Data Engineer Certification Study Guide

Please use this study guide to create your certification self-study plan. We've included the objectives you should meet for each assessed competency, with links to relevant practice assessments.

- **Associate Certification**
  - Exams DE101 and DE102

Associate

Exam DE101: Data Management Theory & SQL and Exploratory Analysis Theory

1.1 Perform data extraction, joining and aggregation tasks (SQL)

- Aggregate numeric, categorical variables and dates by groups using PostgreSQL.
- Interpret a database schema and combine multiple tables by rows or columns using PostgreSQL.
- Extract data based on different conditions using PostgreSQL.
- Use subqueries to reference a second table (e.g. a different table, an aggregated table) within a query in PostgreSQL

1.2 Perform cleaning tasks to prepare data for analysis (SQL)

- Match strings in a dataset with specific patterns.
- Convert values between data types.
- Clean categorical and text data by manipulating strings.
- Clean date and time data.

1.3 Assess data quality and perform validation tasks (SQL)

- Identify and replace missing values.
- Perform different types of data validation tasks (e.g. consistency, constraints, range validation, uniqueness).
- Identify and validate data types in a data set.

Related Assessments

Data Management with SQL

2.1 Interpret a database schema and explain database design concepts (such as normalization, design, schemas, data storage options)

- Explain the design schema of a database
- Identify from a schema how tables are connected and how to join multiple tables
- Explain concepts in database design (normalization, design schemas, data storage options, etc)

2.2 Identify different cloud tools that can be used for storing data and creating and maintaining data pipelines

- Identify the most common cloud tools used for data storage (file storage and databases)
- Identify the most common cloud tools used for creating and managing data pipelines

Related Assessments

Not yet available

3.1 Use data visualization tools to demonstrate characteristics of data (theory)

- Distinguish between different types of data visualizations (bar chart, box plot, line graph, and histogram) in demonstrating the characteristics of data.
- Interpret data visualizations (bar chart, box plot, line graph, and histogram) and summarize the characteristics of the data.

3.2 Read and analyze data visualizations to represent the relationships between features (theory)

- Distinguish between different types of data visualizations (scatterplot, heatmap, and pivot table) in representing the relationships between features.
- Interpret the data visualizations (scatterplot, heatmap, and pivot table) and summarize the relationship between features.

Related Assessments

Exploratory Analysis Theory

Exam DE102: Data Management and Programming in Python

1.1 Perform standard data import, joining and aggregation tasks using Python

- Import data from flat files into Python.

- Import data from databases into Python
- Aggregate numeric, categorical variables and dates by groups using Python.
- Combine multiple tables by rows or columns using Python.
- Filter data based on different criteria using Python.

1.2 Perform cleaning tasks to prepare data for analysis (Python)

- Match strings in a dataset with specific patterns.
- Convert values between data types.
- Clean categorical and text data by manipulating strings.
- Clean date and time data.

1.3 Assess data quality and perform validation tasks (Python)

- Identify and replace missing values.
- Perform different types of data validation tasks (e.g. consistency, constraints, range validation, uniqueness).
- Identify and validate data types in a data set.

1.4 Collect data from non-standard formats (e.g. json) by modifying existing code (Python)

- Adapt provided code to import data from an API using Python.
- Identify the structure of HTML and JSON data and parse them into a usable format for data processing and analysis using Python.

Related Assessments

Importing and Cleaning with Python

2.1 Use common programming constructs to write repeatable production quality code for analysis.

- Define, write and execute functions in Python.
- Use and write control flow statements in Python.
- Use and write loops and iterations in Python.

2.2 Demonstrates best practices in production code including version control, testing, and package development.

- Describe the basic flow and structures of package development in Python.
- Explain how to document code in packages, or modules in Python.
- Explain the importance of the testing and write testing statements in Python.
- Explain the importance of version control and describe key concepts of versioning

Data Engineer Certification Study Guide

2.3 Demonstrates software engineering principles (OOP, profiling, debugging) to write efficient, modular code in Python

- Use object-oriented programming principles to create basic classes and methods
- Identify inefficient or memory/CPU intensive code and be able to suggest approaches to improving efficiency and balancing requirements
- Identify common coding errors and adapt code to remove errors

Related Assessments

Python Programming