Week - 3

Introduction AI and DS using python Labaratory.

Q1 → Write a python program that prompts the user to input a number from 1 to 7. The program should display the corresponding day for the given number. For example, if the user types 1, the output should be Sunday. If the user types 7, the output should be Saturday.

Syntax / code →

```
num = int(input("Enter number (1-7): "))

days = ["Sunday", "Monday", "Tuesday", "Wednesday",
        "Thursday", "Friday", "Saturday"]

print(days[num-1])
```

Output :-

```
Enter number (1-7): 1
Sunday.
```

Expt. No. Date

Page No.

Q2. Write a python program that prompts the user to input the number of calls and calculate the monthly telephone bills as per the following rule:

- Minimum Rs. 200 for up to 100 calls.
- Plus Rs. 0.60 per call for the next 50 calls.
- Plus Rs. 0.50 per call for the next 50 calls.
- Plus Rs. 0.40 per call for any call beyond 200 calls.

Syntax / Code →

```
calls = int( input ("Enter number of calls : "))
bill = 200
if calls >100:
    if calls <=150:
        bill += (calls -100) * 0.60
    elif calls <= 200:
        bill += 50 * 0.60 + (calls- 150)*0.50
    else:
        bill += 50 * 0.6 + 50 * 0.5 + (calls -200)
        * 0.40
print ("Bill:", bill)
```

AJAYA

Teacher's Signature

Output →

```
Enter number of calls :- 220
Bill : 270.0
```

Q3 → Write a python program to calculate the factorial of a number.

Syntax / code →

```
n = int(input("Enter number: "))
fact = 1
for i in range (1, n+1):
    fact *= i
print ("Factorial:", fact)
```

Output :-

```
Enter number :5
Factorial : 120
```

Q4 → Write a python program to calculate the fibonacci sequence till a specific number of terms.

Syntax / code →

```
n = int(input("Enter terms : "))
a,b = 0,1
for i in range(n) :
    print (a, end = " ")
    a,b = b, a+b
```

output →

Enter terms :- 6
0 1 1 2 3 5

Q5 → Write a python program to calculate the factors of a number.

Syntax / code →

```
n = int (input ("Enter number : "))
for i in range (1, n+1):
    if n % i == 0:
        print (i, end = " ")
```

Output → Enter number : 12
1 2 3 4 6 12

Q6 → Write a Python program to calculate the magic square based on a given number.

Syntax / code →

```
n= int (input ("Enter odd number: "))
magic = [[0]*n for _ in range (n)]
i,j = 0, n//2
for num in range (1, n*n+1):
    magic [i][j] = num
    i2, j2 = (i-1) % n, (j+1) % n
```

```
        if magic[i2][j2]:
            i = (i+1)%n
        else:
            i,j = i2, j2
    for row in magic:
        print(row)
```

Output :→ Enter odd number : 3
[8, 1, 6]
[3, 5, 7]
[4, 9, 2]

07. → write a python program to check if a number is a palindrome.

Syntax / code →

```
n = input("Enter number:")
print("Palindrome" if n==n[::-1]
      else "Not Palindrome")
```

Output :- Enter number : 121
Palindrome

08. → Write a Python program to check if a number is an Armstrong number.

Syntax / code :-

```
n= int(input("Enter number:"))
```

```
s = sum (int (d)**3 for d in str (n))
print ("Armstrong" if s==n else "Not
        Armstrong")
```

Output :- Enter number : 153
Armstrong.

Q9 → Write a Python program to check if a number is a krishnamurthy number.

Syntax / code :-

```
import math
n= int(input (" Enter number: "))
s = sum (math.factorial (int(d)) for d
    in str(n))
print ("Krishnamurthy" if s==n
    else "Not Krishnamurthy")
```

Output :-

Enter number : 145
Krishnamurthy.

Q10.→ Write a Python program to find the sum of digits of a number.

