TASK:2 (SIMPLE CALCULATOR)

Design a simple calculator with basic arithmetic operations.Prompt the user to input two numbers and an operation choice.Perform the calculation and display the result.

```
print("Welcome to the Simple Calculator!")
print("Choose an operation:")
print("1. Addition (+)")
print("2. Subtraction (-)")
print("3. Multiplication (*)")
print("4. Division (/)")

# For user input
num1 = eval(input("Enter the first number: "))
num2 = eval(input("Enter the second number: "))
operation = (input("Enter the operation (+, -, *, /): "))

# To perform the calculation
if operation == "+":
    result = num1 + num2
    print("The result is:" ,result)
elif operation == "-":
    result = num1 - num2
    print("The result is:" ,result)
elif operation == "*":
    result = num1 * num2
    print("The result is:" ,result)
elif operation == "/":
    if num2 != 0:
        result = num1 / num2
        print("The result is:" ,result)
    else:
        print("Error: Division by zero is not allowed.")
else:
    print("Invalid operation. Please choose from +, -, *, or /.")


Welcome to the Simple Calculator!
Choose an operation:
1. Addition (+)
2. Subtraction (-)
3. Multiplication (*)
4. Division (/)

Enter the first number:  9
Enter the second number:  0
Enter the operation (+, -, *, /):  /

Error: Division by zero is not allowed.
```

TASK:1 (TO-DO LIST)

A To-Do List application is a useful project that helps users manage and organize their tasks efficiently. This project aims to create acommand-line or GUI-based application using Python, allowingusers to create, update, and track their to-do lists

```
tasks = []

while True:
    print("\n1. View Tasks\n2. Add Task\n3. Delete Task\n4. Exit")
    choice = input("Choose an option: ")

    if choice == "1":
        print("\nTasks:", tasks if tasks else "No tasks yet.")
    elif choice == "2":
        task = input("Enter a task: ")
        tasks.append(task)
        print("Task added!")
    elif choice == "3":
        task = input("Enter task to delete: ")
        if task in tasks:
            tasks.remove(task)
            print("Task deleted!")
        else:
            print("Task not found.")
    elif choice == "4":
        print("Goodbye!")
        break
    else:
        print("Invalid choice, try again.")
```

```
1. View Tasks
2. Add Task
3. Delete Task
4. Exit

Choose an option:  2
Enter a task:  study

Task added!

1. View Tasks
2. Add Task
3. Delete Task
4. Exit

Choose an option:  2
Enter a task:  music
```

```
Task added!

1. View Tasks
2. Add Task
3. Delete Task
4. Exit

Choose an option:  2
Enter a task:  dance

Task added!

1. View Tasks
2. Add Task
3. Delete Task
4. Exit

Choose an option:  1


Tasks: ['study', 'music', 'dance']

1. View Tasks
2. Add Task
3. Delete Task
4. Exit

Choose an option:  4

Goodbye!
```

TASK:5 (CONTACT BOOK)

Contact Information: Store name, phone number, email, and address for each contact.Add Contact: Allow users to add new contacts with their details.View Contact List: Display a list of all saved contacts with names and phone numbers.Search Contact: Implement a search function to find contacts by name or phone number.Update Contact: Enable users to update contact details.Delete Contact: Provide an option to delete a contactUser Interface: Design a user-friendly interface for easy interaction.

```python
contact_info = []

while True:
    print("\nAdd, View, Search, Update, Delete, Exit")
    contact = input("Choose an action: ")

    if contact == "Add":
        Name = input("Enter the name: ")
        Ph_Number = input("Enter the number: ")
        Email = input("Enter the email id: ")
        Address = input("Enter the address: ")
        contact_info.append({"Name": Name, "Phone": Ph_Number,
```

```
"Email": Email, "Address": Address})
        print("Contact information added!")

    elif contact == "View":
        if contact_info:
            print("\nContact List:")
            for c in contact_info:
                print(f"Name: {c['Name']}, Phone: {c['Phone']}, Email:
{c['Email']}, Address: {c['Address']}")
        else:
            print("No contacts available.")

    elif contact == "Update":
        Name_to_Update = input("Enter the name to update: ")
        for c in contact_info:
            if c["Name"] == Name_to_Update:
                c["Name"] = input(f"Enter new name (current:
{c['Name']}): ") or c["Name"]
                c["Phone"] = input(f"Enter new phone (current:
{c['Phone']}): ") or c["Phone"]
                c["Email"] = input(f"Enter new email (current:
{c['Email']}): ") or c["Email"]
                c["Address"] = input(f"Enter new address (current:
{c['Address']}): ") or c["Address"]
                print("Contact updated!")
                break
        else:
            print("Contact not found!")

    elif contact == "Search":
        Search_Name = input("Enter the name to search: ")
        for c in contact_info:
            if c["Name"] == Search_Name:
                print("Contact found:", c)
                break
        else:
            print("Contact not found.")

    elif contact == "Delete":
        Name_to_Delete = input("Enter the name to delete: ")
        for c in contact_info:
            if c["Name"] == Name_to_Delete:
                contact_info.remove(c)
                print("Contact deleted!")
                break
        else:
            print("Contact not found.")

    elif contact == "Exit":
        print("Exiting the program!")
```

```
        break

    else:
        print("Invalid choice, try again.")
```

```
Add, View, Search, Update, Delete, Exit

Choose an action:  Add
Enter the name:  al
Enter the number:  12
Enter the email id:  12
Enter the address:  12

Contact information added!

Add, View, Search, Update, Delete, Exit

Choose an action:  Add
Enter the name:  nk
Enter the number:  7
Enter the email id:  7
Enter the address:  7

Contact information added!

Add, View, Search, Update, Delete, Exit

Choose an action:  View


Contact List:
Name: al, Phone: 12, Email: 12, Address: 12
Name: nk, Phone: 7, Email: 7, Address: 7

Add, View, Search, Update, Delete, Exit

Choose an action:  Update
Enter the name to update:  al
Enter new name (current: al):
Enter new phone (current: 12):
Enter new email (current: 12):  q
Enter new address (current: 12):

Contact updated!

Add, View, Search, Update, Delete, Exit

Choose an action:  View
```

```
Contact List:
Name: al, Phone: 12, Email: q, Address: 12
Name: nk, Phone: 7, Email: 7, Address: 7

Add, View, Search, Update, Delete, Exit
```