NAME – Shaurya Teotia

BATCH – 6th June

DOMAIN – Data Science with ML/AI

ASSESSMENT

1.Mention some of the advantages of Python?

Ans. i) **Simple to Use and Understand**

**ii) Free and Open-Source**

**iii) Productivity has Increased**

**iv) Interpreted Language**

**v) Many in-built libraries**

2. What are local variables and global variables in Python?

Ans. There are two types of variables: global variables and local variables. The scope of global variables is the entire program whereas the scope of local variables is limited to the function where it is defined. Local variables can only be reached within their scope. A global variable can be used anywhere in the program as its scope is the entire program.

3. What are Lambda Functions in Python?

Ans. Lambda Functions in Python are anonymous functions, implying they don't have a name. use the lambda keyword in Python to define an unnamed function. The syntax of the Lambda Function is given below –

lambda arguments: expression

This function accepts any count of inputs but only evaluates and returns one expression. That means it takes many inputs but returns only one output.

Eg. the lambda function to add two numbers

add = lambda num: a+b

print( add(6,4) )

4. What is a Negative Index in Python?

Ans. Negative Indexing is used in Python to begin slicing from the end of the string i.e. the last. Slicing in Python gets a sub-string from a string. For negative indexing, set the start and stop as negative values i.e. slice from the end.

5. What is the difference between tuples and lists?

Ans.

|  |  |
| --- | --- |
| List | Tuple |
| It is mutable | It is immutable |
| The implication of iterations is time-consuming in the list. | Implications of iterations are much faster in tuples. |
| Operations like insertion and deletion are better performed. | Elements can be accessed better. |
| Consumes more memory. | Consumes less memory. |
| Many built-in methods are available. | Does not have many built-in methods. |
| Unexpected errors and changes can easily occur in lists. | Unexpected errors and changes rarely occur in tuples. |

6. What is a dynamically typed language?

Ans. A language is **dynamically typed** if the type of a variable is checked during **run-time**. Common examples of dynamically-typed languages includes JavaScript, Objective-C, PHP, Python, Ruby, and Lisp.

Dynamically-typed languages do not require you to declare the data types of your variables before you use them. Here, variables are bound to objects at run-time by means of assignment statements, and it is possible to bind the same variables to objects of different types during the execution of the program.

7. What are the data types available in Python? Brief about it?

Ans. The following is a list of the Python-defined data types:

### Numbers:-

Int: Whole number worth can be any length, like numbers 10, 2, 29, - 20, - 150, and so on. An integer can be any length you want in Python. Its worth has a place with int.

Float: Float stores drifting point numbers like 1.9, 9.902, 15.2, etc. It can be accurate to within 15 decimal places.

Complex: An intricate number contains an arranged pair, i.e., x + iy, where x and y signify the genuine and non-existent parts separately. The complex numbers like 2.14j, 2.0 + 2.3j, etc.

### Sequence Type

String: The sequence of characters in quotation marks can be used to describe the string. A string can be defined in Python using single, double, or triple quotes.

Lists: Lists in Python are like arrays in C, but lists can contain data of different types. The things put away in the rundown are isolated with a comma (,) and encased inside square sections [].

Tuple: A tuple is like a list. Tuples, like lists, also contain a collection of items from various data types. A parenthetical space () separates the tuple's components from one another. Because we cannot alter the size or value of the items in a tuple, it is a read-only data structure.

Dictionary: It is a key-value pair set arranged in any order. It stores a specific value for each key, like an associative array or a hash table. Value is any Python object, while the key can hold any primitive data type.

### Boolean

True and False are the two default values for the Boolean type. These qualities are utilized to decide the given assertion valid or misleading.

### Set

The data type's unordered collection is Python Set. It is iterable, mutable, and has remarkable components. The elements of a set have no set order; It might return the element's altered sequence. Either a sequence of elements is passed through the curly braces and separated by a comma to create the set or the built-in function set() is used to create the set. It can contain different kinds of values.

### Dictionary

### It is a key-value pair set arranged in any order. It stores a specific value for each key, like an associative array or a hash table. Value is any Python object, while the key can hold any primitive data type. The comma (,) and the curly braces are used to separate the items in the dictionary.

