Project Report: Matrix Calculator Using Python

1. Introduction

The Matrix Calculator is a Python-based interactive tool designed to perform essential matrix operations through a command-line interface. It serves educational as well as practical computational purposes, enabling users to input matrices, validate them, and apply operations such as inverse, transpose, addition, subtraction, multiplication, and identity checks. This project enhances understanding of matrix algebra and Python programming concepts.

---

2. Objectives of the Project

- To develop an interactive matrix computation tool using Python.
- To implement essential matrix operations with proper validation.
- To demonstrate input-handling and error-handling in Python.
- To create a structured and user-friendly command-line interface.

---

3. Technologies / Tools Used

- **Python 3.x** – Core programming language
- **NumPy Library** – For performing mathematical matrix operations
- Terminal/Command Prompt – For interacting with the program

---

4. Project Description

The Matrix Calculator program enables the user to input one or two matrices and perform various operations interactively. It contains modular functions such as: - Input matrix function - Inverse function - Transpose function - Addition, subtraction, multiplication functions - Identical matrix check function

The program starts by taking Matrix A as input. The user can then choose operations such as inverse or transpose. If desired, the user can also input Matrix B to perform combined operations like addition, subtraction, multiplication, and identical matrix check.

This structure makes the program flexible and easy to extend.

---

5. Features of the Matrix Calculator

✔️Matrix Input System

- Accepts custom dimensions
- Validates number of entries
- Detects invalid row/column inputs

✔️Matrix Operations

- **Inverse of a matrix** (only if square and non-singular)
- **Transpose of a matrix**
- **Matrix Addition**
- **Matrix Subtraction**
- **Matrix Multiplication**
- **Identical Matrix Check**

✔️Error Handling

- Incorrect numeric entry
- Dimension mismatch
- Singular matrix detection
- Invalid user choices

---

6. Working of the Program

Step-by-step Flow:

1. User enters Matrix A.
2. User selects whether to calculate:
3. Inverse of A
4. Transpose of A
5. User chooses whether Matrix B is required.
6. If Matrix B is entered, user can perform:
7. Inverse of B
8. Transpose of B
9. Addition (A + B)
10. Subtraction (A − B)
11. Multiplication (A × B)
12. Identical matrix check
13. Program displays results accordingly.
14. Program ends with a thank-you message.

---

7. Code Structure Overview

The project is structured into modular functions:

a. input_matrix_function()

Handles reading matrix dimensions, entries, and validation.

b. inverse_function()

Computes inverse only for square, non-singular matrices.

c. transpose_function()

Returns the transpose of any valid matrix.

d. addition_function(), subtraction_function(), multiplication_function()

Perform basic binary matrix operations with dimension checks.

e. identical_matrix_check()

Checks whether two matrices are identical in both dimensions and values.

f. matrix_calculator()

Main controller function that coordinates all user inputs and operations.

---

8. Steps to Run the Project

1. Install Python 3.x
2. Install NumPy using:

   ```
   pip install numpy
   ```

3. Save the code as:

   ```
   matrix_calculator.py
   ```

4. Run the program using:

   ```
   python matrix_calculator.py
   ```

9. Testing the Project

To test the calculator, perform the following:

✔️Test with valid matrices

- Square matrices (e.g., 2×2, 3×3)
- Non-square matrices for transpose

✔️Test dimension mismatch

- Try addition with different-sized matrices
- Try multiplication with incompatible dimensions

✔️Test singular matrix

- Example: matrix with determinant = 0

✔️Test identical matrix functionality

- Enter same matrix twice
- Enter different matrices

10. Applications

- Educational tool for matrix algebra
- Python programming practice
- Lab experiments for engineering and computer science students
- Basic computational tool for matrix operations

11. Conclusion

The Matrix Calculator is a simple yet powerful Python project demonstrating the application of matrix operations using NumPy. With well-structured functions, interactive input, and robust validation, it serves as an excellent laboratory and learning tool. This project can be expanded further with GUI features, file handling, and advanced matrix operations.

12. Screenshots (Optional)

You may include: - Matrix input screenshot - Output of inverse/transpose - Addition/subtraction/multiplication results - Identical matrix check output

End of Project Report