```python
#1. Create a user defined function to print count for given range.

def count_range(num):
  for i in range(num+1):
    print(i)
  print("Count is:",i)

num = int(input("Enter a number: "))
count_range(num)
```

```
Enter a number: 4
0
1
2
3
4
Count is: 4
```

```python
#2. Write a function that takes an integer n and returns a random integer with exactly n digits.
#Maximum range is 5
#For instance, if n is 3, then 125 and 593 would be valid return values, but 093 would not
#because that is really 93, which is a two-digit number.

from random import *

def random_number(num):
  start = 10 ** (num-1)
  end = (10 ** num) - 1
  return randint(start, end)

num = int(input("Enter a number: "))
random = random_number(num)
print("The random number of", num,"digit is: ",random)
```

```
Enter a number: 4
The random number of 4 digit is:  5375
```

```python
#3. Write a function that converts temperatures from Celsius to Fahrenheit

def convert_CtoF(C):
  F = (C * 1.8) + 32
  return F

C = int(input("Enter a number: "))
CF = convert_CtoF(C)
print("Temperature from Celsius to Fahrenheit is:",CF,"°")
```

```
Enter a number: 40
Temperature from Celsius to Fahrenheit is: 104.0 °
```

```python
#4. Create a user defined function to find the sum of digits of a number.

def DigitofSum(num):
  sum = 0
  while num != 0:
    sum = sum + (num % 10)
    num = num // 10
  return sum

num = int(input("Enter a number: "))
DigitofSum(num)
```

```
Enter a number: 425
11
```

```python
#5. Create a user defined function to find the sum of digits of a number.
#If the sum is more than 9 then call the same function again and again till the sum is between 0 to 9. (i.e Find super digit, use recursion)

def DigitofSum(num):
  sum = 0
  while num != 0:
    sum = sum + (num % 10)
    num = num // 10

  if sum > 9:
    DigitofSum(sum)
  else:
    print(sum)

num = int(input("Enter a number: "))
DigitofSum(num)
```

```
Enter a number: 45
9
```

```python
#6. By using a recursive function find the factorial of a given number.

def factorial(num):
  if num == 1:
    return num
  else:
    return num*factorial(num-1)

num = int(input("Enter a number: "))

if num < 0:
  print("We do not take negative value ")
elif num == 0:
  print("The factorial of 0 is  1")
else:
  print("The factorial of",num,"is",factorial(num))
```

```
Enter a number: 4
The factorial of 4 is 24
```

```python
#7. Write a function called rectangle that takes two integers m and n as arguments and prints out an mx n box consisting of asterisks.
#Shown below is the output of rectangle(2,4)
#****
#****

def star_pyr(m,n):
  for i in range(m):
    print('*'*n)

star_pyr(2,4)
```

