1. What are the key features of Python?

- Python is an interpreted language. That means that, unlike languages like *C* and its variants, Python does not need to be compiled before it is run. Other interpreted languages include *PHP* and *Ruby*.
- Python is dynamically typed; this means that you don't need to state the types of variables when you declare them or anything like that. You can do things like x=111 and then x="I'm a string" without error
- Python is well suited to object orientated programming in that it allows the definition of classes along with composition and inheritance. Python does not have access specifiers (like C++'s public, private).
- In Python, functions are first-class objects. This means that they can be assigned to variables, returned from other functions and passed into functions. Classes are also first class objects
- Writing Python code is quick but running it is often slower than compiled languages. Fortunately, Python allows the inclusion of C-based extensions so bottlenecks can be optimized away and often are. The numpy package is a good example of this, it's really quite quick because a lot of the number-crunching it does isn't actually done by Python
- Python finds use in many spheres – web applications, automation, scientific modelling, big data applications and many more. It's also often used as "glue" code to get other languages and components to play nice. Learn more about Big Data and its applications from the Azure data engineer training course.

2. What is the difference between .py and .pyc files?

The .py files are the python source code files. While the .pyc files contain the bytecode of the python files. .pyc files are created when the code is imported from some other source. The interpreter converts the source .py files to .pyc files which helps by saving time.

3. What is Python? List some popular applications of Python in the world of technology.

Python is a widely-used general-purpose, high-level programming language. It was created by Guido van Rossum in 1991 and further developed by the Python Software Foundation. It was designed with an emphasis on code readability, and its syntax allows programmers to express their concepts in fewer lines of code.

It is used for:

- System Scripting
- Web Development
- Game Development
- Software Development
- Complex Mathematics

4. What are the benefits of using Python language as a tool in the present scenario?

The following are the benefits of using Python language:

- Object-Oriented Language
- High-Level Language

- Dynamically Typed language
- Extensive support Libraries
- Presence of third-party modules
- Open source and community development
- Portable and Interactive
- Portable across Operating systems

5. Is Python a compiled language or an interpreted language?

Actually, Python is a partially compiled language and partially interpreted language. The compilation part is done first when we execute our code and this will generate byte code internally this byte code gets converted by the Python virtual machine (p.v.m) according to the underlying platform (machine operating system).

6. What does the '#' symbol do in Python?

'#' is used to comment on everything that comes after on the line.

7. What is the difference between a Mutable datatype and an Immutable data type?

- Mutable data types can be edited i.e.; they can change at runtime. Eg – List, Dictionary, etc.
- Immutable data types cannot be edited i.e., they cannot change at runtime. Eg – String, Tuple, etc.

8. What are immutable and mutable data types?

Ans. Data types in Python are categorized into mutable and immutable data types.

- Mutable Data Type – A mutable data type is those whose values can be changed. Example: List, Dictionaries, and Set
- Immutable Data Type – An immutable data type is one in which the values can't be changed or altered. Example: String and Tuples

●	Mutable	Immutable
Definition	Data type whose values can be changed after creation.	Data types whose values can't be changed or altered.
Memory Location	Retains the same memory location even after the content is modified.	Any modification results in a new object and new memory location
Performance	It is memory-efficient, as no new objects are created for frequent changes.	It might be faster in some scenarios as there's no need to track changes.
Use-cases	When you need to modify, add, or remove existing data frequently.	When you want to ensure data remains consistent and unaltered.
Example	List, Dictionaries, Set	Strings, Types, Integer

9. Write a code to display the current time?

```
import time
currenttime= time.localtime(time.time())
print ("Current time is", currenttime)
```

10. Is Indentation Required in Python?

Yes, indentation is required in Python. A Python interpreter can be informed that a group of statements belongs to a specific block of code by using Python indentation. Indentations make the code easy to read for developers in all programming languages but in Python, it is very important to indent the code in a specific order.

