Python if else, for loop, and range() Exercises with Solutions

Exercise 1: Print First 10 natural numbers using while loop

Expected output:

```
1
2
3
4
5
6
7
8
9
10
```

Exercise 2: Print the following pattern

Write a program to print the following number pattern using a loop.

```
1
1 2
```

```
1 2 3

1 2 3 4

1 2 3 4 5
```


Exercise 3: Calculate the sum of all numbers from 1 to a given number

Write a program to accept a number from a user and calculate the sum of all numbers from 1 to a given number

For example, if the user entered **10** the output should be **55** (`1+2+3+4+5+6+7+8+9+10`)

Expected Output:

```
Enter number 10

Sum is:  55
```

4: Write a program to print multiplication table of a given number

For example, `num = 2` so the output should be

```
2

4

6

8

10

12
```

```
14

16

18

20
```

```
n = 2
# stop: 11 (because range never include stop number in result)
# run loop 10 times
for i in range(1, 11, 1):
    # 2 *i (current number)
    product = n * i
    print(product)
```

Exercise 5: Display numbers from a list using loop

Write a program to display only those numbers from a list that satisfy the following conditions

- The number must be divisible by five
- If the number is greater than 150, then skip it and move to the next number
- If the number is greater than 500, then stop the loop

Given:

```
numbers = [12, 75, 150, 180, 145, 525, 50]
```

Expected output:

```
75

150

145
```

Exercise 7: Print the following pattern

Write a program to use `for` loop to print the following reverse number pattern

```
5 4 3 2 1

4 3 2 1

3 2 1

2 1

1
```

8: Print list in reverse order using a loop

Given:

AD

```
list1 = [10, 20, 30, 40, 50]
```

Expected output:

```
50

40

30

20

10
```

Exercise 9: Display numbers from -10 to -1 using for loop

Expected output:

```
-10
-9
-8
-7
-6
-5
-4
-3
-2
-1
```

Exercise 10: Use else block to display a message “Done” after successful execution of for loop

For example, the following loop will execute without any error.

Given:

```
for i in range(5):
    print(i)
```

Expected output:

```
0

1

2

3

4

Done!
```