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

```python
#8. Write a function that takes an integer n and returns a random integer with exactly n digits.
#For instance, if n is 3, then 125 and 593 would be valid return values, but 093 would not because that is really 93, which is a two-digit number.

from random import *

def random_number(num):
  start =  10 ** (num-1)
  end = (10 ** num ) - 1
  return randint(start,end)

num = int(input("Enter a number: "))
random = random_number(num)
print("The random number of maximum digit is:",random)
```

```
Enter a number: 4
The random number of maximum digit is: 3183
```

```python
#9. Write a function called root that is given a number x and an integer n and returns x power 1/n.
#In the function definition, set the default value of n to 2.

def power(x,n = 2):
  return x ** (1/n)

power(3,3)
```

```
1.4422495703074083
```

```python
#10. Create a user defined function which takes two user defined parameters, add them and return the sum of it.

def sum_num(n1,n2):
  return n1 + n2

n1 = int(input("Enter a number: "))
n2 = int(input("Enter a number: "))
sum_num(n1,n2)
```

```
Enter a number: 4
Enter a number: 25
29
```

```python
#11
'''
Write a program that generates a list of 20 random numbers between 1 an
d 100.
(a) Print the list.
(b) Print the average of the elements in the list.
(c) Print the largest and smallest values in the list.
(d) Print the second largest and second smallest entries in the list
(e) Print how many even numbers are in the list
'''
#1
print ("a - Print the list.")
list = ["half life", "diablo", "witcher 3"]
for i in list:
  print(i)
print("\n")

#2
print("b - Print the average of the elements in the list.")
num1 = [6,5,3,5,2,3,5]
average = sum(num1) / len(num1)

print("The average of the number is",average)
print("\n")
```

```
#3
print("c - Print largest and smallest value of the list.")
num2 = [6,5,3,5,2,3,5]
print("The largest number of list is",max(num2),"The smallest number fr
om the list is",min(num2))
print("\n")

#4
print("d - Print the second largest and second smallest entries in the
list.")
num3 = [66,55,44,22,33,77,88]
length = len(num3)
num3.sort()
print("The second largest of list is",num3[length - 1])
print("The second smallest number on the list is",num3[1])
print("\n")

#5
print("e - How many even number are even in list.")
num4 = [2,4,36,5,7,56,32]
for  i in num4:
  if i  % 2 == 0:
    print(i)
```

```
a - Print the list.
half life
diablo
witcher 3


b - Print the average of the elements in the list.
The average of the number is 4.142857142857143


c - Print largest and smallest value of the list.
The largest number of list is 6 The smallest number from the list is 2


d - Print the second largest and second smallest entries in the list.
The second largest of list is 88
The second smallest number on the list is 33


e - How many even number are even in list.
2
4
36
56
32
```

```
#12. Write a program that generates a list L of 50 random numbers between 1 and 100.

from random import *

list = []
for i in range(1,100):
  list = randint(0,50)
print("The radom number that generates the list is",list)
```

```
The radom number that generates the list is 32
```

```
#13
'''
Write a program to perform following tasks:
- Read a list of integers from a user.
- Remove duplicate values from this list. We call it a super list.
- In a single scan of the super list, perform following operations:
- If the number is not perfectly divisible by 3, discard it.
- If the number is greater than 30, add 5 to it. Else, subtract 5 from
it.
- Display contents of the list.
'''

print("1.")
lst = []
num = int(input("Enter number of elements: "))
for i in range(0, num):
    num_list = int(input())
    lst.append(num_list) # adding the element
print(lst)
print("\n")

print("2.")
list = [1, 2, 3, 1, 1, 6, 6, 5 ,6, 2]
print("List Before ",list)
super_list = []

for i in list:
    if i not in super_list:
        super_list.append(i)
my_list = super_list
print("List After removing duplicates",my_list)
print("\n")

print("3.")
list = [9,6,18,27,214]
new_list = []
```

```
for  i  in range(len(list)):
  if list[i] % 3 == 0:
    new_list = list[i]
    print(new_list)
print("\n")

print("4.")
list = [44,55,66,77,88]

new_list = []
for i in range(len(list)):
  if list[i]  > 30:
    new_list =  list[i] + 5
    print(new_list)
```

```
1.
Enter number of elements: 4
2
4
5
7
[2, 4, 5, 7]


2.
List Before  [1, 2, 3, 1, 1, 6, 6, 5, 6, 2]
List After removing duplicates [1, 2, 3, 6, 5]


3.
9
6
18
27


4.
49
60
71
82
93
```

```
#14. Create a list with different types of elements. Iterate over the list and print every element and type of element.

list = [1,2,3,4,5,"Hello" , "world "]
for i in list:
  print("The element of the number",i,"The type of the element",type(i))
```

```
#15
'''
Write a program that generates a list of 20 random numbers between 1 an
d 100.
 Print the list.
 Print the average of the elements in the list.
 Print the largest and smallest values in the list.
 Print the second largest and second smallest entries in the list
 Print how many even numbers are in the list.
'''
import random

#genrate random number
list = random.sample(range(1,100),7)
print("Random list of number is",list)
print("\n")

#average of the list
avg = sum(list) / len(list)
print("The average of the list is", avg)
print("\n")

#print the largest and smallest values in the list
print("The largest number in the list",max(list),"The smallest number i
n the list",min(list))
length = len(list)
list.sort()
print("The second largest number of the list",list[length - 1],"The sec
ond smallest number in the list",list[1])
for i in range(len(list)):
  if i % 2 == 0:
    print("The number of",list[i])
print("Count is")
print(len(list))
```

```
Random list of number is [7, 20, 27, 34, 1, 21, 83]


The average of the list is 27.571428571428573


The largest number in the list 83 The smallest number in the list 1
The second largest number of the list 83 The second smallest number in the list 7
The number of 1
The number of 20
The number of 27
The number of 83
Count is
7
```

```
#16
```

```
'''
Perform following functionalities:
Generate a list containing 100 similar words.
Replace each element in a list L with its square.
Count how many items in a list L are greater than 50.
Write a program that generates a list of 20 random numbers between 1 an
d 100.
Print the list.
Print the average of the elements in the list.
Print the largest and smallest values in the list.
Print the second largest and second smallest entries in the list
Print how many even numbers are in the list.
'''

print("A")
value = "Akash "
list = [5 * value]
print(list)
print("\n")

print("B")
L = [1,2,3,4,5]
for i  in  range(len(L)):
   L[i] = L[i] ** 2
print(L)
print("\n")

print("C")
list = [50,60,50,51,66,55]
count = 0
for i in range(len(list)):
  if list[i] > 50:
    count = count + 1
print("The number greater than 50 in ",list,"is",count)
print("\n")

print("D")
import random

randomlist = []
for i in range(0,21):
  n = random.randint(1,100)
  randomlist.append(n)
print(randomlist)
print("\n")

print("E")
list = [1,2,3,4,5,6]
print(list)
```

```
A
['Akash Akash Akash Akash Akash ']


B
[1, 4, 9, 16, 25]


C
The number greater than 50 in  [50, 60, 50, 51, 66, 55] is 4


D
[45, 25, 78, 81, 80, 92, 2, 18, 54, 70, 12, 77, 85, 9, 29, 43, 33, 65, 6, 15, 73]


E
[1, 2, 3, 4, 5, 6]
```

