New Snowflake DSA-C03 Exam Notes & Latest DSA-C03 Exam Discount

BTW, DOWNLOAD part of Actual4Cert DSA-C03 dumps from Cloud Storage: https://drive.google.com/open?id=1s218Fy4vtQDD3hEmbvU3425fV-UJW2jB

The SnowPro Advanced: Data Scientist Certification Exam certification provides beginners and professionals with multiple great career opportunities. The Snowflake Exam DSA-C03 examination is one of the most demanding Snowflake tests. There are multiple benefits you can get after cracking the DSA-C03 test. The top-listed benefits include skill verification, high-paying jobs, bonuses, and promotions in your current organizations. All these benefits of earning the DSA-C03 certificate help you level up your career in the tech sector.

Our DSA-C03 Exams preparation software allows you to do self-assessment. If you have prepared for the DSA-C03 exam, you will be able to assess your preparation with our preparation software. The software provides you the real feel of an exam, and it will ensure 100% success rate as well. You can test your skills in real exam like environment. If you are not getting the desired results, you will get 100% money back guarantee on all of our exam products.

>> New Snowflake DSA-C03 Exam Notes <<

Authentic Snowflake DSA-C03 PDF Dumps - Get Outstanding Results In Exam

When preparing to take the SnowPro Advanced: Data Scientist Certification Exam (DSA-C03) exam dumps, knowing where to start can be a little frustrating, but with Actual4Cert Snowflake DSA-C03 practice questions, you will feel fully prepared. Using our Snowflake DSA-C03 practice test software, you can prepare for the increased difficulty on Snowflake DSA-C03 Exam day. Plus, we have various question types and difficulty levels so that you can tailor your SnowPro Advanced: Data Scientist Certification Exam (DSA-C03) exam dumps preparation to your requirements.

Snowflake SnowPro Advanced: Data Scientist Certification Exam Sample Questions (Q157-Q162):

NEW QUESTION # 157
You are working with a large sales transaction dataset in Snowflake, stored in a table named 'SALES DATA'. This table contains columns such as 'TRANSACTION_ID (unique identifier), 'CUSTOMER_ID', 'PRODUCT_ID, 'TRANSACTION_DATE' , and 'AMOUNT'. Due to a system error, some transactions were duplicated in the table. Your goal is to remove these duplicates efficiently using Snowpark for Python. You want to use the 'window.partitionBy()' and functions. Which of the following code snippets correctly removes duplicates based on all columns, while also creating a new column 'ROW NUM' to indicate the row number within each partition?

- A.
- B.
- C.
- D.
- E.

Answer: E

Explanation:
Option A is the correct answer because it correctly partitions the data by all columns using 'sales_df.columns' within the function. It then assigns a row number within each partition using Finally, it filters the data to keep only the first row (ROW_NUM = 1) within each partition, effectively removing duplicates. The removes the temporary column and saves the unique data to a new table. Option B is incorrect because it uses 'orderBy' instead of 'partitionBy' , which does not group identical rows together for duplicate removal. Option C is incorrect because it uses ' F.rank()' instead of 'rank()' assigns the same rank to identical rows within a partition, potentially keeping more than one duplicate. Option D is incorrect because unpacking the dataframe column in partitionby using sales_df.columns causes TypeError: Column is not iterable. Option E is incorrect because passing the entire sales_df to partitionBy is not valid.

NEW QUESTION # 158
You are training a regression model to predict house prices using a Snowflake dataset. The dataset contains various features, including 'number of_bedrooms', , and You want to use time-based partitioning for your training, validation, and holdout sets. However, you also need to ensure that the dataset is properly shuffled within each time partition to mitigate potential bias introduced by the order of data entry. Which of the following strategies is MOST EFFECTIVE and EFFICIENT for partitioning your data into train, validation, and holdout sets in Snowflake, while also ensuring random shuffling within each partition, and addressing potential data leakage issues?

