elenjical SOLUTIONS

Technical Assignment 2024

Due Date: 5 April 2024

Introduction

This assignment aims to introduce data analysis using Python, SQL and XML. Data analysis is key in our industry – tools to extract information and debug large amounts of data is very important. Even if you do not end up in a very technical role – being able to quickly write an SQL statement or process data in a less tedious way will benefit you. Please make sure that you understand the basics at the end of the assignment.

You may need to find additional material online if you are not familiar with some of the concepts. Some of you come from technical degrees so I encourage you share your knowledge and help your fellow graduates. Please note however that you must submit your own work.

Assignment outputs

- The code for questions 1-7 should be written in Python and should compile.
- The visualizations can be done using tools/add-ins/libraries (except excel).
- Please ensure that you add the Python, SQL and XSL code when you submit.
- You should submit a zipped file containing your code, file outputs and any word/excel documents you used to answer the questions.
- Please ensure that your answers are clearly marked.
- **Due Date: 5th of April**

Python and SQL - Workflow Analysis

The data set provided for this section is extracted from some of the workflow database tables in the Murex Financial database. You don't need any expert knowledge of the workflows in Murex at this point to understand the dataset – all the critical information to complete this assignment will be outlined.

The following files containing the datasets should be used for this assignment:

- WorkflowTaskInfo.csv – provides information for each task in the workflow
- StpFcEntries.csv – contains all the entries for the financial contracts as they move through the contract workflow
- contractPaths.csv – input file for question 6

The aim of this assignment is to use Python and SQL to extract information that would typically be used in a workflow monitoring dashboard. As you can imagine, some of our clients process high volumes of trades and designing efficient workflows are critical. Having metrics readily available is a real use case to understand trade flows, bottlenecks and tasks requiring optimization.

Here is some key information to complete this section:

- *xmlflow_status* in the FC (Financial Contract) dataset joins with *filter_code* in the task data.
- Note that the tasks data has an input and output flag – this is important.
- The status taken flag in the FC data indicates if entry has been processed by a task or not
- The metrics are defined as follows:

Metric	Detail
Wait Time	Time in milliseconds an entry waits at the input node before being processed
Processing Time	Time in milliseconds an entry takes to be processed by the task
Throughput	Number of entries that a task processes per second
Latency	Time in milliseconds between an entry arriving at the input node of a task to when it arrives at the output node

Your tasks are as follows:

1. Understand and cleanup the data – explain the high-level steps you took to prepare the data.
2. Summarize the data by Family, Group and Type and output the number of contracts (not entries) that were processed by the contract workflow.
3. Output the following statistics **per task:**
   a. Wait Time Average
   b. Wait Time Max
   c. Processing Time Average
   d. Processing Time Max
   e. Throughput (per second)
   f. Throughput (per minute)
   g. Latency (per second)
4. Graphically show how many entries are waiting to be processed by each task in a bar chart.
5. Graphically display the top five tasks in terms of processing time in a line graph for the last week.
6. Write code to track the path of a contract in the workflow. The code should track the contract from the entry task to the final task and output all the tasks in the path. You should use the file contractPaths.csv as an input and provide the output:
   Contract 1|Family|Group: Task1 -> Task2 -> Task3
   Contract 2|Family|Group: Task1 -> Task4 -> Task5 -> Task6

   Write the output to a file called contractPaths_output.csv

7. Data Analysis
   a. Which are the top 3 task types that take the longest in general to process an entry – explain how you got to this answer and any assumptions?
   b. Are there any tasks with bottlenecks? Explain your reasoning.
   c. Assume that the traders want to increase the volumes of FX Spot trades – the goals is to process 100 000 trades using a batch upload.
      i. Estimate how long it might take to process all the trades in the workflow?
      ii. Explain which predictive techniques you used to predict the total processing time and how you would implement this in theory.
   d. **Challenge** – implement a prediction algorithm and produce a number.

8. Once the Murex training begins, you will be assigned Murex environments. For this question you should connect to the Murex database. Write SQL to extract:
   a. Summarize the data by Family, Group and Type and output the number of contracts (not entries) that were processed by the contract workflow.
   b. Output the following statistics **per task:**
      i. Processing Time Average
      ii. Processing Time Max
      iii. Processing Time Min
      iv. Throughput (per second)

XML - Market data

Use the futures_price.xml and yieldx_input.xml files for this section. Market data flow into most trading systems (Murex in our example) is carried using xml files. The yieldx_input.xml file contains raw futures prices quoted from the JSE's interest rate exchange (Yieldx) and this xml needs to be transformed or mapped into a valid Murex market data xml file (futures_price.xml).

The goal is to write an XSLT function that will transform the data in yieldx_input.xml into the data in futures_price.xml. Put the function in an XSL file called yieldx_transform.xsl. The function should be able to work for any date. Please take note of the following rules for the transformation:

1. Only map the following futures i.e., Contract codes:

JSE GOVI TR	JSE R2044
JSE IGOV TR	JSE R2048
JSE R186	JSE R207
JSE R2023	JSE R208
JSE R2030	JSE R209
JSE R2037	JSE R213
JSE R2040	JSE R214

2. Only use Uncalculated price
3. All futures have the market ZAR BD GOV, except for JSE GOVI TR and JSE IGOV TR which have ZA JSE as their market.