**Backend Exercise: (CSVs included in Google Drive folder)**

The included CSV file (videos.csv) contains data about different videos. Your job is to write code that will:

1. Load in videos.csv.
2. Analyze the data against the specified rules (listed below).
3. Output the results into two files: valid.csv will contain a list of video ids's that passed the tests and invalid.csv will contain a list of video id's that failed the tests.
4. Output (in the terminal) the total amount of money (in USD) made by all the videos in valid.csv.

For a video to be in valid.csv, it must meet the following rules:

* The video title must be shorter than 30 characters.
* The video must have over 10 likes.
* The video must have over 200 sales.
* The video price must be under 20 Euros or under 25 Canadian Dollars. (videos.csv only lists prices in USD, so we've also provided exchange\_rates.csv, which includes exchange rates from USD to other currencies.)

Rules and guidelines

* You may use any programming language you would like, but you can only use standard libraries. (We should NOT have to install anything other than the language to make your code run.)
* All errors should be handled gracefully--any exceptions that may be thrown based on what rules or files are specified on the command line should be handled appropriately.
* Your code should be sufficiently generic that it would be easy to add additional filtering rules.
* Your implementation should demonstrate your object oriented programming skills and ability to define reusable components.
* Memory usage is an important factor of a well implemented solution to this problem. The lists the program will process may be much larger than the sample provided with the test.
* You have as much time as you need to complete this test. We estimate it will take you around two hours.
* Please do **not** make this code test publicly available on GitHub or Bitbucket.

What to submit

* All source code needed to run your solution (but no compiled binaries).
* A README with instructions on how to compile (if necessary) and run your code (what the command line arguments are, etc.), as well as answers to the following questions:
* Was the question/problem clear? Did you feel like something was missing or not explained correctly?
* What makes your solution awesome?
* Did you have to make any trade-offs in your design? If so, what?
* Is there anything else you want to share about your solution or the problem?