Team Assignment Rubrics

BA780

Due: Sep 30 @ 11:59 PM

This assignment is to create a report of your project's dataset in Python. You must clean and describe your data and provide a helpful summary and simple charts.

You must submit your Team Assignment notebook to me in the team's Slack channel.

What to submit

- A properly named notebook. E.g., B02-Project-Title.ipynb. ("-" = hyphen)
    - B02 represents team 2 from Cobort B.
    - The project title should be hyphenated and capitalized
- The business problem definition + a brief description of your data and its source.
    - This should articulate why the problem is important.
- **Cleaning** phase
- Exploratory **questions** about your data and **answers** to them:
    - You can ask as many questions as you wish with no upper limit. The more thoughtful your questions, the stronger your analysis.
    - Each answer should be supported by a chart or summary table (sometimes both — please note that you have a limit on the number of charts — see the point below).
    - Questions can range from straightforward ("What is the average trip length?") to comparative or hypothesis-driven ("Do weekends show higher average revenue per trip than weekdays?").
    - **Note:** This phase (Team Assignment) should account only for about 50–60% of the total project. The remaining work and analysis should be completed for the final deliverable phase.
- **Five charts** for this phase.
    - Charts should be sized appropriately, axes should be labeled, and legends should be legible.
- Include a summary of your critical observations after each section, and if applicable sub-sections (e.g., after Cleaning, Exploratory Questions, Visualizations, etc.). These should be 1–3 sentences highlighting the main takeaways from that section. Think of them as "mini-conclusions" that make the notebook well-organized and easy to navigate.
- Include a brief **executive summary** of your key observations at the top of the notebook.
    - It appears at the beginning of the report and provides a concise overview of the entire document. It includes key findings, objectives, methods, and major recommendations. The goal is to give decision-makers an understanding of the

most critical information without reading the full report. It is forward-looking, often helping executives quickly grasp the main insights for strategic action.

- Include a **conclusion** at the end of your report.
    - It appears at the end of the report and summarizes the analysis's results. It focuses on interpreting the findings, reflecting on the analysis's outcomes, and reaffirming the recommendations or next steps based on the insights. It closes the discussion by recapping the most important data-driven points, typically informed by the body of the report.
- Any output from executing code in cells should be visible.
- References.
- Generative AI Disclosure: any use of generative AI tools, including for coding assistance/debugging or other purposes, should be disclosed in a statement after the references. How was the AI tool used, and which parts of the notebook? (see below)

Quick submission checklist

- ☐ Filename and header cell correct (team ID, data source, license, access).
- ☐ Executive Summary (4–6 sentences) at the top.
- ☐ Cleaning steps logged with before/after evidence; notebook runs Restart & Run All without errors.
- ☐ 5 labeled, legible charts; each chart answers a stated question.
- ☐ Each section ends with a one-line takeaway.
- ☐ Conclusions list actions, risks/limits, and next steps.
- ☐ References + Generative AI disclosure included.
- ☐ Pushed to Slack as instructed; code outputs visible (no empty cells).

Rubric for Team Assignment:

Category	Wt	Excellent (A / A-); what this *looks like*	Concrete examples
1) Title, data source, filename	1	The file is named exactly `B02-Project-Title.ipynb`; cohort, team, data source, license/terms, and data access instructions.	Filename: `B02-Bike-Sales-Trends.ipynb`. "Data: NYC Citi Bike (2024 Q1), License: CC BY 4.0, Access: `https://…` (CSV, 120MB)."
2) Problem definition (business value)	1	One crisp paragraph that states the business decision, who cares, and success criteria (metric or tangible output).	This project aims to evaluate the distribution, accessibility, and effectiveness of income-restricted housing units across Boston in the context of the broader housing market. We will: ● Identify areas with the highest concentration of income-restricted housing units. ● Compare the trends in income-restricted housing with the general housing market in Boston. ● Examine the impact of various funding sources on the availability of these units. ● Investigate potential disparities in the distribution of these units concerning neighborhood demographics and general housing market trends.
3) Storyline & cohesiveness	2	Clear narrative from problem → data → cleaning → EDA → insights → implications. Includes a 4–6 sentence **Executive Summary** up top and **section-end takeaways**.	Exec summary example (4–6 sentences) and a one-line "So what?" at end of each section: "So what: missingness clusters in payment_type; we standardize before comparisons."
4) Data cleaning	2	Systematic, *reproducible* steps with rationale: types, outliers, missingness, dedupe, joins; each step justified; before or after summaries; data dictionary snippet.	Show `df.info()` before/after; table: "Column, Issue, Action, Justification." E.g., "`age` outliers >100 set to NA (input error); 0.6% rows affected."

5) Python mastery & data visualization	3	Idiomatic, readable code; functions where useful; no silent errors; **5 charts** sized well, labeled axes/units, legible legends, correct encoding; basic stat summaries.	Chart set: histogram of target, bar chart by category, time series, boxplot for distribution, heatmap of correlations. Each has `title`, axis labels, and 1-sentence insight.
6) Notebook organization & Markdown	3	Polished notebook: section headers, short paragraphs, bullets, callouts; hides long dumps; uses code/markdown balance; table renders are readable; links to sources; **reproducible** (runs top-to-bottom), seed set, path-agnostic.	Use this notebook as an example of a well-organized notebook from the previous years. "Repro cell": `!pip freeze > requirements.txt` (optional), `np.random.seed(42)`, tiny "Run me first" setup cell, relative paths, final "Restart & Run All" sanity check note.
7) Quality of questions	2	Questions are specific, hypothesis-driven, and tied to stakeholders; each question maps to a chart/table.	"Q: Do weekend riders generate higher avg revenue per trip than weekdays after controlling for trip length?"
8) Quality of answers	2	Answers are *evidence-based*, cite exact numbers/uncertainty, and include a short interpretation and caveat/limitation.	"A: Yes. Weekend average revenue per trip is $4.10 vs $3.65 weekdays (+12.3%, 95% CI [10.8%, 13.7%]); effect shrinks to +6.1% when controlling for length."
9) Conclusions	3	3–5 prioritized insights linked to business actions, risks/limits, and **next steps** (what to collect/try next).	Bullets labeled Impact/Feasibility. "1) Promote weekend bundles (High impact/Low effort). Risk: seasonal bias; Next: A/B in Oct."
10) Generative AI disclosure	1	Clear, specific, truthful: tools, where used, how validated; no blanket statements.	Example 1 (coding + writing): We used ChatGPT to generate an initial regex for date parsing and to draft an outline for the Executive Summary. All code was tested against the dataset, with corrections made by the team. The summary was edited for clarity and accuracy, ensuring no

			AI-generated text remained unverified. We did not use AI for data analysis or visualization. Example 2 (EDA brainstorming): We used Claude to brainstorm exploratory data analysis (EDA) questions at the start of the project. The suggestions were used to shape our section headers. The actual questions, code, and visuals were created by the team and verified against the dataset. This use saved time in structuring, but all analysis remains original. Example 3 (visualization polish): We used GitHub Copilot for Pandas syntax suggestions when chaining cleaning steps and ChatGPT to refine the labeling of two charts. All generated code was tested by running the notebook top-to-bottom, and all text was edited to align with our team's findings. AI was not used for generating conclusions.
Total	**20**		