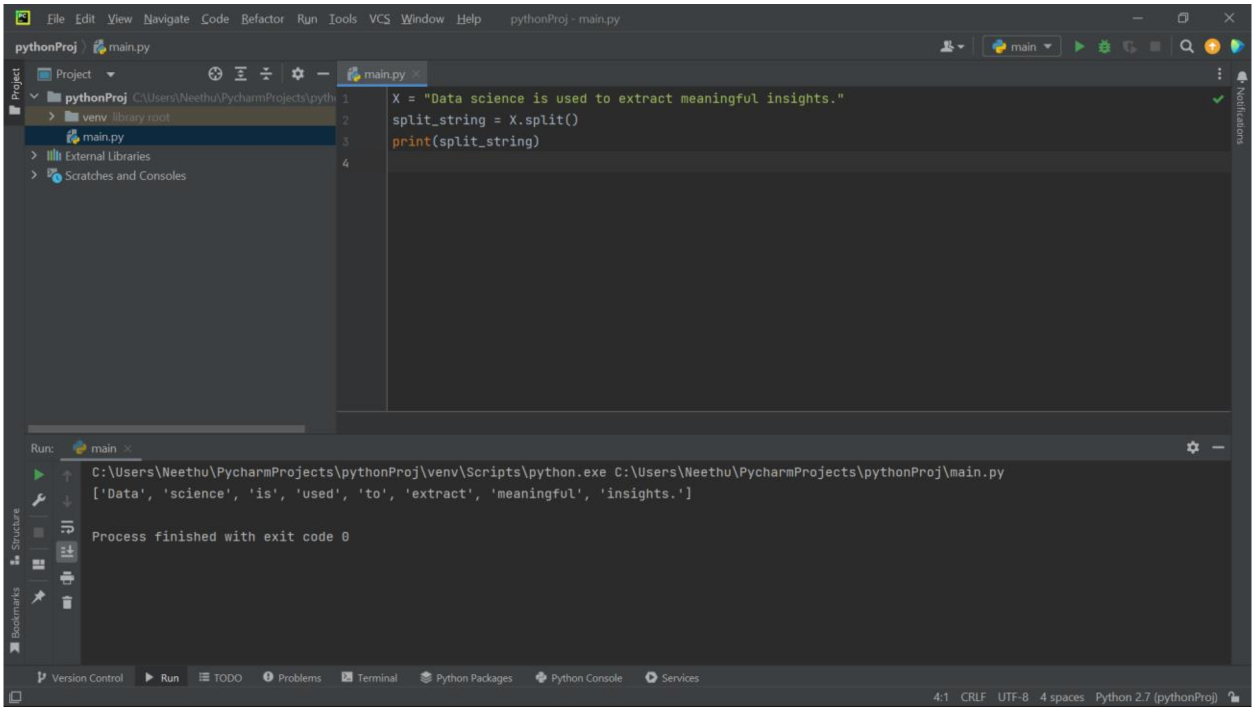
DIGITAL ASSIGNMENT – 1

NAME: GIRISH KUMAR A

REGNO: 20MID0170

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

Code and Output:



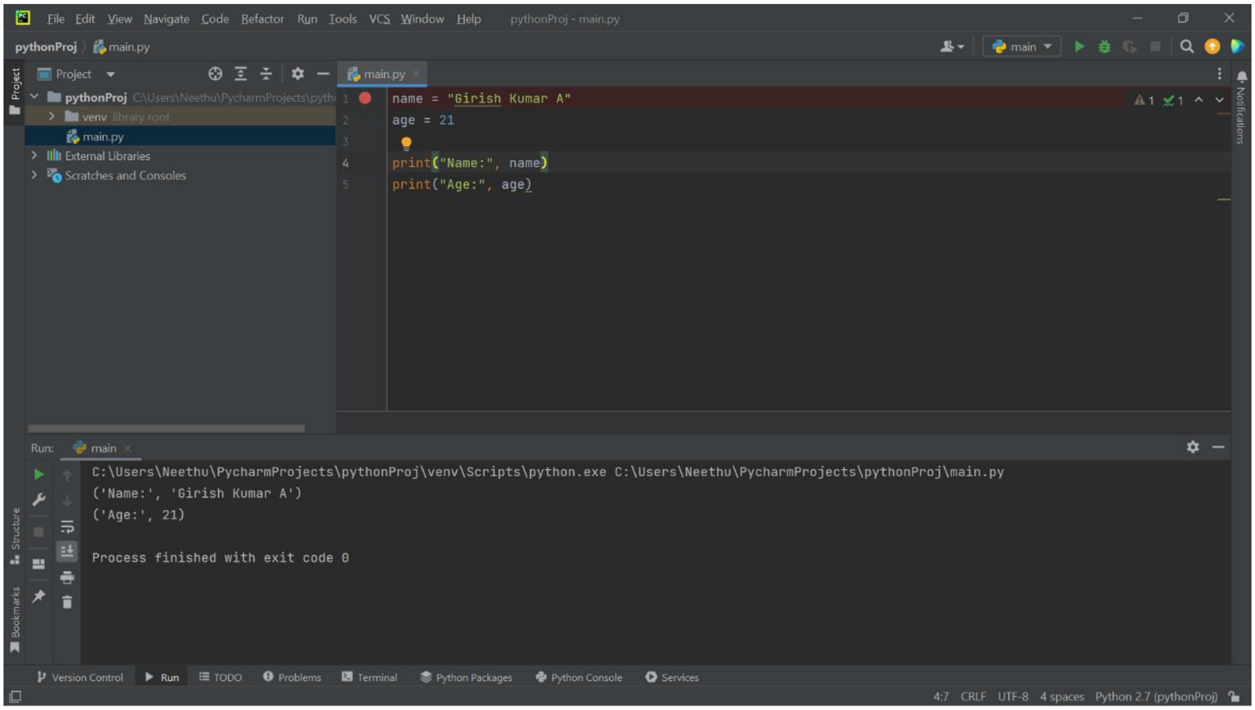
```
X = "Data science is used to extract meaningful insights."
split_string = X.split()
print(split_string)
```

```
C:\Users\Neethu\PycharmProjects\pythonProj\venv\Scripts\python.exe C:\Users\Neethu\PycharmProjects\pythonProj\main.py
['Data', 'science', 'is', 'used', 'to', 'extract', 'meaningful', 'insights.']

Process finished with exit code 0
```

