**1.What is Python? And it’s used for?**

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics developed by Guido van Rossum.

Python is a General purpose programming language we can use it for variety of different tasks.

It Supports structured and unstructured data

Python code is simple, concise and easy to read.

**Used For: used for tasks like:**

data analysis and visualization

Artificial intelligence

Machine learning

Automation

Web apps

Testing

Hacking

Mobile apps

**2.What are the advantages of Python?**

Python Advantages and Disadvantages

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| * Easy to read and learn * Reduces maintenance costs * Easy Memory Management * Wide applicability * Large community * Easy integration with other programming languages * Asynchronous coding * Portable * Enterprise Application Integration | * Slow execution speed * Large memory consumption * Not ideal for mobile and game development * Addiction to easy coding * Weak database access * Design restrictions * More prone to runtime errors |

**3.Is Python a compiled language or an interpreted language?**

Python is an interpreted language, which means the source code of a Python program is converted into bytecode that is then executed by the Python virtual machine. Python is different from major compiled languages, such as C and C + +, as Python code is not required to be built and linked like code for these languages.

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

Mutable data type is those whose values can be changed, whereas immutable data type is one in which the values can't be changed.

**5.What is a lambda function?**

Lambda functions are similar to user-defined functions but without a name. They're commonly referred to as anonymous functions. Lambda functions are efficient whenever you want to create a function that will only contain simple expressions – that is, expressions that are usually a single line of a statement.

**6.Difference between for loop and while loop in Python**

For loop is used to iterate over a sequence of items. While loop is used to repeatedly execute a block of statements while a condition is true. For loops are designed for iterating over a sequence of items.

**7.What is Scope in Python?**

The scope determines the visibility of the names and objects in the program. A namespace is a collection of names. In python, scopes are implemented as dictionaries that maps names to objects. These dictionaries are called namespaces.

**8.What are Built-in data types in Python?**

**Some built-in Python data types are:**

* Numeric data types: int, float, complex.
* String data types: str.
* Sequence types: list, tuple, range.
* Binary types: bytes, bytearray, memoryview.
* Mapping data type: dict.
* Boolean type: bool.
* Set data types: set, frozenset. Python Numeric Data Type.

**9.Differentiate between List and Tuple?**

The primary difference between tuples and lists is that tuples are immutable as opposed to lists which are mutable. Therefore, it is possible to change a list but not a tuple. The contents of a tuple cannot change once they have been created in Python due to the immutability of tuples.

**10.Explain Python Functions?**

A function is a set of statements or a piece of code grouped together to perform a specific task

* In built function
* User defined function
* Lambda function

**Inbuilt functions** are len( ), max( ),abs( ), Sqrt()

We can define our own functions.

Function should be defined before calling.

Syntax:

def Functionname(args):

Statements

Return //null or any value

Functionname(args)

def addition(a,b):

Sum=a+b

Return sum

addition(2,3)

**Recursive function**

Function that calls itself

Function needs to be defined before calling.

def functionName(args):

statements

return functionName(args)

functionName(agrs)

**Lambda function**

* Lambda functions can take any number of arguments, but they can only have one expression

a,b a+b

x y z :x\*y\*z

a a\*\*2

* The expression in a lambda function can be any valid Python expression, including mathematical operations.
* Lambda functions do not have a return statement; the value of the expression is automatically returned.
* Lambda functions do not have a name, so they cannot be referred to after they are defined. They are typically used as arguments to other functions.

**lambda n: n + 1**

**lambda a: a\*\*2**

* **when to use this lambda function?**

When a function is going to be used single time.

* **When it is not advised to use lambda function?**

When the logic is too complex

When function needs to be reused

**11.What are the different types of operators in Python?**

**Types of Python Operators**

* Arithmetic Operators.
* Comparison (Relational) Operators.
* Assignment Operators.
* Logical Operators.
* Bitwise Operators.
* Membership Operators.
* Identity Operators.

