

Documentation for 2. IVS project

Bandasky
29.4.2026

Introduction	2
Installation and uninstallation	3
How to use the calculator	6
Conclusion	12

Introduction

Welcome to our Calculator Application. This calculator provides a user-friendly interface for both basic arithmetic and advanced mathematical operations.

Key features

This calculator is designed with a step-by-step calculation approach. It does not parse complete mathematical expressions like $5 + 3 * 2$, but works with intermediate results and processes one operation at a time. This means:

- You select an operation (addition, subtraction, multiplication, division, etc.) and then enter a number to perform this operation with
- The calculator immediately computes that operation and displays the intermediate result
- You then enter the next operand and select the next operation, building up your calculation step by step

More detailed description of calculator usage is shown in the *How to use* chapter

Supported operations

- Basic arithmetic: Addition (+), Subtraction (−), Multiplication (×), Division (÷)
- Advanced functions: Power (^), Root (√), Modulo (%), Factorial (!), and the constant Pi (π)
- Input controls: Decimal point support, sign toggle (+/−), and digit deletion

Installation and uninstallation

This project is intended to be used on **Ubuntu 24.04 64bit**. The application is distributed as two separate Debian packages:

- **ivs-calculator_1.0.0_all.deb** - graphical calculator application
- **ivs-profiling_1.0.0_all.deb** - command-line profiling utility for sample standard deviation

The calculator package installs a desktop application that can be started from the application menu or from the terminal. The profiling package installs a command-line utility intended to be used with standard input.

Installation

Before installation, make sure that **Python 3** is available on the system. The calculator package also requires the system package *python3-pyqt5*, which is installed automatically as a dependency when using *apt*.

Calculator

To install the calculator package, open a *bash* terminal in the project directory and run:

```
sudo apt install ./install/ivs-calculator_1.0.0_all.deb
```

After installation, the calculator can be started in two ways:

- from the application menu under ****IVS Calculator****
- from the *bash* terminal using:

```
ivs-calculator
```

Profiler

To install the profiling utility, run in *bash*:

```
sudo apt install ./install/ivs-profiling_1.0.0_all.deb
```

After installation, the profiling program can be executed from the *bash* terminal using either of these commands:

```
ivs-profiling
```

or

```
stddev
```

Example usage:

```
echo "1 2 3 4 5" | stddev
```

The program reads numbers from standard input and prints a single result – the sample standard deviation.

Uninstallation

To remove the calculator package, run in *bash*:

```
sudo apt remove ivs-calculator
```

To remove the profiling utility, run in *bash*:

```
sudo apt remove ivs-profiling
```

Removing the packages deletes the installed launchers and application files from the system.

Running from source code

The application can also be started directly from the source repository without using the Debian packages.

Requirements

For manual execution from source code on Ubuntu, install:

```
sudo apt install python3 python3-pip python3-venv python3-pyqt5
```

Running the calculator from source

From the root of the repository, the calculator can be started with:

```
PYTHONPATH=. python3 src/application.py
```

If preferred, a virtual environment can be created first:

```
python3 -m venv .venv
source .venv/bin/activate
pip install -r src/requirements.txt
PYTHONPATH=. python3 src/application.py
```

Running the profiling utility from source

From the root of the repository, the profiling program can be started with:

```
PYTHONPATH=. python3 src/stddev.py < profiling/data10.txt
```

Another example:

```
echo "1 2 3 4 5" | PYTHONPATH=. python3 src/stddev.py
```

Creating a desktop launcher manually

If the application is not installed from the Debian package, a desktop launcher can still be created manually.

First, create the required directories:

```
mkdir -p ~/.local/bin
mkdir -p ~/.local/share/applications
mkdir -p ~/.local/share/icons/hicolor/256x256/apps
```

Copy the calculator icon:

```
cp install/assets/ivs-calculator.png ~/.local/share/icons/hicolor/256x256/apps/ivs-
```

Create the launcher script `~/.local/bin/ivs-calculator` with the following content:

```
#!/usr/bin/env bash
set -euo pipefail
export PYTHONPATH="/absolute/path/to/your/repository"
exec python3 /absolute/path/to/your/repository/src/application.py "$@"
```

Then make it executable:

```
chmod +x ~/.local/bin/ivs-calculator
```

Create the desktop file `~/.local/share/applications/ivs-calculator.desktop` with the following content:

```
[Desktop Entry]
Type=Application
Name=IVS Calculator
Comment=Calculator for IVS project
Exec=/home/USERNAME/.local/bin/ivs-calculator
Icon=ivs-calculator
Terminal=false
Categories=Utility;Calculator;
```

Replace */absolute/path/to/your/repository* and *USERNAME* with your actual values.

After that, the application can be launched from the application menu like a standard desktop program.

Manual removal of source-based setup

If the program was set up manually from the source code, remove the created files and directories:

```
rm -f ~/.local/bin/ivs-calculator
rm -f ~/.local/share/applications/ivs-calculator.desktop
rm -f ~/.local/share/icons/hicolor/256x256/apps/ivs-calculator.png
rm -rf .venv
```

