HW - Earned Value Analysis

Start Assignment

- Due Nov 24 by 11:59pm
- Points 25
- Submitting a file upload
- File Types zip

HW - Earned Value Analysis

NOTE: All calculations and analysis for this assignment must be performed using Python scripts only. Hardcoding or manual calculations will result in a 20% reduction of your score. Ensure your program gracefully handles all potential errors.

You have been provided with three CSV data files, each containing Planned Values (PV), Actual Costs (AC), and Earned Values (EV) data for a project. These files represent the weekly data collected over a 7-week duration for two different modules (Module A and Module B) and their corresponding tasks. Your task is to perform Earned Value Analysis (EVA) to evaluate the project's schedule and cost performance using Python.

The CSV files provided for your analysis:
PV_data.csv (https://canvas.asu.edu/courses/234472/files/112833201?wrap=1) ⤓ (https://canvas.asu.edu/courses/234472/files/112833201/download?download_frd=1)

AC_data.csv (https://canvas.asu.edu/courses/234472/files/112833202?wrap=1) ⤓ (https://canvas.asu.edu/courses/234472/files/112833202/download?download_frd=1)

EV_data.csv (https://canvas.asu.edu/courses/234472/files/112833204?wrap=1) ⤓ (https://canvas.asu.edu/courses/234472/files/112833204/download?download_frd=1)
Each file contains the weekly cumulative costs/values as provided.

Tasks:

(5 points) Data Import and Preparation

- Write a Python script to import the provided CSV data files into a single SQLite database (EVA_data.db).
- Clearly organize the imported data into separate tables (PV, AC, and EV) within the SQLite database.
- Provide clear comments and code readability.

(10 points) Earned Value Analysis Calculations

- Write Python code that reads data from your SQLite database and calculates the following metrics for each week and task:
  - Cost Variance (CV = EV - AC)
  - Schedule Variance (SV = EV - PV)
  - Cost Performance Index (CPI = EV/AC)
  - Schedule Performance Index (SPI = EV/PV)
  - To-Complete Performance Index (TCPI = (BAC - EV)/(BAC - AC)), assuming the Budget at Completion (BAC) is the total planned value at the end of week 7.
- Your script should store these calculated metrics clearly in a pandas DataFrame and export them into a file named EVA_Analysis.csv.

(10 points) Analysis and Interpretation Write a clearly documented markdown file to answer the following questions based on your EVA results:

A. Identify the weeks when the project was ahead of schedule (based on SPI > 1). Provide reasoning with data.

B. Identify the weeks when the project was under budget (based on CPI > 1). Provide reasoning with data.

C. At the end of week 7, summarize the project's overall schedule and cost status clearly (ahead/behind schedule, under/over budget). Explain your findings with specific SPI, CPI, CV, and SV values.

D. Assume Module A and Module B were developed by separate teams. Based on your analysis, determine if one team performed significantly better than the other in terms of schedule adherence and cost management. Justify your answer by clearly comparing relevant EVA metrics between the modules.

Hint: For this, you might want to filter your DataFrame to compare Module A vs. Module B separately.

E. Explain why, in weeks 5 and 6, you expect TCPI (To-Complete Performance Index) to be high (indicating a challenging target to meet budget at completion). Discuss in terms of AC, EV, and PV trends during these weeks.

Submission Details:

Submit a .zip file containing:

- Your well-commented Python scripts (data_import.py, eva_calculations.py, and analysis.md).
- The SQLite database file (EVA_data.db) generated by your Python script.
- CSV output file (EVA_Analysis.csv) with calculated EVA metrics clearly labeled.
- A text or markdown file (analysis.md) containing detailed answers to the interpretation questions (A-E) above.

Ensure that your code runs seamlessly from import to analysis without manual interventions.

HW Rubric - EV (1)

Criteria	Ratings			Pts
Database Creation and Data Storage (Using SQLite Database)	**5 pts** **Full Marks** Data accurately stored into SQLite DB with clear schema design, tables properly created, and all provided data correctly loaded.	**3 pts** **Novice** Data mostly accurate, minor inaccuracies in DB structure or data loading.	**0 pts** **No Marks**	5 pts
Python Script for Data Extraction (Read data from DB into Python)	**4 pts** **Full Marks** Python clearly connects, reads, and imports all data correctly from SQLite database into Pandas or similar structures.	**3 pts** **Novice** EV calculations generally accurate, minor mistakes.	**0 pts** **No Marks**	4 pts
Calculation and Analysis of EV Metrics (PV, AC, EV calculations, Cost/Schedule Variances)	**5 pts** **Full Marks** Correct calculation of all required EV metrics including PV, AC, EV, CV, SV accurately computed for each period/task.	**3 pts** **Novice** EV calculations generally accurate, minor mistakes.	**0 pts** **No Marks**	5 pts
Analytical Responses (Answers to analysis questions)	**5 pts** **Full Marks** All analytical questions answered accurately (ahead/below schedule, cost comparisons, team performance comparison, TCPI analysis) supported clearly by calculated data.	**3 pts** **Novice** Most questions answered with minor inaccuracies or incomplete justification.	**0 pts** **No Marks**	5 pts
Python Code Clarity & Readability	**3 pts** **Full Marks**	**1.5 pts** **Half Marks**		3 pts

Criteria	Ratings		Pts
(Quality and commenting of code)	Code clearly structured, well-commented, readable, and maintainable with logical function breakdown.	Code somewhat structured, limited comments, readability could be improved.	
Final Output Generation and Submission (Correct files in submission, output CSV, databases, scripts)	**3 pts** **Full Marks** Submission clearly includes Python scripts, DB files, output CSV, and comprehensive screenshot of terminal/output.	**1.5 pts** **Novice** Submission has minor missing files or unclear results.	3 pts
			Total Points: 25