1. Python output

In [1]:
```
# Python is a case sensitive language
print('Die World')
```

```
Die World
```

In [2]:
```
print('Data Science')
```

```
Data Science
```

In [5]:
```
print(7)
```

```
7
```

In [6]:
```
print(7.7)
```

```
7.7
```

In [7]:
```
print(True)
```

```
True
```

In [8]:
```
print("hello",1,4.5,True)
```

```
hello 1 4.5 True
```

In [9]:
```
print("Hello",1,4.5,False,sep='-')
```

```
Hello-1-4.5-False
```

In [10]:
```
print('hello')
print('world')
```

```
hello
world
```

In [11]:
```
print('Dead',end='-')
print('World')
```

```
Dead-World
```

2. Data Types

In [12]:
```
# Integer
print(8)
#1*10^308
print(1e310)
```

```
8
inf
```

In [13]:
```
# Decimal / Float
print(8.55)
print(1.7e309)
```

```
8.55
inf
```

In [14]:
```
# Boolean
print(True)
print(False)
```

```
True
False
```

In [15]:
```
# Text / String
print('Dead World')
```

```
Dead World
```

In [16]:
```
# Complex
print(5+6j)
```

```
(5+6j)
```

In [17]:
```
# List -> C -> Array
print([1,2,3,4,5])
```

```
[1, 2, 3, 4, 5]
```

In [18]:
```
# Tuple
print((1,2,3,4,5))
```

```
(1, 2, 3, 4, 5)
```

In [19]:
```
# Sets
print({1,2,3,4,5})
```

```
{1, 2, 3, 4, 5}
```

In [20]:
```
# Dictionary
print({"Fucking":"Shit","Dip":"Shit","Piece":['of','shit']})
```

```
{'Fucking': 'Shit', 'Dip': 'Shit', 'Piece': ['of', 'shit']}
```

In [21]:
```
# Type
type({"Oh":"Shit"})
```

Out[21]:
```
dict
```

3. Variables

In [22]:
```
# Static vs Dynamic typing
# Static vs Dynamic Binding
# Stylish declaration techniques
```

In [23]:
```
# C/C++
name = "Saad"
print(name)

a = 5
b = 6

print(a+b)
```

```
Saad
11
```

In [22]:
```
# Dynamic Typing
a = 5
# Static Typing
#int a = 5
```

In [23]:
```
# Dynamic Binding
a = 5
print(a)
a = "Saad"
print(a)

# Static Binding
#int a = 5
```

```
5
Saad
```

In [24]:
```
a = 1
b = 2
c = 4
print(a,b,c)
```

```
1 2 4
```

In [25]:
```
a,b,c = 1,2,4
print(a,b,c)
```

```
1 2 4
```

In [28]:
```
a=b=c=5
print(a,b,c)
```

```
5 5 5
```

Commments

In [29]:
```
# This a comment
# HOW YOU DOIN'
a = 4
b = 5 #Study Dude
# This also a comment
print(a+b)
```

```
9
```

4. Keywords and Identifiers

In [30]:
```
# Keywords
import keyword

keyword_list = keyword.kwlist
print(keyword_list)
```

```
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'brea
k', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finall
y', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlo
cal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yi
eld']
```

In [31]:
```
# Identifiers
# You can't start with a digit
name1 = "SAAD"
print(name1)
# YOU can use special character -> _
_     = 'saad'
print(_)
# identifiers can not be keywords
```

```
SAAD
saad
```

Temp Heading

5. User Input

In [32]:
```
# Static vs Dynamic
input(" Enter whatever you want to enter here ps. email is preferred :
```

```
 Enter whatever you want to enter here ps. email is preferred :)
```

Out[32]: ''

In [4]:
```
# Take input from users and store them in a variable
first = int(input("give"))
second= int(input("me"))
third = int(input("your"))
fourth= int(input("phone"))
fifth = int(input("number"))
sixth = int(input("and"))
seventh=int(input("have"))
eight = int(input("an ANGEL"))
ninth  = int(input("as your"))
tenth = int(input("contact;)"))
#print(type())
# add the 2 variables
Ten_number_ticket_to_an_angel=first,secd+third+fourth+fifth+sixth+seve
# print the result
print(Ten_number_ticket_to_an_angel)
```

```
give9
me3
your4
phone5
number3
and5
have5
an ANGEL5
as your5
contact;)6
50
```

