29/4/2024
Monday

* SMART BRIDGE *

Assignment - I

D. Swapna
MCA 1Styr II-sem
Vaagdevi Engineer college, Bollikunta

1] Define Artificial Intelligence (AI) & provide example of its applications.

→ A) **Artificial Intelligence :-** Artificial Intelligence refers to the simulation of human intelligence process by machines, especially computer system. These processes includes learning, reasoning, & self-correction this is called as Artificial Intelligence (AI).

Examples of Applications :-

⇒ Autonomous vehicles
⇒ Chatbots
⇒ Self-driving vechicles.
⇒ Virtual Assistants
⇒ Financial services
⇒ language translation.
⇒ Healthcare
⇒ gaming.
⇒ Search engines (Google).

2] Differentiate between supervised & unsupervised learning techniques in ML?

→ Ans:-

Supervised learning	unsupervised learning.
* Supervised learning, the algorithm is trained on using labeled data	* unsupervised learning, the algorithm is trained by using unlabelled data.
* The goal is to learn a mapping from i/p variables to o/p variables based on the labeled example during traning.	* It goal is find hidden structure or patterns within the data.

⇒ Supervised learning model takes direct feedback to check if it is predicting correct o/p or not.	unsupervised learning model doesn't take any feedback.
eg:- Support Vector Machines (SVM), Decision Trees, Random forests, neural networks.	eg:- K-means clustering, Hierarchical clustering, principal component Analysis (PCA), t-distributed stochastic neighbor embedding (t-SNE)

3) What is python? Discuss its main features & advantages.

A) python:- python is a simple, interpreted, high-level & dynamic programming language. It is easy & simple to learn.

Interpreted language:- programming language where the source code is translated into machine code & executed line by line eg:- python, Ruby, Javascript.

Key features of python:-

* Easy to use & learn.
* python has simple syntax that is easy to learn.
* open source - python is open source language.
* large standard library
* object oriented programming language.
* portable.
* Dynamically type language.

Advantages:-

* Easy to read, learn, write

* python is a high-level programming language with a syntax that is similar to that of english
* Improved productivity
* Interpreted language.
* Free & open source.
* portability.

4] What are the advantages of using python as a programming language for AI & ML?

A) python has become one of the most popular programming language for AI & ML due to several advantages they are given below:-

1) Simplicity & Readability:- python syntax is clear, concise & making it easy to read & write. This

* Simplicity facilities faster development & easier maintainence of AI & ML projects.
* Extensive libraries:- python riches ecosystems of libraries & frameworks specifically designed for AI & ML such as Tensorflow, PyTorch, scikit-learn, keras, & NLTK.
* Community support:- It has large & active community of developers, researchers, & enthusiasts.
* Flexibility:- It has versatile language that can be used for a wide range of tasks beyound AI & ML.

Interoperability :- It can easily interface with other languages like C/C++, Java & leveraging performance code written in other language.

Scalability :- It scalability has improved with advancements libraries like Tensort Flow, PyTorch. for computing & GPU acceleration of scaling of AI & ML.

Documentation & resources, Deployment.

5] Discuss the importance of indentation in python code?

Indentation refers to the spaces or tabs at the beginning of the code. It is fundamental to the language Syntax & Structure. the given below several reasons why identation is crucial in python code.

Readability:- It makes code readability, & identation plays a significant role in making code easy to read & understad. It helps in diff block of code, such as loops, conditionals function definitions.

Enforcement of code Blocks:- the identation, programming languages use braces or keywords to denote code of block python uses identation.

Maintaining consistency:- Indentation enforces a consistent coding style across a project or among developers.

Visual Hierarchy:- Indentation visually represents the Hierarchy & nesting of code blocks.

Debugging:- Avoiding Ambiguity.

6) Define a variable in python. provide examples of valid variable names.

