Your grade: 90%

Your latest: **90%** • Your highest: **90%** • To pass you need at least 70%. We keep your highest score.

Next item

1. What is the recommended data storage solution for an online retail company that does not want to use cloud platforms? **1 / 1 point**

- ○ Cloud-based storage
- ○ A NoSQL database
- ○ An external hard drive
- ◉ An on-premises SQL database

Correct
Correct! In the absence of cloud storage, the company should establish an on-premises SQL database to make data accessible to their employees.

2. Which table in the proposed data schema for the online retail store contains the information about the available quantity of items? **1 / 1 point**

- ○ Customer table
- ○ Transaction Invoice table
- ◉ Inventory table
- ○ None of the above

Correct
Correct! The Inventory table contains information about the available quantity of items.

3. In the proposed data schema for the online retail store, which data type is used for the "UnitPrice" column in the Inventory table? **1 / 1 point**

- ○ DATE
- ◉ DECIMAL(10, 2)
- ○ VARCHAR(20)
- ○ INT

Correct
Correct! The Unit price is a floating value with 2 significant decimal places.

4. What infrastructure requirement is necessary for the Data Integration stage of the data workflow based on the proposed data architecture? **0 / 1 point**

- ◉ Data visualization software
- ○ Data connectors

- Data processing frameworks
- Business intelligence tools

Incorrect
Incorrect. Please refer to the Infrastructure requirements section prompt in the final project lab.

5. What is the correct method to read CSV data from a URL in Python using Pandas for the transactional data table, assuming that pandas are imported in the code as 'pd'? **1 / 1 point**

- import_csv(url)
- read_data(url)
- read_csv(url)
- (•) pd.read_csv(url)

Correct
Correct! This is the appropriate method of reading a CSV file available at a URL.

6. Which condition is used to filter out entries in the transactional data table based on the StockCode column in Python? **1 / 1 point**

- Remove entries where StockCode starts with the character 'C'
- Remove entries where StockCode is 'F'
- Remove entries where StockCode is missing
- (•) Remove entries where StockCode is 'M'

Correct
Correct! The entries where StockCode is 'M', 'D', 'C2' or 'POST' were to be removed.

7. What method is used to load the final data into an SQLite database as a table named 'Purchase_transactions' in Python? **1 / 1 point**

- save_to_sqlite('Invoice_Records.db', 'Purchase_transactions')
- (•) data.to_sql('Purchase_transactions', conn, if_exists='replace', index=False)
- to_sql('Invoice_Records.db', 'Purchase_transactions')
- load_data('Invoice_Records.db', 'Purchase_transactions')

Correct
Correct! This is the appropriate format to load data to an SQL database table.

8. What SQL query is used to extract all records from the 'Purchase_transactions' table in the 'Invoice_Records' database where the Country parameter is set to Germany? **1 / 1 point**

- GET * FROM Purchase_transactions WHERE Country = 'Germany'
- SELECT * FROM Purchase_transactions WHERE Country == 'Germany'
- (•) SELECT * FROM Purchase_transactions WHERE Country = 'Germany'
- SEARCH * FROM Purchase_transactions WHERE Country = 'Germany'

Correct
Correct! This is the appropriately framed query for the database to provide the requested response.

9. What is the purpose of grouping records by InvoiceNo and Description with total quantities before applying the Apriori algorithm in Python? 1 / 1 point

- (●) To identify unique items in each invoice
- ( ) To prepare the data for one-hot encoding
- ( ) To extract association rules based on item quantities
- ( ) To calculate the total revenue for each invoice

Correct
Correct! Grouping the information based on InvoiceNo and Description will identify unique items in each unique invoice number.

10. What method is used to apply one-hot encoding on the grouped data table before performing the Apriori algorithm in Python? 1 / 1 point

- (●) applymap(lambda x: True if x > 0 else False)
- ( ) transform_data(lambda x: True if x > 0 else False)
- ( ) apply_encoding(lambda x: True if x > 0 else False)
- ( ) encode_data(lambda x: True if x > 0 else False)

Correct
Correct! This is the appropriate method to assign one-hot encoding to the grouped data.