String Exercises:

```
#1. Write a Python program to count the number of characters in a string.
def number_count(string):
  return len(string)

str = 'mayank'
result = number_count(str)
print(result)
```

```
6
```

```
#2. Write a Python program to reverse a string.
def reverse_string(string):
  return string[::-1]

str='mayank'
result = reverse_string(str)
print(result)
```

```
knayam
```

```
#3. Write a Python program to check if a string is a palindrome.
def palindrome(string):
  return string == string[::-1]

str = 'madam'
result = palindrome(str)
print(result)
```

```
True
```

```
#4. Write a Python program to remove all the vowels from a string.
def remove_vowels(string):
    vowels = "aeiouAEIOU"
    result = ""
    for char in string:
        if char not in vowels:
            result += char
    return result

str = 'mayank'
result = remove_vowels(str)
print(result)
```

```
mynk
```

```
#5. Write a Python program to find the first non-repeating character in a string.
def non_repating(string):
  char_count={}
  for char in string:
    if char in char_count:
      char_count[char] += 1
    else:
      char_count[char] = 1

  for char in char_count:
    if char_count[char] == 1:
        return char

  return char_count

str = 'aaaaabbbbc'
result = non_repating(str)
print('First non-repeating character :',result)
```

```
First non-repeating character : c
```

```
#6. Write a Python program to capitalize the first letter of each word in a string.
def capitalize_first(string):
  new_str = string.title()

  return new_str

str = 'hi mayank,how are you'
```

```
result = capitalize_first(str)
print(result)
```

```
Hi Mayank,How Are You
```

```
#7.Check if a string is an anagram of another string:
def anagram(string1,string2):
  new_str = sorted(string1)==sorted(string2)
  return new_str

str1 = 'race'
str2 = 'care'
result = anagram(str1,str2)
print(result)
```

```
True
```

```
#8.Find the most frequent character in a string:
def frequent_char(string):
  char_count = {}

  for char in string:
    if char in char_count:
      char_count[char] += 1
    else:
      char_count[char] =1

  most_frequent = max(char_count,key=char_count.get)
  return most_frequent

str1 = 'aaaaabbbbbccc'
result = frequent_char(str1)
print(result)
```

```
a
```

```
#9.Check if a string is a valid email address (basic validation):
import re
def valid_email(email):
  pattern = r'^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$'
  if re.match(pattern,email):
    return True
  else:
    return False

email1 = 'mayankparshetye@gmail.com'
email2 = 'invaild.email'
email3 = '12343244gmail.com'

print(f'Is "{email1}" a valid email? {valid_email(email1)}')
print(f'Is "{email2}" a valid email? {valid_email(email2)}')
print(f'Is "{email3}" a valid email? {valid_email(email3)}')
```

```
Is "mayankparshetye@gmail.com" a valid email? True
Is "invaild.email" a valid email? False
Is "12343244gmail.com" a valid email? False
```

```
#10.Find the length of the longest substring without repeating characters:
def longest_substring_length(string):
    char_set = set()
    max_length = 0
    start = 0

    for end in range(len(string)):
        while string[end] in char_set:
            char_set.remove(string[start])
            start += 1
        char_set.add(string[end])
        max_length = max(max_length, end - start + 1)

    return max_length

str = "abcabcbb"
result = longest_substring_length(str)
print(result)
```

```
3
```