- A. Create separate views for train, validation, and holdout sets, filtering by 'sale_date' . Shuffle the entire dataset using 'ORDER BY RANDOM()' before creating the views to ensure randomness across all sets. This does not address shuffling within parition.
- B. Create a new column 'split_group' using a CASE statement based on 'sale_date' to assign each row to 'train', 'validation', or 'holdout'. Then, create temporary tables for each split using 'CREATE TABLE AS SELECT FROM WHERE split_group = ORDER BY RANDOM()'. This can be very slow because of global RANDOM sort and leakage issues with using full dataset for randomness.
- C. Create a user-defined function (UDF) in Python that takes a 'sale_date' as input and returns either 'train', 'validation', or 'holdout' based on pre-defined date ranges. Apply this UDF to each row, creating a 'split_group' column. Then, create temporary tables for each split using 'CREATE TABLE AS SELECT ... FROM . WHERE split_group = ... ORDER BY RANDOM()'. UDF overhead and global RANDOM sort make it very slow.
- D. Create a new column 'split_group' using a CASE statement based on 'sale_date' to assign each row to 'train', 'validation', or 'holdout'. Calculate a random number within each 'split_group' by using OVER (PARTITION BY split_group ORDER BY RANDOM())'. Then create temporary tables for each split using 'CREATE TABLE AS SELECT FROM WHERE split_group = QUALIFY ROW NUMBER() OVER (ORDER BY RANDOM()) (SELECT COUNT( ) FROM transactions WHERE split_group -- ...) (respective split percentage);'
- E. Use Snowflake's SAMPLE clause with a 'REPEATABLE seed for each split (train, validation, holdout), filtering by 'sale_date'. Add an 'ORDER BY RANDOM()' clause within each 'SAMPLE query to shuffle the data within each split. This approach does not guarantee non-overlapping sets and can introduce sampling bias.

Answer: D

Explanation:
Option E is the most effective and efficient because it correctly implements the required partitioning and shuffling while minimizing data leakage and maximizing performance. Here's a breakdown: Time-Based Partitioning: The CASE statement accurately divides the data into train, validation, and holdout sets based on 'sale_date' . Random Shuffling Within Partitions: 'ROW NUMBER() OVER

(PARTITION BY split_group ORDER BY RANDOM())' calculates a random row number within each split group (train, validation, holdout). This ensures that the data is shuffled within each time-based partition, mitigating bias introduced by the order of data entry, without introducing data leakage. Prevents Data Leakage: Shuffling the data within each partition prevents data leakage that could occur if you shuffle the entire dataset before partitioning. Efficiency: Avoids expensive operations like UDFs or sorting the entire dataset.lJses window functions efficiently to calculate random row numbers within partitions. Option A is not suitable since It does not address shuffling within parition and the shuffle will be affected by other filtering operations later.Option B is not suitable because RANDOM does not work inside create table and if it did it will cause data leakage, because all splits influence the randomness. Option C is not ideal because SAMPLE does not guarantee non-overlapping data, which would undermine the integrity of train/validation/holdout sets, moreover 'order by random()' will only apply the sampling on a sorted result not generate a random sampling. Option D is not suitable because it uses UDFs. UDFs in Snowflake generally have performance overhead compared to native SQL functions. Also using a global 'ORDER BY can be very slow on large datasets and will also introduce data leakage.

NEW QUESTION # 159
You are using the Snowflake Python connector from within a Jupyter Notebook running in VS Code to train a model. You have a Snowflake table named 'CUSTOMER DATA' with columns 'ID', 'FEATURE 1', 'FEATURE_2, and 'TARGET. You want to efficiently load the data into a Pandas DataFrame for model training, minimizing memory usage. Which of the following code snippets is the MOST efficient way to achieve this, assuming you only need 'FEATURE 1', 'FEATURE 2, and 'TARGET' columns?

- A.
- B.
- C.
- D.
- E.