Syntax / code :-

```
n = int (input ("Enter number :"))
s = sum (int (d) for d (n str (n))
print ("Sum:" , s)
```

Output : Enter number : 1234
Sum : 10

Q11. → Write a Python program to reverse a given number.

Syntax / code :-

```
n = input ("Enter number :")
print ("Reverse:", n[::-1])
```

Output :- Enter Number : 1234
Reverse : 4321

Q12 → Write a Python program to find the sum of squares of the first n natural numbers.

Syntax / Code :-

```
n = int (input ("Enter n :"))
```

Date

Expt. No. _____ Page No. _____

```
s = sum(i * i for i in range(1, n+1))
print("Sum of Squares :", s)
```

Output : Enter n : 5
Sum of squares : 55

Q13. → Write a python program to convert a decimal number to a binary number.

Syntax / code :-

```
n = int(input("Enter decimal :"))
print("Binary :", bin(n)[2:])
```

Output : Enter decimal : 10
Binary : 1010

Q14 :→ Write a Python program that prompts the user to input a number and prints its multiplication table.

Syntax / Code :-

```
n = int(input("Enter number: "))
for i in range(1, 11):
    print(n, "x", i, "=", n * i)
```

AJAYA

Teacher's Signature _____

Output :-

```
Enter number : 5
5 x 1  =  5
5 x 2  =  10
5 x 3  =  15
5 x 4  =  20
5 x 5  =  25
5 x 6  =  30
5 x 7  =  35
5 x 8  =  40
5 x 9  =  45
5 x 10 =  50
```

Q15 → write a python program to print the first 6 terms of a geometric sequence starting with 2 and having a common ratio.

Syntax / code :-

```
a, r = 2, 2
for i in range(6):
    print(a*(r**i), end=" ")
```

Output :- 2 4 8 16 32 64

Date

Expt. No. Page No.

Week - 4 Programs

Q1 → Print the series up to N terms : 1, 4, 9, 16, 25, 36, ---

Syntax/code :-

```
n = int (input ("Enter N: "))
for i in range (1, n+1):
    print (i ** 2, end = " ")
```

Output :- Enter N : 5
1 4 9 16 25

Q2 → Print the series up to N terms: 2, 4, 8, 16, 32, 64, ---

Syntax/ code :-

```
n = int ( input ("Enter N :"))
for i in range (1, n+1) :
    print (2 **( , end = " ")
```

Output →

Enter N : 6
2 4 8 16 32 64

Q3 → Print the series up to N terms:
1, 3, 7, 13, 21, 31, ---

Syntax / code :-

```
n = int(input("Enter N: "))
num = 1
diff = 2
for _ in range(n):
    print(num, end=" ")
    num += diff
    diff += 2
```

Output ⇒ Enter N: 6
1 3 7 13 21 31

Q4 → Print the series up to N terms:
1, 2, 6, 24, 120, 720 ---

Syntax / code :-

```
n = int(input("Enter N:"))
fact = 1
for i in range(1, n+1):
    fact *= i
    print(fact, end=" ")
```

Output :- Enter N: 6
1 2 6 24 120 720

Date

Expt. No. _____ Page No. _____

Q5→ Compute the sum up to n terms in the series $1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \ldots \pm \frac{1}{n}$

Syntax / Code :-

```
n = int(input("Enter N:"))
total = 0
for i in range(1, n+1):
    if i % 2 == 0:
        total -= 1/i
    else:
        total += 1/i
print("Sum:", total)
```

Output : Enter N : 5
Sum : 0.7833333333

Q6→ Count positives, negatives, and zeros entered by the user until they stop.

Syntax / Code :-

```
pos = neg = zero = 0
while True:
    num = int(input("Enter number (999 to
        stop): "))
    if num > 0:
        pos += 1
```

AJAYA

Teacher's Signature _____

```
    if num == 999:
        break
    if num > 0:
        pos += 1
    elif num < 0:
        neg += 1
    else:
        zero += 1
print ("Positive:", pos, "Negative:", neg,
       "Zeros:", zero)
```

Output :-

```
Enter number (999 to Stop) : 5
Enter number (999 to stop) : -2
Enter number (999 to stop) : 0
Enter number (999 to stop) : 999
Positive : 1  Negative : 1  Zeros : 1
```

Q7.→ Find HCF ~~for~~ of two Numbers.

