Machine learning

Deep learning

Data analysis

Data science

Artificial intelligence

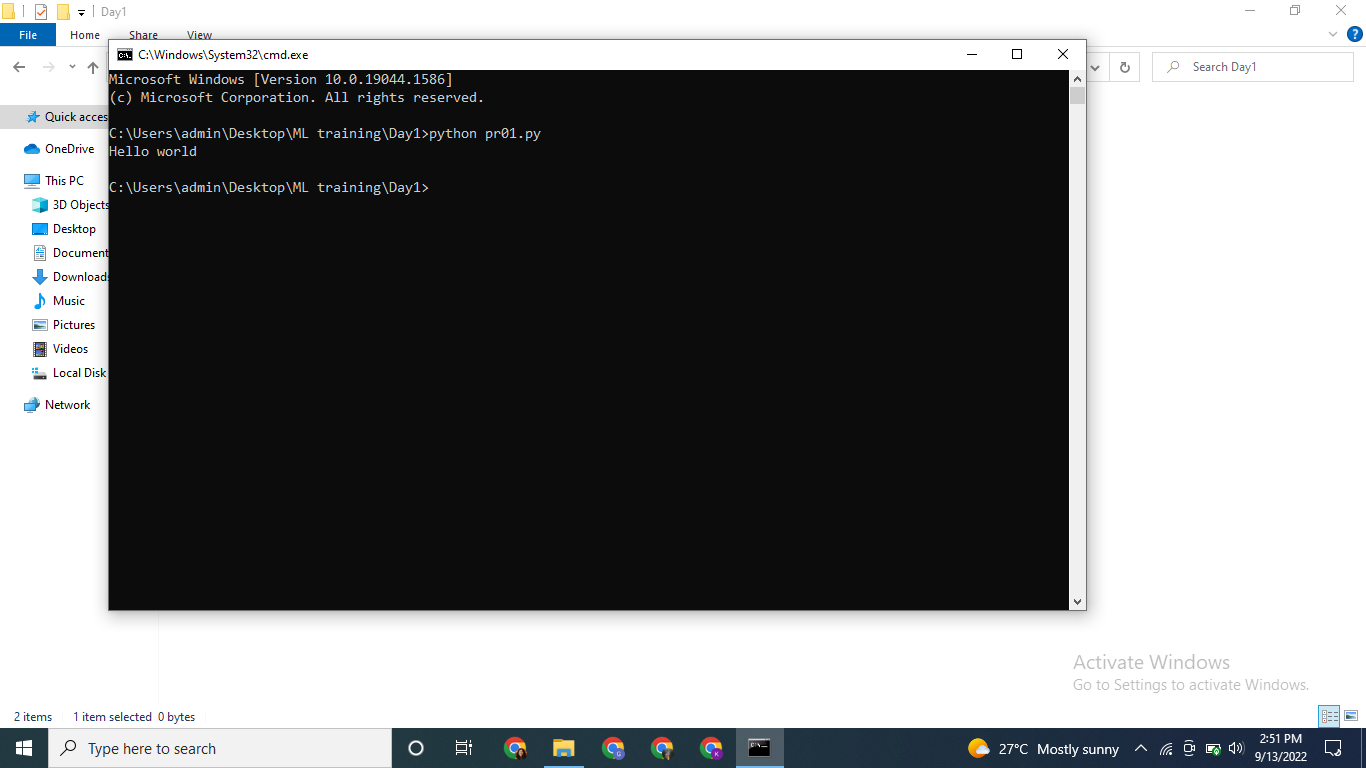
Types of data

1. Unstructured data
2. Structured data

Python libraries

1. Data manipulation – numpy, Pandas
2. Data visualization- Matpotlib, seaborn, Plotly
3. Machine Learning – scikit-leran, sk-learn
4. Deep learning- Tensorflow, keran, Pytorch

Method to access notepad converted python file into the command prompt.



# is used to comment in python.

‘import keyword’ is used to access the library in python

keyword.kwilist is used to access the list items in this library

RULES OF VARIABLES:

1. Can’t start with Number

Ex: 1=’python’ , a1=”python”

1. No Space

Ex: a 1=”python”