12. How are arguments passed by value or by reference in Python?

Everything in Python is an object and all variables hold references to the objects. The reference values are according to the functions; as a result, you cannot change the value of the references. However, you can change the objects if it is mutable.

13. What is the difference between a Set and Dictionary?

The set is an unordered collection of data types that is iterable, mutable and has no duplicate elements.
A dictionary in Python is an ordered collection of data values, used to store data values like a map.

14. What is List Comprehension? Give an Example.

List comprehension is a syntax construction to ease the creation of a list based on existing iterable.
For Example:

```
my_list = [i for i in range(1, 10)]
```

15. What is a lambda function?

A lambda function is an anonymous function. This function can have any number of parameters but, can have just one statement. For Example:

```
a = lambda x, y : x*y
print(a(7, 19))
```

16. What is a pass in Python?

Pass means performing no operation or in other words, it is a placeholder in the compound statement, where there should be a blank left and nothing has to be written there.

17. What is the difference between / and // in Python?

/ represents precise division (result is a floating point number) whereas // represents floor division (result is an integer). For Example:

5//2 = 2

5/2 = 2.5

18. What is the purpose of 'is', 'not' and 'in' operators?

Operators are special functions. They take one or more values and produce a corresponding result.

is: returns true when 2 operands are true (Example: "a" is 'a')

not: returns the inverse of the boolean value

in: checks if some element is present in some sequence

19. What is slicing in Python?

- Slicing is used to access parts of sequences like lists, tuples, and strings.
- The syntax of slicing is-[start:end:step].
- When we write [start:end] this returns all the elements of the sequence from the start (inclusive) till the end-1 element. If the start or end element is negative i, it means the ith element from the end. The step indicates the jump or how many elements have to be skipped.
- Eg. if there is a list- [1,2,3,4,5,6,7,8]. Then [-1:2:2] will return elements starting from the last element till the third element by printing every second element.i.e. [8,6,4].

20. What is Dictionary Comprehension? Give an Example

Dictionary Comprehension is a syntax construction to ease the creation of a dictionary based on the existing iterable.

For Example: my_dict = {i:i+7 for i in range(1, 10)}

21. Is Tuple Comprehension? If yes, how, and if not why?

(i for i in (1, 2, 3))

Tuple comprehension is not possible in Python because it will end up in a generator, not a tuple comprehension.

22. Differentiate between List and Tuple?

Let's analyze the differences between List and Tuple:

List

- Lists are Mutable datatype.
- Lists consume more memory
- The list is better for performing operations, such as insertion and deletion.
- The implication of iterations is Time-consuming

Tuple

- Tuples are Immutable datatype.
- Tuple consumes less memory as compared to the list
- A Tuple data type is appropriate for accessing the elements

- The implication of iterations is comparatively Faster

23. How is Exceptional handling done in Python?

There are 3 main keywords i.e. try, except, and finally which are used to catch exceptions and handle the recovering mechanism accordingly. Try is the block of a code that is monitored for errors. Except block gets executed when an error occurs.

The beauty of the final block is to execute the code after trying for an error. This block gets executed irrespective of whether an error occurred or not. Finally, block is used to do the required cleanup activities of objects/variables.

24. What is the use of try and except block in Python?

The try block is used to check some code for errors i.e the code inside the try block will execute when there is no error in the program. Whereas the code inside the except block will execute whenever the program encounters some error in the preceding try block.

25. Difference between for loop and while loop in Python

The “for” Loop is generally used to iterate through the elements of various collection types such as List, Tuple, Set, and Dictionary. Developers use a “for” loop where they have both the conditions start and the end. Whereas, the “while” loop is the actual looping feature that is used in any other programming language. Programmers use a Python while loop where they just have the end conditions.

26. Can we Pass a function as an argument in Python?

Yes, several arguments can be passed to a function, including objects, variables (of the same or distinct data types), and functions. Functions can be passed as parameters to other functions because they are objects. Higher-order functions are functions that can take other functions as arguments.

