Idris Ratlamwala.
2003146 - C31

Expt - 1.1

1. Aim: Write a python program to print the following:

Twinkle, Twinkle, little star,
"How I wonder what you are!"
Up above the world so high,
Like a diamond in the sky.
Twinkle, Twinkle, little star,
How I wonder what you are

Theory:

print(): This is a function that prints the given object to the std output device (screen).

Syntax:

print(*objects, sep=' ', end='\n', flush=False)

parameters:

1) Objects - objects to be printed. '*' indicates that there may be more than one object.
2) sep - Objects are separated by sep. Default value: ' '
3) end - 'end' is printed at last.
4) flush - If true, the stream is forcibly flused. Default value: False.

printf() doesn't return any value.

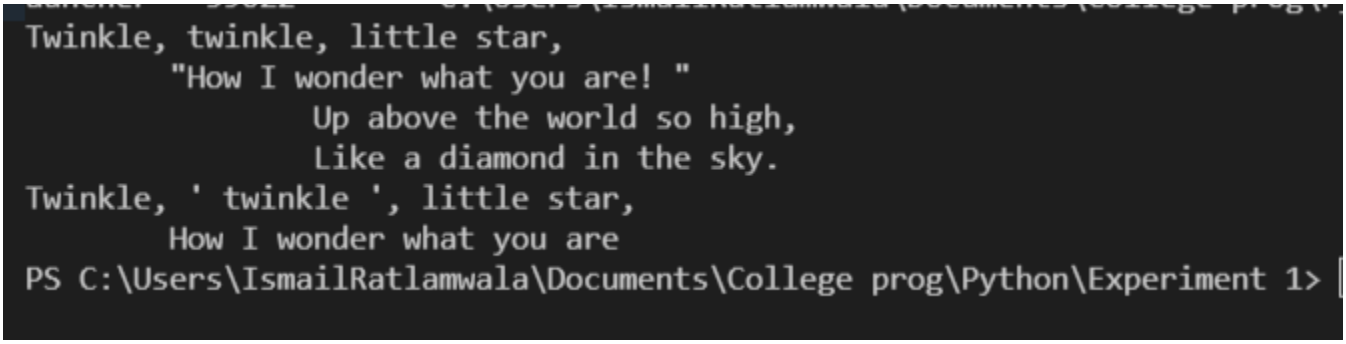
Escape sequences:

code	result
\'	Single quote
\\	Backslash
\n	Newline
\t	Tab
\r	Carriage Return
\f	Form Feed

Program :

```
print('''Twinkle, twinkle, little star,\n\t"How I wonder what you are! "\n\t\tUp
above the world so high,\n\t\tLike a diamond in the sky.\nTwinkle, ' twinkle ',
little star,\n\tHow I wonder what you are''')
```

Output :



Idris Ratlamwala
2003145- C31

Expt-1.2

2. Aim: WAP to show output formatting take 2 values & display them using single printf function.
   - str.format()
   - % operator

Theory:

Str.format(): The format() method formats the specified value(s) and insert them in string's placeholder.

- The placeholder is predefined using curly brackets: {}
- format() method returns the string (formatted)

Syntax:

String.format(value 1, value 2}...)

Parameters: *Required, one or more values that should be formatted and inserted in the string.

- These values are either seperated by comma, a key=value list, or a combination of both.
- Values can be of any data type.

% operator: It is the oldest method of string formatting, using modulo '%' operator.

format specifier	used to inject
%s	strings
%d	integers
%f	binarys
%b	float

Syntax:

string % (value 1, value 2 ...)

eg. "I walked %d km from %s" % (1, 'my home')

Program :

```
name = str(input("Enter your name : "))
age = int(input("Enter age : "))

print("""\nUsing str.format()
Your name is {}, you're {}\n""".format(name,age))

print("""Using modulo operator
Your name is %s, you're %d\n"""%(name,age)
```

Output :

```
Enter your name : Idris Ratlamwala
Enter age : 20

Using str.format()
Your name is Idris Ratlamwala, you're 20

Using modulo operator
Your name is Idris Ratlamwala, you're 20
```

Idris Rotlamwala
2003145 - C31

Expt - 1.3

3. Aim : WAP to find leap year using nested if.

Theory :

Decision making is required when we want to execute a code only if a certain condition is satisfied. The "if... elif... else" statements are used in python for decision making.

Here the program evaluates the test expression & will execute statements only if the test expression is 'True'.

In python, the body of the 'if' statement is indicated by identation. It starts with an identaion & ends before first unidented line.

Syntax :

```
if test expression:
    Body of if....
elif test expression:
    Body of elif....
else:
    Body of else....
```

The elif is short for else if, allowing us to check multiple expressions.

If all the conditions are false, the body of else is executed. The if block can have only 1 else, but multiple elif blocks.