2. X="Datascience is used to extract meaningful insights."

Split the string

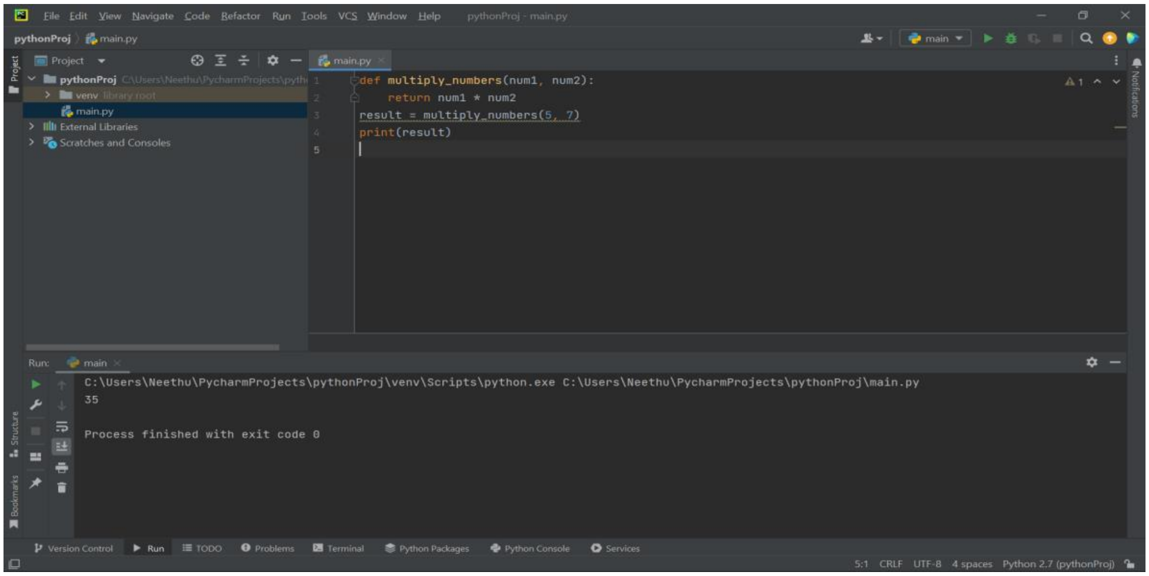
Code and Output:


File Edit View Navigate Code Refactor Run Tools VCS Window Help pythonProj - main.py
pythonProj main.py
Project
pythonProj C:\Users\Neethu\PycharmProjects\pyth
venv library root
main.py
External Libraries
Scratches and Consoles
Run: main
Version Control Run TODO Problems Terminal Python Packages Python Console Services
4:7 CRLF UTF-8 4 spaces Python 2.7 (pythonProj)

```python
name = "Girish Kumar A"
age = 21

print("Name:", name)
print("Age:", age)
```

```
C:\Users\Neethu\PycharmProjects\pythonProj\venv\Scripts\python.exe C:\Users\Neethu\PycharmProjects\pythonProj\main.py
('Name:', 'Girish Kumar A')
('Age:', 21)

Process finished with exit code 0
```

