Q1) Given a range of the first 10 numbers, Iterate from the start number to the end number, and In each iteration print the sum of the current number and previous number

main.py

```
1   def range1(num):
2       lastNum = 0
3       for i in range(num):
4           sum = lastNum + i
5           print("Current Number:", i, "Previous Number: ", lastNum,"
            Sum: ", sum)
6           previousNum = i
7
8   print("The output is: ")
9   range1(10)
```

Console   Shell

```
The output is:
Current Number: 0 Previous Number:  0   Sum:  0
Current Number: 1 Previous Number:  0   Sum:  1
Current Number: 2 Previous Number:  0   Sum:  2
Current Number: 3 Previous Number:  0   Sum:  3
Current Number: 4 Previous Number:  0   Sum:  4
Current Number: 5 Previous Number:  0   Sum:  5
Current Number: 6 Previous Number:  0   Sum:  6
Current Number: 7 Previous Number:  0   Sum:  7
Current Number: 8 Previous Number:  0   Sum:  8
Current Number: 9 Previous Number:  0   Sum:  9
>
```

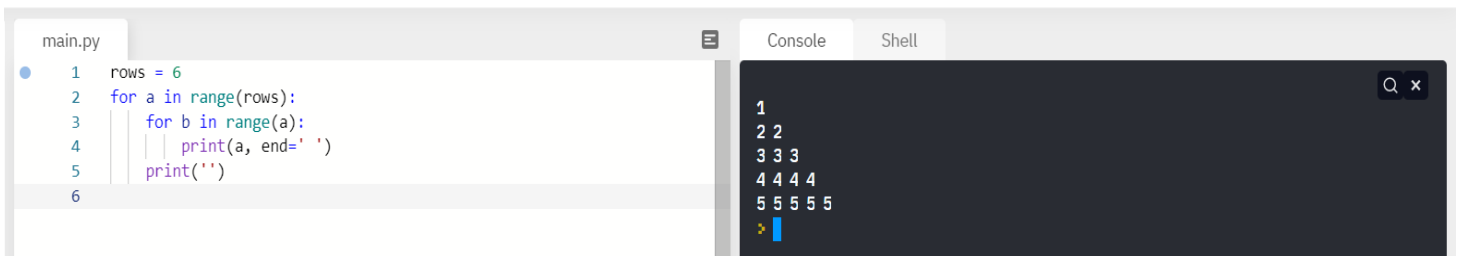**Q2) Take list of numbers from users, return True if first and last number of a list is same.**

main.py

```
1   def quest2(numList):
2       firstElement = numList[0]
3       lastElement = numList[-1]
4       if (firstElement == lastElement):
5           return True
6       else:
7           return False
8
9   n = int(input("ENTER THE NUMBER OF LIST: "))
10  numList = []
11  for i in range(n):
12      element = int(input("ENTER THE ELEMENTS: "))
13      numList.append(element)
14
15  print("The Result is: ", quest2(numList))
16
```

Console   Shell

```
ENTER THE NUMBER OF LIST: 10
ENTER THE ELEMENTS: 12
ENTER THE ELEMENTS: 13
ENTER THE ELEMENTS: 16
ENTER THE ELEMENTS: 98
ENTER THE ELEMENTS: 69
ENTER THE ELEMENTS: 124
ENTER THE ELEMENTS: 56
ENTER THE ELEMENTS: 99
ENTER THE ELEMENTS: 98
ENTER THE ELEMENTS: 12
The Result is:  True
>
```
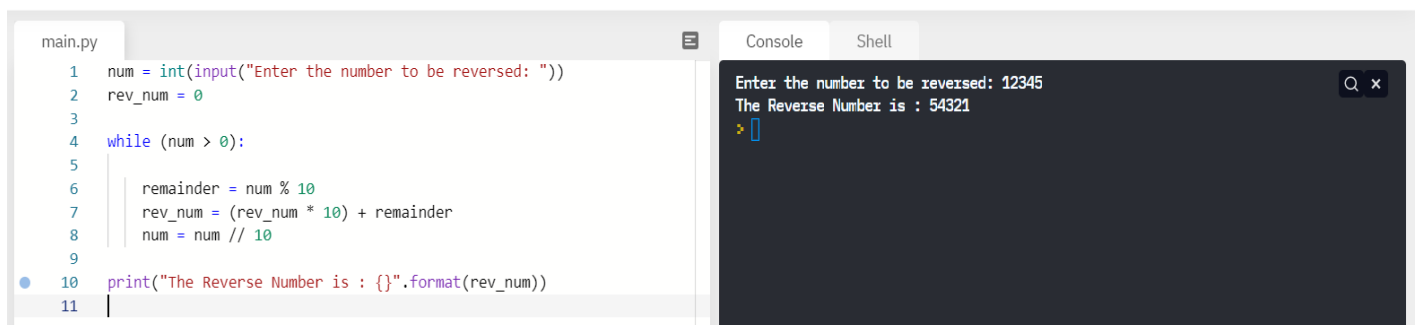
Q3) Print the following pattern