A) In python, a variable is a name that refers to a value stored in the computers memory. Variable as a container that holds data. To define a varible in python, we simply assign a value to name using the "=" operator. variables used to store data values we should not use keywords & special charcters

Syntax:- city_name='warangal'

python
copy code.

eg:- Variable Assigning:-

```
X=5
Y="HEY VEC"
Z= 3.14
print(Z)
print(X)
print(Y)
```

O/P:- 3.14
5
HEY VEC

eg of valid variable names in python:-

python
copy code

```
25 "John" True 85.5
```

Variable names can contain letters (both uppercase & lowercase), digits, underscore, but they must start with a letter or underscore.

Rules for varible names in python.

eg:- multi vaible assign

```
int var, float var, string var=
3, 4.5, "VEC".
print(float var)
```

O/P:- 4.5

1) Variable names can't start with a digit.
2) varible names are case-sensitive (e.g. 'Age' & 'age' are different variables)
3) variable names are can't be as python keywords (eg: 'if', 'else', 'for', 'while' etc).

7) Explain the difference btw a key word & an identifier in python.

A) keywords & identifiers are fundamental concepts, but they are different purposes:-

* Keywords:- Keywords are reserved words that have special meaning & functionalities in python.

** Keywords are the special words that have specific meaning & pupose & you can't use this keywords in variable & functions

Examples of python keywords include if, else, for, while, def, import, class, return, True, False, None.

* Identifiers :-

Identifiers are names given to entities in Python. Such as variables, functions, classes, modules etc. They are user defined names that represents memory locations where values are stored.

eg:- of identifiers :- my_variable, function_name, class_name, module_name etc.

It can consist of letters (both upper & lowercase), digits, & underscore (_) but can't start with a digit.

8) List the basic data types available in python?

A) The Basic datatypes available in python are given below:-

1) Integer (int) :- whole no's without any decimal point.
eg:- 5, -3, 1000.

2) floating point no's (float) :- no's with a decimal point or numbers written in exponential notation.
eg:- 3.14, -0.001, 2.5e2

3) Strings (str) :- Ordered sequence of characters enclosed within single, double, triple quotes
eg:- 'hello', "world", """python""".

4) Boolean (bool) :- Represents truth values. either True or False.

5) Lists :- ordered mutable (changeable) collection of items enclosed within square brackets eg:- [1,2,3], ['apple', banana, 'grapes']

6) Tuples :- ordered immutable (unchangable) collection of items enclosed within ~~curly braces eg:- ('name': 'John', 'age': 30,~~ parentheses. eg- (1,2,3) ('a','b','c')

7) Dictionaries (dict): unodered collection of key-value pairs enclosed within curly braces eg:- {'name': 'John', 'age': '30', 'city': 'new york'}.

8) Sets :- unoodered collection of unique items enclosed within curly braces or created using the set() function eg:- {1,2,3}, set ([4,5,6])

9] Describe the syntax for an if statement in python?

A) In python, the syntax for an if stmnt is quite straight forward. It follows this general structure:-

If statement :- Executes a block of code only if a specified condition is true.

```
x = 10
if x > 5:
    print ("x is greater than 5")
x is greater than 5.
```

If condition :- If the condition evaluates to True, the code block immediately following this line executed. the condition is false, the interpreter moves to the next elif or else block.

* elif another_condition:- This stands for "else if". It allows you to check additional condition if the preceding if condition is false. If another_condition evaluates True, the code block immediately following this line is executed,

* else(optional):- This is default case. If none of the preceding conditions are True, the code block following else statement is executed. eg:- (or) Executes 1 block code if a condition is T & another block if it's false.

eg:-

```
python
copy code
10 if
10.
print ("a is greater than 10")
elif
10:
print ("a is equal to 10")
else
print ("a is less than 10")
```

In this eg:- since a is equal to 10, the o/p would be: "x is equal to 10".

eg:-
```
x=22
if x>50:
    print ("x is greater than 50")
else:
    print ("x is not greater than 50")
```

o/p:-
x. is not greater than 50.

10] Explain the purpose of elif statement in python?

A) Elif statement for "else if". It's used in conjunction with the if statement to to test multiple conditions sequentially.

* Allows you to check multiple conditions & executes different blocks of code accordingly.

eg:-
```
x=22
if x>50;
    print ("x is greater than 50")
elif x==22;
    print ("x is equal to 22")
else:
    print ("x is not greater than 50")
```

x is equal to 22.