**12.What is a dictionary in Python?**

A Python dictionary is a data structure that stores key/value pairs to provide information about the structure.

**13.What is a negative index in Python and why are they used?**

Negative indexing is used in Python to manipulate sequence objects such as lists, arrays, strings, etc. Negative indexing retrieves elements from the end by providing negative numbers as sequence indexes.

**14.What is Data Science?**

Data science is the study of data to extract meaningful insights for business. It is a multidisciplinary approach that combines principles and practices from the fields of mathematics, statistics, artificial intelligence, and computer engineering to analyze large amounts of data.

**15.Differentiate Between Data Analytics and Data Science?**

While both fields involve working with data to gain insights, data analytics tends to focus more on analyzing past data to inform decisions in the present, while data science often involves using data to build models that can predict future outcomes.

**16.How is logistic regression done?**

In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1. Such as values above the threshold value tends to 1, and a value below the threshold values tends to 0.

**17.How can you select k for k-means?**

**Here's how we can do it.**

1. Choose the number of clusters k. The first step in k-means is to pick the number of clusters, k.
2. Select k random points from the data as centroids. ...
3. Assign all the points to the closest cluster centroid. ...
4. Recompute the centroids of newly formed clusters. ...
5. Repeat steps 3 and 4.

**18.How can you calculate accuracy using a confusion matrix?**

Accuracy is calculated as the number of correct predictions divided by the total number of predictions made by the model.

The overall accuracy is calculated by summing the number of correctly classified values and dividing by the total number of values. The correctly classified values are located along the upper-left to lower-right diagonal of the confusion matrix.

**19.What are the differences between supervised and unsupervised learning?**

The main difference between supervised vs unsupervised learning is the need for labelled training data. Supervised machine learning relies on labelled input and output training data, whereas unsupervised learning processes unlabelled or raw data.

**20.How can outlier values be treated?**

Three main methods of dealing with outliers, apart from removing them from the dataset: 1) reducing the weights of outliers (trimming weight) 2) changing the values of outliers (Winsorisation, trimming, imputation) 3) using robust estimation techniques (M-estimation).

**21.Write the equation and calculate the precision and recall rate.**

* Precision = TruePositives / (TruePositives + FalsePositives)
* Recall = TruePositives / (TruePositives + FalseNegatives)

**22.What is a Confusion Matrix?**

A confusion matrix is a table used in machine learning and statistics to assess the performance of a classification model. It summarizes the results of classification by showing the counts of true positive, true negative, false positive, and false negative predictions.

**23.What are the popular libraries used in Data Science?**

NumPy. Picture this: you're diving into the realm of data analysis, and you need a tool that can handle vast amounts of numerical data with lightning-fast speed. ...

Panda. Discover the power of Pandas, the must-have Python library for data science. ...

Matplotlib. ...

Seaborn. ...

Scikit-learn. ...

TensorFlow. ...

PyTorch. ...

Keras.

**24.What information is gained in a decision tree algorithm?**

The information gained in the decision tree can be defined as the amount of information improved in the nodes before splitting them for making further decisions.

**25.What is Deep Learning?**

Deep Learning is a part of Machine Learning used to solve complex problems and build intelligent solutions. The core concept of Deep Learning has been derived from the structure and function of the human brain. Deep Learning uses artificial neural networks to analyze data and make predictions.

**26. Difference between Normalisation and Standardization**

Standardization is divided by the standard deviation after the mean has been subtracted. Data is transformed into a range between 0 and 1 by normalization, which involves dividing a vector by its length.

**27.Explain Naive Bayes and linear regression**

**Naïve Bayes:**

 It is a classification technique based on Bayes' Theorem with an independence assumption among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

The Naïve Bayes classifier is a supervised machine learning algorithm, which is used for classification tasks, like text classification. It is also part of a family of generative learning algorithms, meaning that it seeks to model the distribution of inputs of a given class or category.

**Linear Regression:**

Linear regression is a data analysis technique that predicts the value of unknown data by using another related and known data value. It mathematically models the unknown or dependent variable and the known or independent variable as a linear equation.