3.Make a function that gives multiplication of two numbers

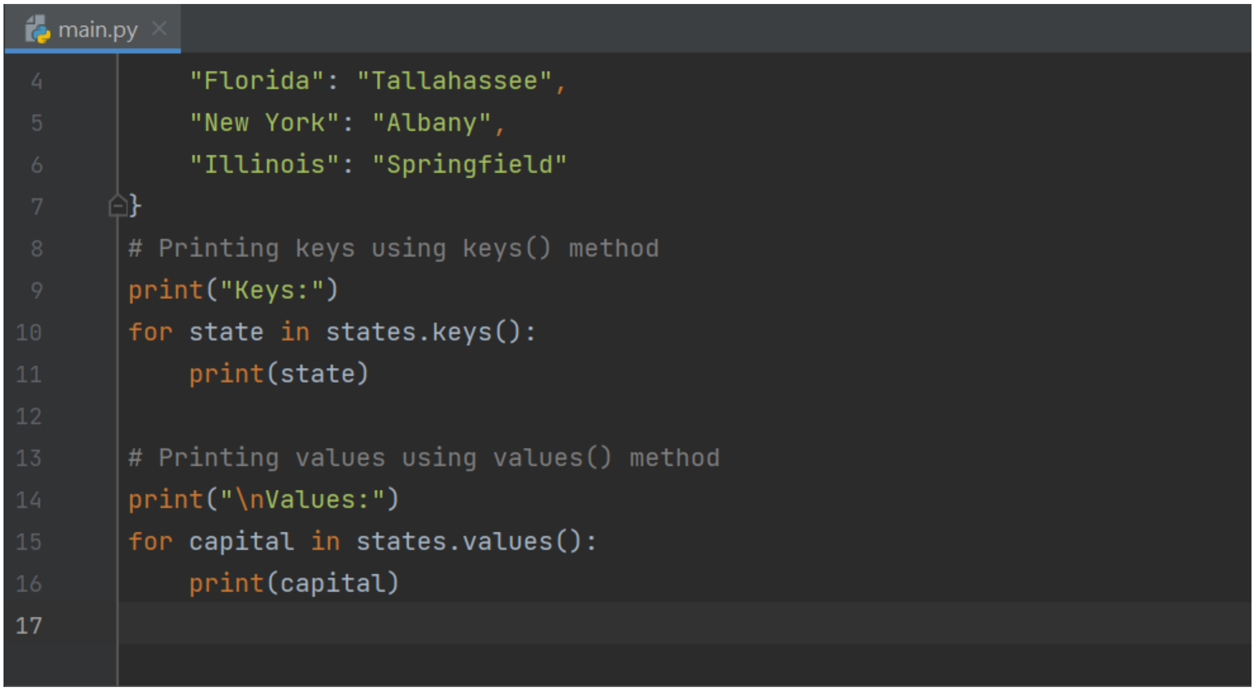

File Edit View Navigate Code Refactor Run Tools VCS Window Help pythonProj - main.py
pythonProj main.py
Project
pythonProj C:\Users\Neethu\PycharmProjects\pyth
venv library root
main.py
External Libraries
Scratches and Consoles
Run: main
Version Control Run TODO Problems Terminal Python Packages Python Console Services
5:1 CRLF UTF-8 4 spaces Python 2.7 (pythonProj)

```python
def multiply_numbers(num1, num2):
    return num1 * num2
result = multiply_numbers(5, 7)
print(result)
```

```
C:\Users\Neethu\PycharmProjects\pythonProj\venv\Scripts\python.exe C:\Users\Neethu\PycharmProjects\pythonProj\main.py
35

Process finished with exit code 0
```

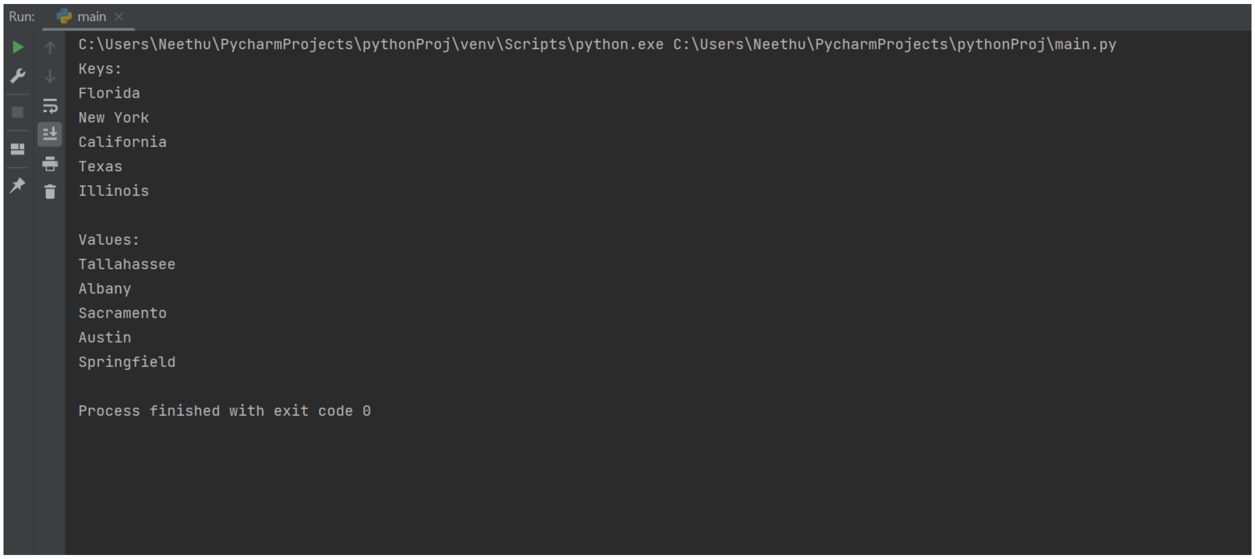
4.Create a Dictionary of 5 States with their capitals. also print the keys and values.

Code:



```python
    "Florida": "Tallahassee",
    "New York": "Albany",
    "Illinois": "Springfield"
}
# Printing keys using keys() method
print("Keys:")
for state in states.keys():
    print(state)

# Printing values using values() method
print("\nValues:")
for capital in states.values():
    print(capital)

```

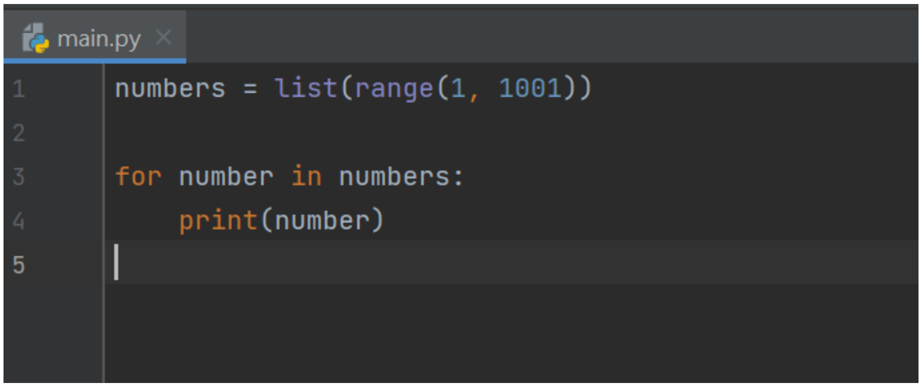
Output:



```
C:\Users\Neethu\PycharmProjects\pythonProj\venv\Scripts\python.exe C:\Users\Neethu\PycharmProjects\pythonProj\main.py
Keys:
Florida
New York
California
Texas
Illinois

Values:
Tallahassee
Albany
Sacramento
Austin
Springfield

Process finished with exit code 0
```