1. Can’t use SPECIAL CHARACTER

Ex: a$=”hello” ; a\_1= ‘hi’ ; \_=200

1. Can’t use KEYWORD

Python operator

1. Arithmetic +,-,\*,/,%,//,\*\*
2. Comparison/relational >,<,>=,<=,!=,==
3. Assignment +,+=,-=,/=,//=

Eg. a=a+23 is same as a+=23

1. Bitwise ^, &, |,
2. Logical and,or,not
3. Identity is, is not
4. Membership in, not in

( ) ----- function , expression , tuple

[ ] ----- list, indexing, slicing

{ } ----- set, Dictionary

< > ---- NO USE

Data types in python

1. Numeric === int

Float

Complex

1. String- str a=’hello’ / a=”hello” /a= ‘’’hi’’’
2. List my\_list = [‘abc’,3,4,5.5,a+ib]
3. Tuple - my\_tuple= (45,4.5, ‘hi’,’bye’)
4. Set – {1,23,56,50}
5. Dictionary – dict { key : value }
6. Boolean ------ True , False

Type () is used to tell the data types of variables

15/09/2022

1. String
2. Indexing ------------- positive and negative

A=’hello’

A [starting point(by default 0) (including): stop(excluding +1): step( by default 1)]

1. list

1. []

2. Homo/hetro

3. Mutable

4. Indexing/slicing,loop

1. Tuple
2. ()
3. Comma sep variable
4. Immutable
5. Homo/hetro
6. Indexing / slicing
7. set
8. unordered
9. cannot use indexing/slicing
10. unique collection

a.capitalize() is used to capitalize the first letters of string ‘temporarily’

a=a.capitalize() will make it permanent.

Some important functions:

A=’hello world’

1. a.capitalize() : 'Hello world'
2. 2. a.center(50): ' Hello world '
3. a.center(50,'#'): '###################Hello world####################'
4. a.count('l'): 3
5. Endswith(‘i’)
6. find(‘j’) ------- to find place of letter J in string
7. get(INDEX) ------ gives that letter at the given index
8. get\_dummies()--------
9. a.lstrip(): 'Hello world' ---------------- to remove left spaces
10. a.rstrip(): 'Hello world'---------------- to remove right spaces
11. len(a): 11
12. a.isupper(): False
13. a.islower(): True
14. a.upper(): 'HELLO WORLD'
15. a=a.upper()
16. a.startswith('HE'): True
17. a.endswith(‘D’) : True
18. b='kj123@gmail.com'
19. b.split('@'): ['kj123', 'gmail.com']
20. b: 'kj123@gmail.com'
21. b=b.split('@')
22. b: ['kj123', 'gmail.com']
23. '@'.join(b): 'kj123@gmail.com'
24. 'hi' in m: True
25. 'hello' in m: False
26. m += ['new value'] or m.append(‘new value’)----------- to add a new value in list at the end of list
27. m: [12, 'hi', 2.3, 300, 'new value']
28. m.extend([‘hi’, ‘hello’, 3]) -------------- to add multiple values in list
29. m.insert(2,'hello')----------------- to add value at certain position in list
30. m.pop(position/nothing): 3 ------------- removes specific posn/ last value and returns it
31. n.sort()---------------- ascending order
32. m.index('hi')-------- to know the posn of value in list
33. d1['name'].insert(1,'bye')-------------- to add value at a position in a dictionary
34. 'hi {} welcome to {}'.format('kinjal','upflairs'): 'hi kinjal welcome to upflairs' (format string)

16/09/2022

Data type dictionary

1. Unordered
2. Key value pair
3. indexing/slicing not possible

Statements

1. condition/ decision making – if-elif-else(suites/indentation)
2. looping – for and while
3. Exception handling --- try, Except

Simple if

If age<18:

Print (‘you win’)

If age>18 and age <20:

Print (‘you win’)

Print(‘a task’)

If age>20:

Print(‘ no gift no task’)

Ladder

If age<18:

Print(‘a gift’)

Elif age>18 and age <20:

Print(‘you win’)

Print(‘a task’)

Else:

Print(‘koi gift nhi h’)

Nested

If today ==’Saturday’

Print(‘half day work’)

Elif today==’Sunday’:

If condition == ‘sick’:

Print(‘rest’)

Else

Print(‘party’)

Range(start (by default 0), stop point (excluding),step (by default 1))

Eg. list(range(2,101,2))

[2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100]

Loop

1. For loop
2. While loop

For I in [1,2,3,4,5]:

Print (i)

Function

1. No i/p no o/p
2. No i/p but o/p
3. I/p but no o/p
4. o/p and i/p
5. taking multiple i/p
6. returning multiple o/p
7. keyword argument
8. default values
9. \*arg
10. \*kwargs

Module---- it’s a .py file; eg. OS module, Math module

Packages – it is a collection of modules ,\_init\_.py

Libraries – it is a collection of packages

Capitalize------ 1st letter capital

Casefold = lower------- lower case conversion

DAY 13

A=[2,3,4,5,6]

B=[4,6,8,10,12]

B=[]