**SOLUTION TO PYTHON EXERCISE**

**1. Odd or Even Checker**

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

if num % 2 == 0:

print(f"{num} is even")

else:

print(f"{num} is odd")

**2. Largest of Three Numbers**

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

b = int(input("Enter second number: "))

c = int(input("Enter third number: "))

if a >= b and a >= c:

print(f"{a} is the largest")

elif b >= a and b >= c:

print(f"{b} is the largest")

else:

print(f"{c} is the largest")

**3. Leap Year Checker**

year = int(input("Enter a year: "))

if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):

print(f"{year} is a leap year")

else:

print(f"{year} is not a leap year")

**4. Grade Calculator**

marks = int(input("Enter your marks: "))

if marks >= 90:

grade = 'A'

elif marks >= 80:

grade = 'B'

elif marks >= 70:

grade = 'C'

elif marks >= 60:

grade = 'D'

else:

grade = 'F'

print(f"Your grade is {grade}")

**5. Vowel or Consonant**

char = input("Enter a character: ").lower()

if char in 'aeiou':

print(f"{char} is a vowel")

else:

print(f"{char} is a consonant")

**6. Positive, Negative, or Zero**

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

if num > 0:

print(f"{num} is positive")

elif num < 0:

print(f"{num} is negative")

else:

print(f"{num} is zero")

**7. Divisibility Test**

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

if num % 5 == 0 and num % 11 == 0:

print(f"{num} is divisible by 5 and 11")

else:

print(f"{num} is not divisible by 5 and 11")

**8. Triangle Validity Checker**

a = int(input("Enter first side: "))

b = int(input("Enter second side: "))

c = int(input("Enter third side: "))

if a + b > c and a + c > b and b + c > a:

print("The given sides form a triangle")

else:

print("The given sides do not form a triangle")

**9. Number in Range**

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

lower = int(input("Enter lower range: "))

upper = int(input("Enter upper range: "))

if lower <= num <= upper:

print(f"{num} is within the range")

else:

print(f"{num} is not within the range")

**10. Number of Days in Month**

month = int(input("Enter month number (1-12): "))

if month == 2:

year = int(input("Enter year: "))

if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):

days = 29

else:

days = 28

elif month in [4, 6, 9, 11]:

days = 30

else:

days = 31

print(f"Number of days: {days}")

**11. Print Numbers from 1 to 100**

for i in range(1, 101):

print(i)

**12. Sum of First N Natural Numbers**

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

total = sum(range(1, N+1))

print(f"Sum of first {N} natural numbers is {total}")

**13. Factorial Calculation**

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

factorial = 1

for i in range(1, num+1):

factorial \*= i

print(f"Factorial of {num} is {factorial}")

**14. Fibonacci Sequence**

N = int(input("Enter number of terms: "))

a, b = 0, 1

for \_ in range(N):

print(a, end=' ')

a, b = b, a + b

**15. Prime Number Checker**

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

if num > 1:

for i in range(2, int(num/2)+1):

if num % i == 0:

print(f"{num} is not a prime number")

break

else:

print(f"{num} is a prime number")

else:

print(f"{num} is not a prime number")

**16. Multiplication Table**

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

for i in range(1, 11):

print(f"{num} x {i} = {num \* i}")

**17. Reverse a Number**

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

reverse = 0

while num != 0:

reverse = reverse \* 10 + num % 10

num //= 10

print(f"Reversed number is {reverse}")

**18. Count Digits in a Number**

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

count = 0

while num != 0:

num //= 10

count += 1

print(f"Number of digits is {count}")

**19. Sum of Digits**

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

total = 0

while num != 0:

total += num % 10

num //= 10

print(f"Sum of digits is {total}")

**20. Armstrong Number Checker**

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

sum\_of\_powers = sum([int(digit)\*\*len(str(num)) for digit in str(num)])

if sum\_of\_powers == num:

print(f"{num} is an Armstrong number")

else:

print(f"{num} is not an Armstrong number")

21. \*\*List Sum\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

print(f"Sum of elements in the list is {sum(lst)}")