List Exercises:

```
#1. Write a Python program to find the sum of all elements in a list.
def sum_element(list1):
  return sum(list1)

list1 = [10,20,30,40]
result = sum_element(list1)
print(result)
```

```
100
```

```
#2. Write a Python program to find the maximum and minimum elements in a list.
def max_min(list1):
  maximum_element = max(list1)
  minimum_element = min(list1)

  return maximum_element,minimum_element

list1 = [10,20,30,40,50]
maximum_element,minimum_element = max_min(list1)
print('maximum element is :',maximum_element)
print('minimum elemenr is : ',minimum_element)
```

```
maximum element is : 50
minimum elemenr is :  10
```

```
#3. Write a Python program to remove duplicates from a list.
def remove_duplicates(list1):
  covert_set = set(list1)
  nlist = list(covert_set)
  new_list = sorted(nlist)

  return new_list

list1 = [10,20,30,10,40,10,50]
result = remove_duplicates(list1)
print(result)
```

```
[10, 20, 30, 40, 50]
```

```
#4. Write a Python program to check if a list is sorted in ascending order.
def sorted_list(lst):
  return lst == sorted(lst)

list1 = [10,20,30]
result = sorted_list(list1)
print(result)
```

```
True
```

```
#5. Write a Python program to find the second largest element in a list.
def second_largest(lst):
  sorted_list = sorted(lst)
  second_largest_element = sorted_list[-2]

  return second_largest_element,sorted_list

list1 = [20,80,30,90,50,110,440,220,400]
second_largest_element,sorted_list = second_largest(list1)
print('ascending order is : ',sorted_list)
print('second largest element is : ',second_largest_element)
```

```
ascending order is :  [20, 30, 50, 80, 90, 110, 220, 400, 440]
second largest element is :  400
```

```
#6. Write a Python program to sort a list of strings in alphabetical order.
def sorted_list(list1):
  return sorted(list1)

list1 = ["apple", "banana", "cherry", "date"]
result = sorted_list(list1)
print(result)
```

```
['apple', 'banana', 'cherry', 'date']
```

```
#7. Write a Python program to find the common elements between two lists.
def common_elements(list1,list2):
  common = set(list1).intersection(set(list2))
  new_list = list(common)
  return new_list

lst1 = [10,20,30,40]
lst2 = [30,20,60,40]
result = common_elements(lst1,lst2)
print(result)
```

```
[40, 20, 30]
```

```
#8. Write a Python program to remove the nth occurrence of an element from a list.
```

```
#9. Write a Python program to find the difference between two lists.
def difference_between(list1,list2):
  diff = list(set(list1).difference(set(list2)))
  return diff

list1 = [1, 2, 3, 4, 5]
list2 = [3, 4, 5, 6, 7]
result = difference_between(list1,list2)
print(result)
```

```
[1, 2]
```

```
#10. Write a Python program to remove the elements of a list that are divisible by 3.
def remove_list(lst):
  num_count = []

  for num in lst:
    if num%3 != 0:
      num_count.append(num)

  return num_count

list1 = [3,6,2,9,12]
result = remove_list(list1)
print(result)
```

```
[2]
```

Tuple Exercises:

```
#1. Write a Python program to find the length of a tuple.
def length_tuple(tup):
  new_tuple = len(tup)

  return new_tuple

tuple1 = (1,2,3,4)
result = length_tuple(tuple1)
print(result)
```

```
4
```

```
#2. Write a Python program to concatenate two tuples.
tuple1 = (1,2,3,4)
tuple2 = (5,6,7,8)
concatenate = tuple(set(tuple1).union(set(tuple2)))
print(concatenate)
```

```
(1, 2, 3, 4, 5, 6, 7, 8)
```

```
#3. Write a Python program to convert a tuple to a list.
def convert_tuple_to_list(tup):
  convert = list(tup)
  return convert

tuple1 = (1,2,3,4,5,6)
result = convert_tuple_to_list(tuple1)
print(result)
```

