CJC

Complete Java Classes

by Kunal Sir

Some Other Functions in NumPy

1. ARANAGE :-

NUMPY.ARANGE() FUNCTION CREATES AN ARRAY OF EVENLY SPACED VALUES WITHIN A GIVEN INTERVAL. IT IS SIMILAR TO PYTHON'S BUILT-IN RANGE() FUNCTION BUT RETURNS A NUMPY ARRAY INSTEAD OF A LIST.

```
ARR= NP.ARANGE(5 , 10)
PRINT(ARR)
```

EX.

```
NP.ARANGE(0,20,2)
```

OP = ARRAY([0, 2, 4, 6, 8, 10, 12, 14, 16, 18])

2. LINSPACE:-

INSPACE() FUNCTION IN NUMPY RETURNS AN ARRAY OF EVENLY SPACED NUMBERS OVER A SPECIFIED RANGE. UNLIKE THE RANGE() FUNCTION IN PYTHON THAT GENERATES NUMBERS WITH A SPECIFIC STEP SIZE. LINSPACE() ALLOWS YOU TO SPECIFY THE TOTAL NUMBER OF POINTS YOU WANT IN THE ARRAY, AND NUMPY WILL CALCULATE THE SPACING BETWEEN THE NUMBERS AUTOMATICALLY.

```
ARRAY = NP.LINSPACE(0, 1, NUM=10)
PRINT(ARRAY)
```

CJC

Complete Java Classes

by Kunal Sir

Syntax: np.linspace(initil, termination, num=no_of_value)

SYNATX: NUMPY.LINSPACE(START, STOP, NUM=50, ENDPOINT=TRUE , RETSTEP=FALSE, DTYPE=NONE, AXIS=0)

PARAMETERS:

START: [OPTIONAL] START OF INTERVAL RANGE. BY DEFAULT START = 0

STOP: END OF INTERVAL RANGE

NUM: [INT, OPTIONAL] NO. OF SAMPLES TO GENERATE

RETSTEP: IF TRUE, STOP IS THE LAST SAMPLE BY DEFAULT RESTEP = FALSE

ENDPOINT: IF TRUE, STOP IS INCLUDED AS THE LAST VALUE. IF FALSE, STOP IS EXCLUDED. BY DEFAULT ENDPOINT=TRUE.

DTYPE: TYPE OF OUTPUT ARRAY

AXIS: IF START AND STOP ARE ARRAYS, AXIS SPECIFIES ON WHAT AXIS WILL THE VALUES BE ADDED. IF AXIS = 0, VALUE IS ADDED TO FRONT, IF AXIS = -1 VALUE IS ADDED AT THE END.

EX.

NP.LINSPACE(0,10,NUM=2)

OP= ARRAY([0., 10.])

TO 2 EQUALLY SPACED VALUE ARE THERE.

NP.LINSPACE(0,10,NUM=3)

OP= ARRAY([0., 5. ,10.])

3. RANDOM.RANDOM & RANDOM.RANDINT

IN NUMPY THERE ARE SUBMODULE CALLED AS RANDOM.

Syntax: np.random.random(number_of_values)

IF WE USE RANDOM.RANDOM IT GENERATES THE RANDOM VALUES IN BETWEEN 0.0 TO 1.0

Syntax: np.random.randint(init, termination, no_of_values)

IF WE USE RANDOM.RANDINT IT GERERATES THE RANDOM VALUES IN BETWEEN THE RANGE WE PROVIDED.

FOR CREATING THE N-DIMENSIONS ARRAY WE PASS ROW AND COLUMN.

Syntax: np.random.randint(init, termi , (no_rows, no_col))

4. SAVE AND LOAD THE RANDOM ARRAY:-

np.save('filename', np_array_to_save)

np.save('CJC_random_Array', np5)

%pwd - Present Working Directory – To check the location

%cd – To change Directory

so you can change directory by using.... %cd foldername

TO LOAD THE ARRAY :-

```
AP_ARRAY = NP.LOAD('CJC_RANDOM_ARRAY')
```

5. DATE TIME:--> IN NumPy:

SYNTAX:->NP.DATETIME64('YYYY-MM-DD')

```
DATE_ARRAY1 = NP.DATETIME64('2024-08-25')
```

IF YOU WANT TO RETRIVE ONLY YEAR IN DATE SO YOU CAN WRITE

```
NP.DATETIME64('2024-08-25', 'Y'          #OP = NUMPY.DATETIME64('2024')
NP.DATETIME64('2024-08-25', 'M')         #OP = NUMPY.DATETIME64('2024-08')
NP.DATETIME64('2024-08-25', 'D')         #OP = NUMPY.DATETIME64('2024-08-25')
```

Complete Java Classes

by Kunal Sir

SO NOW WE SEE HOW TO USE TIME IN THIS THAT COULD BE USED IN PROGRAM

LETS CREATE

SYNTAX : NP.DATETIME64('YYYY-MM-DD HH:MM:SS')

```
DATE_ARRAY2 = NP.DATETIME64('2024-08-21 16:11:33')

HOUR = NP.DATETIME64('2024-08-21 16:11:33', 'H')
#OP = NUMPY.DATETIME64('2024-08-21T16','H')

MIN = NP.DATETIME64('2024-08-21 16:11:33', 'M')
#OP = NUMPY.DATETIME64('2024-08-21T16:11')

SEC = NP.DATETIME64('2024-08-21 16:11:33', 'S')
#OP = NUMPY.DATETIME64('2024-08-21T16:11:33')
```