8. Why do we have to use functions in Python? Explain with an example.

Ans. Functions are an essential part of most programming languages. Functions are reusable pieces of code that can be called using a function's name. Functions can be called anywhere in a Python program, including calling functions within other functions.

Functions provide a couple of benefits:

Functions allow the same piece of code to run multiple times

Functions break long programs up into smaller components

Functions can be shared and used by other programmers

Eg. user-defined function

def my\_function():  
print("Hello from a function")  
  
my\_function()

9. What is the difference between function and generators? Give an example.

Ans. Generator functions are defined as the normal function, but to identify the difference between the normal function and generator function is that in the normal function, we use the [return keyword](https://www.scaler.com/topics/return-in-python/) to return the values, and in the generator function, instead of using the return, we use yield to execute our iterator.

Eg. #generator function

def gen\_fun():

yield 10

yield 20

yield 30

for i in gen\_fun():

print(i)

| Yield | Return |
| --- | --- |
| It is used in generator functions. | It is used in normal functions. |
| It is responsible for controlling the flow of the generator function. After returning the value from yield, it pauses the execution by saving the states. | The return statement returns the value and terminates the function. |

## Difference Between Generator Function & Normal Function

In generator functions, there are one or more yield functions, whereas, in Normal functions, there is only one function

When the generator function is called, the normal function pauses its execution, and the call is transferred to the generator function.

Local variables and their states are remembered between successive calls.

When the generator function is terminated, Stop Iteration is called automatically on further calls.

10. Explain briefly about conditional statements with an example?

Ans. Conditional statements are also called decision-making statements. We use those statements when we want to execute a block of code when the given condition is true or false.

Type of condition statement in Python:

## If statement: It is most usually used as a conditional statement. Eg.

## a = 10

## b = 20

## if a<b:

## print("a is less than b")

## If else statement: It is a conditional statement. The statement itself says that if a given condition is true or false. True means executing the “if” statement to the output. False means executing the “else” statement to the output. Eg.

## a = 10

## b = 20

## if a==b:

## print("a and b are equal")

## else:

## print("a and b are not equal")

## Elif statement: It is a shortcut for else if condition statements. In Python, one or more conditions are used in the elif statement. Eg.

## a = 10

## b = 10

## if a < b:

## print("a is greater than b")

## elif a == b:

## print("a and b are equal")

## else:

## print("b is greater than a")

## Nested if statement: In Python using "if" statements inside other if statements it is called a nested if statement. Eg.

## a= 1001

## if a> 100:

## print("Above 100")

## if a > 1000:

## print("and also above 1000")

## Nested if else statement: In Python using one “if else” statement inside other if else statements it is called a nested if else statement. Eg. a=int(input("enter the a value"))#user give a value

## if a> 100:

## print("Above 100")

## if a > 1000:

## print("and also above 1000")

## else:

## print("and also below 1000")

## else:

## print("below 100")

11. Why we have to use exception handling in Python? What is the keyword used to handle the error, give one example.

Ans. Exception handling is the process of responding to unwanted or unexpected events when a computer program runs. Exception handling deals with these events to avoid the program or system crashing, and without this process, exceptions would disrupt the normal operation of a program. Python uses the following keywords to handle exceptions. These keywords are followed by indented blocks.

try: block contains one or more statements that are likely to encounter an exception.

exception: If the exception does occur, the program flow is transferred to the except block. The statements in the except block are meant to handle the cause of the exception appropriately.

else: the else block gets processed if the try block is found to be exception free.

finally: The final block consists of statements that should be processed regardless of whether an exception occurs in the try block or not.

12. Define class and objects with an example?

Ans. A class is a blueprint that defines some properties and behaviors. An object is an instance of a class that has those properties and behaviors attached. A class is not allocated memory when it is defined. An object is allocated memory when it is created. Class is a logical entity whereas objects are physical entities. Eg.