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

```
#4. Write a Python program to find the index of an element in a tuple.
tuple1 = (1,2,3,4,5,6)
element_find = 3
index = tuple1.index(element_find)
print("Index of", element_find, "in the tuple:", index)
```

```
Index of 3 in the tuple: 2
```

```
#5. Write a Python program to check if an element exists in a tuple.
my_tuple = (1,2,3,4,5)
element_to_check = 10
if element_to_check in my_tuple:
  print(element_to_check, "exists in the tuple.")
else:
    print(element_to_check, "does not exist in the tuple.")
```

```
10 does not exist in the tuple.
```

```
#6. Count the number of occurrences of an element in a tuple:
my_tuple = (1, 2, 2, 3, 4, 2, 5)
element_to_count = 2
count = my_tuple.count(element_to_count)
print("Number of occurrences of", element_to_count, "in the tuple:", count)
```

```
Number of occurrences of 2 in the tuple: 3
```

```
#7. Find the maximum and minimum elements in a tuple:
my_tuple = (10,20,40,60,110,0)
maximum_tuple = max(my_tuple)
minimum_tuple = min(my_tuple)
print("Maximum element in the tuple:", maximum_tuple)
print("Minimum element in the tuple:", minimum_tuple)
```

```
Maximum element in the tuple: 110
Minimum element in the tuple: 0
```

```
#8. Reverse a tuple:
my_tuple = (10,20,30,40,50,60,70,80,90)
reverse_tuple = my_tuple[::-1]
print(reverse_tuple)
```

```
(90, 80, 70, 60, 50, 40, 30, 20, 10)
```

```
#9. Check if all elements in a tuple are the same:
def are_all_elements_same(t):
    return all(x == t[0] for x in t)

my_tuple = (1, 1, 1, 1)
result = are_all_elements_same(my_tuple)
print("All elements are the same:", result)
```

```
All elements are the same: True
```

```
#10. Create a new tuple with elements from two existing tuples:
tuple1 = (1,2,3,4)
tuple2 = (5,6,7,8)
tuple3 = tuple1 + tuple2
print(tuple3)
```

```
(1, 2, 3, 4, 5, 6, 7, 8)
```

Set Exercises:

```
#1. Write a Python program to find the union of two sets.
def union_set(set1,set2):
  new_set = set1.union(set2)
  return new_set

set1 = {1,2,3,4}
set2 = {5,6,7,8,9}
result = union_set(set1,set2)
print("Union of set1 and set2:", result)
```

```
Union of set1 and set2: {1, 2, 3, 4, 5, 6, 7, 8, 9}
```

```
#2. Write a Python program to find the intersection of two sets.
def union_set(set1,set2):
  new_set = set1.intersection(set2)
  return new_set

set1 = {1,2,3,4}
set2 = {3,4,5,6}
result = union_set(set1,set2)
print(result)
```

```
{3, 4}
```

```
#3. Write a Python program to check if a set is a subset of another set.
set1 = {1,2,3,4}
set2 = {1,2,3,4,5}
new_set = set1.issubset(set2)
print(new_set)
```

```
True
```

```
#4. Write a Python program to remove duplicate elements from a set.
set1 = {1,2,2,2,3,4}
print(set1)
```

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

```
#5. Write a Python program to add an element to a set.
set1 = {1,2,3,4}
element = 5
new_set = set1.add(element)
print(set1)
```

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

```
#6. Write a Python program to remove an element from a set.
my_set = {1,2,3,4,54}
remove_element = 54
my_set.remove(remove_element)
print(my_set)
```

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

```
#7. Write a Python program to find the difference between two sets.
set1 = {1,2,3,4,5}
set2 = {2,3,4,6,8}
new_set = set1.difference(set2)
print(new_set)
```

```
{1, 5}
```

```
#8. Write a Python program to check if two sets are disjoint.
set1 = {1,2,3}
set2 = {4,5,6}
new_set = set1.isdisjoint(set2)
if new_set:
    print("The sets are disjoint")
else:
    print("The sets have common elements")
```

```
The sets are disjoint
```

```
#9. Write a Python program to find the symmetric difference between two sets.
set1 = {1,2,3,4}
set2 = {2,3,4,5}
new_set = set1.symmetric_difference(set2)
print(new_set)
```

```
{1, 5}
```

```
#10.Check if a set is empty:
empty_set = set()
if not empty_set:
    print("The set is empty")
else:
    print("The set is not empty")
```

```
The set is empty
```