1
2 2
3 3 3
4 4 4 4
5 5 5 5 5

```
main.py
1   rows = 6
2   for a in range(rows):
3       for b in range(a):
4           print(a, end=' ')
5       print('')
6
```

Console   Shell

```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
>
```
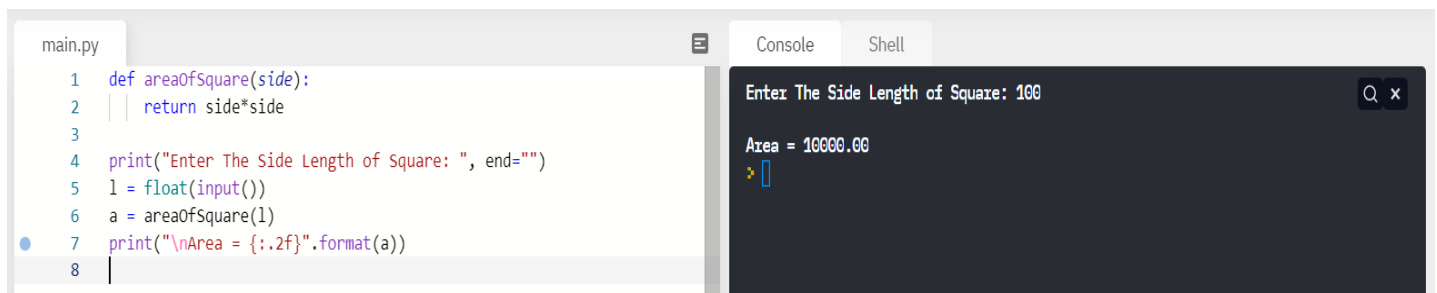
Q4) Write a code to extract each digit from an integer, in the reverse order.

```
main.py
1   num = int(input("Enter the number to be reversed: "))
2   rev_num = 0
3
4   while (num > 0):
5
6       remainder = num % 10
7       rev_num = (rev_num * 10) + remainder
8       num = num // 10
9
10  print("The Reverse Number is : {}".format(rev_num))
11
```

Console   Shell

```
Enter the number to be reversed: 12345
The Reverse Number is : 54321
>
```

Q5) Create a function that returns the area of a square.

```
main.py
1   def areaOfSquare(side):
2       return side*side
3
4   print("Enter The Side Length of Square: ", end="")
5   l = float(input())
6   a = areaOfSquare(l)
7   print("\nArea = {:.2f}".format(a))
8
```

Console   Shell

```
Enter The Side Length of Square: 100

Area = 10000.00
>
```

Q6) Print First 10 prime numbers.

```python
def Prime(n):
    for i in range(2,n//2+1):
        if(n%i==0):
            return(0)
    return(1)

N=int(input("Enter Number:"))
i=2
lst=[]
while(1):
    if(Prime(i)):
        lst.append(i)
        if(len(lst)==N):
            break
    i+=1
print( "Prime numbers are:",end="")
print(*lst)
```

Console | Shell

```
Enter Number:10
Prime numbers are:2 3 5 7 11 13 17 19 23 29
>
```

Q7) Accept number from user and calculate the sum of all numbers from 1 to a given number.

```python
n = int(input("Enter Number whose sum needs to be calculated:"))
sum = 0
for number in range(1, n + 1, 1):
    sum = sum + number
print("Sum from 1 to given number is: ",sum)
```

Console | Shell

```
Enter Number whose sum needs to be calculated:29
Sum from 1 to given number is:  435
>
```

Q8) Given a list, iterate it, and display numbers divisible by five, and if you find a number greater than 150, stop the loop iteration.

```python
list1 = [10, 20, 25, 42, 55, 75, 122, 132, 150, 180, 200, 67, 95]
for num in list1:
    if (num > 150):
        break
    if(num % 5 == 0):
        print(num)
```

Console | Shell

```
10
20
25
55
75
150
>
```

Q9) Write an SQL query to fetch unique values of DEPARTMENT from Worker table.

Ans: SELECT DISTINCT DEPARTMENT FROM WORKER ;

Q10) Write an SQL query to print all Worker details from the Worker table order by FIRST_NAME Ascending.

Ans: SELECT * FROM WORKER ORDER BY FIRST_NAME ASC ;

Q11) Write an SQL query to print all Worker details from the Worker table order by FIRST_NAME Ascending and DEPARTMENT Descending.

Ans: SELECT * FROM WORKER ORDER BY FIRST_NAME ASC, DEPARTMENT DESC ;

Q12) Write an SQL query to print details for Workers with the first name as "Vipul" and "Satish" from Worker table.

Ans: SELECT * FROM WORKER WHERE FIRST_NAME IN ('Vipul' , 'Satish) ;

Q13) Write an SQL query to print details of workers excluding first names, "Vipul" and "Satish" from Worker table.

Ans: SELECT * FROM WORKER WHERE FIRST_NAME NOT IN ('Vipul' , 'Satish') ;

Q14) Write an SQL query to print details of Workers with DEPARTMENT name as "Admin"

Ans: SELECT * FROM WORKER WHERE DEPARTMENT LIKE 'ADMIN%' ;