6. Type Conversion

Implicit vs Explicit

print(5+5.6) print(type(5),type(5.6))

print(4+"4")

In [37]:
```
# Explicit
# str -> int
# int(4+5j)

# int to str
print(str(1))

# float
float(453)
```

```
1
```

Out[37]:
```
453.0
```

7. Literals

In [7]:
```
binary_literal = 0b1010
decimal_literal = 42
octal_literal = 0o52
hexadecimal_literal = 0x2A

print(binary_literal)        # Output: 10
print(decimal_literal)       # Output: 42
print(octal_literal)         # Output: 42
print(hexadecimal_literal)   # Output: 42
```

```
10
42
42
42
```

In [13]:
```
# FLoat Literal
float_1 = 10.5
float_2 = 1.5e4 # 1.5*10^4
float_3 = 1.5e-2# 1.5*10^-2

print(float_1,
      float_2,
      float_3)
```

```
10.5 15000.0 0.015
```

In [15]:
```
# complex Literal
x = 3.14j
print(x,x.imag,x.real)
```

```
3.14j 3.14 0.0
```

In [16]:
```
# Binary
x = 3.14j
print(x.imag)
```

```
3.14
```

In [32]:
```
relative_string = "Beta free rehte ho studies par dhyaan do or success
me_string       = "Seh lo ge aap"
character       =   "U"
multiline_string= """Playing tricks is evilness. Understanding tricks
unicode         = (["Hello, 你好,வணக்கம்"],["\U0001f600\U0001F606\U0
raw_string      = r"raw \n string" #while raw strings treat backslashe
                                   #escape sequences for quotes (' an
print(relative_string)
print(me_string)
print(character)
print(multiline_string)
print(unicode)
print(raw_string)
```

```
Beta free rehte ho studies par dhyaan do or successful insan bano
Seh lo ge aap
U
Playing tricks is evilness. Understanding tricks is cleverness but not do
ing both is innocence.
(['Hello, 你好,வணக்கம்'], ['😀😆🤣'])
raw \n string
```

In [26]:
```
a = True + 4 #True = 1
b = False+10 #False=0

print("a:",a)
print("b:",b)
```

```
a: 5
b: 10
```

In [27]:
```
p = None
a = 4
b=  5
print('Program exe')
```

```
Program exe
```

8. Operators

In [28]:
```
# arithmetic
# relational
# logical
# bitwise
# assignment
# membership
```

In [2]:
```
# Arithmetic Operators
print(5 + 6)   # Addition: 5 + 6 = 11
print(5 - 6)   # Subtraction: 5 - 6 = -1
print(5 * 6)   # Multiplication: 5 * 6 = 30
print(5 / 2)   # Division: 5 / 2 = 2.5
print(5 // 2)  # Floor Division: 5 // 2 = 2 (integer division, discard
print(5 % 2)   # Modulus: 5 % 2 = 1 (remainder of the division)
print(5 ** 2)  # Exponentiation: 5 ** 2 = 25 (5 raised to the power of

```

```
11
-1
30
2.5
2
1
25
```

In [4]:
```
# Relational Operators
print(4 > 5)   # False (4 is not greater than 5)
print(4 < 5)   # True (4 is less than 5)
print(4 >= 4)  # True (4 is greater than or equal to 4)
print(4 <= 4)  # True (4 is less than or equal to 4)
print(4 == 4)  # True (4 is equal to 4)
print(4 != 4)  # False (4 is not not equal to 4, which means it is equ
```

```
False
True
True
True
True
False
```

In [6]:
```
# Logical operators
print(1 and 0)
print(1 or 0)
print(not 0)
print(not 1)
```

```
0
1
True
```