Dictionary Exercises:

```
#1. Write a Python program to iterate over a dictionary and print its keys and values.
my_dict = {'a':1,'b':2,'c':3}
for key,value in my_dict.items():
  print(f'key:{key},value:{value}')
```

```
key:a,value:1
key:b,value:2
key:c,value:3
```

```
#2. Write a Python program to check if a key exists in a dictionary.
my_dict = {1:'a',2:'b',3:'c'}
key_check = 5
if key_check in my_dict:
  print(True)
else:
  print(False)
```

```
False
```

```
#3. Write a Python program to get the value associated with a key in a dictionary.
my_dict = {1:'a',2:'b',3:'c'}
value_check = 'c'
```

```
False
```

```
#4. Write a Python program to remove a key from a dictionary.
```

```
#5. Write a Python program to sort a dictionary by its values.
```

```
#6. Write a Python program to merge two dictionaries.
```

```
#7. Write a Python program to count the frequency of each element in a dictionary.
```

```
#8. Write a Python program to find the length of a dictionary.
```

```
#9. Write a Python program to check if a dictionary is empty.
```

```
#10. Write a Python program to find the keys with the maximum and minimum values in a dictionary.
```

Range Exercises:

```
#1. Write a Python program to iterate over a range of numbers and print them.
start = 1
end = 10

for num in range(start, end+1):
    print(num,end = ' ')
```

```
1 2 3 4 5 6 7 8 9 10
```

```
#2. Write a Python program to find the sum of all numbers in a range.
for i in range(1,11):
  i += i
print(i)
```

```
20
```

```
#3. Write a Python program to print all even numbers in a given range.
for i in range(1,51):
  if i%2==1:
    print(i,end = ' ')
```

```
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49
```

```
#4. Write a Python program to print all odd numbers in a given range.
for i in range(1,51):
  if i%2==0:
    print(i,end=' ')
```

```
2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50
```

```
#5. Write a Python program to find the average of all numbers in a range.
sum = 0
for i in range(1,11):
  sum += i
  average = sum/i
print(f'sum is:{sum}, average is :{average}')
```

```
sum is:55, average is :5.5
```

```
#6. Write a Python program to check if a number is present in a given range.
present_number = 8
for num in range(1,31):
  if num == present_number:
    yes=True
    break
if yes:
    print(f'Number {present_number} is present.')
else:
    print(f'Number {present_number} is not present.')
```

```
Number 8 is present.
```

```
#7. Write a Python program to reverse a range of numbers and print them.
for num in range(20,0,-1):
  print(num,end=' ')
```

```
20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
```

```
#8. Write a Python program to find the product of all numbers in a range.
for num in range(1,11):
  num *= num
print(num)
```

```
100
```

```
#9. Write a Python program to print the squares of all numbers in a range.
start = 1
end = 10
squares = 0
for num in range(start,end+1):
  squares = num*num
  print(f'Number is {num} and its square is {squares}')
```

```
Number is 1 and its square is 1
Number is 2 and its square is 4
Number is 3 and its square is 9
Number is 4 and its square is 16
Number is 5 and its square is 25
Number is 6 and its square is 36
Number is 7 and its square is 49
Number is 8 and its square is 64
Number is 9 and its square is 81
Number is 10 and its square is 100
```

```
#10. Write a Python program to print the cube of all numbers in a range.
cube = 0
for num in range(1,11):
  cube = num ** 3
  print(f'Number is {num} and its square is {cube}')
```

```
Number is 1 and its square is 1
Number is 2 and its square is 8
Number is 3 and its square is 27
Number is 4 and its square is 64
Number is 5 and its square is 125
Number is 6 and its square is 216
Number is 7 and its square is 343
Number is 8 and its square is 512
Number is 9 and its square is 729
Number is 10 and its square is 1000
```

