**1. Hello World**

print("Hello, World!")

**2. Add Two Numbers**

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

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

print("Sum:", a + b)

**3. Find the Square Root**

import math

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

print("Square root:", math.sqrt(num))

**4. Calculate the Area of a Triangle**

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

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

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

s = (a + b + c) / 2

area = (s\*(s - a)\*(s - b)\*(s - c)) \*\* 0.5

print("Area of Triangle:", area)

**5. Swap Two Variables**

x = input("Enter first value: ")

y = input("Enter second value: ")

x, y = y, x

print("Swapped values: x =", x, ", y =", y)

**6. Check Even or Odd**

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

print("Even" if num % 2 == 0 else "Odd")

**7. Factorial of a Number**

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

factorial = 1

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

factorial \*= i

print("Factorial:", factorial)

**8. Generate Fibonacci Sequence**

n = int(input("How many terms? "))

a, b = 0, 1

for \_ in range(n):

print(a, end=" ")

a, b = b, a + b

**9. Prime Number Checker**

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

if num > 1:

for i in range(2, int(num\*\*0.5) + 1):

if num % i == 0:

print("Not Prime")

break

else:

print("Prime")

else:

print("Not Prime")

**10. Multiplication Table**

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

for i in range(1, 11):

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

**11. Sum of Natural Numbers**

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

sum = n \* (n + 1) // 2

print("Sum:", sum)

**12. Convert Celsius to Fahrenheit**

celsius = float(input("Enter Celsius: "))

fahrenheit = (celsius \* 9/5) + 32

print("Fahrenheit:", fahrenheit)

**13. Convert Fahrenheit to Celsius**

fahrenheit = float(input("Enter Fahrenheit: "))

celsius = (fahrenheit - 32) \* 5/9

print("Celsius:", celsius)

**14. Simple Calculator**

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

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

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

if op == '+':

print(a + b)

elif op == '-':

print(a - b)

elif op == '\*':

print(a \* b)

elif op == '/':

print(a / b)

else:

print("Invalid operator")

**15. Find Largest Number**

a = float(input("First: "))

b = float(input("Second: "))

c = float(input("Third: "))

print("Largest:", max(a, b, c))

**16. Check Leap Year**

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

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

print("Leap Year")

else:

print("Not a Leap Year")

**17. Check Armstrong Number**

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

sum = sum(int(d)\*\*len(str(num)) for d in str(num))

print("Armstrong Number" if num == sum else "Not Armstrong")

**18. Find Armstrong Numbers in Interval**

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

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

for num in range(lower, upper + 1):

if num == sum(int(d)\*\*len(str(num)) for d in str(num)):

print(num, end=" ")

**19. Find Sum of Digits**

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

total = 0

while num > 0:

total += num % 10

num //= 10

print("Sum of digits:", total)

**20. Reverse a Number**

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

rev = 0

while num > 0:

rev = rev \* 10 + num % 10

num //= 10

print("Reversed number:", rev)

**21. Reverse a String**

string = input("Enter a string: ")

print("Reversed string:", string[::-1])

**22. Count Vowels in a String**

string = input("Enter a string: ")

vowels = 'aeiouAEIOU'

count = sum(1 for char in string if char in vowels)

print("Vowel count:", count)

**23. Palindrome Checker**

string = input("Enter a string: ")

if string == string[::-1]:

print("Palindrome")

else:

print("Not Palindrome")

**24. Count Words in a String**

string = input("Enter a string: ")

words = len(string.split())

print("Word count:", words)

**25. Remove Punctuation from a String**

import string

text = input("Enter a string: ")

text = text.translate(str.maketrans('', '', string.punctuation))

print("Text without punctuation:", text)

**26. Find ASCII Value of Character**

char = input("Enter a character: ")

print("ASCII value:", ord(char))

**27. Convert String to Uppercase**

string = input("Enter a string: ")

print("Uppercase:", string.upper())

**28. Convert String to Lowercase**

string = input("Enter a string: ")

print("Lowercase:", string.lower())

**29. Find Length of a String**

string = input("Enter a string: ")

print("Length:", len(string))

**30. Replace a Character in a String**

string = input("Enter a string: ")

old\_char = input("Character to replace: ")

new\_char = input("New character: ")

print("Modified string:", string.replace(old\_char, new\_char))

**31. Check Substring in a String**

string = input("Enter a string: ")

substring = input("Enter substring to check: ")

if substring in string:

print("Substring found")

else:

print("Substring not found")

**32. Merge Two Lists**

list1 = [1, 2, 3]

list2 = [4, 5, 6]

merged = list1 + list2

print("Merged list:", merged)

**33. Remove Duplicate Elements from a List**

lst = [1, 2, 2, 3, 4, 4, 5]

unique\_lst = list(set(lst))

print("List without duplicates:", unique\_lst)

**34. Find Largest Element in a List**

lst = [1, 2, 3, 0, 5]

print("Largest element:", max(lst))

**35. Sort List in Ascending Order**

lst = [5, 2, 9, 1, 5, 6]

lst.sort()

print("Sorted list:", lst)