22. \*\*List Maximum and Minimum\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

print(f"Maximum element is {max(lst)}, Minimum element is {min(lst)}")

23. \*\*List Average\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

print(f"Average of elements in the list is {sum(lst) / len(lst)}")

24. \*\*String Reversal\*\*

s = input("Enter a string: ")

print(f"Reversed string is {s[::-1]}")

25. \*\*String Palindrome Checker\*\*

s = input("Enter a string: ")

if s == s[::-1]:

print(f"{s} is a palindrome")

else:

print(f"{s} is not a palindrome")

26. \*\*Count Vowels in a String\*\*

s = input("Enter a string: ").lower()

count = sum(1 for char in s if char in 'aeiou')

print(f"Number of vowels in the string is {count}")

27. \*\*Dictionary Key-Value Swap\*\*

d = {key: value for key, value in input("Enter dictionary items (key:value) separated by commas: ").split(',')}

swapped = {v: k for k, v in d.items()}

print(f"Swapped dictionary: {swapped}")

28. \*\*Merge Two Dictionaries\*\*

d1 = {key: value for key, value in input("Enter first dictionary items (key:value) separated by commas: ").split(',')}

d2 = {key: value for key, value in input("Enter second dictionary items (key:value) separated by commas: ").split(',')}

d1.update(d2)

print(f"Merged dictionary: {d1}")

29. \*\*Set Operations\*\*

s1 = set(input("Enter first set elements separated by space: ").split())

s2 = set(input("Enter second set elements separated by space: ").split())

print(f

"Union: {s1 | s2}")

print(f"Intersection: {s1 & s2}")

print(f"Difference: {s1 - s2}")

30. \*\*Tuple Operations\*\*

t = tuple(map(int, input("Enter tuple elements separated by space: ").split()))

print(f"Sum: {sum(t)}, Min: {min(t)}, Max: {max(t)}")

31. \*\*Login Authentication System\*\*

stored\_username = "admin"

stored\_password = "password"

username = input("Enter username: ")

password = input("Enter password: ")

if username == stored\_username and password == stored\_password:

print("Login successful")

else:

print("Login failed")

32. \*\*Quadratic Equation Roots\*\*

import cmath

a = float(input("Enter coefficient a: "))

b = float(input("Enter coefficient b: "))

c = float(input("Enter coefficient c: "))

d = b\*\*2 - 4\*a\*c

root1 = (-b + cmath.sqrt(d)) / (2\*a)

root2 = (-b - cmath.sqrt(d)) / (2\*a)

print(f"Roots are {root1} and {root2}")

33. \*\*Grade Categorization\*\*

marks = int(input("Enter your marks: "))

if marks >= 90:

grade = 'A'

elif marks >= 80:

grade = 'B'

elif marks >= 70:

grade = 'C'

elif marks >= 60:

grade = 'D'

else:

grade = 'F'

print(f"Your grade is {grade}")

34. \*\*Simple Calculator\*\*

num1 = float(input("Enter first number: "))

num2 = float(input("Enter second number: "))

op = input("Enter operation (+, -, \*, /): ")

if op == '+':

result = num1 + num2

elif op == '-':

result = num1 - num2

elif op == '\*':

result = num1 \* num2

elif op == '/':

result = num1 / num2

else:

result = "Invalid operation"

print(f"Result: {result}")

35. \*\*BMI Calculator\*\*

weight = float(input("Enter your weight (kg): "))

height = float(input("Enter your height (m): "))

bmi = weight / (height \*\* 2)

if bmi < 18.5:

category = "Underweight"

elif 18.5 <= bmi < 24.9:

category = "Normal weight"

elif 25 <= bmi < 29.9:

category = "Overweight"

else:

category = "Obesity"

print(f"Your BMI is {bmi:.2f}, which is considered {category}")