Answer: B

Explanation:
Option B, using is the most efficient. The method directly retrieves the data as a Pandas DataFrame, leveraging Snowflake's internal optimizations for transferring data to Pandas. It's significantly faster than fetching rows individually or all at once and then creating the DataFrame. Also, it only selects the needed Columns. Option A fetches all columns and then tries to build dataframe from the list which is less effective. Option C would require additional setup with sqlalchemy and may introduce extra dependencies. Option D is also correct, but option B utilizes snowflake's internal optimizations for pandas retrieval making it best choice. Option E is also not effective as it only fetches 1000 records.

NEW QUESTION # 160
A data scientist is tasked with predicting house prices using Snowflake. They have a dataset stored in a Snowflake table called 'HOUSE PRICES' with columns such as 'SQUARE FOOTAGE, 'NUM BEDROOMS, 'LOCATION_ID, and 'PRICE. They choose a Random Forest Regressor model. Which of the following steps is MOST important to prevent overfitting and ensure good generalization performance on unseen data, and how can this be effectively implemented within a Snowflake-centric workflow?

- A. Increase the number of estimators (trees) in the Random Forest to the maximum possible value to capture all potential patterns, without cross validation.
- B. Train the Random Forest model on the entire 'HOUSE PRICES table without splitting into training and validation sets, as this will provide the model with the most data.
- C. Randomly select a small subset of the features (e.g., only use 'SQUARE FOOTAGE and 'NUM BEDROOMS) to simplify the model and prevent overfitting.
- D. Eliminate outliers without understanding the data properly to reduce noise.
- E. Tune the hyperparameters of the Random Forest model (e.g., 'max_deptm, 'n_estimators') using cross-validation. You can achieve this by splitting the 'HOUSE PRICES table into training and validation sets using Snowflake's 'QUALIFY clause or temporary tables, then train and evaluate the model within a loop or stored procedure.

Answer: E

Explanation:
Hyperparameter tuning with cross-validation is crucial to prevent overfitting. By splitting the data into training and validation sets, we can evaluate the model's performance on unseen data and adjust the hyperparameters accordingly. Snowflake's 'QUALIFY' clause and temporary tables can be used to efficiently manage these splits. Using a maximum number of estimators without validation is prone to overfitting. Training on the entire dataset without validation provides no indication of generalization performance. Randomly

selecting a subset of features may remove important predictors and eliminating outliers without proper investigation can skew your data and reduce the efficacy of the model.

NEW QUESTION # 161
You are tasked with building a model to predict customer churn. You have a table named in Snowflake with the following relevant columns: 'customer_id', 'login_date', , 'orders_placed', , and 'churned' (binary indicator). You want to engineer features that capture customer engagement over time using Snowpark for Python. Which of the following feature engineering steps, applied sequentially, are MOST effective in creating features indicative of churn risk?

- A. 1. Calculate the number of days since the customer's last login, and use nulls instead of negative numbers to indicate inactivity. 2. Calculate the rolling 7-day average of 'orders_placed' using a window function, partitioning by 'customer_id' and ordering by 'login_date'. 3. Calculate the slope of a linear regression of page_views' over time for each customer, indicating the trend in engagement using Snowpark ML. 4. Calculate the percentage of weeks the customer logged in. 5. Create a feature showing standard deviation of page_views per customer over the last 90 days.
- B. 1. Calculate the average 'page_views' per week for each customer over the last 3 months using a window function. 2. Calculate the recency of the last order (days since last order) for each customer. 3. Create a feature indicating the change in average daily page views over the last month compared to the previous month. 4. Create a feature showing standard deviation of page_views per customer over the last 90 days.
- C. 1. Calculate the total 'page_views' and 'orders_placed' for each customer without considering time. 2. Use one-hot encoding for the 'subscription_type' column.
- D. 1. Calculate the maximum 'page_views' in a single day for each customer. 2. Calculate the total number of days with no 'login_date' for each customer. 3. Create a feature indicating if a customer has ever placed an order. 4. Use a simple boolean for the 'subscription_type' column.
- E. 1. Calculate the average 'page_views' per day for each customer. 2. Calculate the total number of for each customer. 3. Create a feature indicating whether the customer has a premium subscription ('subscription_type' = 'premium').

