ASSIGNMENT-1

Name:Gouni Sushma Rollno:21UK1A0505

1.Define Artifical Intelligence(AI) and provide examples of its & Applications

A. Artifical Intelligence is the capability of a computer program or a machine to think, learn, and perform tasks like humans.

It is the ability of a digital computer or computer controlled robot to perform tasks commonly associated with intelligent beings.

It is also defined as a set of technologies that enables computers to perform a variety of advanced functions.

Examples:

1.Maps and Navigation .

2.Facial Detection and Recognition.

3.Text editors or Autocorrect.

4.Chatbots.

5.Digital Assistants.

6.Self driving cars.

7.E-payments.

Applications of AI

1. Healthcare: AI assists in medical diagnosis, personalized treatment plans, drug discovery, and patient monitoring.

2. Finance: AI algorithms are used for fraud detection, risk assessment, algorithmic trading, and personalized financial services.

3. Autonomous Vehicles : AI powers self-driving cars and drones, enabling them to perceive their environment, make decisions, and navigate safely.

4. Cybersecurity: AI helps in threat detection, anomaly detection, and identifying patterns of cyberattacks in real-time.

5. Education: AI assists in personalized learning, adaptive tutoring, grading automation, and educational content creation.

6. Manufacturing: AI enhances process automation, quality control, predictive maintenance, and supply chain optimization in manufacturing industries.

2. Differentiate between supervised and unsupervised learning techniques in ML.

Supervised learning	Unsupervised learning
Supervised learning algorithms are trained using labelled data	Unsupervised learning algorithms are trained using unlabeled data
Supervised learning model takes direct feedback to check if it is predicting correct output or not	Unsupervised learning model does not take care any feedback
Supervised learning model predicts the output	Unsupervised model finds the hidden patterns in data
In supervised learning ,input data is provided to the model along with the output	In unsupervised learning only input data is provided to the model
The goal of supervised learning is to train the model so that it can predict the output when it is given new data.	The goal of unsupervised learning is to find the hidden patterns and useful insights from the unknown dataset
Supervised learning can be categorized in classification and regression problems	Unsupervised learning can be classified in clustering and Association problems

3. What is Python? Discuss its main features and advantages.

A. Python is a high-level programming language known for its simplicity and readability. It's widely used in various domains such as web development, data science, artificial intelligence, and more. Python emphasizes code reusability and simplicity,making it a popular choice for both beginners and experienced programmers.

Features of python:

1. Easy to Learn and Readability: Python emphasizes readability and simplicity, making it accessible for beginners and enjoyable for experienced programmers.

2. Versatility: Python is a versatile language with a wide range of applications, from web development and data analysis to artificial intelligence and scientific computing.

3. Large Standard Library: Python comes with a comprehensive standard library that provides support for various tasks without requiring additional installations.

4. Platform Independence: Python is available on multiple platforms, including Windows, macOS, and Linux, ensuring compatibility across different operating systems.

5. Object-Oriented: Python supports object-oriented programming paradigms, allowing for the creation and manipulation of objects to model real-world entities.

6. Dynamic Typing: Python uses dynamic typing, meaning you don't need to declare variable types explicitly, making code writing more flexible and concise.

Advantages:

1.Ease of use
2.Readability
3.Concise and straightforward syntax,and vast libraries.
4.Simplicity
5.Interpreted language
6.Platform independence
7.Versatility
8.Community and support.

4.What are the advantages of using Python as a programming language for AI and ML?

A. *Python is a popular choice for AI and ML because it has some awesome advantages.

*It has a simple and readable syntax, making it easy to understand and write code.

*It has a vast collection of libraries like TensorFlow and PyTorch that provide powerful tools for AI and ML tasks.

*Python has a strong community support, which means there are plenty of resources and tutorials available. It's a great language to dive into AI and ML.

* it surpasses java in popularity and has many advantages,such as good visualization options, Great library ecosystem,Allows entry barrier,community support,flexibility,Readability, and platform independence.

---

5. Discuss the importance of indentation in Python code.

A. The primary purpose of integration in Python is to define the scope of statements such as those within loops, conditionals, functions and classes. Consistent and proper intendation is crucial for the interpreter to understand the logical structure of the code. It is used to define the structure and hierarchy of the code. Intendation helps to improve code readability and ensures the Code structured correctly.

1. Syntax Structure:

In Python, indentation is used to define the structure of the code. Unlike languages like C++ or Java, where braces ({}) are used to indicate code blocks, Python uses indentation. This means that incorrect indentation can lead to syntax errors or, worse, alter the logical structure of the code.