27. What are *args and *kwargs?

To pass a variable number of arguments to a function in Python, use the special syntax *args and **kwargs in the function specification. It is used to pass a variable-length, keyword-free argument list. By using the *, the variable we associate with the * becomes iterable, allowing you to do operations on it such as iterating over it and using higher-order operations like map and filter.

28. What is docstring in Python?

Python documentation strings (or docstrings) provide a convenient way of associating documentation with Python modules, functions, classes, and methods.

- **Declaring Docstrings:** The docstrings are declared using ”’triple single quotes”’ or “”””triple double quotes””””” just below the class, method, or function declaration. All functions should have a docstring.

- **Accessing Docstrings:** The docstrings can be accessed using the __doc__ method of the object or using the help function.

29. What is a dynamically typed language?

Typed languages are the languages in which we define the type of data type and it will be known by the machine at the compile-time or at runtime. Typed languages can be classified into two categories:

- **Statically typed languages:** In this type of language, the data type of a variable is known at the compile time which means the programmer has to specify the data type of a variable at the time of its declaration.
- **Dynamically typed languages:** These are the languages that do not require any pre-defined data type for any variable as it is interpreted at runtime by the machine itself. In these languages, interpreters assign the data type to a variable at runtime depending on its value.

30. What is a break, continue, and pass in Python?

The **break** statement is used to terminate the loop or statement in which it is present. After that, the control will pass to the statements that are present after the break statement, if available.

Continue is also a loop control statement just like the break statement. continue statement is opposite to that of the break statement, instead of terminating the loop, it forces to execute the next iteration of the loop.

Pass means performing no operation or in other words, it is a placeholder in the compound statement, where there should be a blank left and nothing has to be written there.

31. What are Iterators in Python?

In Python, iterators are used to iterate a group of elements, containers like a list. Iterators are collections of items, and they can be a list, tuples, or a dictionary. Python iterator implements __itr__ and the next() method to iterate the stored elements. We generally use loops to iterate over the collections (list, tuple) in Python.

32. What is self in Python?

Self is an instance or an object of a class. In Python, this is explicitly included as the first parameter. However, this is not the case in Java where it's optional. It helps to differentiate between the methods and attributes of a class with local variables.

The self variable in the init method refers to the newly created object while in other methods, it refers to the object whose method was called.

33. Does Python supports multiple Inheritance?

Python does support multiple inheritances, unlike Java. Multiple inheritances mean that a class can be derived from more than one parent class.

34. What is Polymorphism in Python?

Polymorphism means the ability to take multiple forms. So, for instance, if the parent class has a method named ABC then the child class also can have a method with the same name ABC having its own parameters and variables. Python allows polymorphism.

35. How do you do data abstraction in Python?

Data Abstraction is providing only the required details and hides the implementation from the world. It can be achieved in Python by using interfaces and abstract classes.

36. How to delete a file using Python?

We can delete a file using Python by following approaches:

- os.remove()
- os.unlink()

37. What is PIP?

PIP is an acronym for Python Installer Package which provides a seamless interface to install various Python modules. It is a command-line tool that can search for packages over the internet and install them without any user interaction.

38. What is a zip function?

Python zip() function returns a zip object, which maps a similar index of multiple containers. It takes an iterable, converts it into an iterator and aggregates the elements based on iterables passed. It returns an iterator of tuples.

39. What is __init__() in Python?

Equivalent to constructors in OOP terminology, __init__ is a reserved method in Python classes. The __init__ method is called automatically whenever a new object is initiated. This method allocates memory to the new object as soon as it is created. This method can also be used to initialize variables.

40. What is the difference between Python Arrays and lists?

Arrays and lists, in Python, have the same way of storing data. But, arrays can hold only a single data type elements whereas lists can hold any data type elements.