In [8]:
```
# Bitwise Operators :

# Bitwise and
print(2 & 3)

# Bitwise or
print(2|3)

#Bitwise xor
# Bitwise XOR
print(2 ^ 3)   # Output: 1 (Binary: 0010 XOR 0011 = 0001)

# Bitwise NOT
print(~3)      # Output: -4 (Binary: NOT(0011) = 1100, the result is a

# Bitwise Right Shift
print(4 >> 2)  # Output: 1 (Binary: 0100 >> 2 = 0001)

# Bitwise Left Shift
print(5 << 2)  # Output: 20 (Binary: 0101 << 2 = 10100)
```

```
2
3
1
-4
1
20
```

In [9]:
```
# Assignment Operators

a  = 2 # Assigns the value 2 to variable a

a %= 2 # Modifies a to be equal to its value modulo 2 (which  is 0)

# increament a by 1 using +=
a += 1 # Equivalent to a = a+1

print(a)  # Output will be 1
```

```
1
```

In [13]:
```
# Membership Operators
# in/not in

print('peace' not in "Peace is not in my Life.")

print('dark' in "World is dark, atleast for me.")

print(1 in [3,4,2,4,2,4,5,6,7])
```

```
True
True
False
```

In [21]:
```
# Program - Find the sum of the last 3 digit number entered by the use

# Prompting the user to enter a 3 digit number
number = int(input('Enter any number it will ony calculate the additio

# Extracting individual digits using modulo and integer division
a = number % 10
number = number // 10

b = number % 10
number = number // 10

c = number % 10

# Computing and printing the sum of the digits
print("Sum of the digits:", a + b + c)

```

```
Enter any number it will ony calculate the addition of last 3 digit : 334
4654376
Sum of the digits: 16
```

if - else in python

In [6]:
```
 #                    [  Ultimate Jaadu   ]
#Program - Find the sum of the 3 digit number entered by the user

def add_three_digits():
    num1 = int(input("pehle ek no. daal:"))
    num2 = int(input("ab dusra daalde:"))
    num3 = int(input("teesra bhi daal hi de:"))
    # Check if all numbers are three digits
    if 0<=num1<=9 and 0<=num2<=9 and 0<=num3<=9:
        jaadu = num1+num2+num3
        return jaadu
    else:
        return "never-Try Again"

# Example usage:
print("Teen number daal aur jaadu dekh:")
print("aaila jaadu:", add_three_digits())
```

```
Teen number daal aur jaadu dekh:
pehle ek no. daal:3
ab dusra daalde:4
teesra bhi daal hi de:5
aaila jaadu: 12
```

In [4]:
```
#                                                    [Gaand faadu code]
# login program and indentation
# email --> datadudesaadkhan@gmail.com
# password --> Baigan

email = input("apna email bataiye :")
password = input("kuch huwa tow hamari responsibility nahi hai:")

if email == "datadudesaadkhan@gmail.com" and password =="Baigan":
    print("Welcome user")
elif email == "datadudesaadkhan@gmail.com" and password !="Baigan":
    # tell the user
    print("Dimak ke andhe password galat dala hai tunne")
    password=input("Abki baar dhang se daal")
    if password =="Baigan":
        print("Subha ka bhula , shaam ko wapas agaya")
    else:
        print("Tum se na ho paega")
else:
    print("Gian he aap ==> [p.s. Doremon waala]")
```

```
apna email bataiye :datadudesaadkhan@gmail.com
kuch huwa tow hamari responsibility nahi hai:TheSearch
Dimak ke andhe password galat dala hai tunne
Abki baar dhang se daalTheMockingbird
Tum se na ho paega
```

In [ ]:
```
# menu driven calculater
menu = input("""
Aur bhai calculate karne ke liye paisa bhi tow hona chaiye :(
1. Pin change karega tow 1 daba
2.Paisa kitna hai pata karne ka hai tow 2 daba
3.Paisa nikal na hai tow 3 daba ps. agar hoga tabhi niklega
4.Wapas jaana hai tow 4 daba
""")

if menu =='1':
    print('pin change')
elif menu=='2':
    print('le balance dekh')
else:
    print('Wapas bhejo isko')
```

Modules in Python

math

keywords

random

datetime

In [57]:
```
# math
import math
math.sqrt(225)
```

Out[57]:
```
15.0
```

In [58]:
```
# Keyword
import keyword
print(keyword.kwlist)
```

```
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'brea
k', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finall
y', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlo
cal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yi
eld']
```

