20BDS0211:Assignment-1 (Externship:week 1)

Q) Assign your Name to the variable name and Age to the variable age. Make a Python program that prints your name and age.

```
name = "Your Name"
age = 25
print("Name:", name)
print("Age:", age)
```

```
Name: Your Name
Age: 25
```

Q2) X = "Datascience is used to extract meaningful insights." Split the string

```
X = "Datascience is used to extract meaningful insights."
split_string = X.split()
print(split_string)
```

```
['Datascience', 'is', 'used', 'to', 'extract', 'meaningful', 'insights.']
```

Q3) Make a function that gives the multiplication of two numbers:

```
def mul_num(a, b):
    return a * b

result = mul_num(5, 10)
print("Result:", result)
```

```
Result: 50
```

Q4) Create a Dictionary of 5 States with their capitals. Also, print the keys and values

```
states = {
    "Telengana": "Hyderabad",
    "Tamil Nadu": "Chennai",
    "Kerla": "Cochin",
    "Gujrat": "Surat",
    "Karnataka": "Bengaluru"
}

print("Keys:", list(states.keys()))
print("Values:", list(states.values()))
```

```
Keys: ['Telengana', 'Tamil Nadu', 'Kerla', 'Gujrat', 'Karnataka']
Values: ['Hyderabad', 'Chennai', 'Cochin', 'Surat', 'Bengaluru']
```

Q5) Create a list of 1000 numbers using the range function:

```
numbers = list(range(1, 1001))
print(numbers)
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35,
```

Q6) Create an identity matrix of dimension 4 by 4:

```
import numpy as np

identity_matrix = np.identity(4)
print(identity_matrix)
```

```
[[1. 0. 0. 0.]
 [0. 1. 0. 0.]
```

```
[0. 0. 1. 0.]
[0. 0. 0. 1.]]
```

Q7) Create a 3x3 matrix with values ranging from 1 to 9:

```
import numpy as np

matrix = np.arange(1, 10).reshape(3, 3)
print(matrix)
```

```
[[1 2 3]
 [4 5 6]
 [7 8 9]]
```

Q8) Create two similar dimensional arrays and perform the sum on them:

```
array1 = np.array([[1, 2], [3, 4]])
array2 = np.array([[5, 6], [7, 8]])

sum_array = array1 + array2
print(sum_array)
```

```
[[ 6  8]
 [10 12]]
```

Q9) Generate the series of dates from 1st Feb 2023 to 1st March 2023 (both inclusive):

```
import pandas as pd

date_range = pd.date_range(start='2023-02-01', end='2023-03-01')
print(date_range)
```

```
DatetimeIndex(['2023-02-01', '2023-02-02', '2023-02-03', '2023-02-04',
               '2023-02-05', '2023-02-06', '2023-02-07', '2023-02-08',
               '2023-02-09', '2023-02-10', '2023-02-11', '2023-02-12',
               '2023-02-13', '2023-02-14', '2023-02-15', '2023-02-16',
               '2023-02-17', '2023-02-18', '2023-02-19', '2023-02-20',
               '2023-02-21', '2023-02-22', '2023-02-23', '2023-02-24',
               '2023-02-25', '2023-02-26', '2023-02-27', '2023-02-28',
               '2023-03-01'],
              dtype='datetime64[ns]', freq='D')
```

Q10) Given a dictionary, convert it into a corresponding dataframe and display it:

```
dictionary = {'Brand': ['Maruti', 'Renault', 'Hyundai'], 'Sales': [250, 200, 240]}
df = pd.DataFrame(dictionary)
print(df)
```

```
      Brand  Sales
0   Maruti    250
1  Renault    200
2  Hyundai    240
```

✓ 0s completed at 3:47 PM