41. What advantages do NumPy arrays offer over (nested) Python lists?

1. Python’s lists are efficient general-purpose containers. They support (fairly) efficient insertion, deletion, appending, and concatenation, and Python’s list comprehensions make them easy to construct and manipulate.

2. They have certain limitations: they don't support "vectorized" operations like elementwise addition and multiplication, and the fact that they can contain objects of differing types mean that Python must store type information for every element, and must execute type dispatching code when operating on each element.
3. NumPy is not just more efficient; it is also more convenient. You get a lot of vector and matrix operations for free, which sometimes allow one to avoid unnecessary work. And they are also efficiently implemented.
4. NumPy array is faster and You get a lot built in with NumPy, FFTs, convolutions, fast searching, basic statistics, linear algebra, histograms, etc.

42. What do you know about Pandas in Python?

Pandas is a data manipulation package in Python for tabular data. That is, data in the form of rows and columns, also known as DataFrames. Intuitively, you can think of a DataFrame as an Excel sheet. Pandas' functionality includes data transformations, like sorting rows and taking subsets, to calculating summary statistics such as the mean, reshaping DataFrames, and joining DataFrames together. pandas works well with other popular Python data science packages, often called the PyData ecosystem, including

- **NumPy** for numerical computing
- **Matplotlib**, **Seaborn**, **Plotly**, and other data visualization packages
- **scikit-learn** for machine learning

pandas is used throughout the data analysis workflow. With pandas, you can:

- Import datasets from databases, spreadsheets, comma-separated values (CSV) files, and more.
- Clean datasets, for example, by dealing with missing values.
- Tidy datasets by reshaping their structure into a suitable format for analysis.
- Aggregate data by calculating summary statistics such as the mean of columns, correlation between them, and more.
- Visualize datasets and uncover insights.

43. Why split is used?

In Python, the split() function is used to split a string.

44. What is NumPy?

NumPy (Numerical Python) is a fundamental package for scientific computing in Python. It provides support for arrays, matrices, and many mathematical functions to operate on these arrays.

45. How do you create a NumPy array?

You can create a NumPy array using numpy.array() by passing a list or a tuple.

46. What are the advantages of using NumPy arrays over Python lists?

NumPy arrays are more efficient in terms of memory and performance. They provide support for mathematical and logical operations, and they are optimized for large amounts of data.

47. What are broadcasting and how does it work in NumPy?

Broadcasting allows NumPy to perform operations on arrays of different shapes. It "stretches" the smaller array so that it has the same shape as the larger array.

48. How do you generate a single random number using NumPy?

You can generate a single random number using numpy.random.rand() for a float between 0 and 1.

49. How do you generate random integers within a specific range?

You can generate random integers using numpy.random.randint(low, high=None, size=None), where low is the inclusive lower bound, high is the exclusive upper bound, and size is the shape of the output array.

50. How do you generate random numbers from a normal (Gaussian) distribution?

You can use numpy.random.normal(loc=0.0, scale=1.0, size=None), where loc is the mean, scale is the standard deviation, and size is the shape of the output array.

51. What are the main data structures in Pandas?

The main data structures in Pandas are Series (1-dimensional) and DataFrame (2-dimensional).

52. How do you read and write data in Pandas from/to different file formats?

You can read data using functions like pd.read_csv(), pd.read_excel(), pd.read_json(), and write data using to_csv(), to_excel(), to_json(), etc.

53. How do you handle missing data in a DataFrame?

You can handle missing data using methods like dropna() to remove missing values or fillna() to fill them with a specified value.

54. How do you group data in a DataFrame?

You can group data using the groupby() method and then perform aggregation functions like sum(), mean(), etc.

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

loc is label-based indexing, which means you have to specify the name of the rows and columns you want to filter out. iloc is integer position-based indexing, which means you have to specify the integer index of the rows and columns you want to filter out.

56. What is Matplotlib?

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. It is highly customizable and supports various types of plots.