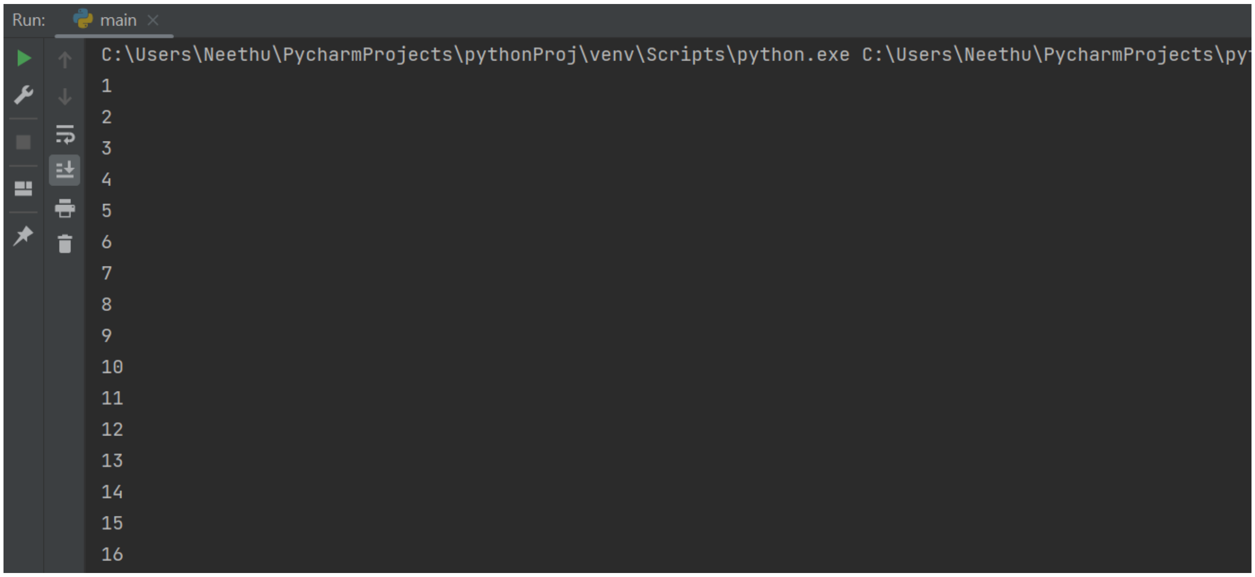
5. Create a list of 1000 numbers using range function.

Code:



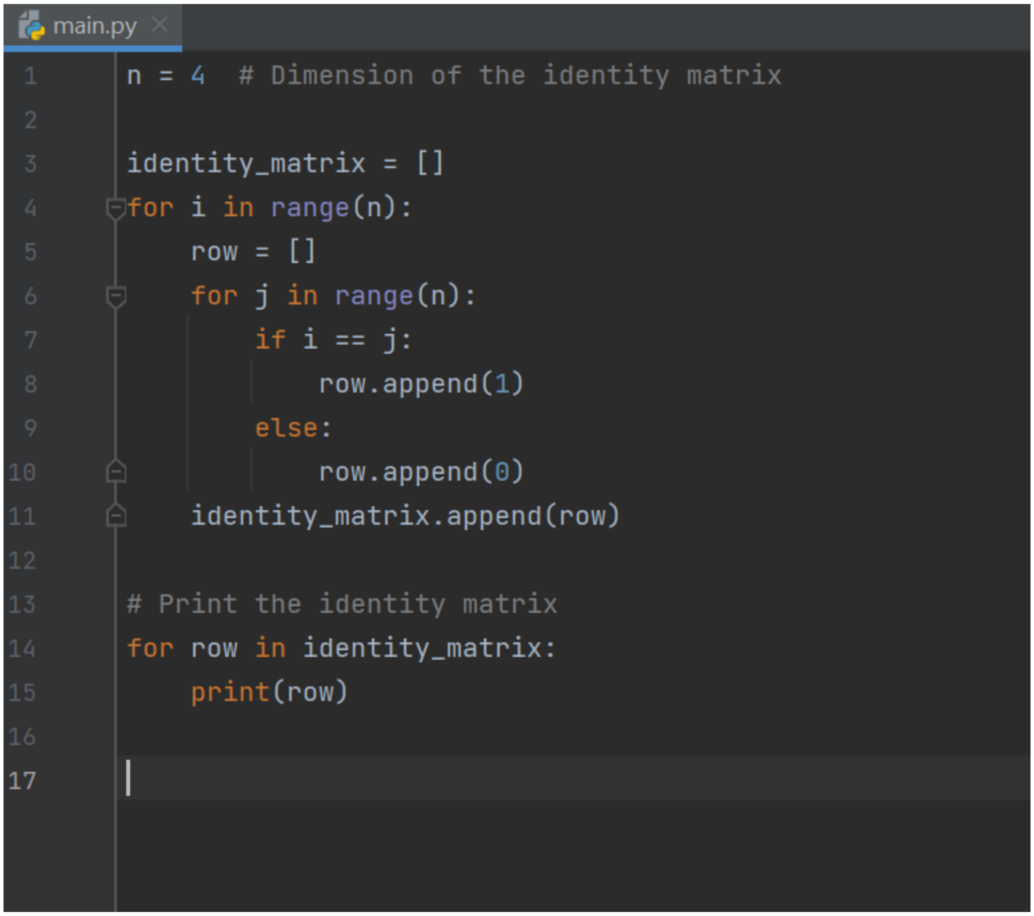
```python
numbers = list(range(1, 1001))

for number in numbers:
    print(number)
```

Output:



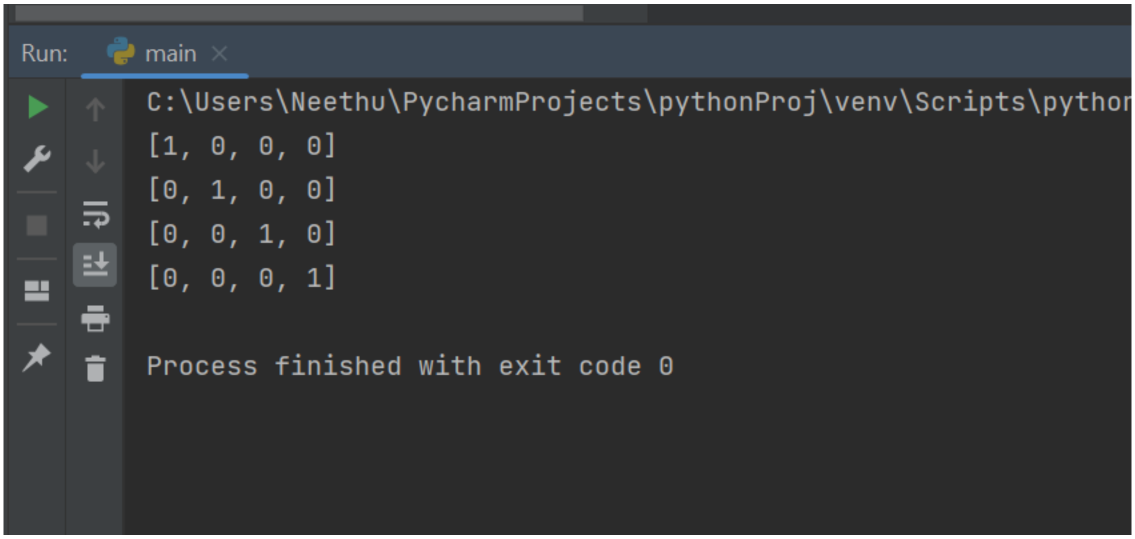
```
C:\Users\Neethu\PycharmProjects\pythonProj\venv\Scripts\python.exe C:\Users\Neethu\PycharmProjects\py
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
```

6.Create an identity matrix of dimension 4 by 4



```python
n = 4  # Dimension of the identity matrix

identity_matrix = []
for i in range(n):
    row = []
    for j in range(n):
        if i == j:
            row.append(1)
        else:
            row.append(0)
    identity_matrix.append(row)

# Print the identity matrix
for row in identity_matrix:
    print(row)

```

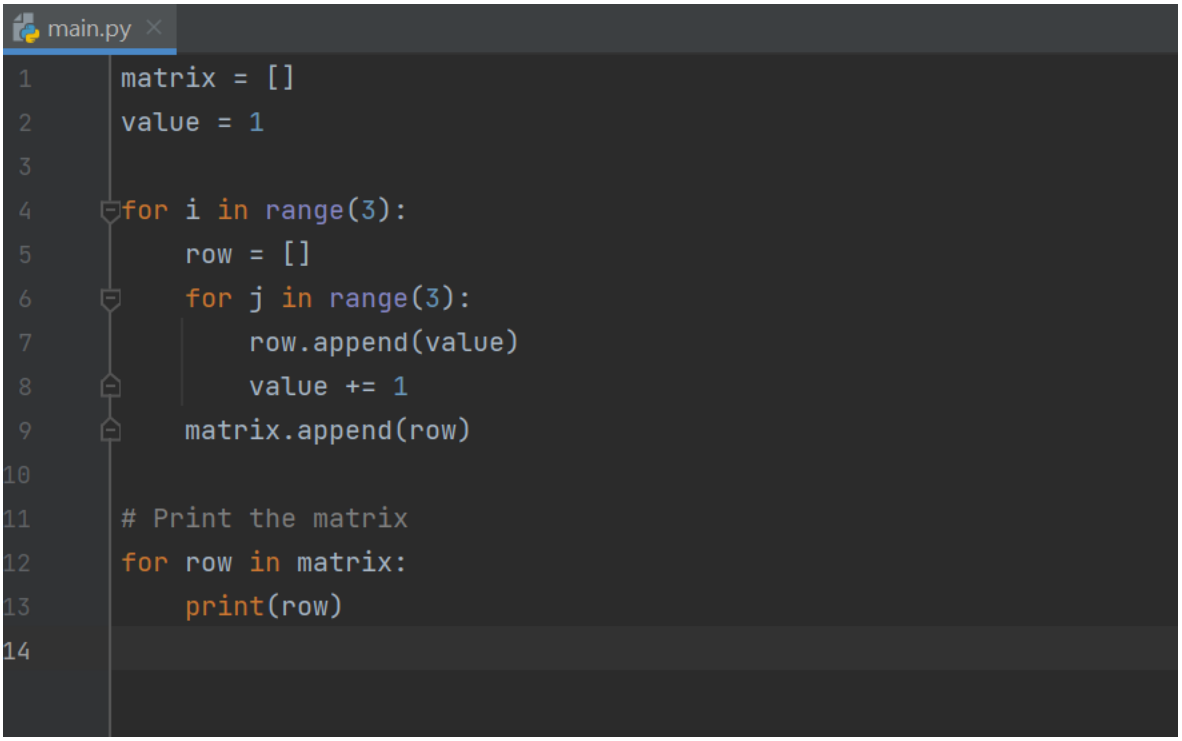
Output:



```
C:\Users\Neethu\PycharmProjects\pythonProj\venv\Scripts\python
[1, 0, 0, 0]
[0, 1, 0, 0]
[0, 0, 1, 0]
[0, 0, 0, 1]

Process finished with exit code 0
```

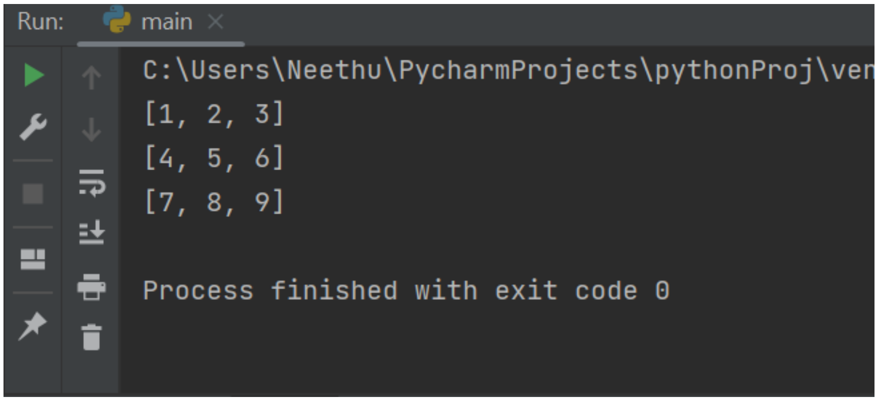
7.Create a 3x3 matrix with values ranging from 1 to 9