**28.Differentiate between Machine Learning, Data Science, and AI.**

Data Science involves analysis, visualization, and prediction. It uses different statistical techniques, while AI and Machine Learning implements models to predict future events and makes use of algorithms

**29.Difference between Regression and Classification**

The key distinction between Classification vs Regression algorithms is Regression algorithms are used to determine continuous values such as price, income, age, etc. and Classification algorithms are used to forecast or classify the distinct values such as Real or False, Male or Female, Spam or Not Spam, etc..

**30.Explain lambda function with example**

Refer Q.NO:5

**31.Explain exception handling.**

The requirement for handling exceptions in Python arises when an error occurs that can cause the program to terminate. Errors interrupt the flow of the program at the point where they appear, so any further code stops executing. This error is called an exception.

**39.Is it possible to handle indentation and syntax errors using exception handling.**

Yes, exception handling in Python allows you to catch and handle various types of errors, including indentation errors and syntax errors. However, it's important to note that syntax errors are typically detected by the Python interpreter before the program even starts running, so they cannot be caught and handled using the **try** and **except** blocks. Indentation errors are also detected during the parsing phase, so they can't be caught at runtime.

Here's a simple example illustrating exception handling for runtime errors:

try:

# Some code that might raise an exception

x = 10 / 0 # This will raise a ZeroDivisionError

except ZeroDivisionError as e:

print(f"Error: {e}")

except Exception as e:

print(f"Another type of error occurred: {e}")

else:

print("No error occurred.")

finally:

print("This block will always be executed.")

In this example, if a **ZeroDivisionError** occurs, it will be caught and a message will be printed. If any other type of exception occurs, it will be caught by the generic **Exception** block. The **else** block will be executed if no exception occurs, and the **finally** block will always be executed, whether an exception occurred or not.

Remember that syntax errors and indentation errors are usually fixed during the development phase, as they prevent the code from being executed in the first place. Once the code is correct in terms of syntax and indentation, you can use exception handling for runtime errors that might occur during the execution of the program.

**43.Is it possible to build a Python program that reverses a number?**

Yes, it is certainly possible to build a Python program that reverses a number. Here's a simple example:

def reverse\_number(number):

# Convert the number to a string, reverse it, and convert back to an integer

reversed\_number = int(str(number)[::-1])

return reversed\_number

# Example usage

original\_number = 12345

reversed\_number = reverse\_number(original\_number)

print(f"Original number: {original\_number}")

print(f"Reversed number: {reversed\_number}")

**44.What is the best way to add items to a Python array?**

In Python, there are several ways to add items to a list (Python does not have a built-in array type, but lists are commonly used as dynamic arrays). Here are some common methods:

**1.append() method:**

* This method adds an element to the end of the list.

my\_list = [1, 2, 3]

my\_list.append(4)

1. After this operation, **my\_list** will be **[1, 2, 3, 4]**.
2. **2.extend() method:**
   * This method is used to append the elements of an iterable (e.g., another list) to the end of the list.

my\_list = [1, 2, 3]

my\_list.extend([4, 5, 6])

1. After this operation, **my\_list** will be **[1, 2, 3, 4, 5, 6]**.
2. **3.insert() method:**
   * This method is used to insert an element at a specific position in the list.

my\_list = [1, 2, 3]

my\_list.insert(1, 4) # Insert 4 at index 1

1. After this operation, **my\_list** will be **[1, 4, 2, 3]**.
2. **4.Using the + operator:**
   * You can concatenate two lists using the **+** operator.

my\_list = [1, 2, 3]

my\_list = my\_list + [4, 5, 6]

1. After this operation, **my\_list** will be **[1, 2, 3, 4, 5, 6]**.
2. **5.List comprehension:**
   * You can use list comprehension to generate a new list and then assign it to the original list.

my\_list = [1, 2, 3]

my\_list = my\_list + [x for x in range(4, 7)]