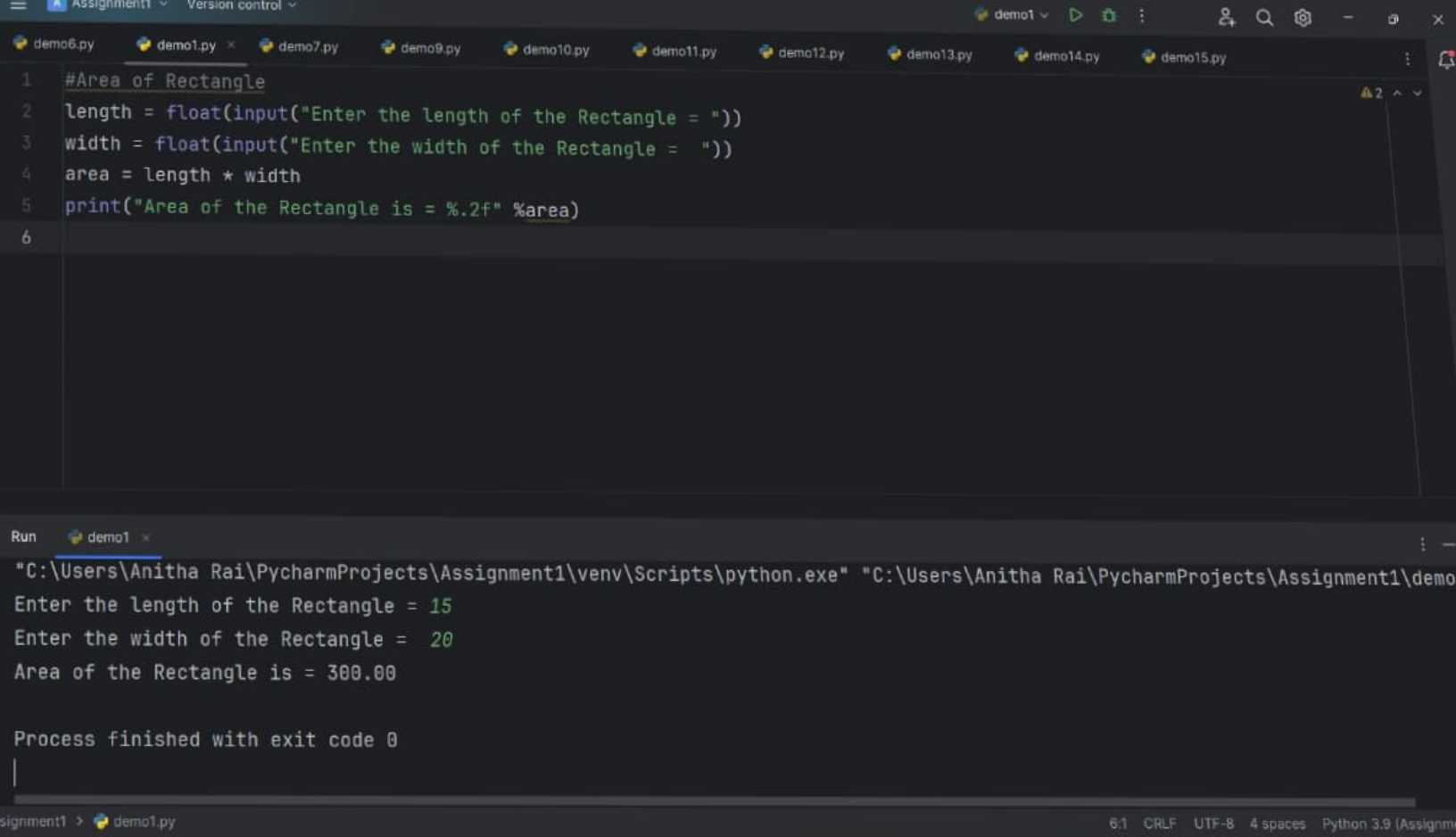
Assignment1
Version control
demo1
demo6.py
demo1.py
demo7.py
demo9.py
demo10.py
demo11.py
demo12.py
demo13.py
demo14.py
demo15.py
#Area of Rectangle
length = float(input("Enter the length of the Rectangle = "))
width = float(input("Enter the width of the Rectangle = "))
area = length * width
print("Area of the Rectangle is = %.2f" %area)
Run
demo1
"C:\Users\Anitha Rai\PycharmProjects\Assignment1\venv\Scripts\python.exe" "C:\Users\Anitha Rai\PycharmProjects\Assignment1\demo
Enter the length of the Rectangle = 15
Enter the width of the Rectangle = 20
Area of the Rectangle is = 300.00
Process finished with exit code 0
signment1 > demo1.py
6:1 CRLF UTF-8 4 spaces Python 3.9 (Assignm

