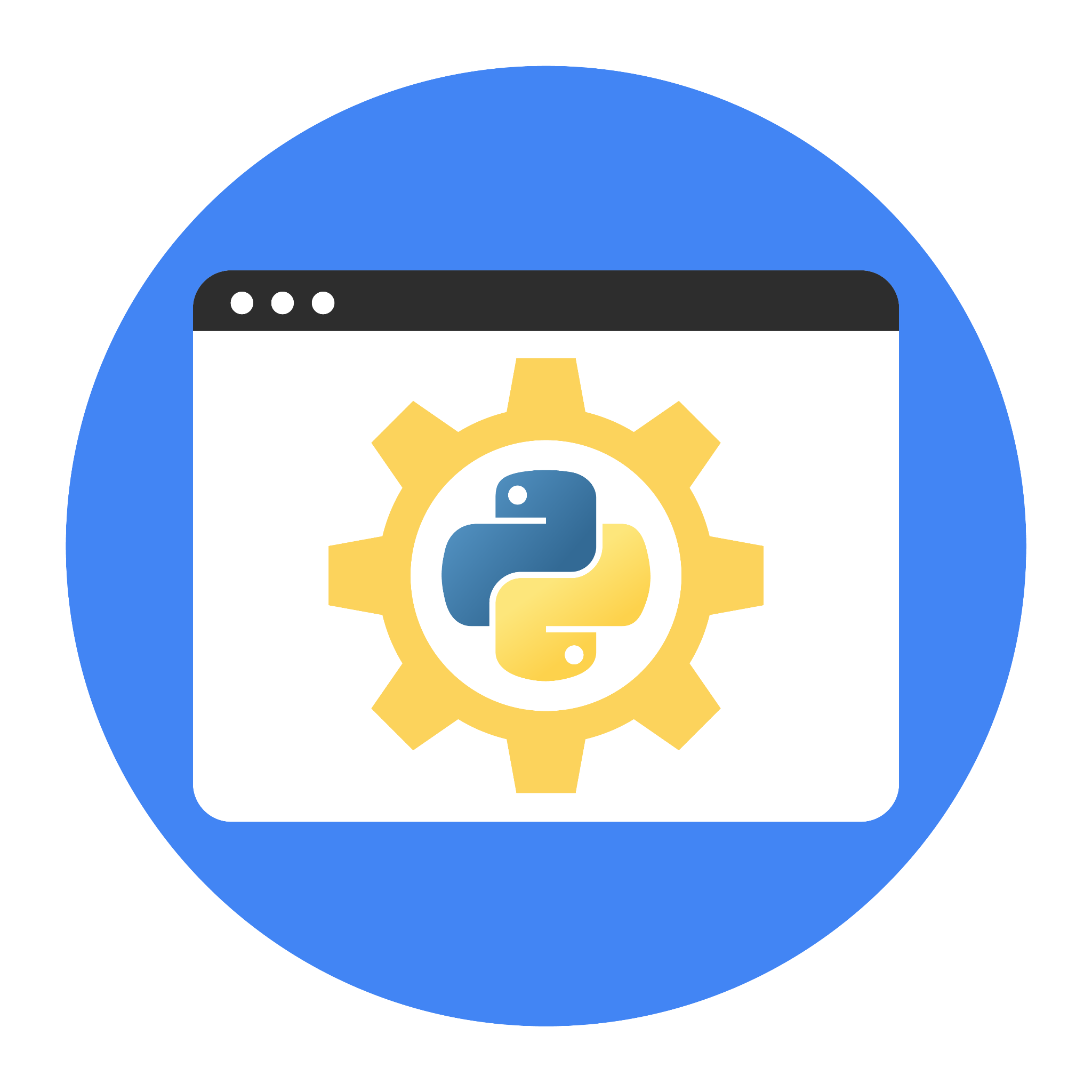
**Course Two**

# Get Started with Python



# Instructions

Use this PACE strategy document to record decisions and reflections as you work through this end-of-course project. You can use this document as a guide to consider your responses and reflections at different stages of the data analytical process. Additionally, the PACE strategy documents can be used as a resource when working on future projects.