# create class

Class Bike:

name=””

gear=0

#create objects of class

bike1=Bike()

13. Write a program to swap the two numbers in Python?

Ans. #code to swipe 2 numbers:

a= int( input("Please enter value for a: "))

b = int( input("Please enter value for b: "))

temp = a

a = b

b = temp

print ("The Value of a after swapping: ", a)

print ("The Value of b after swapping: ", b)

14. Write a palindrome program in Python?

Ans. #palindrome code

n=int(input("Enter number:"))

temp=n

rev=0

while(n>0):

dig=n%10

rev=rev\*10+dig

n=n//10

if(temp==rev):

print("The number is a palindrome!")

else:

print("The number isn't a palindrome!")

15 What is the use of NumPy array over list?

Ans. Advantages of using Numpy Arrays Over Python Lists:

Consumes less memory.

Fast as compared to the Python List.

Convenient to use.

16. Explain briefly about reshaping in NumPy with an example?

Ans. Reshaping means changing the shape of an array. The shape of an array is the number of elements in each dimension. By reshaping we can add or remove dimensions or change a number of elements in each dimension. We can reshape array into any shape as long as the elements required for reshaping are equal in both shapes. Eg. #1D array into 3D array:

import numpy as np  
arr = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12])  
newarr = arr.reshape(2, 3, 2  
print(newarr)

17. what is the use of pandas explain in brief?

Ans. pandas is a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labeled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real-world data analysis in Python. It is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data. It has the following uses:

Pandas allow us to analyze big data and make conclusions based on statistical theories.

Pandas can clean messy data sets, and make them readable and relevant.

Relevant data is very important in data science.

18 What do you mean by supervised machine learning and unsupervised machine learning? Explain in brief.

Ans. Supervised learning is a machine learning method in which models are trained using labeled data. In supervised learning, models need to find the mapping function to map the input variable (X) with the output variable (Y).

Y=f(x)

Supervised learning needs supervision to train the model, which is similar to how a student learns things in the presence of a teacher. Supervised learning can be used for two types of problems: Classification and Regression.

Unsupervised learning is another machine learning method in which patterns are inferred from the unlabeled input data. The goal of unsupervised learning is to find the structure and patterns from the input data. Unsupervised learning does not need any supervision. Instead, it finds patterns from the data on its own. Unsupervised learning can be used for two types of problems: **Clustering** and **Association**.

19 Differentiate between Decision tree and Random forest?

Ans.

|  |  |
| --- | --- |
| Decision Tree | Random Forest |
| A decision tree is a tree-like model of decisions along with possible outcomes in a diagram. | A classification algorithm consisting of many decision trees combined to get a more accurate result as compared to a single tree. |
| There is always a scope for overfitting, caused due to the presence of variance. | Random forest algorithm avoids and prevents overfitting by using multiple trees. |
| The results are not accurate. | This gives accurate and precise results. |
| Decision trees require low computation, thus reducing time to implement and carrying low accuracy. | This consumes more computation. The process of generation and analysis is time-consuming. |
| It is easy to visualize. The only task is to fit the decision tree model. | This has complex visualization as it determines the pattern behind the data. |

20. What do you mean by forward propagation and backward propagation in ANN?

Ans. Forward propagation is where input data is fed through a network, in a forward direction, to generate an output. The data is accepted by hidden layers and processed, as per the [activation function](https://h2o.ai/wiki/activation-function), and moves to the successive layer.

Backpropagation is just a way of propagating the total loss back into the neural network to know how much of the loss every node is responsible for, and subsequently updating the weights in a way that minimizes the loss by giving the nodes with higher error rates lower weights, and vice versa.

21. Explain the working flow of ANN and CNN?

Ans. Working of ANN:

Artificial Neural Network can be best represented as a weighted directed graph, where the artificial neurons form the nodes. The association between the neurons outputs and neuron inputs can be viewed as the directed edges with weights. The Artificial Neural Network receives the input signal from the external source in the form of a pattern and image in the form of a vector. These inputs are then mathematically assigned by the notations x(n) for every n number of inputs.

Afterward, each of the inputs is multiplied by its corresponding weights ( these weights are the details utilized by the artificial neural networks to solve a specific problem ). In general terms, these weights normally represent the strength of the interconnection between neurons inside the artificial neural network. All the weighted inputs are summarized inside the computing unit.