demo6.py demo1.py demo2.py demo7.py demo10.py demo11.py demo12.py demo13.py demo14.py demo15.py

```python
# Miles to kilometer (1km = 0.621371 miles)
miles = int(input("Enter miles = "))
km = miles / 0.621371
print("Mile = ",miles)
print("kilometeres = ",km)

```

Run demo2

```
"C:\Users\Anitha Rai\PycharmProjects\Assignment1\venv\Scripts\python.exe" "C:\Users\Anitha Rai\PycharmProjects\Assignment1\demo
Enter miles = 10
Mile =  10
kilometeres =  16.093444978925632

Process finished with exit code 0
```

signment1 > demo2.py 7:1 CRLF UTF-8 4 spaces Python 3.9 (Assignm

2 usages

```python
def is_palindrome(text):
    length = len(text)

    for i in range(0, length // 2):
        if(text[i] != text[length - i - 1]):
            return False

    return True
string1 = "racecar"
print(is_palindrome(string1))

string1 = "abceba"
print(is_palindrome(string1))
```

Assignment1

demo6.py demo1.py demo2.py demo3.py demo7.py demo11.py

```python
#using function cheking of a given string palidrom or not
2 usages
def is_palindrome(text):
    length = len(text)

    for i in range(0, length // 2):
        if(text[i] != text[length - i - 1]):
            return False

    return True
string1 = "racecar"
print(is_palindrome(string1))
```

Run demo3

```
"C:\Users\Anitha Rai\PycharmProjects\Assignment1\venv\Scripts\python.exe" "C:\Users\Anitha Rai\PycharmProjects\Assignment1\demo3
True
False

Process finished with exit code 0
```

demo6.py demo1.py demo2.py demo3.py demo4.py demo7.py demo12.py demo13.py demo14.py demo15.py

```python
# second largest element in a list
list = [1, 4, 5, 6, 7, 99, 90, 100]
list.sort()
print(list)
print(list[-2])
```

Run demo4

```
"C:\Users\Anitha Rai\PycharmProjects\Assignment1\venv\Scripts\python.exe" "C:\Users\Anitha Rai\PycharmProjects\Assignment1\demo4
[1, 4, 5, 6, 7, 90, 99, 100]
99

Process finished with exit code 0
```

signment1 > demo4.py 4:12 CRLF UTF-8 4 spaces Python 3.9 (Assignmen

demo6.py demo1.py demo2.py demo3.py demo4.py demo5.py demo7.py demo13.py demo14.py demo15.py

```python
# operations in set
set1 = {"ani", "prami", "nani","Ronaldo"}
set2 = {"Ronaldo", "zendeya", "sonakshi"}
print(set1.union(set2))
print(set1.intersection(set2))
print(set1.difference(set2))
print(set1.symmetric_difference(set2))
```

Run demo5

```
{'zendeya', 'sonakshi', 'prami', 'nani', 'ani', 'Ronaldo'}
{'Ronaldo'}
{'prami', 'nani', 'ani'}
{'sonakshi', 'zendeya', 'prami', 'nani', 'ani'}

Process finished with exit code 0
```