Program :

```
year = int(input("Enter the Year to be checked: "))

if(year % 4 == 0):
        if(year % 100 !=0):
            print("%d is a Leap Year" % year)
        else :
            if year % 400==0 :
                print("%d is a Leap Year" % year)
            else :
                print("%d is Not the Leap Year" % year)
else:
    print("%d is Not the Leap Year" % year)
```

Output :

```
Enter the Year to be checked: 123
123 is Not the Leap Year
PS C:\Users\IsmailRatlamwala\Documents\College prog\Python\Experiment 1>
lamwala\Documents\College prog\Python\Experiment 1'; & 'C:\Users\IsmailRa
rams\Python\Python310\python.exe' 'c:\Users\IsmailRatlamwala\.vscode\exte
1.12.1559732655\pythonFiles\lib\python\debugpy\launcher' '64771' '--' 'c:
uments\College prog\Python\Experiment 1\3.py'
Enter the Year to be checked: 2004
2004 is a Leap Year
PS C:\Users\IsmailRatlamwala\Documents\College prog\Python\Experiment 1>
```

Expt-1.4

Idris Ratlamwala
2003145 - C31

4. Aim: WAP to print all armstrong numbers in range 1 to 1000.

Theory:

The for loop is used to iterate over a sequence such as list, tuple or string, or other iterable objects.

Syntax:

```
for val in range(start, stop, step_size)
    loop body
```

Here, 'val' is the variable that takes value of the item inside the sequence (range) on each iteration. Loop continues until we reach last item in the sequence. The body of for is seperated from rest of code using identation.

range():

We can generate a sequence of numbers using range() function. eg

eg. range(10) will generate numbers from 0 to 9 (10 num)

We can also define start, stop & size of step as range(start, stop, step_size)

start defaults to 0 & step_size to 1, if not provided.

Hence,

```
for i range(5)
    print(i, end=" ")
```

will print: 0 1 2 3 4

Program :

```
for num in range(1, 1001):

    n = len(str(num))
    sum = 0

    temp = num
    while temp > 0:
        digit = temp % 10
        sum += digit ** n
        temp = int(temp/10)
    if num == sum:
        print(num)
```

Output :

```
1
2
3
4
5
6
7
8
9
153
370
371
407
```

Idris Rattonwala
2003145 - C31

Expt - 1.5

5. Aim: Write a Program to find fibbnacci series of n terms.

Theory:

While loop in python is used to iterate over a blocks of code until, the test expression is true.

This loop is generally used when we don't know how many times the loop is going to get executed. (beforehand)

Syntax:

```
while test_expression:
    Body of while ....
```

In the while loop, the test exp is tested first, The body of while is entered only if the test exp is evaluates to true. After 1 iteration the loop is checked again. This process continues until the test exp evaluates to false.

Python interprets any non zero value as 'True'. 'None' and '0' are interpreted as 'false'.

Program :

```
n = int(input("Enter the value of 'n': "))
a = 0
b = 1
sum = 0
count = 1

print("Fibonacci Series: ", end=" ")
while(count <= n):
    print(sum, end=" ")
    a = b
    b = sum
    sum = a + b
    count +=1
```

Output :

```
Enter the value of 'n': 10
Fibonacci Series:  0 1 1 2 3 5 8 13 21 34 PS C:\U
\Experiment 1>
```

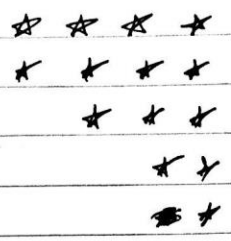
Idris Ratlamwala
2003145- C31

Expt-1.6

6. Aim : Write Programs on pattern

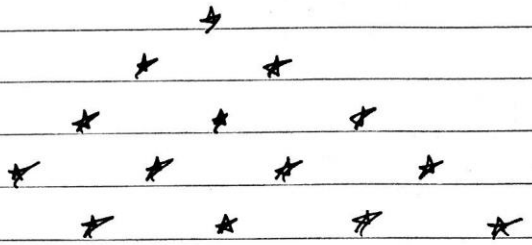
a)
```
A
B B
C C C
D D D D
E E E E E
```

b)



c)
```
        1
      1 2 1
    1 2 3 2 1
  1 2 3 4 3 2 1
1 2 3 4 5 4 3 2 1
```

d)



Theory :

for loop is used to print the various patterns The multiple loops are used to print the patterns where the first outer loop is used to print the number of columns.

- The outer loop to print the number of rows
- The inner loops to print the number of columns.
- Sometimes an extra variable to keep track of count.

Program /Output :

A.

```
for i in range(5):
    for j in range(i+1):
        print(chr(i+65),end=" ")
    print()
```

```
A
B B
C C C
D D D D
E E E E E
```

B.

```
for i in range(5):
    for j in range(i):
        print(end=" ")
    for j in range(5-i):
        print("*",end="")
    print()
```

```
*****
 ****
  ***
   **
    *
```

C.

```
for i in range(6):
    for j in range(5-i):
        print(end=" ")
    for j in range(i+1):
        print(j+1,end="")
    for j in range(i,0,-1):
        print(j,end="")
    print()
```

D.

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