If the weighted sum is equal to zero, then bias is added to make the output non-zero or something else to scale up to the system's response. Bias has the same input, and weight equals 1. Here the total of weighted inputs can be in the range of 0 to positive infinity. Here, to keep the response within the limits of the desired value, a certain maximum value is benchmarked, and the total of weighted inputs is passed through the activation function.

The activation function refers to the set of transfer functions used to achieve the desired output. There is a different kind of activation function, but primarily either linear or non-linear sets of functions. Some of the commonly used sets of activation functions are the Binary, linear, and Tan hyperbolic sigmoidal activation functions.

Working of CNN:

Convolutional neural networks are composed of multiple layers of artificial neurons. Artificial neurons, a rough imitation of their biological counterparts, are mathematical functions that calculate the weighted sum of multiple inputs and output an activation value. When you input an image in a ConvNet, each layer generates several activation functions that are passed on to the next layer.

The first layer usually extracts basic features such as horizontal or diagonal edges. This output is passed on to the next layer which detects more complex features such as corners or combinational edges. As we move deeper into the network it can identify even more complex features such as objects, faces, etc.

Based on the activation map of the final convolution layer, the classification layer outputs a set of confidence scores (values between 0 and 1) that specify how likely the image is to belong to a “class.” For instance, if you have a ConvNet that detects cats, dogs, and horses, the output of the final layer is the possibility that the input image contains any of those animals.

22. what is the use of activation functions and optimizers?

Ans. Use of activation function:

The activation function decides whether a neuron should be activated or not by calculating the weighted sum and further adding bias to it. The purpose of the activation function is to introduce non-linearity into the output of a neuron.

Use of optimizers:

An optimizer is an algorithm that helps another algorithm to reach its peak performance without delay. With respect to machine learning (neural network), we can say an optimizer is a mathematical algorithm that is used to help our loss function reach its convergence point with minimum delay (and most importantly, reduce the posibility of gradient explosion). Examples include, adam, stochastic gradient descent (SGD), adadelta, rmsprop, adamax, adagrad, nadam etc.

Optimizers are a key part of the training process for deep learning models. They are responsible for **adjusting the model parameters** to minimize the loss function, which measures how well the model can make predictions on a given dataset. Different optimization algorithms are available, and choosing which can significantly impact the model's performance. They are used to adjust the parameters of a model to minimize a loss function. The choice of optimizer can greatly affect the performance and speed of training a model.

23. What is the difference between R square and adjusted R square?

Ans. The difference between the R squared and adjusted R squared value is that R squared value assumes that all the independent variables considered affect the result of the model, whereas the adjusted R squared value considers only those independent variables which actually have an effect on the performance of the model.

24. What do you mean by LSTM in RNN?

Ans. The Long Short-Term Memory (short: LSTM) model is a subtype of Recurrent Neural Networks (RNN). It is used to recognize patterns in data sequences, such as those that appear in sensor data, stock prices, or natural language. RNNs can do this because, in addition to the actual value, they also include its position in the sequence in the prediction.

The problem with Recurrent Neural Networks is that they simply store the previous data in their “short-term memory”. Once the memory in it runs out, it simply deletes the longest retained information and replaces it with new data. The LSTM model attempts to escape this problem by retaining selected information in long-term memory. This long-term memory is stored in the so-called Cell State. In addition, there is also the hidden state, which we already know from normal neural networks and in which short-term information from the previous calculation steps is stored. The hidden state is the short-term memory of the model. This also explains the name Long Short-Term Networks.

25. What is the use of image processing?

Ans. Various application of image processing are as follow –

### Image Enhancement

The process of improving the quality of an image is known as **Image Enhancement**.

In the image enhancement process, we perform various operations on the pixel of the given image.

Image enhancement techniques are classified into two category

**Image Enhancement in Spatial Domain**

**Image enhancement in Frequency Domain**

### Image Restoration

**Image Restoration** is the process of obtaining the original image from a degraded image. In Image processing applications various types of image restoration functions are used.