**36. Sort List in Descending Order**

lst = [5, 2, 9, 1, 5, 6]

lst.sort(reverse=True)

print("Sorted list (descending):", lst)

**37. Sum of All Elements in a List**

lst = [1, 2, 3, 4, 5]

print("Sum of elements:", sum(lst))

**38. Count Occurrences of Element in List**

lst = [1, 2, 3, 1, 1, 4]

print("Occurrences of 1:", lst.count(1))

**39. Find Index of an Element in List**

lst = [10, 20, 30, 40, 50]

print("Index of 30:", lst.index(30))

**40. Find Second Largest Element in a List**

lst = [10, 20, 4, 45, 99]

lst.sort()

print("Second largest element:", lst[-2])

**41. Check if a List is Empty**

lst = []

if not lst:

print("List is empty")

else:

print("List is not empty")

**42. Append an Item to a List**

lst = [1, 2, 3]

lst.append(4)

print("List after append:", lst)

**43. Extend a List with Another List**

lst1 = [1, 2]

lst2 = [3, 4]

lst1.extend(lst2)

print("Extended list:", lst1)

**44. Insert an Element in a List**

lst = [1, 2, 3]

lst.insert(1, 4)

print("List after insertion:", lst)

**45. Remove an Element from a List**

lst = [1, 2, 3, 4]

lst.remove(3)

print("List after removal:", lst)

**46. Pop an Element from a List**

lst = [1, 2, 3, 4]

lst.pop(2)

print("List after pop:", lst)

**47. Clear a List**

lst = [1, 2, 3, 4]

lst.clear()

print("List after clearing:", lst)

**48. Copy a List**

lst = [1, 2, 3]

copied\_lst = lst.copy()

print("Copied list:", copied\_lst)

**49. Concatenate Two Lists**

lst1 = [1, 2]

lst2 = [3, 4]

concatenated = lst1 + lst2

print("Concatenated list:", concatenated)

**50. Count Elements Greater Than a Certain Value**

lst = [10, 20, 30, 40, 50]

count = sum(1 for x in lst if x > 25)

print("Count of elements greater than 25:", count)

**51. Reverse a List**

lst = [1, 2, 3, 4]

lst.reverse()

print("Reversed list:", lst)

**52. Convert List to String**

lst = ['a', 'b', 'c']

string = ''.join(lst)

print("String:", string)

**53. Convert String to List**

string = "hello"

lst = list(string)

print("List:", lst)

**54. Multiply All Elements in a List**

lst = [1, 2, 3, 4]

product = 1

for num in lst:

product \*= num

print("Product of elements:", product)

**55. Find Minimum Element in a List**

lst = [10, 20, 5, 100]

print("Minimum element:", min(lst))

**56. Find Sum of Odd Numbers in a List**

lst = [1, 2, 3, 4, 5]

odd\_sum = sum(x for x in lst if x % 2 != 0)

print("Sum of odd numbers:", odd\_sum)

**57. Find Sum of Even Numbers in a List**

lst = [1, 2, 3, 4, 5]

even\_sum = sum(x for x in lst if x % 2 == 0)

print("Sum of even numbers:", even\_sum)

**58. Find All Prime Numbers in a List**

lst = [10, 11, 12, 13, 14, 15]

primes = [x for x in lst if all(x % i != 0 for i in range(2, x))]

print("Prime numbers:", primes)

**59. Merge Two Dictionaries**

dict1 = {'a': 1, 'b': 2}

dict2 = {'c': 3, 'd': 4}

dict1.update(dict2)

print("Merged dictionary:", dict1)

**60. Check if Key Exists in a Dictionary**

my\_dict = {'a': 1, 'b': 2}

key = 'a'

if key in my\_dict:

print(f"'{key}' exists in dictionary")

else:

print(f"'{key}' does not exist in dictionary")

**61. Create a Dictionary with Multiple Key-Value Pairs**

my\_dict = {'a': 1, 'b': 2, 'c': 3}

print("Dictionary:", my\_dict)

**62. Remove a Key from a Dictionary**

my\_dict = {'a': 1, 'b': 2, 'c': 3}

my\_dict.pop('b')

print("Dictionary after removing 'b':", my\_dict)

**63. Merge Two Lists into a Dictionary**

keys = ['a', 'b', 'c']

values = [1, 2, 3]

my\_dict = dict(zip(keys, values))

print("Merged dictionary:", my\_dict)

**64. Create a Set**

my\_set = {1, 2, 3}

print("Set:", my\_set)

**65. Add an Element to a Set**

my\_set = {1, 2, 3}

my\_set.add(4)

print("Set after adding 4:", my\_set)

**66. Remove an Element from a Set**

my\_set = {1, 2, 3, 4}

my\_set.remove(3)

print("Set after removing 3:", my\_set)

**67. Union of Two Sets**

set1 = {1, 2, 3}

set2 = {3, 4, 5}

union\_set = set1.union(set2)

print("Union of sets:", union\_set)

**68. Intersection of Two Sets**

set1 = {1, 2, 3}

set2 = {3, 4, 5}

intersection\_set = set1.intersection(set2)

print("Intersection of sets:", intersection\_set)