In [69]:
```
# random
import random
# Generate a random integer between a specified range (inclusive)
random_number = random.randint(1, 100)
print(random_number)
# Generate a random floating-point number between 0 and 1
random_float = random.random()
print(random_float)
# Generate a random floating-point number between a specified range
random_float_range = random.uniform(10.5,31.5)
print(random_float_range)
```

```
17
0.5383122957066993
10.627510561420028
```

In [72]:
```
# Datetime
import datetime
# Get the current date and time
current_datetime = datetime.datetime.now()
print(current_datetime)
# Format a datetime object
formatted_datetime = current_datetime.strftime("%Y-%m-%d %H:%M:%S")
print(formatted_datetime)
# Parse a string into a datetime object
parsed_datetime = datetime.datetime.strptime("2024-03-07 12:30:45", "%
print(parsed_datetime)
```

```
2024-03-07 16:17:19.617753
2024-03-07 16:17:19
2024-03-07 12:30:45
```

In [21]:
```
1 help('modules')
```

```
Please wait a moment while I gather a list of all available modules...


WARNING: AstropyDeprecationWarning: The private astropy._erfa module has
been made into its own package, pyerfa, which is a dependency of astropy
and can be imported directly using "import erfa" [astropy._erfa]
C:\Users\mrsaa\anaconda3\Lib\site-packages\paramiko\transport.py:219: Cry
ptographyDeprecationWarning: Blowfish has been deprecated
  "class": algorithms.Blowfish,

KeyboardInterrupt
```

Loops in Python

Need for Loops

While Loop

For Loop

In [74]:
```
1 # While loop example -> program to print the table
2 # Program -> Sum of all digits of a given number
3 # Program -> keep accepting numbers from users till he/she enters a 0
```

In [78]:
```
1 number = int(input('enter the number'))
2
3 i = 1
4
5 while i<11:
6   print(number,'*',i,'=',number * i)
7   i += 1
```

```
enter the number3
3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15
3 * 6 = 18
3 * 7 = 21
3 * 8 = 24
3 * 9 = 27
3 * 10 = 30
```

In [4]:

```
# while loop with else
x = 1
while x<12:
 print(x)
 x += 1

else:
 print('Bohot ho gaya')
```

```
1
2
3
4
5
6
7
8
9
10
11
Bohot ho gaya
```

In [5]:
```python
# kon bane ga crorepati
 # generate a random integer between 1 to 100
import random
jaadu = random.randint(1,100)
guess = int(input("bolo bacha"))
counter = 1
while guess != jaadu:
    if guess < jaadu:
      print('galat jawab! high ho jao')
    else:
      print('galat jawab! low jao')

    guess = int(input('bolo bacha'))
    counter+=1
else:
    print('sahi jawab')
    print('your_chanches',counter)
```

```
bolo bacha12
galat jawab! high ho jao
bolo bacha45
galat jawab! low jao
bolo bacha35
galat jawab! low jao
bolo bacha16
galat jawab! high ho jao
bolo bacha25
galat jawab! high ho jao
bolo bacha29
galat jawab! low jao
bolo bacha34
galat jawab! low jao
bolo bacha32
galat jawab! low jao
bolo bacha30
galat jawab! low jao
bolo bacha27
sahi jawab
your_chanches 10
```

In [7]:
```python
# For loop demo

for i in {1,2,3,4,5}:
    print(i)
```

```
1
2
3
4
5
```

In [8]:
```python
# For loop examples
```

Program - The current population of a town is 10000. The population of the town is increasing at the rate of 10% per year. You have to write a program to find out the population at the end of each of the last 10 years.

