ASSIGNMENT

1.Define Artificial Intelligence (AI) and provide examples of its applications.

Ans: AI refers to the simulation of human intelligence in machines that are programmed to think and act like humans. It involves the development of algorithms and computer programs that can perform tasks that typically require human intelligence such as visual perception, speech recognition, decision-making, and language translation.

Uses:

Artificial Intelligence has many practical applications across various industries and domains, including:

1. **Healthcare:** AI is used for medical diagnosis, drug discovery, and predictive analysis of diseases.
2. **Finance:** AI helps in credit scoring, fraud detection, and financial forecasting.
3. **Retail:** AI is used for product recommendations, price optimization, and supply chain management.
4. **Manufacturing:** AI helps in quality control, predictive maintenance, and production optimization.
5. **Transportation:** AI is used for autonomous vehicles, traffic prediction, and route optimization.
6. **Customer service:** AI-powered chatbots are used for customer support, answering frequently asked questions, and handling simple requests.
7. **Security:** AI is used for facial recognition, intrusion detection, and cybersecurity threat analysis.
8. **Marketing:** AI is used for targeted advertising, customer segmentation, and sentiment analysis.
9. **Education:** AI is used for personalized learning, adaptive testing, and intelligent tutoring systems.

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

Ans:

SUPERVISED LEARNING	UNSUPERVISED LEARNING
Uses Known and labeled Data as input	Uses Unknown Data as input
Less Computational Complexity	More Computational Complex
Uses off-line analysis	Uses Real-Time Analysis of Data
The number of Classes is known	The number of Classes is not known
Accurate and Reliable Results	Moderate Accurate and Reliable Result

The desired output is given.	The desired, output is not given.
In supervised learning it is not possible to learn larger and more complex models than in unsupervised learning	In unsupervised learning it is possible to learn larger and more complex models than in supervised learning
In supervised learning training data is used to infer model	In unsupervised learning training data is not used.
Supervised learning is also called classification.	Unsupervised learning is also called clustering.
We can test our model.	We cannot test our model
Optical Character Recognition	Find a face in an image.

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

Ans: Python is a dynamic, high-level, free open source, and interpreted programming language. It supports object-oriented programming as well as procedural-oriented programming. In Python, we don't need to declare the type of variable because it is a dynamically typed language.

Features

1. Free and Open Source
2. Easy to code
3. Easy to Read
4. Object-Oriented Language
5. GUI Programming Support
6. High-Level Language
7. Large Community Support
8. Easy to Debug
9. Python is a Portable language
10. Large Standard Library

Advantages

1: Presence of third-party modules
2. Extensive support libraries
3. Open source and large active community base
4. Versatile, easy to read, learn, and write
5. User-friendly data structures:
6. High-level language
7. Dynamically typed language
8. Object-Oriented and Procedural programming language
9. Portable and interactive

10. Highly efficient
11. Interpreted language

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

Ans:

1. A great library ecosystem

A great choice of libraries is one of the main reasons Python is the most popular programming language used for AI. A library is a module or a group of modules published by different sources like PyPi which include a pre-written piece of code that allows users to reach some functionality or perform different actions. Python libraries provide base level items so developers don't have to code them from the very beginning every time.

ML requires continuous data processing, and Python's libraries let you access, handle and transform data.

2. A low entry barrier

Working in the ML and AI industry means dealing with a bunch of data that you need to process in the most convenient and effective way. The low entry barrier allows more data scientists to quickly pick up Python and start using Python for AI development without wasting too much effort on learning the language.

3. Flexibility

Python for machine learning is a great choice, as this language is very flexible:

It offers an option to choose either to use OOPs or scripting.

There's also no need to recompile the source code, Python developers can implement any changes and quickly see the results.

Programmers can combine Python and other languages to reach their goals.

4. Platform independence

The next advantage of python for AI and ML development is platform independence. Python is not only comfortable to use and easy to learn but also very versatile. What we mean is that Python for machine learning development can run on any platform including Windows, MacOS, Linux, Unix, and twenty-one others. To transfer the process from one platform to another, developers need to implement several small-scale changes and modify some lines of code to create an executable form of code for the chosen platform. Developers can use packages like PyInstaller to prepare their code for running on different platforms.

5. Readability

Python is very easy to read so every Python developer can understand the code of their peers and change, copy or share it. There's no confusion, errors or conflicting paradigms, and this leads to more a efficient exchange of algorithms, ideas, and tools between AI and ML professionals.