2. Readability:

Proper indentation makes code more readable. When indentation is consistent and logical, it's easier for developers to understand the flow of control in the code. This is particularly important when working on collaborative projects or when revisiting code after some time.

3.Debugging:

Well-indented code is easier to debug. When there's a bug in your code, correct indentation can help you quickly identify the scope of the issue and narrow down the section of code that needs to be examined.

4.Maintainability:

Consistent indentation makes code easier to maintain. When adding new features, making modifications, or debugging issues, developers can navigate and understand the code more efficiently if it follows a clear indentation style.

5.Define a variable in Python. Provide examples of valid variable names.

A.* A variable is the name given to a memory location a value holding python variable is also known as an identifier.

*Variables names must begin with a letter or underscore, but they can be a group of both letters and digits.The name of the variable should be written in lower case.

*Variable names can be any length can have capitalized, lower case (start to finish ,a to z) ,the digit (0-9), and highlight character(_).

Examples of valid variable names:

1.name=”rubi”

Age =20

marks=80.50

print(name)

print(age)

print(marks)

2._age=20

print(_age)

3.name1=”aadi”

print(name1)

4. x6=2

print(x6)

o/p:

rubi

20

80.5

5.marks_of_sanju=40

print(marks_of_sanju)

6.Explain the difference between a keyword and an identifier in Python.

Keyword	Identifier
A python keyword is a reserved word which can’t use as a name of variable, class ,function, module and object	Python identifiers are user defined names they are used to specify the names of

	variables ,functions class, module and objects
It has a predefined meaning and purpose in the language	Identifiers must follow certain rules such as starting with a letter or underscore and can't be the same as python keywords
Ex; If ,else. For, while, del,class ,import from ,return, and ,del, assert ,or, not, try	Ex; Variable names: 'x',' count', 'total_ amount'. function names :'print _result'. class names :'customer' module names :'math', 'random'

7.List the basic data types available in Python.

Numeric types

1.Integer

2.float

Sequence type

1. string

2.list

3.tuple

Mapping type

1. Dictionary

2.set types

Boolean type

1.Bool

2. nun type

NUMERIC TYPES

Numeric values are stored in numbers the whole numbers,float, and complex qualities have a place with a python numbers data type.

Ex;

A=5

Print("the type of a",type(a))

B=40.5

Print("the type of b",type(b))

o/p:

The type of A< class ‘ int’>

The type of B<class’float’>

SEQUENCE TYPE

1.String:

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

Ex: S=”hello everyone pleasure meeting you all”

Print(S)

2.List:

List 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 [ ] .

To gain access to the list data we can use slice[:] operators like how they worked with strings ,the list is handled by the concatenation operator(+) and the repetition operator(*)

Ex:

List=[1,”hi”,”well”,2]

Print(type(list))

Print(list[3:])

Print(list[0:2])

o/p:

[1,’hi’,’well’,2]

[2]

[1,’hi’,’well’]

3.Tuple:

In many ways a tuple is like a list tuples like list. 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.

Ex:

tuple =(“Hi”,” Python”, 2)

Print(type(tuple))

MAPPING TYPES

1.Dictionary:

A dictionary 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 curly braces are used to separate the items in the dictionary.

Ex:

A={1,:'alexa',2:'mike'}

Print(d)

Print(d.keys())

2.Set types:

Sets are commonly enclosed within Curly Braces .{ } sets don't allow repeats if you try to add something already there nothing changes. You can add or remove things from your set after creating it .Sets can hold different types of things.

Ex:

set ={7,9,4,2,6}

set

o/p:

{2,4,6,7,9}

BOOLEAN TYPE:

1.bool:

Represents truth values either true or false.

2. None type :

none represents the absence of value or a null value.

In [ ]:

8.Describe the syntax for an if statement in Python.

A. if statement:

Executes a block of code only if a specified condition is true.

Syntax:

If condition

#code

Ex:

a=10

if a>0

print("a is positive number")

o/p:

a is positive number

---

9. Explain the purpose of the elif statement in Python.

A. elif Stands for Elseif and is used in python programming to test multiple conditions .It is written following an if statement in python to check an alternative condition .If the first condition is false the code block under the if statement will be executed only if its condition is true.

Purpose of elif statement is to allow for structured and multilevel decision making processes with in the program.

Syntax:

```
If condition
    #code
elif condition
    #code
else:
    #code
```

Ex:

```
  e=20
  If e>50
     Print(“ e is greater than 50”)
  elif e==20
     Print(“e is  equal to 20”)
  else:
     Print(“ e is not greater than  50”)
```