In [9]:
```
curr_pop = 10000

for i in range(10,0,-1):
    print(i,curr_pop)
    curr_pop = curr_pop-0.1*curr_pop
```

```
10 10000
9 9000.0
8 8100.0
7 7290.0
6 6561.0
5 5904.9
4 5314.41
3 4782.969
2 4304.6721
1 3874.20489
```

Sequence sum

In [12]:
```
import math
```

In [13]:
```
math.factorial(5)
```

Out[13]:
```
120
```

In [19]:
```
sequence = [(1/math.factorial(1))+(2/math.factorial(2))+(3/math.factor
total = sum(sequence)
print("Sum of the sequence:", total)
```

```
Sum of the sequence: 2.5
```

In [15]:
```
for i in range(5):
    print(math.factorial(i))
```

```
1
1
2
6
24
```

In [20]:
```
sequence = [1, 2, 3, 4, 5]
total = sum(sequence)
print("Sum of the sequence:", total)
```

```
Sum of the sequence: 15
```

In [4]:
```
n = int(input('enter n'))

result = 0
fact = 1

for i in range (1,n+1):
    fact = fact*i
    result = result+i/fact

print(result)
```

```
enter n6
2.7166666666666663
```

Nested Loops

In [5]:
```
# Examples --> unique pairs

for i in range(1,5):
    for j in range(1,5):
        print(i,j)
```

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

Pattern 1

-

**

In [13]:
```
# Code here
rows = int(input('enter the number of rows'))

for i in range(1,rows+1):
    for j in range(1,i+1):
        print('*',end='')
    print()
```

```
enter the number of rows4
*
**
***
****
```

pattern 2

1

121

12321

1234321

In [17]:
```
# Code here
rows = int(input('enter number of rows'))

for i in range(1,rows+1):
    for j in range(1,i+1):
        print(j,end="")
    for k in range(i-1,0,-1):
        print(k,end="")

    print()

```

```
enter number of rows4
1
121
12321
1234321
```

Loop Control Statement

Break

Continue

Pass

In [18]:
```
for i in range(1,10):
    if i == 5:
        break
    print(i)
```

```
1
2
3
4
```

In [21]:
```
lower = int(input('Enter lower range: '))
upper = int(input('Enter upper range: '))

for i in range(lower, upper + 1):
    if i > 1:
        for j in range(2, int(i ** 0.5) + 1):
            if i % j == 0:
                break
        else:
            print(i)

```

```
Enter lower range: 2
Enter upper range: 100
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
53
59
61
67
71
73
79
83
89
97
```

In [26]:
```
# Continue
for i in range (1,10):
    if i==5:
        continue
    print(i)
#when i is equal to 5, the print(i) statement is skipped due to the co
```

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

In [31]:
```
for i in range (1,10):
    pass
print(i)
#This behavior is because i retains its value after the loop ends due
```

```
9
```

```
Strings are sequence of characters

In python specifically , strings are a sequence of unicode
characters
Creating Strings
Accessing Strings
Adding Chars to String
Editing Strings
Deleting Strings
Operations on Strings
String Functions

```

Creating Strings

In [33]:
```
s = "hello"
s = 'hello'
#Multiline strings
s = '''hello'''
s = """hello"""
s = str('hello')
print(s)
```

```
hello
```

In [38]:
```
'Skin to Skin is not enough , I want to take a nap inside your ribcage
```

Out[38]: 'Skin to Skin is not enough , I want to take a nap inside your ribcage.'

Accessing Substrings from a String

In [39]:
```
# Positive Indexing
s = 'Dead World'
print(s[2])
```

```
a
```

In [40]:
```
# Negative Indexing
s = 'Dead World'
print(s[-3])
```

```
r
```

In [41]:
```
# Slicing
s = 'Dead World'
print(s[6:0:-2])
```

```
o a
```

In [44]:
```
s= 'devil neve reven lived'
print(s[::-1])
```

```
devil never even lived
```

In [46]:
```
s = 'World is Vivid'
print(s[-1:-6:-1])
```

```
diviV
```

Editing and Deleting in Strings

In [96]:
```
s = ['hello world',32,43] #Strings are immutable
del s[0]
print(s)
```

```
[32, 43]
```

In [99]:
```
s = ['Dead World','Suuiii','timro pratiksha',0,32,4,54,4]
del s[-1:-5:2]
print(s)
```

```
['Dead World', 'Suuiii', 'timro pratiksha', 0, 32, 4, 54, 4]
```

