**Python:**

1. What are Python Keywords?

Solution- Python keywords are reserved words. These keywords cannot be used as identifiers (such as variable names or function names). Examples of Python keywords include **if**, **else**, **for**, **while**, **def**, **class**, **import**, **from**, **and**, **or**, **not**, **return**, **True**, **False**, **None**, etc.

1. How do you handle a specific exception in your code?If you encounter a ‘FileNotFoundError’, how would you catch and handle it gracefully in your code?

Solution- You can handle a specific exception, such as **FileNotFoundError**, by using a **try** statement with an **except** block that specifies the type of exception you want to catch. Here's how you would catch and handle a **FileNotFoundError**

try:

with open('nonexistent\_file.txt','r') as file:

data=file.read()

except FileNotFoundError:

print("Error: File not Found!")

except Exception as e:

print(f"An error occured: {e}")

1. What is a lambda function in Python and where is it useful?

Solution- A lambda function in Python is a small, anonymous function defined using the **lambda** keyword. Lambda functions are particularly useful in situations where you need a small function for a short period of time, such as when passing a function as an argument to higher-order functions like **map()**, **filter()**, and **sorted()**, or when using functions like **reduce()** from the **functools** module.

1. How do you handle exceptions in Python and what is the reason for using the exceptions?

Solution- exceptions are used to handle errors and unexpected events that occur during the execution of a program. Exceptions are raised when an error occurs, and they can be caught and handled using try-except blocks.

1. What distinguishes the Python '==' and 'is' operators?

Solution- == compares the value or equality of two objects

Is operator checks whether 2 variables point to the same object in memory.

**EDA:**

1. Explain the concept of correlation and Which function is used to check the correlation between features?

Solution- Correlation is a statistical concept that measures the strength and direction of the relationship between two variables.

**Strength**

**Direction**

1. Explain the different types of transformation?

Solution-Function Transformation.(Best Preferred)

Power Transformation.

1-Box-Cox Transformation

2-Yeo-Jhonson Transformation

Quantile Transformation.(Least Preferred)

1. What is the formula for calculating Skewness and which python function is used to get the skewness value?
2. What does X - axis and Y - axis represent in a Histogram?

Solution- On the x-axis, you typically find the bins or intervals into which the data has been grouped.

The y-axis of a histogram represents the frequency or count of observations falling into each bin.

1. Which function is used to get a horizontal bar plot?

Solution- To create a horizontal bar plot in Python, you can use the **barh()** function from the **matplotlib.pyplot** module.

To create a vertical bar plot in Python, you can use the **bar()** function from the **matplotlib.pyplot**

**Data preprocessing:**

1. How do you handle skewed distributions in data preprocessing?

Solution-To handle skewed distribution in data preprocessing we should go for transformations. In transformation there are 3 types of transformation function transformation , power transformation, quantile transformation.Also you can remove outliers .

1. Sometimes in data, null values play hide and seek. How will you identify null values ?

Solution- **isnull()**

**notnull()**

**info()**

**isna()**

1. What challenges can arise during the data preprocessing phase and how do you overcome those challenges?
2. **Solution**- **Missing Data**:
   * **Challenge**: Missing data can be problematic as it can lead to biased analysis and inaccurate results.
   * **Solution**:
     + One approach is to remove observations with missing data if they form only a small portion of the dataset.
     + Another approach is to impute missing values using methods like mean, median, mode, or more sophisticated techniques such as interpolation or machine learning-based imputation.
3. **Outliers**:
   * **Challenge**: Outliers can skew statistical measures and affect the performance of some machine learning algorithms.
   * **Solution**:
     + Detect outliers using statistical methods like z-score, IQR (Interquartile Range), or visualization techniques such as box plots.
     + Decide whether to remove outliers or transform them using techniques like winsorization or robust scaling.
4. **Feature Scaling**:
   * **Challenge**: Features in the dataset may have different scales, which can lead to issues in algorithms that are sensitive to the scale of features.
   * **Solution**:
     + Standardize features by subtracting the mean and dividing by the standard deviation (z-score normalization).
     + Alternatively, normalize features to a fixed range, such as [0, 1] or [-1, 1], using Min-Max scaling.
5. **Categorical Data**:
   * **Challenge**: Many machine learning algorithms cannot directly handle categorical data, so it needs to be encoded into numerical format.
   * **Solution**:
     + One-hot encoding: Convert categorical variables into binary vectors where each category becomes a binary feature.
     + Label encoding: Map each category to a unique integer.
     + Target encoding: Encode categorical variables based on the target variable's mean or other statistics.
6. **Data Transformation**:
   * **Challenge**: Sometimes, data may need transformation to make it suitable for analysis or modeling.
   * **Solution**:
     + Logarithmic transformation: Useful for data that follows a skewed distribution.
     + Polynomial transformation: Introduce polynomial features to capture nonlinear relationships.
     + Box-Cox transformation: Generalizes the logarithm by allowing for negative values and a parameter to adjust the skewness.
7. **Data Integration**:
   * **Challenge**: When working with multiple datasets, integrating them into a single cohesive dataset can be challenging.
   * **Solution**:
     + Ensure consistency in variable names and data formats across datasets.
     + Merge datasets using common identifiers or keys.
     + Handle conflicts and missing values appropriately during integration.
8. **Computational Efficiency**:
   * **Challenge**: Preprocessing large datasets can be computationally intensive and time-consuming.
   * **Solution**:
     + Utilize parallel processing techniques to speed up computations.
     + Consider sampling techniques to work with smaller representative subsets of the data during development.
     + Optimize code and use efficient libraries and data structures.
9. In One Hot Encoding technique sometimes results come as sparse Matrix, What is the reason for it ?

Solution- if you have multiple classes, lets say a column with 10 rows are there, for that 10 column will be made and many 0’s we will be getting only because of one hot encoding .To avoid this we should go for label encoding.

1. What is the purpose of data normalisation and what methods can we use to normalise the data?

**Solution**- The purpose of data normalization is to transform the features of a dataset into a similar scale, typically between 0 and 1, **Min-Max Scaling (Normalization)**: This method scales the data to a fixed range, typically between 0 and 1.