**Task-2:Calculator**

def add(x, y):

return x + y

def subtract(x, y):

return x - y

def multiply(x, y):

return x \* y

def divide(x, y):

if y == 0:

return "Cannot divide by zero"

return x / y

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

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

operation = input("Choose an operation (+, -, \*, /): ")

if operation == '+':

result = add(num1, num2)

elif operation == '-':

result = subtract(num1, num2)

elif operation == '\*':

result = multiply(num1, num2)

elif operation == '/':

result = divide(num1, num2)

else:

result = "Invalid operation"

print("Result:",result)

**Task-2.Todolist**

tasks = []

def add\_task(task):

tasks.append(task)

print("Task added successfully!")

def display\_tasks():

if not tasks:

print("No tasks in the list.")

else:

for i, task in enumerate(tasks, 1):

print(f"{i}. {task}")

def complete\_task(task\_index):

if 1 <= task\_index <= len(tasks):

task = tasks[task\_index - 1]

print(f"Completed: {task}")

tasks.remove(task)

else:

print("Invalid task index.")

while True:

print("\nTo-Do List Menu:")

print("1. Add Task")

print("2. Display Tasks")

print("3. Complete Task")

print("4. Quit")

choice = input("Enter your choice: ")

if choice == "1":

task = input("Enter the task: ")

add\_task(task)

elif choice == "2":

display\_tasks()

elif choice == "3":

task\_index = int(input("Enter the task index to mark as completed: "))

complete\_task(task\_index)

elif choice == "4":

print("Goodbye!")

break

else:

print("Invalid choice. Please choose again.")

**Task-3.Password**

import random

import string

def generate\_password(length):

characters = string.ascii\_letters + string.digits + string.punctuation

password = ''.join(random.choice(characters) for i in range(length))

return password

def main():

try:

length = int(input("Enter the desired length of the password: "))

if length < 1:

print("Password length should be at least 1 character.")

else:

password = generate\_password(length)

print("Generated Password:", password)

except ValueError:

print("Please enter a valid length (a positive integer).")

if\_name\_=="\_\_main":

main()