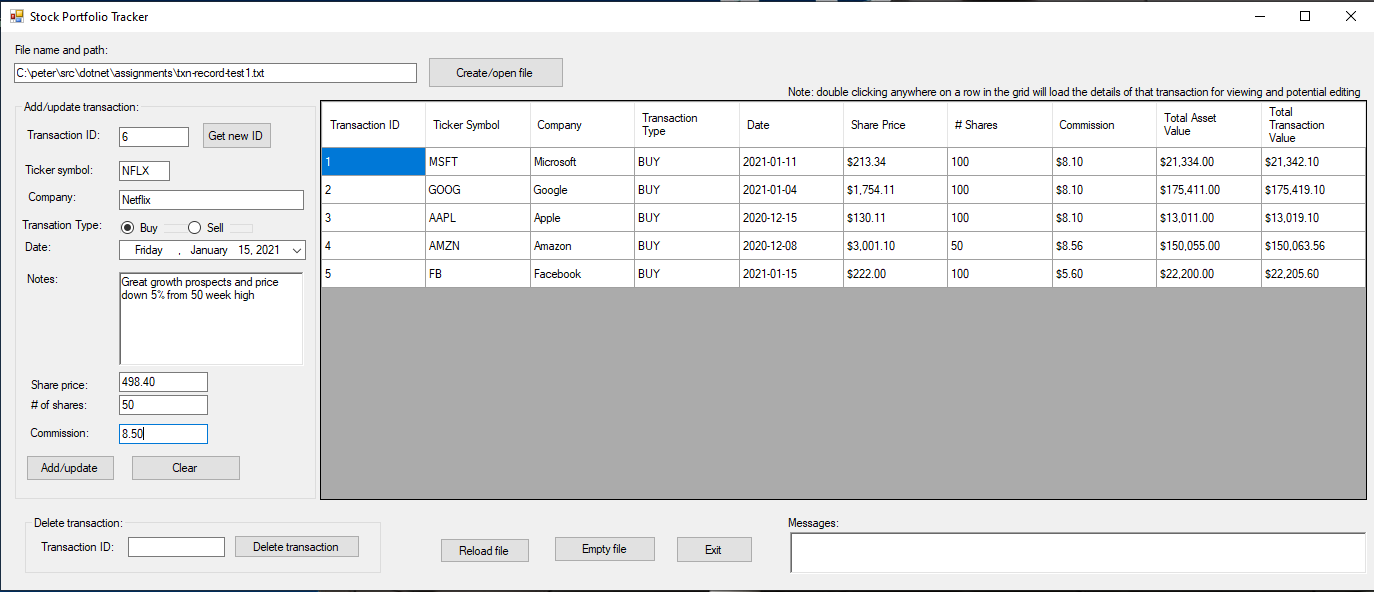
~~This is an exercise in file I/O management through an encapsulated class, not an exercise in List or Array management. If you declare a collection such as an array or List outside of a method or as a property, I will delete the object before I start marking. You have to think of a multi-user environment: if you retain your data in memory and just dump it to file periodically, then you will obliterate data added or updated by others … and they will do the same to you.~~

## Assignment 3 – File I/O

~~An investor wants an app to keep track of the stock transactions that he or she makes in their personal investment portfolio. The transactions (e.g. the investor bought 100 shares of MSFT at $216.34 per share) will be stored in a text file where each row represents a single transaction. The fields within the file will be separated by a delimiter of your choice~~.

