# Batch Calculator

This assignment is designed to give you experience using exception handling, which is a technique for anticipating and handling runtime errors. Generally, these are errors that happen as a result of user interaction or when processing data-driven transactions.

Develop a program that will:

* Read the batch.txt transaction file and for each line in the file
  + Parse the arithmetic problem presented.
    - If the line cannot be parsed, print an error message and continue execution
    - If the line involves an unknown mathematical operation, print an error message and continue education
    - Note that Python includes an eval() function that you must **not** use. This is so you get experience parsing strings to find and isolate data.
  + Process the arithmetic problem and compute the answer.
  + Print the problem and answer on the screen.

**Features**

This assignment features material that we have covered. Specifically:

* Functions
* Loops
* Decisions
* File I/O
* String Manipulation
* Command Line Arguments (possibly)
* Exception Handling
  + FileNotFoundError
  + ValueError
  + ZeroDivisionError

**Requirements**

* D Effort:
  + Deliver a non-working program that was headed in the right direction.
* C Effort:
  + Implement the program as described above.
* B Effort
  + Implement all of the D and C levels of effort. Gather the name of the transaction (text) file from a command line argument. You may have to research command line arguments if we did not cover this in the lecture.
* A Effort:
  + Implement all of the D, C, and B levels of effort. Then add the ability to handle two more types of transaction. The first being a comment. If the first character of a line begins with a pound sign (#) then ignore the line. The second being a print command. If the first character of a line begins with an exclamation point (!) then print the rest of the line without any mathematical processing. The test file batch2.txt has been provided to include these features. If you've implemented the B level, then you should be able to process either batch file by specifying which file you want to process on the command line, similar to the following:  
      
    python3 assign-08.py batch2.txt

**Hints**

* FileNotFoundError is thrown when you attempt to open a file that is not available. If you're working on the B Effort requirements, then you must be able to handle the case where the user gives you a filename for a file that does not exist.
* ValueError is thrown when you attempt to convert a non-number to a number. That is, you cannot convert "ABC" into a number. This can happen when there is bad data in the transaction file. (And there will be bad data in the transaction file. Count on it.)
* ZeroDivisionError happens when you attempt to divide by zero. There is no defined value for this situation, so you need to handle this exception, and print out something like "Undefined" as the answer to problems where division by zero is requested.