### Medical Field

The common application of image processing in the field of medicine are as follow

Gamma-ray imaging

PET scan

X-Ray Imaging

Medical CT

UV imaging

### Pattern Recognition

Pattern Recognition is a part of image processing that involves the study of image processing along with the use of artificial intelligence and machine learning.

Some examples of pattern recognition-based applications are computer-aided diagnosis, handwriting recognition, and image recognition can be easily implemented.

### Video Processing

Video is basically a fast movement of images. Various image processing techniques are used in Video Processing.

**Some methods of image processing used in Video Processing are noise removal, image stabilization, frame rate conversion, detail enhancement, and much more.**

### Computer / Machine Vision

One of the most interesting and useful applications of Image Processing is in Computer Vision.

[**Computer Vision**](https://www.ibm.com/in-en/topics/computer-vision#:~:text=Computer%20vision%20is%20a%20field,recommendations%20based%20on%20that%20information.) can also be seem as high level image processing. Computer vision is used to make the computer see, identify things, and process the whole environment as a whole.

An important use of Computer Vision is Self Driving cars, Drones etc. Computer vision helps in obstacle detection, path recognition, and understanding the environment.

26.What are the steps involved in NLP?

Ans. There are general five steps −

Lexical Analysis − It involves identifying and analyzing the structure of words. Lexicon of a language means the collection of words and phrases in a language. Lexical analysis is dividing the whole chunk of txt into paragraphs, sentences, and words.

Syntactic Analysis (Parsing) − It involves the analysis of words in the sentence for grammar and arranging words in a manner that shows the relationship among the words. A sentence such as “The school goes to the boy” is rejected by an English syntactic analyzer.

Semantic Analysis − It draws the exact meaning or the dictionary meaning from the text. The text is checked for meaningfulness. It is done by mapping syntactic structures and objects in the task domain. The semantic analyzer disregards sentences such as “hot ice cream”.

Discourse Integration − The meaning of any sentence depends upon the meaning of the sentence just before it. In addition, it also brings about the meaning of the immediately succeeding sentence.

Pragmatic Analysis − During this, what was said is re-interpreted on what it actually meant. It involves deriving those aspects of language which require real-world knowledge.

27. What do you mean by bias and variance?

Ans. Bias:

A machine learning model analyses the data, finds patterns in it, and makes predictions. While training, the model learns these patterns in the dataset and applies them to test data for prediction*.*While making predictions, a difference occurs between prediction values made by the model and actual values/expected values*,*and this difference is known as bias errors or Errors due to bias*.* It can be defined as an inability of machine learning algorithms such as Linear Regression to capture the true relationship between the data points. Each algorithm begins with some amount of bias because bias occurs from assumptions in the model, which makes the target function simple to learn. A model has either:

Low Bias: A low-bias model will make fewer assumptions about the form of the target function.

High Bias: A model with a high bias makes more assumptions, and the model becomes unable to capture the important features of our dataset. A high-bias model also cannot perform well on new data.

Variance:

It would specify the amount of variation in the prediction if the different training data was used*.* In simple words*,*variance tells how much a random variable is different from its expected value.Ideally, a model should not vary too much from one training dataset to another, which means the algorithm should be good at understanding the hidden mapping between inputs and output variables. Variance errors are either of **low variance or high variance.**

**Low variance** means there is a small variation in the prediction of the target function with changes in the training data set. At the same time, **High variance** shows a large variation in the prediction of the target function with changes in the training dataset.

28. What are the steps involved in machine learning to create a model?

Ans. following are the steps to create a machine learning model:

Data Ingestion: collect and read data.

EDA (Exploratory Data Analysis): Understanding the dataset using NumPy, pandas, matplotlib, and seaborn and making the analysis.

Pre-Processing (Feature Engineering): Make required changes in the dataset. eg. , remove duplicate values, fill null values, detect outliers, do the encoding (label, one-hot or ordinal encoding), split dataset applies to scale on the dataset (eg. standard, min, max, unique scaling), etc.

Model Creation: Apply algorithms like simple linear regression, multi-linear regression, poly linear regression, decision tree, random forest, naïve Bayes, SVC, etc.

Evaluation & Validation: Check the accuracy and score of each algorithm & find the best algorithm that suits the dataset.