Answer: A,B

Explanation:
Options B and E are the MOST effective because they incorporate time-based features and indicators of engagement trends. Recency (days since last order) captures the time elapsed since the customer's last interaction. Calculating changes in page views, the number of login days and linear regression slope identifies trends in engagement. Rolling averages smooth out daily fluctuations and capture longer-term patterns. Standard deviation of page views indicates a trend in page view variance, and thus overall customer engagement variance. Option A lacks recency and trend information. Option C misses temporal analysis. Option D has less relevance features and can be used however it is more useful to compare how well a customer is engaged with previous activity.

NEW QUESTION # 162
.....

They always keep the accuracy of questions and answers.

Enthusiastic Reseller!

- New DSA-C03 Exam Notes - 100% Pass Quiz Snowflake - DSA-C03 - First-grade Latest SnowPro Advanced: Data Scientist Certification Exam Exam Discount □ Easily obtain ➤ DSA-C03 □ for free download through ➥ www.prepawaypdf.com □ □High DSA-C03 Quality
- Free PDF Quiz Snowflake - DSA-C03 Latest New Exam Notes □ Enter □ www.pdfvce.com □ and search for □ DSA-C03 □ to download for free □DSA-C03 Sample Questions
- Download DSA-C03 Demo ➽ DSA-C03 Exam Paper Pdf □ Training DSA-C03 For Exam □ Immediately open [www.troytecdumps.com] and search for ▸ DSA-C03 ◂ to obtain a free download □New DSA-C03 Dumps Ebook
- New DSA-C03 Exam Notes - 100% Pass Quiz Snowflake - DSA-C03 - First-grade Latest SnowPro Advanced: Data Scientist Certification Exam Exam Discount □ Go to website ➥ www.pdfvce.com □ open and search for ➤ DSA-C03 □ to download for free □DSA-C03 Valid Test Forum
- Questions DSA-C03 Exam □ DSA-C03 Valid Dumps Book □ DSA-C03 Certification Torrent □ Easily obtain ➥ DSA-C03 □ for free download through { www.examcollectionpass.com } □DSA-C03 Certification Torrent
- Training DSA-C03 Tools □ DSA-C03 Certification Torrent □ DSA-C03 Valid Exam Testking □ Copy URL □ www.pdfvce.com □ open and search for □ DSA-C03 □ to download for free □DSA-C03 Dump File
- Passing DSA-C03 Score □ DSA-C03 Valid Exam Testking □ DSA-C03 Dump File □ Search for ⇛ DSA-C03 ⇚ on □ www.exam4labs.com □ immediately to obtain a free download □High DSA-C03 Quality
- High DSA-C03 Quality □ DSA-C03 Exam Paper Pdf □ DSA-C03 Latest Dumps Ebook □ Simply search for （ DSA-C03 ） for free download on ➽ www.pdfvce.com □ □DSA-C03 Dump File
- Free PDF Quiz Snowflake - DSA-C03 Latest New Exam Notes □ The page for free download of ➠ DSA-C03 □ on [www.practicevce.com] will open immediately □High DSA-C03 Passing Score
- Download DSA-C03 Demo ◂ Download DSA-C03 Demo □ Training DSA-C03 Tools □ Enter ➠ www.pdfvce.com □ and search for □ DSA-C03 □ to download for free □Download DSA-C03 Demo
- Go With Snowflake DSA-C03 Exam Questions For 100% Success □ Search for ⇛ DSA-C03 ⇚ and download it for free on □ www.vce4dumps.com □ website □High DSA-C03 Passing Score
- www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, Disposable vapes

2026 Latest Actual4Cert DSA-C03 PDF Dumps and DSA-C03 Exam Engine Free Share: https://drive.google.com/open?id=1s218Fy4vtQDD3hEmbvU3425fV-UJW2jB