# Course Project Recap

Regardless of which track you have chosen to complete, your goals for this project are:

* ~~Complete the questions in the Course 2 PACE strategy document~~
* ~~Answer the questions in the Jupyter notebook project file~~
* ~~Complete coding prep work on project’s Jupyter notebook~~
* ~~Summarize the column Dtypes~~
* ~~Communicate important findings in the form of an executive summary~~

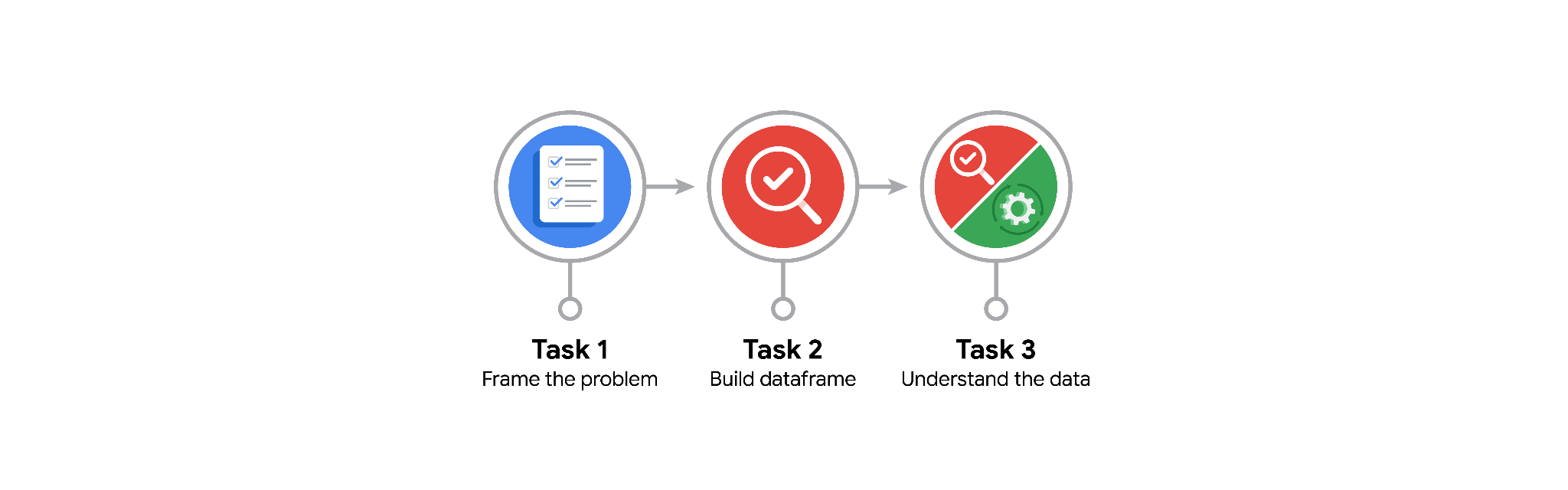
# Relevant Interview Questions

Completing the end-of-course project will help you respond these types of questions that are often asked during the interview process:

* Describe the steps you would take to clean and transform an unstructured data set.
* What specific things might you look for as part of your cleaning process?
* What are some of the outliers, anomalies, or unusual things you might look for in the data cleaning process that might impact analyses or ability to create insights?

**Reference Guide**

This project has three tasks; the visual below identifies how the stages of PACE are incorporated across those tasks.



**Data Project Questions & Considerations**

**PACE: Plan Stage**

* How can you best prepare to understand and organize the provided information?

To best prepare, I first reviewed the project scenario, team expectations, and the data dictionary describing each column. I also explored the dataset structure using tools like df.info(), df.describe(), and df.head() to understand column types, missing values, and potential anomalies. Organizing the data involved identifying which variables are relevant for predictive modeling, converting datetime fields to proper formats, and noting which columns require cleaning or transformation. This preparation ensures the dataset is structured, consistent, and ready for deeper analysis or modeling.