6. Good visualization options

We've already mentioned that Python offers a variety of libraries, and some of them are great visualization tools. However, for AI developers, it's important to highlight that in artificial intelligence, deep learning, and machine learning, it's vital to be able to represent data in a human-readable format.

5. Discuss the importance of indentation in Python code.

Ans: Indentation is a very important concept of Python because without properly indenting the Python code, you will end up seeing IndentationError and the code will not get compiled.
Python Indentation
Python indentation refers to adding white space before a statement to a particular block of code. In another word, all the statements with the same space to the right, belong to the same code block.
Python indentation is a way of telling a Python interpreter that the group of statements belongs to a particular block of code. A block is a combination of all these statements. Block can be regarded as the grouping of statements for a specific purpose. Most programming languages like C, C++, and Java use braces { } to define a block of code. Python uses indentation to highlight the blocks of code. Whitespace is used for indentation in Python. All statements with the same distance to the right belong to the same block of code. If a block has to be more deeply nested, it is simply indented further to the right.

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

Ans: Python Variable is containers that store values. Python is not "statically typed". We do not need to declare variables before using them or declare their type. A variable is created the moment we first assign a value to it. A Python variable is a name given to a memory location. It is the basic unit of storage in a program.

Rules for Python variables

- A Python variable name must start with a letter or the underscore character.
- A Python variable name cannot start with a number.
- A Python variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _).
- Variable in Python names are case-sensitive (name, Name, and NAME are three different variables).

- The reserved words(keywords) in Python cannot be used to name the variable in Python.

Example:

```
1. # declaring the var
   Number = 100
2. A string
   name = "John"
3. # A floating point
   salary = 1456.8
```

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

Ans**:**

Keywords in Python

Python Keywords are some predefined and reserved words in Python that have special meanings. Keywords are used to define the syntax of the coding. The keyword cannot be used as an identifier, function, or variable name. All the keywords in Python are written in lowercase except True and False. There are 35 keywords in Python 3.11.

In Python, there is an inbuilt keyword module that provides an iskeyword() function that can be used to check whether a given string is a valid keyword or not. Furthermore, we can check the name of the keywords in Python by using the kwlist attribute of the keyword module.

Identifiers in Python

Identifier is a user-defined name given to a variable, function, class, module, etc. The identifier is a combination of character digits and an underscore. They are case-sensitive i.e., 'num' and 'Num' and 'NUM' are three different identifiers in python. It is a good programming practice to give meaningful names to identifiers to make the code understandable.

We can also use the Python string isidentifier() method to check whether a string is a valid identifier or not.

Rules for Naming Python Identifiers

- It cannot be a reserved python keyword.
- It should not contain white space.
- It can be a combination of A-Z, a-z, 0-9, or underscore.
- It should start with an alphabet character or an underscore (_).
- It should not contain any special character other than an underscore ( ).

8. List the basic data types available in Python.

Ans: Python Data types are the classification or categorization of data items. It represents the kind of value that tells what operations can be performed on a particular data. Since everything is an object in Python programming, Python data types are classes and variables are instances (objects) of these classes. The following are the standard or built-in data types in Python:

- **Numeric**
  - Integer
  - float
- **Sequence Type**
  - String
  - List
  - tuple
- **Boolean**
- **Set**
- **Dictionary**
- **Binary Types**

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

Ans: we use the if statement to execute a block of code based on a specified condition.

Simple if:

The syntax of the if statement is as follows:

```
if  condition:
        if-block;
```

If the condition evaluates to True, it executes the statements in the if-block. Otherwise, it ignores the statements.

If..else

typically, we want to perform an action when a condition is True and another action when the condition is False.

To do so, we use the if...else statement.

The following shows the syntax of the if...else statement:

```
if condition:
        if-block;
else:
        else block;
```

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

Ans: The if statements are executed from the top down. As soon as one of the conditions controlling the if is true, the statement associated with that if is

executed, and the rest of the ladder is bypassed. If none of the conditions is true, then the final “else” statement will be executed.

Syntax:

```
if condition:
        if-block;
elif condition:
        elif-block:
elif condition:
        elif block:
else:
        else-block;
```

At first the first if condition is evaluated, if it is true then the block of code is executed. Otherwise, it checks elif block and then if it is true it executes the code else it checks next elif statement. This process repeats until all the elif conditions are checked if none of them becomes true then it goes to the else block.