Code:



```python
matrix = []
value = 1

for i in range(3):
    row = []
    for j in range(3):
        row.append(value)
        value += 1
    matrix.append(row)

# Print the matrix
for row in matrix:
    print(row)
```

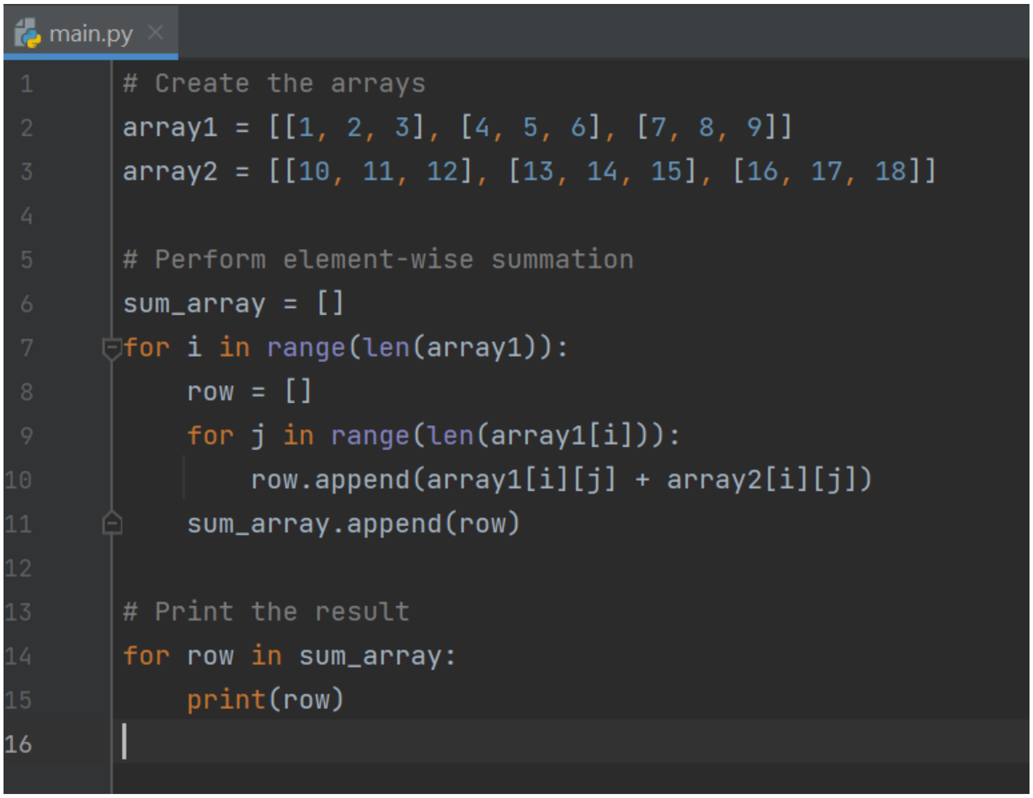
Output:



```
C:\Users\Neethu\PycharmProjects\pythonProj\ven
[1, 2, 3]
[4, 5, 6]
[7, 8, 9]

Process finished with exit code 0
```

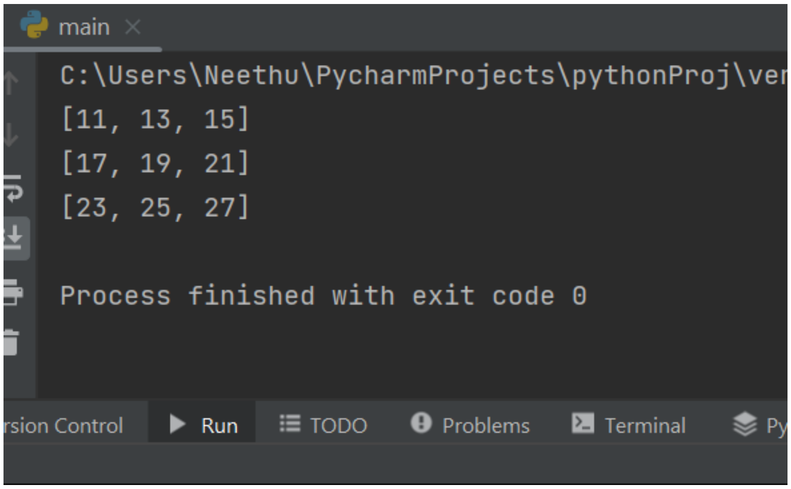
8. Create 2 similar dimensional array and perform sum on them.

Code:



```
# Create the arrays
array1 = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
array2 = [[10, 11, 12], [13, 14, 15], [16, 17, 18]]

# Perform element-wise summation
sum_array = []
for i in range(len(array1)):
    row = []
    for j in range(len(array1[i])):
        row.append(array1[i][j] + array2[i][j])
    sum_array.append(row)

# Print the result
for row in sum_array:
    print(row)
```

