Questions & Answers

Question 1: What is Pandas, and why is it commonly used in data cleaning tasks?

- Pandas is python library for data manipulation & analysis
- It is used in data cleaning task because it offers helpful tools for handling missing data, merging, joining tables and to perform other data manipulation
- It is very flexibles to use thus make is easy to use

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

```
# Check missing values in each column
    missing_values = df.isnull().sum()
# Count total number of missing values
    total_missing = df.isnull().sum().sum()
# Display the missing values
    print("Missing values in each column:")
    print(missing_values)
    print("\nTotal number of missing values:", total_missing)
```

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

- We can use drop_duplicates() method in pandas toremove duplicates from DataFrame while retaining first occurance of each unique row

```
#Remove duplicate & retain first occurrence of each unique row
Duplicates=df.drop_duplicates()
#Display DataFrame without duplicates
Print(Duplicates)
```

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?

- We can use str.lower()

```
#convert all values to in given column to lowercase
df['New_Column_Name']=df['Your_Column'].str.lower(inplace=True)
print(df)
```

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

- We can use fillna() method in pandas

```
#Replace missing values with 0 in given column
df['New_Column_Name']=df['Your_Column'].fillna(0)
print(df)
```

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

- We can use dt in accessor in pandas to extract year, month, day

```
#Extract Year, Month, Day into separate column
    df['Year]=df['Your_datetime_Column'].dt.year
    df['month']=df['Your_datetime_Column'].dt.month
```

```
df['day']=df['Your_datetime_Column'].dt.day
#Display
print(df)
```

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)?

- We can use Boolean indexing to filter rows in DataFrame based on condition

```
#to find row where age greater than 30
Filtered_df=df[df['age']>30]
#Display
print(df)
```

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?

- .apply() function used to apply function
- Eg. We want to transform Column & Row

```
#create function
Def calculate_column(row):
Return row['column1']+row['column2']
#Apply function
df['new_column']=df.apply(calculate_column,axis=1)
#Display
Print(df)
```

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?

- We can use Merge() function in pandas

```
#Merge dataframe
df=pd.merge(df1, df2, on='Key')
#display
Print(df)
```

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?

- We can used User defined function in python & apply() function

```
# Function for clean and standardize text
    def clean_text(text):
    text = text.lower()
    text = ''.join([char for char in text if char not in string.punctuation])
    return text
# Apply function
    df['text_column'] = df['text_column'].apply(clean_text)
# Display
    print(df)
```

Data Quality Report

Following Data Preprocessing steps taken

- Check duplicated values in dataset
- Check null values in dataset
- Dropped unnecessary column
- Replaced RAM GB with empty string
- Replaced Weight KG with empty string
- Used lambda function- Touchscreen 1 or Non-touchscreen 0
- Used lambda function- IPS Display 1 or Non-IPS Display 0
- Used lambda function- for CPU Column
- Perform other similar functions to do some minor data cleaning process
- Used Seaborn Library for visualization
    - Distplot to check price density
    - Bar charts to check majority of companies for laptops (Since Project is about Laptop)
    - Bar plot to compare prices Vs laptop companies
    - Bar plot to check laptop price Vs price
    - Scatter plot to check Inches Vs Price
    - Bar plot for Touchscreen and Non-Touch Screen
    - Bar plot for CPU Brand and Non-Touch Price
    - Heat Map to check Correlation between varbales