Syntax / Code :-

```
a = int( input ("Enter first number: "))
b = int(input("Enter second number : "))
while b :
    a, b = b, a % b
print ("HCF:" , a)
```

Output :-

```
Enter first number : 36
Enter second number : 60
HCF : 12
```

Date

Expt. No. Page No.

Q8 → Sum of first 7 terms of the series $1 + \frac{2}{3} + \frac{3}{5} + \cdots$

Syntax / code :-

```
total = 0
num = 1
den = 1
for _ in range(7):
    total += num / den
    num += 1
    den += 2
print("Sum:", total)
```

Output :-

Sum: 10.207142857142857

Q9 → Print the pattern up to N lines (special number arrangement)

Syntax / codes :-

```
n = int(input("Enter N: "))
num = 1
matrix = [[0]*n for _ in range(n)]

top, bottom, left, right = 0, n-1, 0, n-1
while left <= right and top <= bottom:
    for i in range(left, right+1):
        matrix[top][i] = num
```

AJAYA

Teacher's Signature

```
            num += 1
        top += 1
        for i in range(top, bottom+1):
            matrix[i][right] = num
            num += 1
        right -= 1
        for i in range(right, left-1, -1):
            matrix[bottom][i] = num
            num += 1
        bottom -= 1
        for i in range(bottom, top-1, -1):
            matrix[i][left] = num
            num += 1
        left += 1

for row in matrix:
    print(" ".join(str(x) for x in row))
```

Output :- Enter N : 3

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

Q10 → write programs to print following patterns for N=5.

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

Date

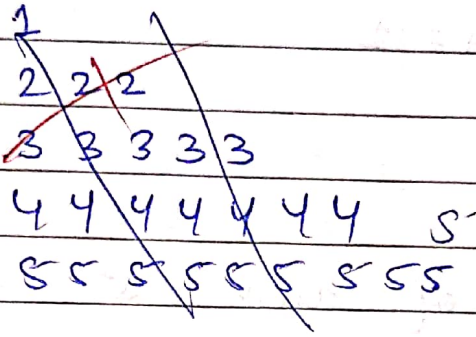
Expt. No.

Page No.

Syntax / code →

```
n = int (input ("Enter N: "))
for i in range (1, n+1):
    print (str(i) * (2*i - 1)).
```

output → Enter N: 5



output :-

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

Q11 → Pascal's Triangle up to N lines.

Syntax / code →

```
n = int (input ("Enter N:"))
for i in range (n):
    num = 1      row = []
    print (" " * (n-i), end = " ")
    for j in range (i+1):
        print (num, end = " ")
        row.append (str (num))
        num = num * (i-j) // (j+1)
    print (" ".join (row).centre (N*4))
```

Output :- Enter N: 5

```
    1
   1 1
  1 2 1
 1 3 3 1
1 4 6 4 1
```

AJAYA

Teacher's Signature

Q12.→ Floyd's Triangle

Syntax / code :-

```
n = int (input ("Enter number of rows: "))
num = 1
for i in range (1, n+1):
    for j in range (i):
        print (num, end = " ")
        num += 1
    print ()
```

Output :-

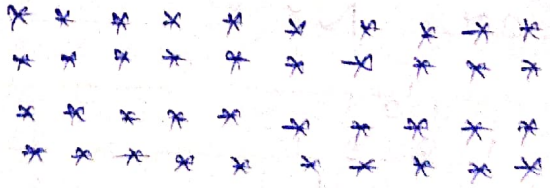
```
Enter number of rows : 5
1
2  3
4  5  6
7  8  9  10
11 12 13 14 15
```

Q13 → Star Pattern

Syntax / code :-

```
print (" * * * * * * * * * ")
print (" * * * * * * * * * ")
print (" * * * * * * * * * ")
print (" * * * * * * * * * ")
```