57. What is Seaborn?

Seaborn is a Python data visualization library based on Matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

58. How do you customize the appearance of plots in Matplotlib?

You can customize plots in Matplotlib by setting various parameters like colors, labels, titles, and legends.

```
plt.plot(x, y, color='red', linestyle='--', marker='o')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.title('Customized Line Plot')
plt.legend(['Data Line'])
plt.xticks(rotations=90)
plt.grid(True)
plt.show()
```

59. What are some common styles available in Seaborn for plots?

Seaborn provides several built-in themes and color palettes like darkgrid, whitegrid, dark, white, ticks.

60. How do you open a file in Python?

You can open a file using the open() function, which returns a file object.

```
file = open('example.txt', 'r')  # Open file in read mode
```

61. What are the different file modes available in Python?

- **The common file modes are:**
    - 'r' - Read (default mode)
    - 'w' - Write (creates a new file or truncates the existing file)
    - 'a' - Append (adds content to the end of the file)
    - 'b' - Binary mode
    - '+' - Read and write

62. How do you read the contents of a file?

You can read the contents of a file using read(), readline(), or readlines().

```
with open('example.txt', 'r') as file:
    content = file.read()
```

63. How do you write data to a file in Python?

You can write data to a file using the write() or writelines() method.

```
with open('example.txt', 'w') as file:
    file.write('Hello, World!')
```

64. How do you append data to an existing file?

You can append data using the 'a' mode with the open() function.

65. How do you handle exceptions while working with files in Python?

You can handle exceptions using a try...except block.

```
try:
    with open('example.txt', 'r') as file:
        content = file.read()
except FileNotFoundError:
    print('File not found')
except IOError:
    print('An I/O error occurred')
```

66. What is EDA in Python?

Exploratory Data Analysis (EDA) in Python is an approach to analyzing data sets to summarize their main characteristics, often using visualization methods.

67. What are the key libraries in Python for performing EDA?

Key libraries include Pandas, NumPy, Matplotlib, Seaborn, and Plotly.

68. How do you load a CSV file in Python using Pandas?

You can use the read_csv() function from the Pandas library, like this:

```
import pandas as pd
data = pd.read_csv('file.csv')
```

69. What is the purpose of the head() function in Pandas?

The head() function is used to display the first few rows of a DataFrame, providing a quick overview of its structure and content.

70. How do you check for missing values in a DataFrame?

You can use the isnull() method followed by sum() to count missing values in each column, like this:

```
print(data.isnull().sum())
```

71. How do you handle missing values in Python?

You can handle missing values by either removing them, filling them with a specific value (like the mean or median), or using more advanced techniques like interpolation.

72. What are the main types of plots used in EDA?

Common types include histograms, scatter plots, box plots, bar plots, and heatmaps.

73. How do you create a histogram in Python using Matplotlib?

You can use the hist() function from Matplotlib, like this:

```
import matplotlib.pyplot as plt
plt.hist(data['column_name'], bins=10)
plt.show()
```

74. What is a box plot? How is it useful in EDA?

A box plot (or box-and-whisker plot) is a graphical summary of the distribution of a dataset. It displays the median, quartiles, and potential outliers. It's useful in EDA for identifying central tendency, variability, and outliers in the data.

75. How do you create a box plot in Python using Seaborn?

You can use the boxplot() function from Seaborn, like this:

```
import seaborn as sns
sns.boxplot(x='column_name', data=data)
plt.show()
```

76. What is correlation? How do you calculate it in Python?

Correlation measures the strength and direction of the relationship between two variables. In Python, you can calculate it using the corr() function from Pandas.

77. How do you visualize correlation matrices in Python?

You can use a heatmap to visualize correlation matrices. Seaborn's heatmap() function is commonly used for this purpose.

78. What is the purpose of outlier detection in EDA?

Outlier detection helps identify data points that deviate significantly from the rest of the data. These outliers can sometimes indicate errors or anomalies in the data.

