

1. How do you load a CSV file into a Pandas DataFrame?

```
import pandas as pd
```

```
df = pd.read_csv(r'Path where the CSV file is stored\File name.csv')  
print(df)
```

2. How do you check the data type of a column in a Pandas DataFrame?

To check the data type in pandas DataFrame we can use the “dtype” attribute. The attribute returns a series with the data type of each column. And the column names of the DataFrame are represented as the index of the resultant series object and the corresponding data types are returned as values of the series object.

3. How do you select rows from a Pandas DataFrame based on a condition?

```
[2]: import pandas as pd  
  
[3]: # importing pandas  
import pandas as pd  
  
record = {  
    'Name': ['Ankit', 'Amit', 'Aishwarya', 'Priyanka', 'Priya', 'Shaurya' ],  
    'Age': [21, 19, 20, 18, 17, 21],  
    'Stream': ['Math', 'Commerce', 'Science', 'Math', 'Math', 'Science'],  
    'Percentage': [88, 92, 95, 70, 65, 78]}  
  
# create a dataframe  
dataframe = pd.DataFrame(record, columns = ['Name', 'Age', 'Stream', 'Percentage'])  
  
print("Given Dataframe :\n", dataframe)  
  
# selecting rows based on condition  
rslt_df = dataframe.loc[dataframe['Percentage'] != 95]  
  
print('\nResult dataframe :\n', rslt_df)
```

```
Given Dataframe :  
      Name  Age  Stream  Percentage  
0    Ankit   21    Math          88  
1     Amit   19  Commerce          92  
2 Aishwarya   20   Science          95  
3  Priyanka   18    Math          70  
4     Priya   17    Math          65  
5   Shaurya   21   Science          78
```

```
Result dataframe :  
      Name  Age  Stream  Percentage  
0    Ankit   21    Math          88  
1     Amit   19  Commerce          92  
3  Priyanka   18    Math          70  
4     Priya   17    Math          65  
5   Shaurya   21   Science          78
```

```
[ ]:
```

4. How do you rename columns in a Pandas DataFrame?

To rename columns in a Pandas DataFrame, you can use the `DataFrame.rename()` method and pass a dictionary containing the old column names as keys and the new column names as values. For example:

```
[2]: import pandas as pd

[4]: import pandas as pd

# create a sample DataFrame with columns 'A' and 'B'
df = pd.DataFrame({'A': [1, 2, 3], 'B': [4, 5, 6]})

# rename columns 'A' and 'B' to 'X' and 'Y'
df.rename(columns={'A': 'X', 'B': 'Y'}, inplace=True)

# print the DataFrame
print(df)
```

	X	Y
0	1	4
1	2	5
2	3	6

```
[ ]:
```

5. How do you drop columns in a Pandas DataFrame?

Pandas DataFrame drop() Method

The `drop()` method removes the specified row or column. By specifying the column axis (`axis='columns'`), the `drop()` method removes the specified column. By specifying the row axis (`axis='index'`), the `drop()` method removes the specified row.

6. How do you find the unique values in a column of a Pandas DataFrame?

You can get unique values in column (multiple columns) from pandas DataFrame using `unique()` or `Series.unique()` functions. `unique()` from Series is used to get unique values from a single column and the other one is used to get from multiple columns.

7. How do you find the number of missing values in each column of a Pandas DataFrame?

You can use the following syntax to count NaN values in Pandas DataFrame:

- (1) Count NaN values under a single DataFrame column: `df['column name'].isna().sum()`
- (2) Count NaN values under an entire DataFrame: `df.isna().sum().sum()`
- (3) Count NaN values across a single DataFrame row: `df.loc[[index value]].isna().sum().sum()`

8. How do you fill missing values in a Pandas DataFrame with a specific value?

Use the `fillna()` Method

The `fillna()` function iterates through your dataset and fills all empty rows with a specified value. This could be the mean, median, modal, or any other value.

9. How do you concatenate two Pandas DataFrames?

To concatenate an arbitrary number of pandas objects (DataFrame or Series), use `concat`.

10. How do you merge two Pandas DataFrames on a specific column?

You can join pandas Dataframes in much the same way as you join tables in SQL.

The `concat()` function can be used to concatenate two Dataframes by adding the rows of one to the other.