Output :-



Date

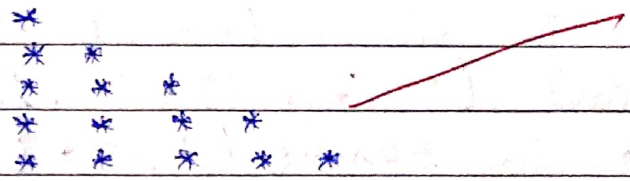
Expt. No. _____ Page No. _____

Q14 → Star pattern (left-aligned)

syntax / code :-

```
n = 5
for i in range (1, n+1):
    print ("*" * i)
```

output :-

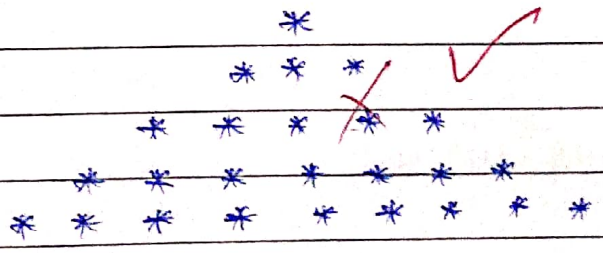


Q15 → Centre-Aligned Pyramid

Syntax / code :-

```
n = 5
for i in range (1, n+1):
    spaces = n - i
    stars = 2 * i - 1
    print (" " * spaces + "*" * stars)
```

output :-



Q16 - Diamond pattern

Syntax / code :-

```
n = 3   # middle row height
for i in range (1, n+1):
    print (" "*(n-i) + " ".join(str(x) for
    x in range (1, 2*i)))

for i in range (n-1, 0, -1):
    print (" "*(n-i) + " ".join (str(x) for
    x in range (1, 2*i)))
```

Output :-

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

28/09/25

Date

Expt. No. Page No.

Week - 5 practicals

Q1→ Implement the following functions / methods which operates on lists in python with suitable examples:

(a) list ()

code →

```
my_list = list ((1, 2, 3, 4))
print (my_list)
```

output → [1, 2, 3, 4]

(b) len()

code :-

```
my_list = [10, 20, 30, 40]
print (len (my_list))
```

output :
4

(c) count ()

code :

```
my_list = [1, 2, 2, 3, 2, 4]
print (my_list.count (2))
```

output :
3

(d) index ()

code :

```
my_list = [5, 10, 15, 20]
print (my_list.index (15))
```

output :
2

AJAYA

Teacher's Signature

(e) append ()

code:

```
my_list = [1,2,3]
my_list.append (4)
print (my_list)
```

output:

[1, 2, 3, 4]

(f) insert ()

code:

```
my_list = [1,2,4]
my_list.insert (2,3)
print (my_list)
```

output:

[1, 2, 3, 4]

(g) extend ()

code:

```
my_list = [1,2]
my_list.extend ([3,4,5])
print (my_list)
```

output:

[1, 2, 3, 4, 5]

(h) remove ()

code:

```
my_list = [10,20,30,40]
my_list.remove (30)
print (my_list)
```

output:

[10, 20, 40]

(i) pop () code:

```
my_list = [100,200,300]
my_list.pop ()
print (my_list)
```

output:

[100, 200]

Date

Expt. No. Page No.

(j) reverse()

code:
```
my_list = [1, 2, 3, 4]
my_list.reverse()
print(my_list)
```

output:
[4, 3, 2, 1]

(k) sort()

code:
```
my_list = [50, 10, 30, 20]
my_list.sort()
print(my_list)
```

output:
[10, 20, 30, 50]

(l) copy()

code:
```
my_list = [5, 10, 15]
new_list = my_list.copy()
print(new_list)
```

output:
[5, 10, 15]

(m) Clear()

code:
```
my_list = [1, 2, 3]
my_list.clear()
print(my_list)
```

output:
[]