79. How do you detect outliers in Python?

You can use statistical methods like z-score or IQR (Interquartile Range) to detect outliers in Python.

80. Explain z-score method for outlier detection.

Z-score method involves calculating the z-score for each data point, which measures how many standard deviations it is from the mean. Data points with z-scores beyond a certain threshold (commonly ±3) are considered outliers.

81. How do you handle outliers in Python?

Outliers can be handled by either removing them, transforming the data, or using more robust statistical techniques that are less sensitive to outliers.

82. What is skewness? How do you detect skewness in a dataset?

Skewness measures the asymmetry of the probability distribution of a real-valued random variable about its mean. You can detect skewness in a dataset by calculating its skewness coefficient using libraries like SciPy.

83. How do you handle skewness in Python?

Skewness can be handled by transforming the data using techniques like log transformation, square root transformation.

84. What are the main steps in EDA?

The main steps include data collection, data cleaning, data exploration, and visualization, statistical analysis, and drawing conclusions.

85. What is the purpose of data transformation in EDA?

Data transformation is used to convert the original data into a format that is more suitable for analysis. It can involve normalization, standardization, or transformations to correct skewness.

86. What is the purpose of a correlation matrix in EDA?

A correlation matrix is used to examine the relationships between multiple variables in a dataset, helping to identify patterns and dependencies.

87. How do you create a correlation matrix in Python?

You can use the corr() function from Pandas to calculate the correlation matrix for a DataFrame.

88. What does a correlation coefficient value of 0 indicate?

A correlation coefficient value of 0 indicates no linear relationship between the two variables.

89. What is the difference between positive and negative correlation?

Positive correlation means that as one variable increases, the other variable also tends to increase, while negative correlation means that as one variable increases, the other variable tends to decrease.

90. How do you interpret the strength of correlation coefficients?

The strength of correlation coefficients is typically interpreted as follows: close to 1 or -1 indicates strong correlation, close to 0 indicates weak correlation.

91. What is the purpose of a scatter plot in EDA?

A scatter plot is used to visualize the relationship between two continuous variables, helping to identify patterns such as correlations, clusters, or outliers.

92. What is the purpose of a pair plot in EDA?

A pair plot is used to visualize pairwise relationships between variables in a dataset. It creates scatter plots for numerical variables and histograms for categorical variables along the diagonal.

93. What is the purpose of a bar plot in EDA?

A bar plot is used to visualize the distribution of a categorical variable, often showing the frequency or proportion of each category.

94. What is the purpose of a count plot in EDA?

A count plot is a specialized form of a bar plot used to count the occurrences of each category in a categorical variable.

94. What is the lifecycle of a data science project?

Data Collection: Gathering relevant data from various sources.

Exploratory Data Analysis (EDA): Understanding the data, identifying patterns, and visualizing relationships.

Model Training and Testing: Building predictive models using machine learning algorithms.

96. Differentiate between Univariate, Bivariate, and Multivariate analysis:

- Univariate Analysis: Examining one variable at a time. It aims to describe the variable and find patterns within it (e.g., analyzing student heights).
- Bivariate Analysis: Involves two different variables. It explores relationships and causes between them (e.g., temperature vs. ice cream sales).

97. What are the two kinds of target variables for predictive modeling?

- Numerical/Continuous Variable: Values lie within a range (e.g., student heights). Predictions can be any value within that range.
- Categorical Variable: Takes on a limited, fixed number of possible values (e.g., class labels).

98. Explain about sampling distribution?

Sampling distribution refers to the distribution of statistics, like the mean or proportion, calculated from multiple samples drawn from the same population. It helps understand how these statistics vary across different samples and is essential for making inferences about population parameters. The central limit theorem states that the sampling distribution of the sample mean tends towards a normal distribution as the sample size increases, regardless of the population distribution, making it a fundamental concept in statistical inference.