▾ If-Else Loop Exercises:

```
#1. Write a Python program to check if a number is positive, negative, or zero.
def positive_negative_zero(num):
  if num > 0:
    return 'number is positive'
  elif num < 0:
    return 'number is negative'
  else:
    return 'number is zero'

num1 = 0
num2 = 9
num3 = -1
print(positive_negative_zero(num1))
print(positive_negative_zero(num2))
print(positive_negative_zero(num3))
```

```
number is zero
number is positive
number is negative
```

```
#2. Write a Python program to check if a number is even or odd.
num = int(input('enter a number :'))
if num % 2 == 0:
  print('number is odd')
else:
  print('number is even')
```

```
enter a number :34
number is odd
```

```
#3. Write a Python program to check if a year is a leap year or not.
num = int(input('enter a year : '))
if num%400 == 0 or num%4 == 0:
  print(f'{num} this is leap year')
else:
  print(f'{num} this is not leap year')
```

```
enter a year : 2024
2024 this is leap year
```

```
#4. Write a Python program to find the maximum of three numbers using if-else.
num1 = int(input('Enter the first number: '))
num2 = int(input('Enter the second number: '))
num3 = int(input('Enter the third number: '))

if num1 > num2 > num3:
    print(f'{num1} is greater than {num2} and {num3}')
elif num2 > num1 > num3:
    print(f'{num2} is greater than {num1} and {num3}')
else:
    print(f'{num3} is greater than {num2} and {num1}')
```

```
Enter the first number: 89
Enter the second number: 90
Enter the third number: 34
90 is greater than 89 and 34
```

```
#5. Write a Python program to check if a number is prime.
def prime_number_or_not(num):
  if num == 1:
    return 'number is not prime number '
  if num > 1:
    for i in range(2,num):
      if num%i == 0:
        return 'number is not prime number'
        break
    else:
     return 'number is prime number'

number = 8
result = prime_number_or_not(number)
print(result)
```

```
number is not prime number
```

```
#6. Write a Python program to check if a number is divisible by both 3 and 5.
num = int(input('enter a number : '))
if num%3 == 0 and num%5 == 0:
  print(f'{num} are divisible by both 3 and 5')
```

```
else:
  print(f'{num} are not divisible by both 3 and 5')
```

```
enter a number : 15
15 are divisible by both 3 and 5
```

```
#7. Write a Python program to check if a character is a vowel or consonant.
char = input('enter a string : ')
if char in 'aeiouAEIOU':
  print(f'{char},is vowel')
else:
  print(f'{char},is consonant')
```

```
enter a string : iou
iou,is vowel
```

```
#8. Write a Python program to check if a given string is a palindrome using if-else.
char = input('enter a string : ')
if char == char[::-1]:
  print(f'{char} is palindrome')
else:
  print(f'{char} is not palindrome')
```

```
enter a string : madam
madam is palindrome
```

```
#9. Write a Python program to determine the largest among three numbers using nested if-else
num1 = float(input("Enter first number: "))
num2 = float(input("Enter second number: "))
num3 = float(input("Enter third number: "))

if num1 >= num2:
    if num1 >= num3:
        print("Largest number:", num1)
    else:
        print("Largest number:", num3)
else:
    if num2 >= num3:
        print("Largest number:", num2)
    else:
        print("Largest number:", num3)
```

```
Enter first number: 30
Enter second number: 20
Enter third number: 90
Largest number: 90.0
```

```
#10. Write a Python program to check if a triangle is equilateral, isosceles, or scalene based on its side lengths using if-else.
side_one = float(input('Enter first side : '))
side_two = float(input('Enter second side : '))
side_three = float(input('Enter third side : '))
if side_one == side_two == side_three:
  print('this is equilateral triangle')
elif side_one != side_two == side_three or side_one == side_two != side_three or side_one != side_three == side_two :
  print('this is isosceles triangle')
else:
  print('this is scalene triangle')
```

```
Enter first side : 14
Enter second side : 14
Enter third side : 8
this is isosceles triangle
```

