

**Name: Ahmad Raza**

**Roll No: 093**

**import re**

* This line brings in Python's **regular expressions module**.
* It helps us search and replace patterns in the input (like adding \* between numbers and parentheses).

**import random**

* Imports Python’s **random number module**.
* ❗ You are **not using** it in this code — so it’s **not needed** unless you plan to generate random numbers later.

**print("This is Dynamic Calculator")**

* Prints a welcome message to the user.
* Let’s the user know they are using the calculator.

**while True:**

* Starts an **infinite loop**.
* This loop will keep running until the user types "exit".

**user\_input = input("Enter expression: ")**

* Shows a prompt asking the user to type a math expression (like 2+3\*4 or (5+3)/2).
* Stores what the user typed into the variable user\_input.

**if user\_input.lower() == "exit":**

* Checks if the user typed "exit" (in any case like Exit, EXIT, etc.)
* If yes, the loop stops using break.

**user\_input = user\_input.replace("×", "\*").replace("÷", "/")**

* Replaces the **multiplication (×)** and **division (÷)** symbols with Python-friendly symbols:
  + × becomes \*
  + ÷ becomes /
* This allows users to use common calculator symbols.

**user\_input = re.sub(r'(\d)(\ ()', r'\1\*(', user\_input)**

* This line fixes **implicit multiplication** like:
  + 2(3+4) → 2\*(3+4)
* It inserts a \* between a number and an opening parenthesis if missing.

**user\_input = user\_input.replace(")(", ")\*(")**

* Handles cases like (2+3)(4+5) which need a \* between the parentheses:
  + Changes to (2+3)\*(4+5)

**try:**

* Start of the **try block**. It tries to **evaluate** the expression.
* If the expression is valid, it goes to the next line.
* If something goes wrong (error), it skips to the except part.

**result = eval(user\_input)**

* This is where the expression is **calculated**.
* eval() runs the math expression entered by the user.
  + For example: "2+3\*4" becomes 2+3\*4 → 14

**print("Result:", result)**

* Shows the **result** of the calculation to the user.

**except Exception as e:**

* If the input was invalid (like typing 2+\*5), this catches the error.
* Prevents the program from crashing.

**print("Invalid Expression!")**

* Tells the user they entered something incorrect or not understandable.

