SteelEye Data Python Engineer Assessment Test

Thank you for your interest in the Data Python Engineer teams at SteelEye . As you may be aware SteelEye have 3 core Data teams.

- Data Intelligence
- Data Integration
- Data Platform

We use this exercise to help determine which team you will be a good fit for, and also at what level your skills match within our team.

We hope the test will not only test your abilities but also give you a little glimpse into some of the tasks we carry out. You are welcome to use any resources that would be available to you during the course of a normal day: e.g. Google is permitted, as are textbooks. You may include code developed elsewhere so long as they are licensed for commercial use without restriction. You should clearly identify and attribute sources within your code to their appropriate ownership.

One final word of warning - at a startup like SteelEye you will regularly gather your own requirements directly or receive vague details by email. Sometimes it is not possible to get a very detailed specification simply because the user doesn't know or isn't available. When this happens you have to use your initiative and work things out for yourself. This test has ambiguities and vague definitions. It is up to you to make reasonable and valid assumptions. Do not ask your recruitment consultant for help or advice as they know no more than you do.

And one final tip: KISS – Keep It Short and Simple. Don't overcomplicate it, make it simple, efficient, easy to maintain and easy for the user to use.

Brief:

1. The requirement needs to be developed in Python 3
2. Code should follow pep8 standards and should include pydoc, logging & unit tests
3. Please provide github link for review.

Requirement:

1. Download the xml from this link
2. From the xml, please parse through to the first download link whose file_type is DLTINS and download the zip
3. Extract the xml from the zip.
4. Convert the contents of the xml into a CSV with the following header:

- FinInstrmGnlAttrbts.Id
- FinInstrmGnlAttrbts.FullNm
- FinInstrmGnlAttrbts.ClssfctnTp
- FinInstrmGnlAttrbts.CmmdtyDerivInd
- FinInstrmGnlAttrbts.NtnlCcy
- Issr

5. Store the csv from step 4) in an AWS S3 bucket
6. The above function should be run as an AWS Lambda (Optional)

Assessment criteria:

1. Percentage of requirements satisfied
2. How clean the code is - in particular simplicity, adhering to python code style conventions and error handling.
3. Follows PEP 8 guidelines
4. We expect pydoc for each class and function with optional type hints(nice to have)
5. Follows standard logging (no print statements). Logs are essential part of troubleshooting application.
6. Unit tests with good code coverage