For Loop Exercises:

```
#1. Write a Python program to print the numbers from 1 to 10 using a for loop.
for num in range(1 , 11):
  print(num,end=' ')
```

```
1 2 3 4 5 6 7 8 9 10
```

```
#2. Write a Python program to calculate the sum of all numbers in a list using a for loop.
sum = 0
for num in range(1,11):
  sum += num
print('sum : ',sum)
```

```
     sum :  55
```

```
#3. Write a Python program to find the factorial of a number using a for loop.
import math

for num in range(1, 11):
    factorial = math.factorial(num)
    print(f"Factorial of {num} is {factorial}")
```

```
     Factorial of 1 is 1
     Factorial of 2 is 2
     Factorial of 3 is 6
     Factorial of 4 is 24
     Factorial of 5 is 120
     Factorial of 6 is 720
     Factorial of 7 is 5040
     Factorial of 8 is 40320
     Factorial of 9 is 362880
     Factorial of 10 is 3628800
```

```
#4. Write a Python program to print all the even numbers between 1 and 50 using a for loop.
for num in range(1,51):
  if num%2==1:
    print(num,end=" ")
```

```
     1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49
```

```
#5. Write a Python program to iterate over a string and print each character using a for loop.
string = 'mayank'
for char in string:
  print(char)
```

```
     m
     a
     y
     a
     n
     k
```

```
#6. Write a Python program to iterate over a list of tuples and print each element using a for loop.
list1 = [(1,'mayank'),(2,'digvijay')]
for char in list1:
  print(char)
```

```
     (1, 'mayank')
     (2, 'digvijay')
```

```
#7. Write a Python program to find the largest element in a list using a for loop.
list1 = [10,30,90,50,110]
maximum_number = list1[0]
for num in list1:
  if num>maximum_number:
    maximum_number = num
print(num)
```

```
     110
```

```
#8. Write a Python program to check if all elements in a list are even using a for loop.
my_list = [1,3,5,7]
all_even = True
for num in my_list:
  if num%2==1:
    all_even = False
    break
if all_even:
  print("All elements in the list are even.")
else:
    print("Not all elements in the list are even.")
```

```
     Not all elements in the list are even.
```

```
#9. Write a Python program to find the common elements between two lists using a for loop.
list1 = [10,20,30,40]
list2 = [20,30,40,50]
common_list = []
for num1 in list1:
  for num2 in list2:
    if num1 == num2 :
      common_list.append(num1)
print(common_list)
```

```
[20, 30, 40]
```

```
#10. Write a Python program to calculate the sum of the digits of a number using a for loop.
num = 121
sum_of_digits = 0

num_str = str(num)

for digit in num_str:
    sum_of_digits += int(digit)

print("Sum of the digits:", sum_of_digits)
```

```
Sum of the digits: 4
```