99. Explain in 2 lines about statistical terms?

Mean: It's the average value of a dataset calculated by summing all values and dividing by the total number of observations, providing a measure of central tendency.

Median: The middle value of a dataset when arranged in ascending order; it's robust to extreme values and gives another measure of central tendency.

Mode: The most frequently occurring value in a dataset, providing another measure of central tendency.

Standard deviation: It measures the average deviation of data points from the mean, providing a measure of dispersion in the dataset.

Range: The difference between the maximum and minimum values in a dataset, providing a simple measure of variability.

100. Define the term 'Data Wrangling in Data Analytics.

Data Wrangling is the process wherein raw data is cleaned, structured, and enriched into a desired usable format for better decision making. It involves discovering, structuring, cleaning, enriching, validating, and analyzing data. This process can turn and map out large amounts of data extracted from various sources into a more useful format. Techniques such as merging, grouping, concatenating, joining, and sorting are used to analyze the data. Thereafter it gets ready to be used with another dataset.

101. What are the various steps involved in any analytics project?

This is one of the most basic data analyst interview questions. The various steps involved in any common analytics projects are as follows:

Understanding the Problem

- Understand the business problem, define the organizational goals, and plan for a lucrative solution.

Collecting Data

- Gather the right data from various sources and other information based on your priorities.

Cleaning Data

- Clean the data to remove unwanted, redundant, and missing values, and make it ready for analysis.

Exploring and Analyzing Data

- Use data visualization and business intelligence tools, data mining techniques, and predictive modeling to analyze data.

Interpreting the Results

- Interpret the results to find out hidden patterns, future trends, and gain insights.

102. What is the significance of Exploratory Data Analysis (EDA)?

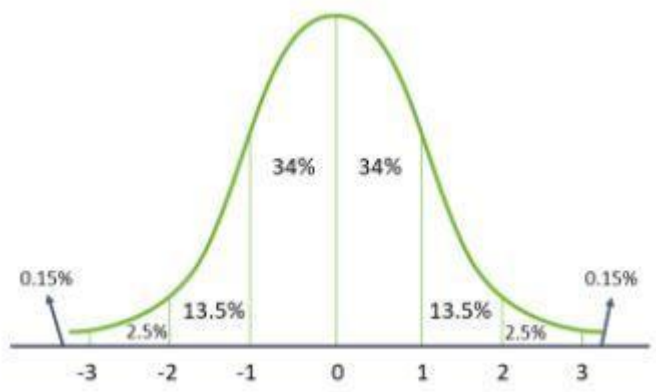
- Exploratory data analysis (EDA) helps to understand the data better.
- It helps you obtain confidence in your data to a point where you're ready to engage a machine learning algorithm.
- It allows you to refine your selection of feature variables that will be used later for model building.
- You can discover hidden trends and insights from the data.

103. What are the best methods for data cleaning?

- Create a data cleaning plan by understanding where the common errors take place and keep all the communications open.
- Before working with the data, identify and remove the duplicates. This will lead to an easy and effective data analysis process.
- Focus on the accuracy of the data. Set cross-field validation, maintain the value types of data, and provide mandatory constraints.
- Normalize the data at the entry point so that it is less chaotic. You will be able to ensure that all information is standardized, leading to fewer errors on entry.

104. Explain the term Normal Distribution.

Normal Distribution refers to a continuous probability distribution that is symmetric about the mean. In a graph, normal distribution will appear as a bell curve.



- The mean, median, and mode are equal
- All of them are located in the center of the distribution
- 68% of the data falls within one standard deviation of the mean

- 95% of the data lies between two standard deviations of the mean
- 99.7% of the data lies between three standard deviations of the mean

105. Mention the differences between Data Mining and Data Profiling?

Data Mining	Data Profiting
Data mining is the process of discovering relevant information that has not yet been identified before.	Data profiling is done to evaluate a dataset for its uniqueness, logic, and consistency.
In data mining, raw data is converted into valuable information.	It cannot identify inaccurate or incorrect data values.