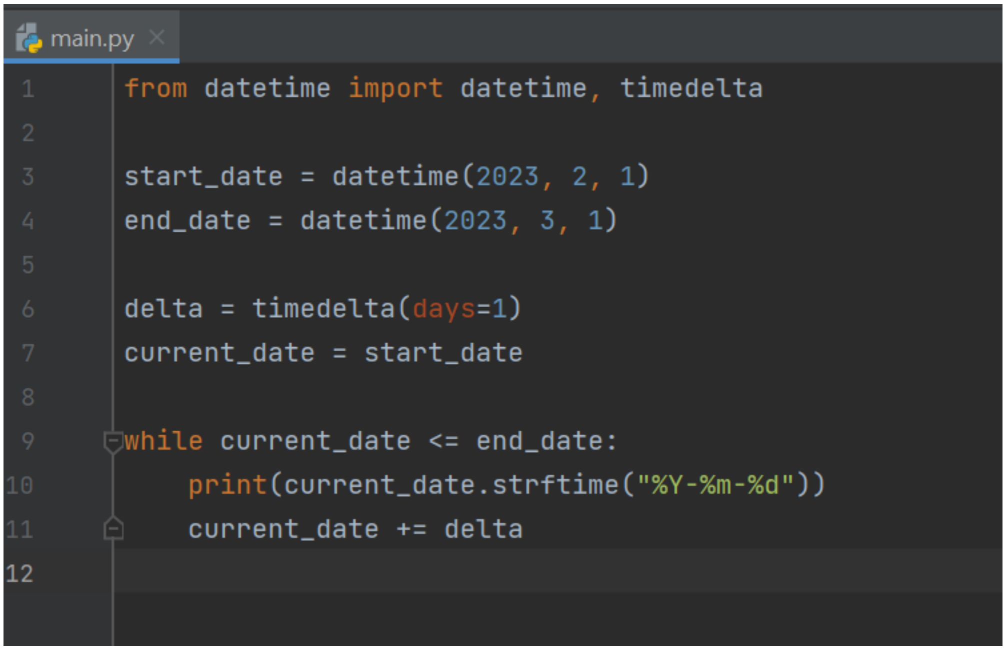
Output:



```
C:\Users\Neethu\PycharmProjects\pythonProj\ver
[11, 13, 15]
[17, 19, 21]
[23, 25, 27]

Process finished with exit code 0
```

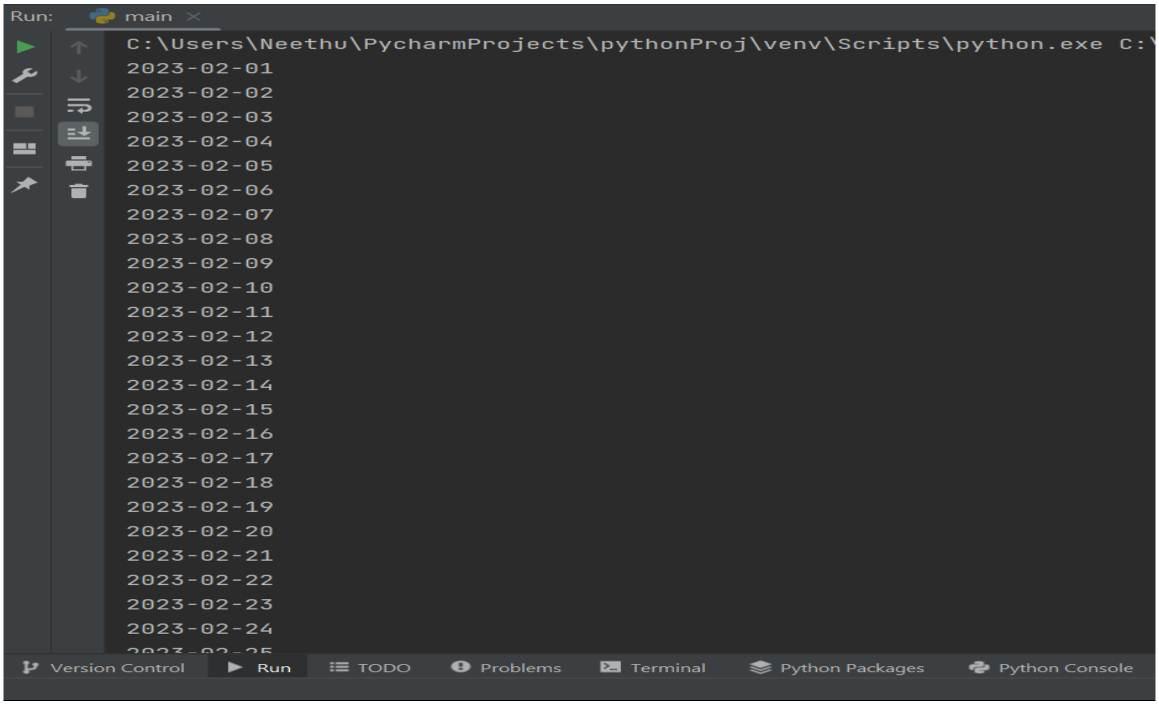
9. Generate the series of dates fror 1st Feb, 2023 to 1st March, 2023 (both inclusive)

Code:

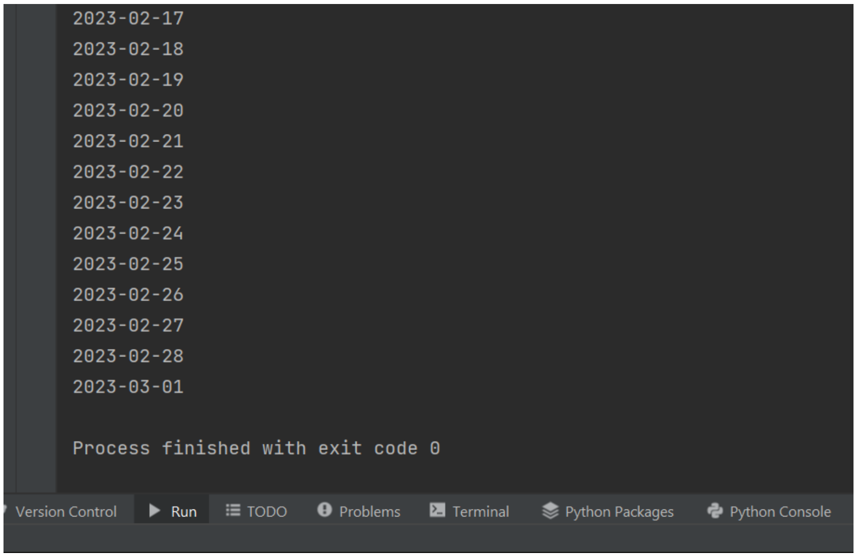


```python
from datetime import datetime, timedelta

start_date = datetime(2023, 2, 1)
end_date = datetime(2023, 3, 1)

delta = timedelta(days=1)
current_date = start_date

while current_date <= end_date:
    print(current_date.strftime("%Y-%m-%d"))
    current_date += delta