Operations on strings

Arithmetic Operations

Relational Operations

Logical Operations

Loops on Strings

Membership Operations

In [102]:
```
1 print('Nagpur'+' '+"Bhusawal")
```

```
Nagpur Bhusawal
```

In [106]:
```
1 print('Gian hai aap - '*5)
```

```
Gian hai aap - Gian hai aap - Gian hai aap - Gian hai aap - Gian hai aap
-
```

In [107]:
```
1 print("*"*50)
```

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

In [112]:
```
1 "Pandas" != "Pandas" and "Pandas" == "Pandas"
```

Out[112]:
```
False
```

In [114]:
```
1 "pune">"PUNE"
2 #When comparing strings, lowercase letters are considered to be greate
```

Out[114]:
```
True
```

In [116]:
```
1 "mumbai" > "pune" #lexicographic (dictionary) order. i.e m comes befor
```

Out[116]:
```
False
```

In [117]:
```
1 'Dead' and "World" # It converts the string into bytes and the highest
```

Out[117]:
```
'World'
```

In [120]:
```
1 'Shitty' or 'world' # It converts the string into bytes and the lowest
```

Out[120]:
```
'Shitty'
```

In [122]:
```
1 '' and "world"
```

Out[122]:
```
''
```

In [124]:
```
1 '' or 'world'
```

Out[124]:
```
'world'
```

In [125]:
```
1 not 'World'
```

Out[125]:
```
False
```

In [127]:
```
for i in'Dead':
    print(i)
```

```
D
e
a
d
```

In [128]:
```
for i in 'A Being':
    print("Survivor")
```

```
Survivor
Survivor
Survivor
Survivor
Survivor
Survivor
Survivor
```

In [129]:
```
"D" in 'delhi'
```

Out[129]:
```
False
```

Common Functions

len

max

min

sorted

In [131]:
```
len('Dead World') # length of the object
```

Out[131]:
```
10
```

In [132]:
```
max("Dead World") # alphabetically last alphabet
```

Out[132]:
```
'r'
```

In [135]:
```
min("deadworld") # alphabetically first alphabet
```

Out[135]:
```
'a'
```

In [136]:
```
sorted('dead world',reverse=True)
```

Out[136]:
```
['w', 'r', 'o', 'l', 'e', 'd', 'd', 'd', 'a', ' ']
```

Capitalize/Title/Upper/Lower/Swapcase

In [140]:
```
s= "may flowers grow in the saddest parts of you"
print(s.capitalize())
print(s)
```

```
May flowers grow in the saddest parts of you
may flowers grow in the saddest parts of you
```

In [141]:
```
s.title()
```

Out[141]: 'May Flowers Grow In The Saddest Parts Of You'

In [142]:
```
s.upper()
```

Out[142]: 'MAY FLOWERS GROW IN THE SADDEST PARTS OF YOU'

In [143]:
```
'MAY FLOWERS GROW IN THE SADDEST PARTS OF YOU'.lower()

```

Out[143]: 'may flowers grow in the saddest parts of you'

In [144]:
```
'May Flowers Grow In The Saddest Parts Of You'.swapcase()

```

Out[144]: 'mAY fLOWERS gROW iN tHE sADDEST pARTS oF yOU'

Count/Find/Index

In [148]:
```
a = "I am datadude saad khan."
```

In [149]:
```
a.count('a')  # How many times a is used in sentence.
```

Out[149]: 6

In [151]:
```
a.find('x')  # If the substring 'x' is not found in 'a' variable the -
```

Out[151]: -1

In [153]:
```
a.find('m') # Index postion of 'm' in variable.
```

Out[153]: 3

In [155]:
```
a.index('k') #Index postion of 'k' in variable but throws error if sub
```

Out[155]: 19

endswith / startswith

In [157]:
```
1 e = 'Baby I dont need no dollar bills to have fun tonight , I love che
```

In [159]:
```
1 e.endswith('rills.')
```

Out[159]: True

In [161]:
```
1 e.startswith('Bab')
```

Out[161]: True

Format

In [163]:
```
1 feeling  = 'solitude'
2 heart    = 'shattered'
3 "A sprout of darkness is {0} and it {1} me from inside.".format(feelin
```