106. Define the term 'Data Wrangling in Data Analytics.

Data Wrangling is the process wherein raw data is cleaned, structured, and enriched into a desired usable format for better decision making. It involves discovering, structuring, cleaning, enriching, validating, and analyzing data. This process can turn and map out large amounts of data extracted from various sources into a more useful format. Techniques such as merging, grouping, concatenating, joining, and sorting are used to analyze the data. Thereafter it gets ready to be used with another dataset.

107. What is the significance of Exploratory Data Analysis (EDA)?

- Exploratory data analysis (EDA) helps to understand the data better.
- It helps you obtain confidence in your data to a point where you’re ready to engage a machine learning algorithm.
- It allows you to refine your selection of feature variables that will be used later for model building.
- You can discover hidden trends and insights from the data.

108. What are the different types of sampling techniques used by data analysts?

Sampling is a statistical method to select a subset of data from an entire dataset (population) to estimate the characteristics of the whole population.

There are majorly five types of sampling methods:

- Simple random sampling
- Systematic sampling
- Cluster sampling
- Stratified sampling
- Judgmental or purposive sampling

109. What are some common data visualization tools you have used?

You should name the tools you have used personally, however here's a list of the commonly used data visualization tools in the industry:

- Tableau
- Microsoft Power BI
- QlikView
- Google Data Studio
- Plotly
- Matplotlib (Python library)
- Excel (with built-in charting capabilities)

110. How can you handle missing values in a dataset?

This is one of the most frequently asked data analyst interview questions, and the interviewer expects you to give a detailed answer here, and not just the name of the methods. There are four methods to handle missing values in a dataset.

Listwise Deletion

- In the listwise deletion method, an entire record is excluded from analysis if any single value is missing.

Average Imputation

- Take the average value of the other participants' responses and fill in the missing value.

Regression Substitution

- You can use multiple-regression analyses to estimate a missing value.

Multiple Imputations

- It creates plausible values based on the correlations for the missing data and then averages the simulated datasets by incorporating random errors in your predictions.

111. How do you treat outliers in a dataset?

An outlier is a data point that is distant from other similar points. They may be due to variability in the measurement or may indicate experimental errors.

The graph depicted below shows there are three outliers in the dataset.

To deal with outliers, you can use the following four methods:

- Drop the outlier records
- Cap your outliers data
- Assign a new value
- Try a new transformation

112. How can pandas be used for data analysis?

Pandas is one of the most widely used Python libraries for data analysis. It has powerful tools and data structure which is very helpful in analyzing and processing data. Some of the most useful functions of pandas which are used for various tasks involved in data analysis are as follows:

1. **Data loading functions:** Pandas provides different functions to read the dataset from the different-different formats like read_csv, read_excel, and read_sql functions are used to read the dataset from CSV, Excel, and SQL datasets respectively in a pandas DataFrame.
2. **Data Exploration:** Pandas provides functions like head, tail, and sample to rapidly inspect the data after it has been imported. In order to learn more about the different data types, missing values, and summary statistics, use pandas .info and. describe functions.
3. **Data Cleaning:** Pandas offers functions for dealing with missing values (fillna), duplicate rows (drop_duplicates), and incorrect data types (astype) before analysis.
4. **Data Transformation:** Pandas may be used to modify and transform data. It is simple to do actions like selecting columns, filtering rows (loc, iloc), and adding new ones. Custom transformations are feasible using the apply and map functions.
5. **Data Aggregation:** With the help of pandas, we can group the data using groupby function, and also apply aggregation tasks like sum, mean, count, etc., on specify columns.
6. **Time Series Analysis:** Pandas offers robust support for time series data. We can easily conduct date-based computations using functions like resample, shift etc.
7. **Merging and Joining:** Data from different sources can be combined using Pandas merge and join functions.