Strings

```
#1
#Creating String #simple string program

my_string = 'Hello'
print(my_string)

my_string = "Hello"
print(my_string)

my_string = '''Hello, my name is Akash'''
print(my_string)

#triple quoted string can extend multiple lines
my_string = """Hello, welcome to the world of Python"""
print(my_string)
```

```
Hello
Hello
Hello, my name is Akash
Hello, welcome to the world of Python
```

```
#2
#String Operations
str1 = 'Hello '
str2 =' World!'
# using +
print('str1 + str2 =',str1 + str2)
# using *
print('str1 * 3 =',3*str1)
str1 + str2
str1 * 3
```

```
str1 + str2 = Hello  World!
str1 * 3 = Hello Hello Hello
'Hello Hello Hello '
```

```python
#3
word1 = 'Wow'
word2 = 'Wow'
print('Equality:', word1 == word2, ' Alias:', word1 is word2)
word1 = [1,2,3]
word2 = [1,2,3]
print('Equality:', word1 == word2, ' Alias:', word1 is word2)
```

```
Equality: True  Alias: True
Equality: True  Alias: False
```

```python
#4
mystr='Python Programming by Akash Bhavsar'
mystr1="Python"
myst="abaababc29"
print(mystr.isalnum())
print(mystr.isalnum())
str1='2000'
print(str1.isdigit())
```

```
False
False
True
```

```python
#5
class Student:
  def __init__(self,f_name,l_name,course):
    self.FName=f_name
    self.LName=l_name
    self.course=course
  def display(self):
    print("Student",self.FName,self.LName,"is studying in the course",self.course)

s1=Student("Akash","Bhavsar","BSc CA & IT.")
s1.display()
s2=Student("Jiyan","Prajapati","DS")
s2.display()
```

```
Student Akash Bhavsar is studying in the course BSc CA & IT.
Student Jiyan Prajapati is studying in the course DS
```