36. \*\*Day of the Week\*\*

```python

import datetime

date\_str = input("Enter date (YYYY-MM-DD): ")

date\_obj = datetime.datetime.strptime(date\_str, '%Y-%m-%d')

print(f"The day of the week is {date\_obj.strftime('%A')}")

37. \*\*Character Type Checker\*\*

char = input("Enter a character: ")

if char.isupper():

print(f"{char} is an uppercase letter")

elif char.islower():

print(f"{char} is a lowercase letter")

elif char.isdigit():

print(f"{char} is a digit")

else:

print(f"{char} is a special character")

38. \*\*Traffic Light Simulation\*\*

light = input("Enter traffic light color (red, yellow, green): ").lower()

if light == 'red':

action = "Stop"

elif light == 'yellow':

action = "Slow down"

elif light == 'green':

action = "Go"

else:

action = "Invalid light color"

print(action)

39. \*\*Month to Number Conversion\*\*

month = input("Enter month name: ").lower()

months = ["january", "february", "march", "april", "may", "june", "july", "august", "september", "october", "november", "december"]

if month in months:

print(f"Month number is {months.index(month) + 1}")

else:

print("Invalid month name")

40. \*\*Password Strength Checker\*\*

```python

import re

password = input("Enter a password: ")

if len(password) < 8:

print("Password is too short")

elif not re.search(r'[a-z]', password):

print("Password must contain at least one lowercase letter")

elif not re.search(r'[A-Z]', password):

print("Password must contain at least one uppercase letter")

elif not re.search(r'[0-9]', password):

print("Password must contain at least one digit")

elif not re.search(r'[!@#$%^&\*(),.?":{}|<>]', password):

print("Password must contain at least one special character")

else:

print("Password is strong")

41. \*\*Right Triangle Pattern\*\*

n = int(input("Enter the number of rows: "))

for i in range(1, n+1):

print('\*' \* i)

42. \*\*Inverted Triangle Pattern\*\*

n = int(input("Enter the number of rows: "))

for i in range(n, 0, -1):

print('\*' \* i)

43. \*\*Diamond Shape Pattern\*\*

n = int(input("Enter the number of rows: "))

for i in range(1, n+1):

print(' ' \* (n-i) + '\*' \* (2\*i-1))

for i in range(n-1, 0, -1):

print(' ' \* (n-i) + '\*' \* (2\*i-1))

44. \*\*Prime Numbers in Range\*\*

start = int(input("Enter the start of the range: "))

end = int(input("Enter the end of the range: "))

for num in range(start, end + 1):

if num > 1:

for i in range(2, int(num/2) + 1):

if num % i == 0:

break

else:

print(num, end=' ')

45. \*\*Sum of Even and Odd Numbers\*\*

start = int(input("Enter the start of the range: "))

end = int(input("Enter the end of the range: "))

sum\_even = sum\_odd = 0

for num in range(start, end + 1):

if num % 2 == 0:

sum\_even += num

else:

sum\_odd += num

print(f"Sum of even numbers: {sum\_even}, Sum of odd numbers: {sum\_odd}")

46. \*\*GCD of Two Numbers\*\*

def gcd(a, b):

while b:

a, b = b, a % b

return a

num1 = int(input("Enter first number: "))

num2 = int(input("Enter second number: "))

print(f"GCD of {num1} and {num2} is {gcd(num1, num2)}")

47. \*\*LCM of Two Numbers\*\*

def lcm(a, b):

def gcd(a, b):

while b:

a, b = b, a % b

return a

return abs(a \* b) // gcd(a, b)

num1 = int(input("Enter first number: "))

num2 = int(input("Enter second number: "))

print(f"LCM of {num1} and {num2} is {lcm(num1, num2)}")

48. \*\*Hollow Square Pattern\*\*

n = int(input("Enter the size of the square: "))

for i in range(n):

if i == 0 or i == n-1:

print('\*' \* n)

else:

print('\*' + ' ' \* (n-2) + '\*')

49. \*\*Floyd’s Triangle\*\*

n = int(input("Enter the number of rows: "))

num = 1

for i in range(1, n+1):

for j in range(1, i+1):

print(num, end=' ')

num += 1

print()

50. \*\*Count Prime Numbers\*\*

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

count = 0

for num in range(2, N+1):

for i in range(2, int(num/2)+1):

if num % i == 0:

break

else:

count += 1

print(f"Number of prime numbers up to {N} is {count}")