```

Output:



```
C:\Users\Neethu\PycharmProjects\pythonProj\venv\Scripts\python.exe C:\
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
```

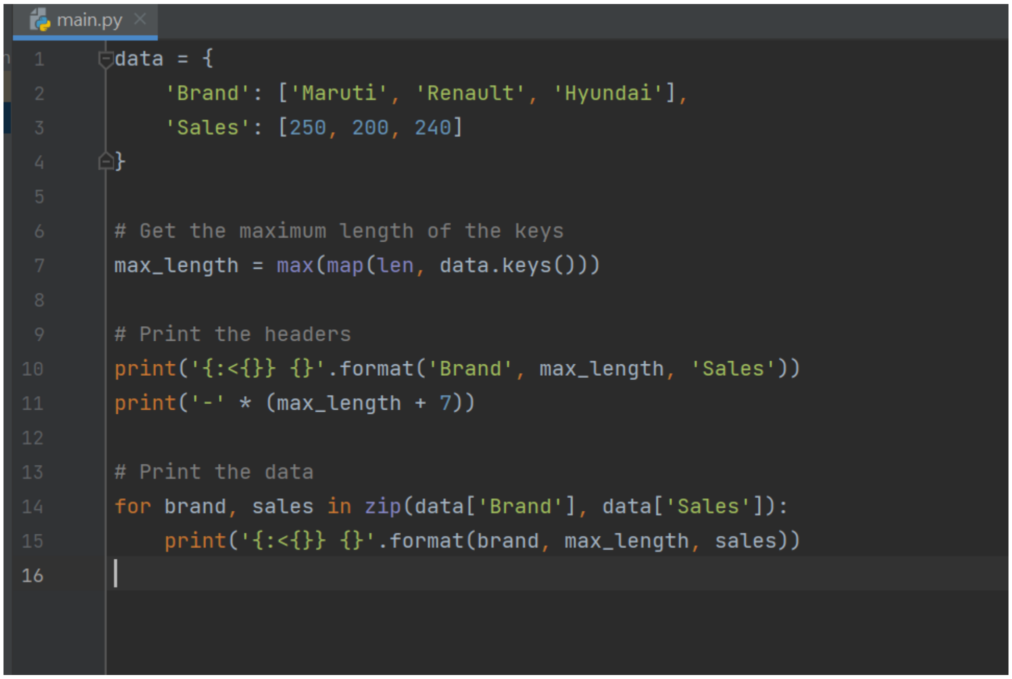


```
2023-02-17
2023-02-18
2023-02-19
2023-02-20
2023-02-21
2023-02-22
2023-02-23
2023-02-24
2023-02-25
2023-02-26
2023-02-27
2023-02-28
2023-03-01

Process finished with exit code 0
```

10. Given a dictionary, convert it into corresponding dataframe and display it

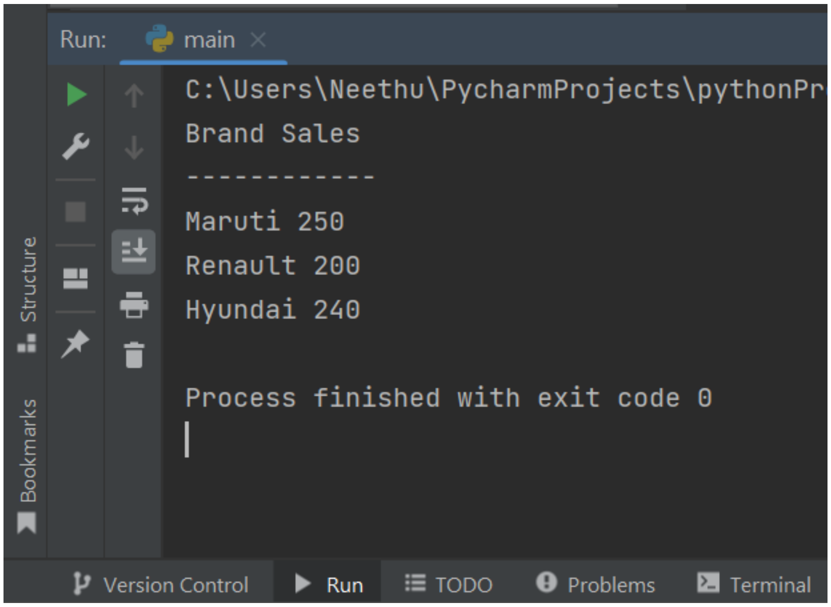
dictionary="Brand': ['Maruti', 'Renault', "Hyndai'], 'Sales':[250, 200,240])

Code:



```python
data = {
    'Brand': ['Maruti', 'Renault', 'Hyundai'],
    'Sales': [250, 200, 240]
}

# Get the maximum length of the keys
max_length = max(map(len, data.keys()))

# Print the headers
print('{:<{}} {}'.format('Brand', max_length, 'Sales'))
print('-' * (max_length + 7))

# Print the data
for brand, sales in zip(data['Brand'], data['Sales']):
    print('{:<{}} {}'.format(brand, max_length, sales))
```

Output:



```
C:\Users\Neethu\PycharmProjects\pythonPr
Brand Sales
------------
Maruti 250
Renault 200
Hyundai 240

Process finished with exit code 0
```