Inheritance

```python
#1
#Simple Inheritance
class Robot:

    def __init__(self, name):# constructor, one mandatory parameter self
        self.name = name  #object variable

    def say_hi(self):  # class method,one mandatory parameter self
        print("Hi, I am " + self.name) # object variable referenced with self.variablename

class PhysicianRobot(Robot):
    pass  #Do not add or modify anything

x = Robot("Akash")
y = PhysicianRobot("Bhavsar")

print(x, type(x))
print(y, type(y))
x.say_hi()
y.say_hi()
```

```python
#2
#Overriding
class Robot:
    def __init__(self, name):
        self.name = name
    def say_hi(self):
        print("Hi, I am " + self.name)

class PhysicianRobot(Robot):
  #  def say_hi(self): #overrideent
   #     print("Say hi from child class")
    def say_hi_PR(self):  # Only child method
        print("Everything will be okay! ")
        print(self.name + " takes care of you!")

x=Robot("Akash")
x.say_hi()
y = PhysicianRobot("G1 robo")
y.say_hi()
y.say_hi_PR()
```

```
Hi, I am Akash
Hi, I am G1 robo
Everything will be okay!
G1 robo takes care of you!
```

```python
#3
#Multi-Level
class Animal:
    def speak(self):
        print("Animal Speaking")
#The child class Dog inherits the base class Animal
class Dog(Animal):  # two methods, speak & bark
    def bark(self):
        print("dog barking")
#The child class Dogchild inherits another child class Dog
class DogChild(Dog):  #will have 3 methods speak,bark, eat
    def eat(self):
        print("Eating bread...")
d = DogChild()
d.bark()
d.speak()
d.eat()
d1=Dog()
d1.speak()
d1.bark()
```

```
dog barking
Animal Speaking
Eating bread...
Animal Speaking
dog barking
```

```python
#4
#Multiple
class Calculation1:
    def Summation(self,a,b):
        return a+b;
    def printmsg(self):
      print("from class1 which does summation only")
class Calculation2:
    def Multiplication(self,a,b):
        return a*b;
    def printmsg(self):
      print("from class2 which does multiplication")
class Derived(Calculation2,Calculation1):  #multiplication,printmsg,summation, printmesg,divide
    def Divide(self,a,b):
        return a/b;
d = Derived()
print(d.Summation(10,20))
print(d.Multiplication(10,20))
print(d.Divide(10,20))
d.printmsg()
```

```
30
200
0.5
from class2 which does multiplication
```

```python
#5
#Multilevel & super()
class Calculation1:
    def Summation(self,a,b):
        return a+b;
    def printmsg(self):
      print("from class1 which does summation only, printmsg")
class Calculation2(Calculation1):  #summation, multiplication,printmsg
    def Multiplication(self,a,b):
        return a*b;
    def printmsg(self):
      print("from class2 which does multiplication only")
      super().printmsg()
class Derived(Calculation2):  #multiplication,printmsg calc2,summation, divide
    def Divide(self,a,b):
        return a/b;
d = Calculation2()
print(d.Summation(10,20))
print(d.Multiplication(10,20))
#print(d.Divide(10,20))
d.printmsg()
```

```
30
200
from class2 which does multiplication only
from class1 which does summation only, printmsg
```

Exception Handling

```python
#1
try:
  a = eval(input('Enter a number: '))
  print (3/a)
except NameError:
  print('Please enter a number.')
except ZeroDivisionError:
  print("Can't enter 0.")
```

```
Enter a number: 4
0.75
```

```
#2
try:
  file = open('Akash.txt', 'r')
except IOError:
  print('Could not open file')
else:
  s = file.read()
  print(s)
```

```
Could not open file
```

```
#3
f = open('akash.txt', 'w')
s = 'hi'
try:
  a = eval(input('Enter a number: '))
  print (a/0)
finally:
  f.close()
```

```
Enter a number: 4
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-11-a411e82e50c7> in <module>()
      4 try:
      5   a = eval(input('Enter a number: '))
----> 6   print (a/0)
      7 finally:
      8   f.close()

ZeroDivisionError: division by zero

 SEARCH STACK OVERFLOW
```

```
#4
try:
  a = eval(input('Enter a number: '))
  print (x/a)
except:
  print('A problem occurred.')
```

```
Enter a number: 4
A problem occurred.
```

```
#5
try:
    age = int(input("Enter the age:"))
    if(age<18):
        raise ValueError
    else:
        print("The age is valid")
except ValueError:
    print("The age is not valid")
```

```
Enter the age:23
The age is valid
```

File Handling

```
#1
lines = [line.strip() for line in open('/content/sample_data/Akash.txt')]
print(lines)
```

```
['i am akash', 'i study at indus university']
```

```
#2
"""Write a program that reads a list of temperatures from a file called temps.txt, con-
verts those temperatures to Fahrenheit, and writes the results to a file called ftemps.txt."""
file1 = open('/content/sample_data/ftemps.txt', 'w')
temperatures = [line.strip() for line in open('/content/sample_data/temps.txt')]
for t in temperatures:
  print((float(t)*9/5+32), file=file1)
file1.close()
```

temps.txt

```
34
35
36
```

temps.txt ftemps.txt

```
93.2
95.0
96.8
```

```
#3
#read first 5 characters
my_file = open("/content/sample_data/Akash.txt", "r")
print(my_file.read(10))
print(my_file.read())
print(my_file.readline(2))
print(my_file.readlines())
```

```
i am akash

i study at indus university

[]
```

```
#4
#Reading a specific line
line_number = 2
currentline = 1
my_file = open("/content/sample_data/Akash.txt", "r")
for line in my_file:
  if(currentline == line_number):
    print(line)
    break
  currentline = currentline + 1
```

```
i study at indus university
```

```
#5
my_file = open("/content/sample_data/Akash.txt", "r")
print(my_file.readlines())
print(my_file.tell())
my_file.seek(2)
print(my_file.tell())
my_file.seek(0)
print(my_file.tell())
my_file.seek(31)
print(my_file.tell())
print(my_file.read())
```

```
['i am akash\n', 'i study at indus university']
39
2
0
31
iversity
```