Out[163]: 'A sprout of darkness is solitude and it shattered me from inside.'

isalnum/ isalpha/ isdigit/ isidentifier

In [164]:
```
1 'Self-Destroyer69'.isalnum()
```

Out[164]: False

In [166]:
```
1 "FredrickNeitschze".isalpha()
```

Out[166]: True

In [167]:
```
1 '77afn'.isdigit()
```

Out[167]: False

In [168]:
```
1 name1 = "variable_name"
2 print(name1.isidentifier())  # Output will be True
```

True

Split/Join

In [170]:
```
1 'hi my name is Saad'.split()
```

Out[170]: ['hi', 'my', 'name', 'is', 'Saad']

In [171]:
```
1 " ".join([" ".join(['hi', 'my', 'name', 'is', 'Saad'])])
```

Out[171]: 'hi my name is Saad'

Replace

In [173]:
```
'hi', 'my', 'name', 'is', 'Saad'.replace('Saad','datadude')
```

Out[173]:
```
('hi', 'my', 'name', 'is', 'datadude')
```

Strip

In [174]:
```
'Besabriyaan           '.strip()
```

Out[174]:
```
'Besabriyaan'
```

Example Programs

In [175]:
```
## Find the length of a given string without using the len() function

s = input('enter the string')

counter = 0

for i in s:
    counter += 1

print('length of the string',counter)

```

```
enter the stringSaad is a bitch
length of the string 15
```

In [176]:
```
 # Extract username from a given email.
# Eg if the email is nitish24singh@gmail.com
# then the username should be nitish24singh

s = input('Apna email dalia.')

pos = s.index('@')
print(s[0:pos])


```

```
enter the emailPapakiPari@gmail.com
PapakiPari
```

In [182]:
```
# Count the frequency of a particular character in a provided string.
# Eg 'hello how are you' is the string, the frequency of h in this str

s = input('apna email dalia.')
term = input('kiski frequency janna chahenge aap.')

counter = 0
for i in s:
  if i == term:
    counter += 1

print('frequency',counter)

```

```
apna email dalia.PhoenixRisingFromTheAshes@comeback.com
kiski frequency janna chahenge aap.e
frequency 4
```

In [1]:
```
# Write a program which can remove a particular character from a strin
s = input('enter the string')
term = input('what would you like to remove')

result = ''

for i in s:
  if i != term:
    result = result + i

print(result)
```

```
enter the stringLucid Dreamer
what would you like to removed
Luci Dreamer
```

In [188]:
```
# Write a program that can check whether a given string is palindrome
# abba
# malayalam

s = input('enter the string')
flag = True
for i in range(0,len(s)//2):
  if s[i] != s[len(s) - i -1]:
    flag = False
    print('Haule, galat daala hai tuh.')
    break

if flag:
  print('Palindrome')


```

```
enter the stringqueen
Haule, galat daala hai tuh.
```

In [192]:
```
# Write a program to count the number of words in a string without spl

s = input('enter the string')
L = []
temp = ''
for i in s:

  if i != ' ':
    temp = temp + i
  else:
    L.append(temp)
    temp = ''

L.append(temp)
print(L)

```

```
enter the stringThey pierced a pen through my heart and I became a poet.
['They', 'pierced', 'a', 'pen', 'through', 'my', 'heart', 'and', 'I', 'be
came', 'a', 'poet.']
```

In [190]:
```
# Write a python program to convert a string to title case without usi
s = input('enter the string')

L = []
for i in s.split():
  L.append(i[0].upper() + i[1:].lower())

print(" ".join(L))
```

```
enter the stringBe careful about rushing God's timing. You never know who
or what HE is protecting you from.
Be Careful About Rushing God's Timing. You Never Know Who Or What He Is P
rotecting You From.
```

In [191]:
```
# Write a program that can convert an integer to string.

number = int(input('enter the number'))

digits = '0123456789'
result = ''
while number != 0:
  result = digits[number % 10] + result
  number = number//10

print(result)
print(type(result))
```

```
enter the number22345
22345
<class 'str'>
```