51. \*\*Remove Duplicates from List\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

unique\_lst =

list(set(lst))

print(f"List after removing duplicates: {unique\_lst}")

52. \*\*Second Largest in List\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

lst = list(set(lst)) # Remove duplicates

lst.sort()

if len(lst) > 1:

print(f"Second largest element is {lst[-2]}")

else:

print("Not enough unique elements")

53. \*\*Rotate List\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

k = int(input("Enter number of positions to rotate: "))

k %= len(lst)

rotated\_lst = lst[-k:] + lst[:-k]

print(f"Rotated list: {rotated\_lst}")

54. \*\*Frequency of Elements in List\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

freq = {i: lst.count(i) for i in lst}

print(f"Frequency of elements: {freq}")

55. \*\*Sum of List Elements\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

print(f"Sum of list elements is {sum(lst)}")

56. \*\*Smallest Element in List\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

print(f"Smallest element in the list is {min(lst)}")

57. \*\*Replace List Element\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

old\_value = int(input("Enter the value to be replaced: "))

new\_value = int(input("Enter the new value: "))

lst = [new\_value if x == old\_value else x for x in lst]

print(f"Updated list: {lst}")

58. \*\*Cumulative Sum of List\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

cum\_sum = [sum(lst[:i+1]) for i in range(len(lst))]

print(f"Cumulative sum of the list: {cum\_sum}")

59. \*\*Unique Elements in List\*\*

lst = [int(x) for x in input("Enter list elements separated by space: ").split()]

unique\_lst = [x for x in lst if lst.count(x) == 1]

print(f"Unique elements in the list: {unique\_lst}")

60. \*\*Matrix Addition\*\*

rows = int(input("Enter number of rows: "))

cols = int(input("Enter number of columns: "))

matrix1 = [[int(input(f"Enter element for matrix1[{i}][{j}]: ")) for j in range(cols)] for i in range(rows)]

matrix2 = [[int(input(f"Enter element for matrix2[{i}][{j}]: ")) for j in range(cols)] for i in range(rows)]

result = [[matrix1[i][j] + matrix2[i][j] for j in range(cols)] for i in range(rows)]

print("Resultant matrix after addition:")

for row in result:

print(row)

61. \*\*Matrix Multiplication\*\*

rows1 = int(input("Enter number of rows in first matrix: "))

cols1 = int(input("Enter number of columns in first matrix: "))

rows2 = int(input("Enter number of rows in second matrix: "))

cols2 = int(input("Enter number of columns in second matrix: "))

if cols1 != rows2:

print("Matrix multiplication is not possible")

else:

matrix1 = [[int(input(f"Enter element for matrix1[{i}][{j}]: ")) for j in range(cols1)] for i in range(rows1)]

matrix2 = [[int(input(f"Enter element for matrix2[{i}][{j}]: ")) for j in range(cols2)] for i in range(rows2)]

result = [[sum(matrix1[i][k] \* matrix2[k][j] for k in range(cols1)) for j in range(cols2)] for i in range(rows1)]

print("Resultant matrix after multiplication:")

for row in result:

print(row)

62. \*\*Matrix Transpose\*\*

rows = int(input("Enter number of rows: "))

cols = int(input("Enter number of columns: "))

matrix = [[int(input(f"Enter element for matrix[{i}][{j}]: ")) for j in range(cols)] for i in range(rows)]

transpose = [[matrix[j][i] for j in range(rows)] for i in range(cols)]

print("Transpose of the matrix:")

for row in transpose:

print(row)

63. \*\*Upper Triangular Matrix\*\*

n = int(input("Enter the size of the matrix: "))

matrix = [[int(input(f"Enter element for matrix[{i}][{j}]: ")) for j in range(n)] for i in range(n)]

print("Upper triangular matrix:")

for i in range(n):

for j in range(n):

if i > j:

print(0, end=' ')

else:

print(matrix[i][j], end=' ')

print()