8:1 CRLF UTF-8 4 spaces Python 3.9 (Assignment1)

signment1 > demo5.py

ENG IN 13:28 24-02-2024

demo6.py | demo1.py | demo2.py | demo3.py | demo4.py | demo5.py | demo7.py | demo13.py | demo14.py | demo15.py

```python
#using while loop printing of 1 to 10 numbers
i = 1
while i <= 10:
    print(i)
    i = i + 1
```

while i <= 10

Run demo6

```
1
2
3
4
5
6
7
```

ssignment1 > demo6.py

5:14 CRLF UTF-8 4 spaces Python 3.9 (Assignment

88°F Haze

Search

ENG IN

13:29 24-02-2024

demo6.py demo1.py demo2.py demo3.py demo4.py demo5.py demo7.py demo13.py demo14.py demo15.py

```python
#factorial of a number using while loop
num = int(input("enter a number:"))
fact = 1
while(num > 0):
    fact = fact * num
    num = num - 1
print("factorial of a number:",fact)

```

Run demo7

```
"C:\Users\Anitha Rai\PycharmProjects\Assignment1\venv\Scripts\python.exe" "C:\Users\Anitha Rai\PycharmProjects\Assignment1\demo
enter a number:5
factorial of a number: 120

Process finished with exit code 0
```

demo6.py demo1.py demo2.py demo3.py demo4.py demo5.py demo7.py demo8.py demo14.py demo15.py

```python
#cheking of positive,negative and zero using if_elif_else
num = int(input("enter a number:"))
if num > 0:
    print("positive number")
elif num < 0:
    print("negative number")
else:
    print("zero")
```

else

Run demo8

```
"C:\Users\Anitha Rai\PycharmProjects\Assignment1\venv\Scripts\python.exe" "C:\Users\Anitha Rai\PycharmProjects\Assignment1\demo
enter a number:10
positive number

Process finished with exit code 0
```

7:6 CRLF UTF-8 4 spaces Python 3.9 (Assign

signment1 > demo8.py

demo6.py demo1.py demo2.py demo3.py demo4.py demo5.py demo7.py demo8.py demo9.py demo15.py

```python
#finding largest among three numbers uding conditional statements
n1 = int(input("enter a number:"))
n2 = int(input("enter a number:"))
n3 = int(input("enter a number:"))
if n1 > n2 and n1 > n3:
    largest = n1
elif n2 > n1 and n2 > n3:
    largest = n2
else:
    largest = n3
print("largest among {} {} and {} is {}".format( *args: n1, n2, n3, largest))
```

else

Run demo9

```
enter a number:10
enter a number:20
enter a number:30
largest among 10 20 and 30 is 30

Process finished with exit code 0
```

9:6 CRLF UTF-8 4 spaces Python 3.9 (Assignment1)

Assignment1 > demo9.py

ENG IN 13:30 24-02-2024

demo6.py demo1.py demo2.py demo3.py demo4.py demo5.py demo7.py demo8.py demo9.py demo10.py

```python
#create a numpy array filled with ones
import numpy as np
1 usage
def create_ones_array(shape):
    return np.ones(shape)


# Example usage:
shape = (3, 4)  # Shape of the array
ones_array = create_ones_array(shape)
print("Array of ones:")
print(ones_array)
```

Run demo10

```
Array of ones:
[[1. 1. 1. 1.]
 [1. 1. 1. 1.]
 [1. 1. 1. 1.]]

Process finished with exit code 0
```

```
5
6
7  # Example usage:
8  shape = (3, 4)  # Shape of the array
9  random_array = create_random_array(shape)
10 print("2D Array initialized with random integers:")
11 print(random_array)
12
```

Run demo11

```
2D Array initialized with random integers:
[[6 4 9 3]
 [5 6 8 5]
 [3 9 4 9]]

Process finished with exit code 0
```