While Loop Exercises:

```
#1. Write a Python program to print the numbers from 1 to 10 using a while loop.
num = 1
while num <= 10:
  print(num)
  num+=1
```

```
1
2
3
4
5
6
7
8
9
10
```

```
#2. Write a Python program to calculate the sum of all numbers from 1 to 100 using a while loop.
num = 1
sum = 0
while num <=100:
  sum += num
  num += 1
print("Sum:", sum)
```

```
Sum: 5050
```

```
#3. Write a Python program to find the factorial of a number using a while loop.
import math
num = 1
factorial = 0
while num <= 10:
  factorial = math.factorial(num)
  print(f"the number :{num} anf their factorial :{factorial}")
  num += 1
```

```
the number :1 anf their factorial :1
the number :2 anf their factorial :2
the number :3 anf their factorial :6
the number :4 anf their factorial :24
the number :5 anf their factorial :120
the number :6 anf their factorial :720
the number :7 anf their factorial :5040
the number :8 anf their factorial :40320
the number :9 anf their factorial :362880
the number :10 anf their factorial :3628800
```

```
#4. Write a Python program to print all the even numbers between 1 and 50 using a while loop.
num = 2
while num <=50:
  print(num,end = " ")
  num +=2
```

```
2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50
```

```
#5. Write a Python program to iterate over a string and print each character using a while loop.
string = 'mayank'
index = 0
```

```
while index<len(string):
  print(string[index])
  index += 1
```

```
m
a
y
a
n
k
```

```
#6. Write a Python program to iterate over a list of tuples and print each element using a while loop.
truple = (1,'mayank',2,'prachi')
index = 0
while index<len(truple):
    print(truple[index])
    index += 1
```

```
1
mayank
2
prachi
```

```
#7. Write a Python program to find the largest element in a list using a while loop.
my_list = [900,1000,388,2000]
index = 0
max_num = my_list[0]
while index < len(my_list):
  if my_list[index] > max_num:
    max_num = my_list[index]
  index += 1
print(max_num)
```

```
2000
```

```
#9. Write a Python program to find the common elements between two lists using a while loop.
list1 = [1,2,3,4,5]
list2 = [3,4,2,6,9]
common = []
i = 0
while i < len(list1):
  j =0
  while j < len(list2):
    if list1[i]==list2[j]:
      common.append(list1[i])
    j += 1
  i += 1
print(common)
```

```
[2, 3, 4]
```

```
#10. Write a Python program to calculate the sum of the digits of a number using a while loop.
num = 29092002
sum = 0
while num > 0:
  digit = num % 10
  sum += num
  num //= 10
print(sum)
```

```
32324444
```

Break Statement Exercises:

```
#1. Write a Python program to find the first occurrence of a number in a list using a for loop and break statement.
list1 = [1,2,3,4,5]
find_number = 5
for num in list1:
  if num == find_number:
    print("Found", find_number, "at index", list1.index(i))
    break
```

```
Found 5 at index 4
```

```
#2. Search for a specific element in a list using a while loop and break statement:
list1 = [10,20,60,40]
search_num = 60
index = 0
```

```
while index < len(list1):
  if list1[index] == search_num:
    print("Element found at index", index)
    break
  index += 1
```

```
Element found at index 2
```

```
#3. Write a Python program to find the prime numbers between 1 and 100 using a for loop and break statement.
num = int(input('Enter a number : '))

if num == 1:
  print('Number is not prime number')
for i in range(2,num):
  if num%i == 0:
    print('Number is not prime number')
    break
else:
  print('number is prime number')
```

```
Enter a number : 7
number is prime number
```

```
#4. Write a Python program to check if a number is present in a list using a while loop and break statement.
list1 = [1,2,3,4,5,6]
present_not = 4
index = 0
while index < len(list1):
  if list1[index] == present_not:
    print(f"{present_not} is present in the list.")
    break
  index += 1
```

```
4 is present in the list.
```

```
#5. Largest Palindrome Number in a Given Range using a For Loop and Break:
```

```
True
```

```
#6. First Negative Number in a List using a While Loop and Break:
list1 = [2,5,-1,9,7,-8]
i = 0
while i < len(list1):
  if list1[i] < 0:
    print("First negative number:", list1[i])
    break
  i += 1
```

```
First negative number: -1
```

```
list1 = [2, 5, -1, 9, 7, -8]
i = 0
while i < len(list1):
    if list1[i] < 0:
        print("First negative number:", list1[i])
        break
    i += 1
```

```
First negative number: -1
```

```
#7. Write a Python program to print the elements of a list until a specific condition is met using a for loop and break statement.
list1 = [10, 20, 30, 40]
list2 = [80, 50, 30, 100]
sum_num = 30

for num in list1 + list2:
    if num == sum_num:
        break
    print(num)
```

```
10
20
```

```
#8. Write a Python program to search for a specific character in a string using a while loop and break statement
string = 'Prachi'
char = 'a'
index = 0
```

```
while index < len(string):
  if string[index] == char:
    print(f"The character '{char}' is found at index {index}.")
    break
  index += 1
```

```
The character 'a' is found at index 2.
```

```
#9. Write a Python program to find the first occurrence of a vowel in a string using a for loop and break statement.
def vowel(string):
  total_vowel = "aeiouAEIOU"
  for char in string:
    if char in total_vowel:
      print(f"The first vowel '{char}' is found at index {index}.")
      break
str = 'Prachi'
vowel(str)
```

```
The first vowel 'a' is found at index 2.
```

```
#10. Write a Python program to find the index of the first occurrence of a substring in a string using a while loop and break statement.
```