* What follow-along and self-review codebooks will help you perform this work?

The follow-along labs and practice notebooks from the Python course were especially useful. They helped reinforce key concepts like loading data with Pandas, exploring datasets using .info() and .describe(), handling missing data, and grouping by variables. I also referred to earlier exercises that showed how to filter data, calculate summary statistics, and sort values — all of which were essential for analyzing variables like trip\_distance, payment\_type, and tip\_amount.

* What are some additional activities a resourceful learner would perform before starting to code?

A resourceful learner would begin by carefully reading the project brief and understanding the end goals and audience. They would examine the data dictionary to clarify each column’s meaning and identify which variables are most relevant. Before coding, they might sketch out a plan for data cleaning and transformation, look up examples of similar regression tasks, and review any anomalies or edge cases in the dataset. They might also check for prior feedback or common pitfalls in similar projects and ensure their Python environment is set up with all necessary packages.

**PACE: Analyze Stage**

* Will the available information be sufficient to achieve the goal based on your intuition and the analysis of the variables?

Yes, the dataset provides a wide range of relevant variables such as trip\_distance, fare\_amount, payment\_type, and tip\_amount, which are useful for understanding ride costs and predicting fares. While some fields (like tip\_amount) are incomplete for cash payments, the majority of the data is complete and structured. As long as we handle anomalies and outliers, the dataset is sufficient to begin developing a predictive model for estimating taxi fares.

* How would you build summary dataframe statistics and assess the min and max range of the data?

I would use the .describe() method in Pandas to generate summary statistics for all numeric variables. This includes the count, mean, standard deviation, minimum, 25th percentile, median (50%), 75th percentile, and maximum values. This helps identify the normal range of data, detect outliers (such as negative fares or unusually high tips), and understand data distribution. For specific columns like trip\_distance or total\_amount, I would sort the DataFrame to directly inspect the rows with the highest and lowest values and verify if they are valid or errors.

* Do the averages of any of the data variables look unusual? Can you describe the interval data?

Yes, a few averages stand out as potentially unusual. For instance, the average fare\_amount is around $13.03, which is reasonable, but there are some extreme values like -$120 and $999.99 that skew the range. The average tip\_amount is about $1.84, but this is only recorded for credit card transactions, making it incomplete overall. The trip\_distance average is around 2.91 miles, which seems typical for city rides. Most numeric columns in the dataset represent **interval data**, meaning they have measurable differences between values but no true zero — such as fare, distance, and time-based charges — which are appropriate for regression analysis.

**PACE: Construct Stage**

**Note**: The Construct stage does not apply to this workflow. The PACE framework can be adapted to fit the specific requirements of any project.

**PACE: Execute Stage**

* Given your current knowledge of the data, what would you initially recommend to your manager to investigate further prior to performing exploratory data analysis?

I would recommend investigating records with negative or extremely high values in fare\_amount, total\_amount, and tip\_amount to determine if they are data entry errors, canceled trips, or special cases. Additionally, I’d suggest converting the datetime columns to proper datetime objects for time-based analysis in future steps. Finally, I’d recommend reviewing outliers in RatecodeID (e.g., values like 99) to see if they are valid or misclassified

* What data initially presents as containing anomalies?

Several variables contain anomalies. For example, fare\_amount and total\_amount have negative values, which are not realistic for taxi fares. There are also trips with trip\_distance equal to zero but non-zero fares, which may represent errors or cancellations. The tip\_amount field has unusually high values (up to $200), which could be incorrect or unrepresentative outliers. Additionally, RatecodeID includes values like 99, which fall outside the documented valid range (1–6), suggesting possible data entry issues.

* What additional types of data could strengthen this dataset?

Including weather conditions, traffic data, or time-of-day traffic congestion levels could help explain fare variability and better model total charges. Adding detailed geographic data (such as pickup/dropoff neighborhood names) would improve interpretability beyond numeric location IDs. Customer ratings or ride duration in minutes could also offer valuable context. Lastly, including cash tip estimates, if available, would provide a more complete picture of total trip cost and tipping behavior.