Q2 → Implement the following functions / methods which operates on tuples in Python with suitable examples:

(a) len ()

```
my_tuple = (10,20,30,40)
print(len(my_tuple))
```

output :

4

(b) count ()

```
my_tuple = (1,2,2,3,2)
print (my_tuple.count (2))
```

output :

3

(c) index ()

```
my_tuple = (5,10,15,20)
print (my_tuple.index (15))
```

output :

2

(d) sorted ()

```
my_tuple = (40,10,30,20)
print (sorted (my_tuple))
```

output :

[10,20,30,40]

(e) min ()

```
my_tuple = (7,3,9,1)
print (min (my_tuple))
```

output :

1

(f) max ()

```
my_tuple = (7,3,9,1)
print (max (my_tuple))
```

output :

9

Date

Expt. No.

Page No.

(g) cmp():

Note: cmp() is not available in Python 3. we can implement using comparision operators.

code:

```
a = (1, 2, 3)
b = (1, 2, 4)
if a == b:
    print ("Both are equal")
elif a < b:
    print ("a is smaller than b")
else:
    print ("a is greater than b")
```

output:

a is smaller than b

(h) reversed ()

output:

```
my_tuple = (1, 2, 3, 4)
print(tuple(reversed(my_tuple)))
```

(4, 3, 2, 1)

Q3 → Write a python program that accepts a string from user. Your program should count and display number of vowels is that string.

AJAYA

Teacher's Signature

Code :-

```
string = input ("Enter a string : ")
vowels = "AEIOUaeiou"
count = 0

for char in string:
    if char in vowels:
        count += 1

print ("Number of Vowels :", count )
```

Output :-

```
Enter a string : HELLO
Number of vowels : 2
```

Q4 → write a python program that reads a string from the keyboard and display :

- The number of uppercase letters.
- The number of lowercase letters.
- The number of digits.
- The number of whitespace characters.

Code:

```
string = input ("Enter a string :")

upper = lower = digit = space = 0
```

Date

Expt. No. Page No.

```
for ch in string:
    if ch.isupper():
        upper += 1
    elif ch.islower():
        lower += 1
    elif ch.isdigit():
        digit += 1
    elif ch.isspace():
        space += 1

print("Uppercase letters:", upper)
print("Lowercase letters:", lower)
print("Digits:", digit)
print("Whitespaces:", space)
```

Output:

```
Enter a string: Hello World 123
Uppercase letters: 2
Lowercase letters: 8
Digits : 3
Whitespaces : 2
```

Q5 → Write a python program that accepts a string from user. Your program should create a new string in reverse of first string and display it.

AJAYA

Teacher's Signature

code :

```
string = input ("Enter a string: ")
new_string = string [::-1]
print ("Reversed string:", new_string)
```

Output :

```
Enter a string : EXAM
Reversed String : MAXE
```

Q6 → Write a program that display following output :

```
SHIFT
HIFTS
IFTSH
FTSHI
TSHIF
SHIFT
```

code:

```
String = "SHIFT"
for i in range (len (string) + 1):
    rotated = string [i:] + string [:i]
    print (rotated)
```

Output :

```
SHIFT
HIFTS
IFTSH
FTSHI
TSHIF
SHIFT
```

Date

Expt. No. _____ Page No. _____

07 → write a python program that accepts a string from user. Your program should create and display a new string where the first and last characters have been exchanged.

Code:

```
string = input("Enter a string:")
new_string = string[-1] + string[1:-1] +
             string[0]
print("New string", new_string)
```

output:

```
Enter a string : HELLO
New string : OELLH
```

08 → Write a program that asks the user to input his name and print its initials (without using split()).

Code:

```
name = input("Enter your full name:")

initials = ""
initials += name[0] + "."

for i in range(1, len(name)):
    if name[i-1] == " ":
```

AJAYA

Teacher's Signature _____

```
        initials += name[i] + ". "

print ("Initials:", initials)
```

output :

Enter your full name: Pranay keshwani

Initials : P. K.