Continue Statement Exercises:

```
#1. Write a Python program to print all the even numbers between 1 and 20 except for the number 10 using a for loop and continue statemen
for num in range(1,20):
  if num == 10:
    continue
  if num % 2 == 0:
    print(num)
```

```
2
4
6
8
12
14
16
18
```

```
#2. Write a Python program to print the elements of a list skipping the negative numbers using a while loop and continue statement.
list1 = [1,2,3,-1,-2,-3,4,5]
index = 0
postive_list = []
while index < len(list1):
  if list1[index] < 0:
    index += 1
    continue
  if list1[index] > 0:
    postive_list.append(list1[index])
  index+= 1
print(postive_list)
```

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

```
#3. Write a Python program to print the first 10 multiples of 3 except for the number 9 using a for loop and continue statement.
for num in range(3,31):
  if num == 9:
    continue
  if num % 3 == 0:
    print(num)
```

```
3
6
12
15
18
21
24
27
30
```

```
#4. Write a Python program to iterate over a string and print only the consonants using a for loop and continue statement.
def only_consonants(string):
  vowel ="aeiouAEIOU"
  for char in string:
    if char in vowel:
      continue
    print(char,end='')

string = 'mayank'
only_consonants(string)
```

```
mynk
```

```
#5. Write a Python program to print the elements of a list skipping the even numbers using a while loop and continue statement.
list1 = [1,2,3,4,5,6,7,8,9,10]
index = 0
while index < len(list1):
  if list1[index]%2 == 0:
    index += 1
    continue
  print(list1[index],end= ' ')
  index += 1
```

```
1 3 5 7 9
```

```
#6. Write a Python program to find the sum of all numbers between 1 and 100, excluding the multiples of 5 using a for loop and continue s
sum = 0
for num in range(1,100):
  if num % 5 == 0:
    continue
  sum += num
print("Sum of numbers between 1 and 100 (excluding multiples of 5):", sum)
```

```
Sum of numbers between 1 and 100 (excluding multiples of 5): 4000
```

```
#7. Write a Python program to print the uppercase letters in a string using a while loop and continue statement.
string = 'HelloWorld'
index = 0
while index < len(string):
  char = string[index]
  if char.isupper():
    print(char ,end ='')
  index += 1
```

```
HW
```

```
#8. Write a Python program to print the elements of a list skipping the elements divisible by 3 using a for loop and continue statement.
def list_skipping(lst):
  for num in lst:
    if num % 3 == 0:
      continue
    print(num)
list1 = [1,2,3,4,5]
list_skipping(list1)
```

```
1
2
4
5
```

```
list1 = [1,2,3,4,5,6,7,8,9,10]
for num in list1:
  if num % 3 == 0:
    continue
  print(num)
```

```
1
2
4
5
7
8
10
```

```
#9. Write a Python program to iterate over a list of tuples and print only the tuples with a specific condition using a while loop and co
tuples = [(1, 'apple'), (2, 'banana'), (3, 'cherry'), (4, 'date')]

i = 0

while i < len(tuples):
    if tuples[i][1].startswith('b'):
        print(tuples[i])
    i += 1
```

```
(2, 'banana')
```

```
#10. Print Numbers from 1 to 50, Skipping Multiples of 7 using a For Loop and Continue:
for num in range(1,50):
  if num % 7 == 0:
    continue
  print(num,end=',')
```

```
1,2,3,4,5,6,8,9,10,11,12,13,15,16,17,18,19,20,22,23,24,25,26,27,29,30,31,32,33,34,36,37,38,39,40,41,43,44,45,46,47,48,
```