter the product code. The dialog box looks like below:

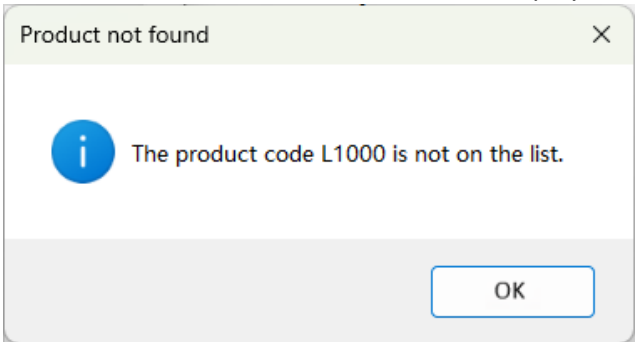


3. If the code that the user entered can be found in the table, display a message box as below:



Note: "N1351" should be the code that the user entered in the input box and 34.99 is the corresponding unit price of this product.

4. If the code that the user entered cannot be found in the table, display a message box as below:



Note: "L1000" should be the code that the user entered in the input box.

Feel free to declare the additional variables and the data type of the variables if necessary.