64. \*\*Lower Triangular Matrix\*\*

n = int(input("Enter the size of the matrix: "))

matrix = [[int(input(f"Enter element for matrix[{i}][{j}]: ")) for j in range(n)] for i in range(n)]

print("Lower triangular matrix:")

for i in range(n):

for j in range(n):

if i < j:

print(0, end=' ')

else:

print(matrix[i][j], end=' ')

print()

65. \*\*Sparse Matrix Checker\*\*

rows = int(input("Enter number of rows: "))

cols = int(input("Enter number of columns: "))

matrix = [[int(input(f"Enter element for matrix[{i}][{j}]: ")) for j in range(cols)] for i in range(rows)]

zero\_count = sum(row.count(0) for row in matrix)

if zero\_count > (rows \* cols) / 2:

print("The matrix is a sparse matrix")

else:

print("The matrix is not a sparse matrix")

66. \*\*List of Tuples to Dictionary\*\*

n = int(input("Enter number of tuples: "))

tuples\_list = [tuple(input(f"Enter tuple {i+1} elements separated by space: ").split()) for i in range(n)]

dictionary = {k: v for k, v in tuples\_list}

print(f"Dictionary: {dictionary}")

67. \*\*Frequency of Words in String\*\*

s = input("Enter a string: ")

words = s.split()

frequency = {word: words.count(word) for word in words}

print(f"Frequency of words: {frequency}")

68. \*\*String to List of Characters\*\*

s = input("Enter a string: ")

char\_list = list(s)

print(f"List of characters: {char\_list}")

69. \*\*Convert List to String\*\*

lst = input("Enter list elements separated by space: ").split()

string = ' '.join(lst)

print(f"String: {string}")

70. \*\*Count Substring Occurrence\*\*

s = input("Enter a string: ")

sub = input("Enter substring: ")

print(f"Number of occurrences of '{sub}': {s.count(sub)}")

71. \*\*Remove Vowels from String\*\*

s = input("Enter a string: ")

no\_vowels = ''.join([char for char in s if char.lower() not in 'aeiou'])

print(f"String without vowels: {no\_vowels}")

72. \*\*String to Title Case\*\*

s = input("Enter a string: ")

title\_case = s.title()

print(f"Title case string: {title\_case}")

73. \*\*Capitalize First and Last Character of Each Word\*\*

s = input("Enter a string: ")

capitalized = ' '.join([word[0].upper() + word[1:-1] + word[-1].upper() if len(word) > 1 else word.upper() for word in s.split()])

print(f"Capitalized string: {capitalized}")

74. \*\*Count Occurrences of Each Character\*\*

s = input("Enter a string: ")

frequency = {char: s.count(char) for char in set(s)}

print(f"Character frequency: {frequency}")

75. \*\*Reverse Each Word in String\*\*

s = input("Enter a string: ")

reversed\_words = ' '.join([word[::-1] for word in s.split()])

print(f"Reversed words string: {reversed\_words}")

76. \*\*Remove Duplicates from String\*\*

s = input("Enter a string:

")

result = ''.join(sorted(set(s), key=s.index))

print(f"String after removing duplicates: {result}")

77. \*\*Check if String is Palindrome\*\*

s = input("Enter a string: ")

if s == s[::-1]:

print(f"'{s}' is a palindrome")

else:

print(f"'{s}' is not a palindrome")

78. \*\*Longest Word in String\*\*

s = input("Enter a string: ")

words = s.split()

longest\_word = max(words, key=len)

print(f"Longest word: {longest\_word}")

79. \*\*Check Anagram\*\*

str1 = input("Enter first string: ")

str2 = input("Enter second string: ")

if sorted(str1) == sorted(str2):

print(f"'{str1}' and '{str2}' are anagrams")

else:

print(f"'{str1}' and '{str2}' are not anagrams")

80. \*\*Replace Substring in String\*\*

s = input("Enter a string: ")

old\_sub = input("Enter the substring to replace: ")

new\_sub = input("Enter the new substring: ")

result = s.replace(old\_sub, new\_sub)

print(f"String after replacement: {result}")