1. ~~The starting point for the app is being able to open an existing transaction record file, or create the file if it doesn’t exist. If one didn’t exist, display a message to the screen notifying that a new file was created. If the file does exist, the program will display its records on a grid view when it is opened. Furthermore, any new data will be appended to it and the grid view will be kept in sync.~~
2. ~~The investor can add new, update existing, and delete transactions and all these changes need to be reflected immediately in both the text file and the grid view. More specifically:~~
   1. The list of existing transactions will be displayed on a grid view.
      1. ~~For the grid view, make sure you do not~~ *~~Allow Users To Resize Columns~~* ~~or~~ *~~Allow Users to Add Rows~~* ~~to the grid. Additionally, the grid should be~~ *~~Read Only~~* ~~and the~~ *~~Row Headers Visibility~~* ~~should be set to false~~.
      2. ~~Additionally, notice that there is~~ *~~not~~* ~~a 1 to 1 mapping between every field that will be stored in the file and the fields to be shown on the grid view. For example, the notes the investor records when tracking the transaction are not shown in the grid view but need to be stored in the file. The columns that your grid view should display are specified in the image below.~~
   2. ~~Format the Date as shown on the image below (e.g. 2021-02-21).~~
   3. ~~Display the Share Price, Commission Fee, Total Asset Value, and Total Transaction Value as currency.~~
   4. ~~When data is being inserted, updated, or deleted, a~~ *~~Filename~~* ~~needs to have already been specified.~~
   5. ~~Double-clicking anywhere on a row in the grid view will load the details for the transaction that row represents in the add/update transaction fields so that the user can potentially edit that transaction~~.
3. ~~Catch all file I/O exceptions (open/read/write) and display them as error messages on the form~~.
4. ~~All new/update transaction input fields are required~~. ~~If the transaction does not exist in the file, insert the record. Otherwise, update the record~~. ~~Any errors during the insert or update, are displayed all at once, one line per error, and give the focus to the first field in error. Also, the following constraints must be met:~~
   1. ~~The transaction ID must be a unique integer, and greater than zero.~~
   2. ~~The transaction date cannot be in the future.~~
   3. ~~The transaction notes may contain commas and can be multi-line.~~
   4. ~~The share price needs to be greater than zero.~~
   5. ~~The total asset & transaction values will not be entered by the user. Instead, the program will calculate them.~~
5. ~~In order to delete a transaction, a valid transaction ID needs to be entered.~~
6. ~~The “Reload file” button should load to the grid view the transactions found in the current record file in case the user suspects the data currently displayed in the grid view is stale in comparison to what’s stored in the file.~~
7. ~~The “Exit” button exits the program. The “Empty record file” button removes all transaction entries from the record file.~~



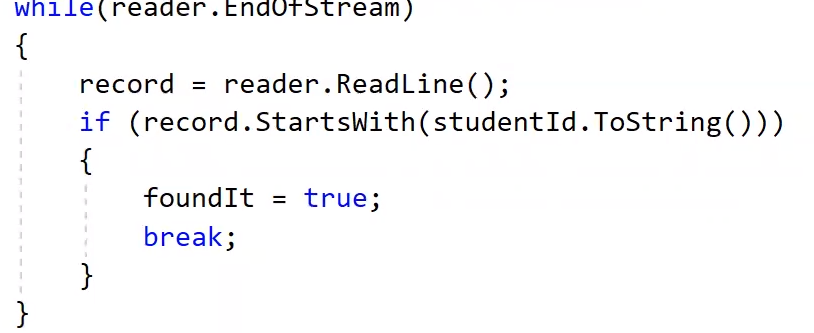
## Hand In

1. Bundle and upload your solution folder to the assignment drop-box on eConestoga as a ***single*** zip file.

~~Total, total -> doesn’t get to be saved in the file (check)~~

~~Don’t store dollar sign~~

~~Check tap~~



Fixed:

Date format on the grid view (mm/dd/yyyy -> yyyy-mm-dd)

Validation for numbers (error message for digits and required together -> separate)

Total asset & total transaction saved to record -> only appears on the gride view (not saved on the file)

$ sign stored on the file -> only appears on the grid view (not saved on the file)

Directory creation fixed

Variable name change (container -> records)

Deleting & updating a record (File.Delete + append: true -> append: false)

When Add or update (transaction fields left not cleared -> get cleared)