12:1 CRLF UTF-8 4 spaces Python 3.9 (Assignment1)

Assignment1 > demo11.py

ENG IN 13:36 24-02-2024

demo6.py demo3.py demo4.py demo5.py demo7.py demo8.py demo9.py demo10.py demo11.py demo12.py

```python
#generate an array of evenly spaced numbers over a specified range using linespace
import numpy as np
1 usage
def generate_linspace(start, stop, num):
    return np.linspace(start, stop, num)

# Example usage:
start = 0  # Start value of the sequence
stop = 10  # End value of the sequence
num = 5  # Number of samples
linspace_array = generate_linspace(start, stop, num)
print("Array of evenly spaced numbers using linspace:")
```

Run demo12

```
"C:\Users\Anitha Rai\PycharmProjects\Assignment1\venv\Scripts\python.exe" "C:\Users\Anitha Rai\PycharmProjects\Assignment1\demo12
Array of evenly spaced numbers using linspace:
[ 0.   2.5  5.   7.5 10. ]

Process finished with exit code 0
```

demo6.py demo4.py demo5.py demo7.py demo8.py demo9.py demo10.py demo11.py demo12.py demo13.py

```python
#python program generate an array off 10 equally spaced values between 1 and 100 using linspace
import numpy as np
equally_spaced_values = np.linspace( start: 1,  stop: 100,  num: 10)

# Print the result
print("Array of 10 equally spaced values between 1 and 100:")
print(equally_spaced_values)

```

Run demo13

```
"C:\Users\Anitha Rai\PycharmProjects\Assignment1\venv\Scripts\python.exe" "C:\Users\Anitha Rai\PycharmProjects\Assignment1\demo1
Array of 10 equally spaced values between 1 and 100:
[  1.  12.  23.  34.  45.  56.  67.  78.  89. 100.]

Process finished with exit code 0
```

Assignment1 > demo13.py 8:1 CRLF UTF-8 4 spaces Python 3.9 (Assignment

demo6.py demo5.py demo7.py demo8.py demo9.py demo10.py demo11.py demo12.py demo13.py demo14.py

```python
#write a python program to create an array containing even numbers from 2 to 20 using arrange
import numpy as np

# Create an array containing even numbers from 2 to 20
even_numbers = np.arange(2, 21, 2)

# Print the result
print("Array containing even numbers from 2 to 20:")
print(even_numbers)
```

Run demo14

```
"C:\Users\Anitha Rai\PycharmProjects\Assignment1\venv\Scripts\python.exe" "C:\Users\Anitha Rai\PycharmProjects\Assignment1\demo1
Array containing even numbers from 2 to 20:
[ 2  4  6  8 10 12 14 16 18 20]

Process finished with exit code 0
```

Assignment1 > demo14.py

10:1 CRLF UTF-8 4 spaces Python 3.9 (Assignment1)

demo6.py demo7.py demo8.py demo9.py demo10.py demo11.py demo12.py demo13.py demo14.py demo15.py

```python
#python program to create an array containing numbers from 1 to 10 using with a step size of 0.5 using arrange
import numpy as np
numbers = np.arange(1, 10.5, 0.5)

# Print the result
print("Array containing numbers from 1 to 10 with a step size of 0.5:")
print(numbers)

```

Run demo15

```
"C:\Users\Anitha Rai\PycharmProjects\Assignment1\venv\Scripts\python.exe" "C:\Users\Anitha Rai\PycharmProjects\Assignment1\demo15
Array containing numbers from 1 to 10 with a step size of 0.5:
[ 1.   1.5  2.   2.5  3.   3.5  4.   4.5  5.   5.5  6.   6.5  7.   7.5
  8.   8.5  9.   9.5 10. ]

Process finished with exit code 0
```

Q]Explain indentation means in python.
In essence, indentation helps Python know the structure of your code, like where loops, conditionals, functions, and other blocks of code begin and end. It's a fundamental part of how Python organizes and understands your program.