Data Analytic Question & Answer

Question 1: What is Pandas, and why is it commonly used in data cleaning tasks?
Answer :- Pandas is an open-source Python library used for data manipulation and analysis. It provides data structures like Series and DataFrame, which are designed for efficiently working with structured data. Pandas is widely used in data science, machine learning, and data analysis tasks for tasks such as cleaning, transforming, and analyzing data. Pandas is commonly used in data cleaning tasks due to its powerful tools for handling missing data, reshaping data, removing duplicates, and facilitating various transformations. Its intuitive syntax and efficient data structures, such as DataFrames, make it a preferred choice for data scientists and analysts when cleaning and preparing data for analysis.

Question 2: Given a DataFrame with missing values, how would you check for missing values in each column and count the total number of missing values?

Answer :- SYNTAX:-

```
missing_values_per_column = data.isnull().sum()

total_missing_values = data.isnull().sum().sum()
```

This code uses the isnull() method to create a boolean mask indicating missing values, and then sum() is used to count the True values (missing values) along each column.

Question 3: How can you remove duplicates from a DataFrame while retaining the first occurrence of each unique row?

Answer :- SYNTTAX:- `data.drop(columns='Outlet_Location_Type', inplace=True)`

Question 4: If you have a DataFrame with a column containing string values, how can you convert all the values in that column to lowercase?

Answer :- To convert all values in a column of a DataFrame to lowercase.

SYNTAX :- SYNTAX :-

```
data=data[‘Item_Weight’] .str.lower()
```

Question 5: How do you replace missing values in a DataFrame with a specific value, like 0, for a particular column?

Answer :- To replace missing values in a DataFrame with a specific value, such as 0, for a particular column, you can use the fillna() method in Pandas.

SYNTAX :- data=data[' Item_Weight’’].fillna(0)

Question 6: If you have a DataFrame with a datetime column, how can you extract the year, month, and day into separate columns?

Answer :- data['Year'] = data[' Outlet_Establishment_Year'].dt.year

Question 7: How can you filter rows in a DataFrame where a specific column's values meet a certain condition (e.g., all rows where 'age' is greater than 30)?

Answer :- SYNTAX :- condition = data['Item_Weight'] > 30

filtered_data = data.loc[condition]

Question 8: What is the purpose of the .apply() function in Pandas, and how would you use it to create a new column based on values from existing columns?

Answer :- The .apply() function in Pandas is used to apply a given function along the axis of a DataFrame. It is commonly used to transform data by applying a function to each element, row, or column of the DataFrame.

Question 9: Suppose you want to merge two DataFrames, 'df1' and 'df2,' on a common column 'key.' How would you perform this merge operation in Pandas?

Answer :- To merge two DataFrames, 'df1' and 'df2,' on a common column 'key' in Pandas, you can use the merge function.

For example :- merged_df = pd.merge(df1, df2, on='key', how='inner')

Question 10: You have a DataFrame with a column containing messy text data. How can you clean and standardize the text data (e.g., remove punctuation and convert to lowercase) in that column

Answer :- SYNTAX :-

```
def clean_text(text):
# Convert to lowercase
text = text.lower()
# Remove punctuation
text = text.translate(str.maketrans(" ", " ", string.punctuation))
```

```
df['Text_Column'] = df['Text_Column'].apply(clean_text)
```