Q1. How do you load a CSV file into a Pandas DataFrame?

Answer: import pandas as pd

Df=pd.read\_csv(‘filename.csv’)

Q2. How do you check the data type of a column in a Pandas DataFrame?

Answer: df.dtypes

Q3. How do you select rows from a Pandas DataFrame based on a condition?

Answer: df[1:10]

Q4. How do you rename columns in a Pandas DataFrame?

Answer: We can rename() method

Q5. How do you drop columns in a Pandas DataFrame?

Answer: df.drop()

Q6. How do you find the unique values in a column of a Pandas DataFrame?

Answer:df[‘columnname’].unique()

Q7. How do you find the number of missing values in each column of a Pandas DataFrame?

Answer:

Q8. How do you fill missing values in a Pandas DataFrame with a specific value?

Answer: df[‘column name’].isna().sum()

Q9. How do you concatenate two Pandas DataFrames?

Answer: By using concat() method

List4=[list1,list2,list3]

Results=pd.concat(List4)

Q10. How do you merge two Pandas DataFrames on a specific column?

Answer:By using merge() method

Q11. How do you group data in a Pandas DataFrame by a specific column and apply an aggregation function?

Answer: df.groupby() method

Q12. How do you pivot a Pandas DataFrame?

Answer:df.pivot()

Q13. How do you change the data type of a column in a Pandas DataFrame?

Answer:By using astype() method

df.astype(type name)

Q14. How do you sort a Pandas DataFrame by a specific column?

Answer:By using sort\_values() method

Q15. How do you create a copy of a Pandas DataFrame?

Answer:df.copy() method

Q16. How do you filter rows of a Pandas DataFrame by multiple conditions?

Answer:By using loc or we can use df.query(condition)

Q17. How do you calculate the mean of a column in a Pandas DataFrame?

Answer: data[‘column name’].mean() or data.describe() where we can see mean of every column

Q18. How do you calculate the standard deviation of a column in a Pandas DataFrame?

Answer:df[‘columnname’].std()

Q19. How do you calculate the correlation between two columns in a Pandas DataFrame?

Answer:df[‘first\_column’].corr(df[‘second\_column’])

Q20. How do you select specific columns in a DataFrame using their labels?

Answer:df.loc[1:10]

Q21. How do you select specific rows in a DataFrame using their indexes?

Answer:df.iloc[[0]]

Q22. How do you sort a DataFrame by a specific column?

Answer: df.sort\_values(‘column name’)

Q23. How do you create a new column in a DataFrame based on the values of another column?

Answer:df.apply()

Q24. How do you remove duplicates from a DataFrame?

Answer:df.drop\_duplicates() method

Q25. What is the difference between .loc and .iloc in Pandas?

Answer:

The loc() function  is label based data selecting method which means that we have to pass the name of the row or column which we want to select. This method includes the last element of the range passed in it, unlike iloc(). loc() can accept the boolean data unlike iloc().

The iloc() function  is an indexed-based selecting method which means that we have to pass an integer index in the method to select a specific row/column. This method does not include the last element of the range passed in it unlike loc(). iloc() does not accept the boolean data unlike loc().