|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FinTech Unit 7 SQL Homework Grading Rubric** | | | | |
| **Criteria** | **Ratings** | | | |
| **Data Modeling** • Database model defined  • PostgreSQL database created using defined model | **20 Points Mastery** • Completed 2 out of 2 requirements • Code runs without error and produces the assigned results • Code accounts for all possible scenario  • Code is free of bugs | **19 > 15 Points Approaching Mastery** • Completed 1 out of 2 of requirements • Code runs without error • Code produces results as expected 80% of the time | **15 > 13 Points Progressing** • Completed fewer than 1 out of 2 requirements • Code runs without error  • Code produces results, but not necessarily the correct results | **13 > 0 Emerging** • Completed 0 out of 2 requirements • No submission • Code runs with error |
| **Data Engineering** • Database schema for each table and relationships. • Data types specified • Primary keys • Foreign Keys | **20 Points Mastery** • Completed 4 out of 4 requirements • Code runs without error and produces the assigned results • Code accounts for all possible scenario  • Code is free of bugs | **19 > 15 Points Approaching Mastery** • Completed 3 out of 4 of requirements • Code runs without error • Code produces results as expected 80% of the time | **15 > 13 Points Progressing** • Completed 2 out of 4 requirements • Code runs without error  • Code produces results, but not necessarily the correct results | **13 > 0 Emerging** • Completed 1 or none out of the 4 requirements • No submission • Code runs with error |
| **Data Analysis** • Fraudulent transactions identified. • SQL and Pandas DataFrames utilized for report within Jupyter Notebook.  • Visual data analysis of fraudulent transactions using Pandas, Plotly Express, hvPlot, and SQLAlchemy to create the visualizations. | **30 Points Mastery** • Completed 3 out of 3 requirements • Code runs without error and produces the assigned results • Code accounts for all possible scenario  • Code is free of bugs | **29 > 25 Points Approaching Mastery** • Completed 2 out of 3 of requirements • Code runs without error • Code produces results as expected 80% of the time | **25 > 20 Points Progressing** • Completed 1 out of 3 requirements • Code runs without error  • Code produces results, but not necessarily the correct results | **20 > 0 Emerging** • Completed 0 out of the 3 requirements • No submission • Code runs with error |
| **Coding Conventions/Formatting**  • Appropriate header, name, short description at top of the notebook  • Imports are at the top of the file, just after any headers or subheads.  • Files read in from relative file path  • Functions and variable names are descriptive, lowercase, with words separated by underscores  • Clean code, no repetition, maintainable and highly reusable code.  • Appropriate code wrapping and cell sizes  • Appropriate subheads as needed | **10 Points Mastery** | **8 Points - Approaching Mastery** | **5 Points - Progressing** | **0 Points - Emerging** |
| **Deployment/Submission**  • Files submitted in personal repo • Appropriate directory structure with correct files needed to run scripts  • Appropriate commit messages  • Appropriate README | **10 Points Mastery** | **8 Points - Approaching Mastery** | **5 Points - Progressing** | **0 Points - Emerging** |
| **Documentation/Comments**  • Code is well commented with concise, relevant comments | **10 Points Mastery** | **8 Points - Approaching Mastery** | **5 Points - Progressing** | **0 Points - Emerging** |