81. \*\*Count Uppercase, Lowercase, Digits, and Special Characters\*\*

s = input("Enter a string: ")

uppercase = sum(1 for char in s if char.isupper())

lowercase = sum(1 for char in s if char.islower())

digits = sum(1 for char in s if char.isdigit())

special = len(s) - uppercase - lowercase - digits

print(f"Uppercase: {uppercase}, Lowercase: {lowercase}, Digits: {digits}, Special characters: {special}")

82. \*\*Convert String to Uppercase\*\*

s = input("Enter a string: ")

print(f"Uppercase string: {s.upper()}")

83. \*\*Convert String to Lowercase\*\*

s = input("Enter a string: ")

print(f"Lowercase string: {s.lower()}")

84. \*\*Remove Leading and Trailing Whitespaces\*\*

s = input("Enter a string: ")

print(f"String after removing leading and trailing whitespaces: '{s.strip()}'")

85. \*\*String to ASCII Values\*\*

s = input("Enter a string: ")

ascii\_values = [ord(char) for char in s]

print(f"ASCII values: {ascii\_values}")

86. \*\*ASCII Values to String\*\*

ascii\_values = [int(x) for x in input("Enter ASCII values separated by space: ").split()]

string = ''.join(chr(value) for value in ascii\_values)

print(f"String: {string}")

87. \*\*Concatenate Strings\*\*

s1 = input("Enter first string: ")

s2 = input("Enter second string: ")

print(f"Concatenated string: {s1 + s2}")

88. \*\*Find Common Elements in Two Strings\*\*

s1 = input("Enter first string: ")

s2 = input("Enter second string: ")

common\_elements = set(s1) & set(s2)

print(f"Common elements: {common\_elements}")

89. \*\*Find Non-Common Elements in Two Strings\*\*

s1 = input("Enter first string: ")

s2 = input("Enter second string: ")

non\_common\_elements = set(s1) ^ set(s2)

print(f"Non-common elements: {non\_common\_elements}")

90. \*\*Reverse String\*\*

s = input("Enter a string: ")

print(f"Reversed string: {s[::-1]}")

91. \*\*Convert String to List of Words\*\*

s = input("Enter a string: ")

words = s.split()

print(f"List of words: {words}")

92. \*\*Swap First and Last Character of String\*\*

s = input("Enter a string: ")

if len(s) > 1:

s = s[-1] + s[1:-1] + s[0]

print(f"String after swapping first and last character: {s}")

93. \*\*Sum of Digits in String\*\*

s = input("Enter a string: ")

digits\_sum = sum(int(char) for char in s if char.isdigit())

print(f"Sum of digits in the string: {digits\_sum}")

94. \*\*Count Words in String\*\*

s = input("Enter a string: ")

words\_count = len(s.split())

print(f"Number of words in the string: {words\_count}")

95. \*\*Split String by Comma\*\*

s = input("Enter a string: ")

parts = s.split(',')

print(f"Split string: {parts}")

96. \*\*Join List of Words into String\*\*

words = input("Enter words separated by space: ").split()

result = ' '.join(words)

print(f"Joined string: {result}")

97. \*\*Palindrome Check with Case Ignored\*\*

s = input("Enter a string: ")

s = s.lower().replace(" ", "")

if s == s[::-1]:

print(f"'{s}' is a palindrome")

else:

print(f"'{s}' is not a palindrome")

98. \*\*Count Occurrences of Substring Ignoring Case\*\*

s = input("Enter a string: ")

sub = input("Enter substring: ")

count = s.lower().count(sub.lower())

print(f"Number of occurrences of '{sub}': {count}")

99. \*\*Find Position of Substring\*\*

s = input("Enter a string: ")

sub = input("Enter substring: ")

position = s.find(sub)

if position != -1:

print(f"Substring '{sub}' found at position {position}")

else:

print(f"Substring '{sub}' not found")

100. \*\*Check if String Contains Only Alphabets\*\*

s = input("Enter a string: ")

if s.isalpha():

print(f"'{s}' contains only alphabets")

else:

print(f"'{s}' does not contain only alphabets")