**69. Difference Between Two Sets**

set1 = {1, 2, 3}

set2 = {3, 4, 5}

difference\_set = set1.difference(set2)

print("Difference between sets:", difference\_set)

**70. Check if a Set is Subset of Another Set**

set1 = {1, 2}

set2 = {1, 2, 3}

print("set1 is subset of set2:", set1.issubset(set2))

**71. Check if a Set is Superset of Another Set**

set1 = {1, 2, 3}

set2 = {1, 2}

print("set1 is superset of set2:", set1.issuperset(set2))

**72. Find Length of a Set**

my\_set = {1, 2, 3, 4}

print("Length of set:", len(my\_set))

**73. Clear All Elements in a Set**

my\_set = {1, 2, 3, 4}

my\_set.clear()

print("Set after clearing:", my\_set)

**74. Create a Tuple**

my\_tuple = (1, 2, 3)

print("Tuple:", my\_tuple)

**75. Access Elements of a Tuple**

my\_tuple = (1, 2, 3, 4)

print("Element at index 2:", my\_tuple[2])

**76. Convert Tuple to List**

my\_tuple = (1, 2, 3)

my\_list = list(my\_tuple)

print("List from tuple:", my\_list)

**77. Convert List to Tuple**

my\_list = [1, 2, 3]

my\_tuple = tuple(my\_list)

print("Tuple from list:", my\_tuple)

**78. Find Length of a Tuple**

my\_tuple = (1, 2, 3, 4)

print("Length of tuple:", len(my\_tuple))

**79. Check if Element Exists in Tuple**

my\_tuple = (1, 2, 3, 4)

print("3 exists in tuple:", 3 in my\_tuple)

**80. Unpack a Tuple**

my\_tuple = (1, 2, 3)

a, b, c = my\_tuple

print("Unpacked values:", a, b, c)

**81. Create a Function**

def greet(name):

return f"Hello, {name}!"

print(greet("Alice"))

**82. Function with Default Parameter**

def greet(name="Guest"):

return f"Hello, {name}!"

print(greet()) # Default value

print(greet("Bob")) # Custom value

**83. Function with Multiple Arguments**

def add(a, b, c):

return a + b + c

print(add(1, 2, 3))

**84. Return Multiple Values from Function**

def calculate(x, y):

return x + y, x - y

result = calculate(5, 3)

print("Sum:", result[0], "Difference:", result[1])

**85. Function to Check Even or Odd**

def even\_odd(num):

if num % 2 == 0:

return "Even"

else:

return "Odd"

print(even\_odd(10))

**86. Lambda Function for Addition**

add = lambda x, y: x + y

print(add(5, 3))

**87. Filter Even Numbers from List Using Lambda**

nums = [1, 2, 3, 4, 5, 6]

even\_nums = list(filter(lambda x: x % 2 == 0, nums))

print("Even numbers:", even\_nums)

**88. Map Function to Square Numbers**

nums = [1, 2, 3, 4]

squares = list(map(lambda x: x\*\*2, nums))

print("Squared numbers:", squares)

**89. Reduce Function to Sum a List**

from functools import reduce

nums = [1, 2, 3, 4]

total = reduce(lambda x, y: x + y, nums)

print("Total sum:", total)

**90. Create a Class**

class Person:

def \_\_init\_\_(self, name, age):

self.name = name

self.age = age

def greet(self):

print(f"Hello, I'm {self.name} and I'm {self.age} years old.")

person = Person("Alice", 25)

person.greet()

**91. Inheritance in Python**

class Animal:

def speak(self):

print("Animal speaks")

class Dog(Animal):

def bark(self):

print("Dog barks")

dog = Dog()

dog.speak()

dog.bark()

**92. Method Overriding in Python**

class Animal:

def sound(self):

print("Animal sound")

class Dog(Animal):

def sound(self):

print("Bark")

dog = Dog()

dog.sound()

**93. Encapsulation in Python**

class Person:

def \_\_init\_\_(self, name, age):

self.\_\_name = name

self.\_\_age = age

def get\_name(self):

return self.\_\_name

def set\_name(self, name):

self.\_\_name = name

person = Person("Alice", 25)

print(person.get\_name())

person.set\_name("Bob")

print(person.get\_name())

**94. Abstraction in Python**

from abc import ABC, abstractmethod

class Animal(ABC):

@abstractmethod

def speak(self):

pass

class Dog(Animal):

def speak(self):

print("Bark")

dog = Dog()

dog.speak()

**95. Python Try-Except for Error Handling**

try:

x = 1 / 0

except ZeroDivisionError:

print("Cannot divide by zero!")

**96. Python Finally Block**

try:

x = 1 / 1

except ZeroDivisionError:

print("Cannot divide by zero!")

finally:

print("This block always runs.")

**97. Read a File**

with open("sample.txt", "r") as file:

content = file.read()

print(content)

**98. Write to a File**

with open("output.txt", "w") as file:

file.write("Hello, file writing!")

print("File written successfully!")

**99. Append to a File**

with open("output.txt", "a") as file:

file.write("\nAppending this text.")

print("Text appended to file!")

**100. Delete a File**

import os

os.remove("output.txt")

print("File deleted successfully!")