`concat()` can also combine Dataframes by columns but the `merge()` function is the preferred way.

11. How do you group data in a Pandas DataFrame by a specific column and apply an aggregation function?

To group data in a Pandas DataFrame by a specific column and apply an aggregation function, we can use the `groupby()` method followed by the desired aggregation function. For example, to group data in a DataFrame by the "Category" column and calculate the mean of the "Sales" column, we can use the following code:

```
[5]: import pandas as pd

[7]: import pandas as pd

# load data into a DataFrame
df = pd.read_csv("data.csv")

# group data by "Category" column and calculate mean of "Sales" column
grouped_df = df.groupby("Category")["Sales"].mean()
```

This will return a DataFrame with the grouped data and the calculated mean for each group. We can also apply multiple aggregation functions to the same group by passing a list of functions to the `agg()` method, like so:

```
[8]: import pandas as pd

# load data into a DataFrame
df = pd.read_csv("data.csv")

# group data by "Category" column and apply multiple aggregation functions
grouped_df = df.groupby("Category")["Sales"].agg(["mean", "max", "min"])
```

12. How do you pivot a Pandas DataFrame?

Pandas DataFrame: `pivot()` function

The `pivot()` function is used to reshaped a given DataFrame organized by given index / column values. This function does not support data aggregation, multiple values will result in a MultiIndex in the columns.

13. How do you change the data type of a column in a Pandas DataFrame?

The best way to convert one or more columns of a DataFrame to numeric values is to use `pandas.to_numeric()`. This function will try to change non-numeric objects (such as strings) into integers or floating-point numbers as appropriate.

14. How do you sort a Pandas DataFrame by a specific column?

To sort the DataFrame based on the values in a single column, you'll use `.sort_values()`. By default, this will return a new DataFrame sorted in ascending order. It does not modify the original DataFrame.

15. How do you create a copy of a Pandas DataFrame?

Pandas DataFrame `copy()` Method

The `copy()` method returns a copy of the DataFrame. By default, the copy is a "deep copy" meaning that any changes made in the original DataFrame will NOT be reflected in the copy.

16. How do you filter rows of a Pandas DataFrame by multiple conditions?

Using Loc to Filter With Multiple Conditions

The loc function in pandas can be used to access groups of rows or columns by label. Add each condition you want to be included in the filtered result and concatenate them with the & operator. You'll see our code sample will return a pd.dataframe of our filtered rows.

17. How do you calculate the mean of a column in a Pandas DataFrame?

To calculate the mean of whole columns in the DataFrame, use pandas.Series.mean() with a list of DataFrame columns. You can also get the mean for all numeric columns using DataFrame.mean(), use axis=0 argument to calculate the column-wise mean of the DataFrame.

18. How do you calculate the standard deviation of a column in a Pandas DataFrame?

Standard deviation is calculated using the function .std(). However, the Pandas library creates the Dataframe object and then the function .std() is applied on that Dataframe.

19. How do you calculate the correlation between two columns in a Pandas DataFrame?

By using corr() function we can get the correlation between two columns in the dataframe.

20. How do you select specific columns in a DataFrame using their labels?

This is the most basic way to select a single column from a dataframe, just put the string name of the column in brackets. Returns a pandas series. Passing a list in the brackets lets you select multiple columns at the same time.

21. How do you select specific rows in a DataFrame using their indexes?

If you'd like to select rows based on integer indexing, you can use the .iloc function. If you'd like to select rows based on label indexing, you can use the .loc function.

22. How do you sort a DataFrame by a specific column?

To sort the DataFrame based on the values in a single column, you'll use .sort_values(). By default, this will return a new DataFrame sorted in ascending order. It does not modify the original DataFrame.

23. How do you create a new column in a DataFrame based on the values of another column?

Using `apply()` method

If you need to apply a method over an existing column in order to compute some values that will eventually be added as a new column in the existing DataFrame, then pandas DataFrame. `apply()` method should do the trick.

24. How do you remove duplicates from a DataFrame?

Pandas `drop_duplicates()` method helps in removing duplicates from the Pandas Dataframe In Python.

25. What is the difference between `.loc` and `.iloc` in Pandas?

The main distinction between the two methods is: `loc` gets rows (and/or columns) with particular labels. `iloc` gets rows (and/or columns) at integer locations.