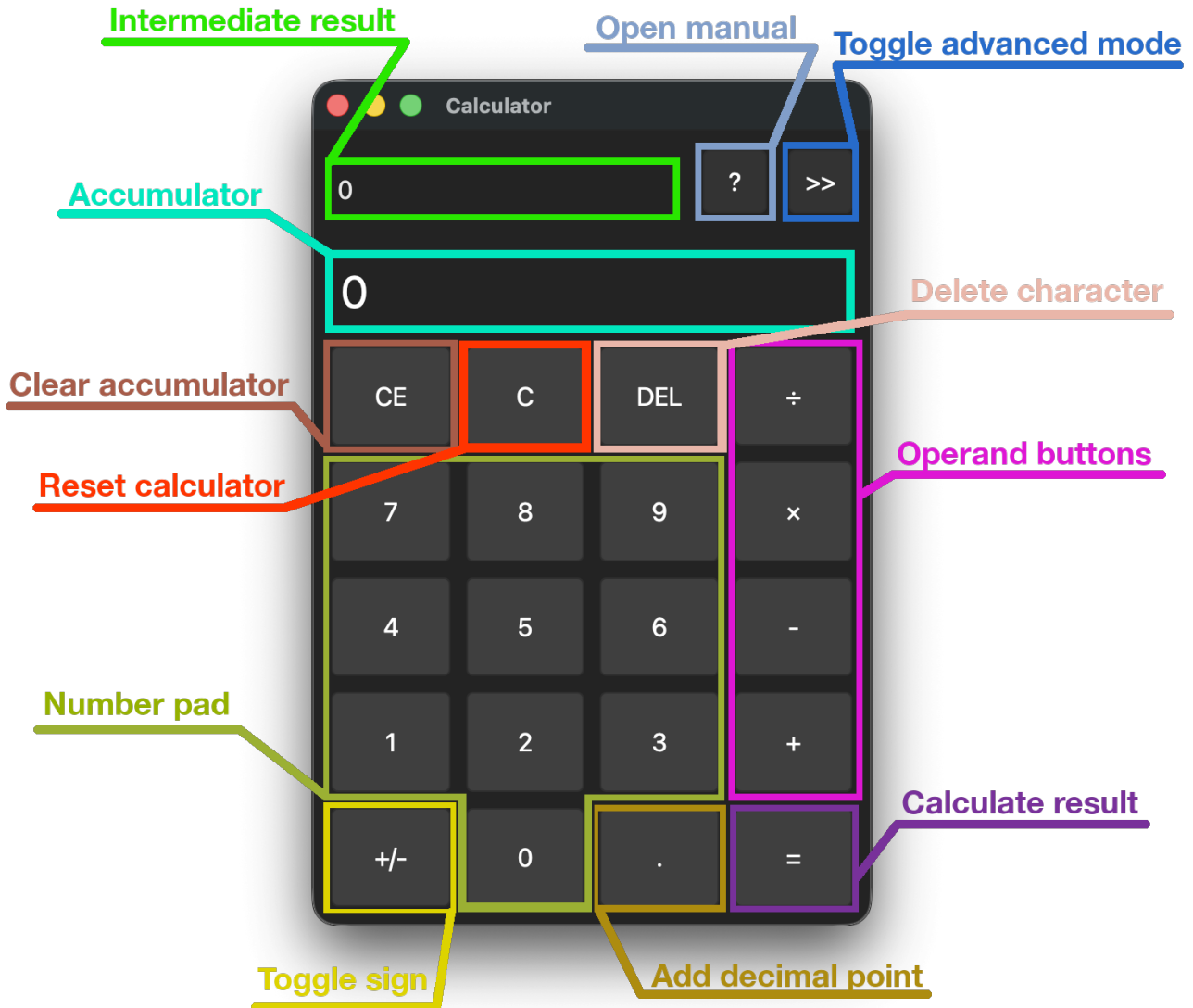
This removes the manually created launcher, desktop entry, icon, and optional virtual environment.

How to use the calculator

Simple Layout

Upon opening the calculator app, the user will be presented with simple layout mode. Here, basic operations can be performed, like addition, subtraction, multiply and divide.

Calculator buttons diagram



Accumulator

The accumulator is the primary display area where you see your current input. It consists of up to three components that change dynamically based on your actions:

1. **Operand** - The arithmetic operation pending from your last button press (+, -, ×, ÷, ^, √, %)
2. **Sign** - Indicates whether your current number is positive or negative (Omitted for positive numbers). Can be toggled with the *Toggle Sign Button*.
3. **Number** - The digits you are entering.

Intermediate result

The intermediate summary label displays the cumulative result of all operations performed so far. In each step, it is modified with the accumulator based on the selected operation.

Normal State: Shows the running result in regular text (light gray color)

- Updates after each operation you perform
- Reflects the result of the previous calculation step

Highlighted State (After Pressing =): When you press the equals button (=), the result appears with an orange background and black text

- This visual highlight signals an important state change
- It warns you that the next number you enter will start a fresh calculation, not continue the current one
- The highlight persists until you press another operation button, after which it will be replaced by entered number

Open manual

Clicking the ? button will open this user manual (manual.pdf) in your default PDF viewer.

Number pad

Number buttons (**0** through **9**). Press to enter digits into the calculator. You can build multi-digit numbers by pressing the buttons in sequence.

Add decimal point

Decimal point button (.). Press to add a decimal point to your input, allowing you to enter fractional numbers (e.g., 3.14). Only one decimal point can be added per number.

Toggle sign

Sign Toggle button (+/-). Changes the sign of the current input between positive and negative. When used with addition or subtraction, it switches between + and – operations.

Operand buttons

- ÷ Division button. Press to divide the intermediate result by accumulator. Returns NaN if dividing by zero.
- × Multiplication button. Press to multiply the intermediate result by accumulator.
- Subtraction button. Press to subtract accumulator from the intermediate result.
- + Addition button. Press to add the intermediate result and accumulator.

Pressing them performs the pending operation and selects the operand for new operation.

Calculate result

Equals button (=). Executes the pending operation and displays the final result.

Clear accumulator

Clear entry button (**CE**). Clears the accumulator (sets it to +0) without changing the selected operand and intermediate result.

Reset calculator

Clear All button (**C**). Completely resets the calculator, clearing the input, result and pending operation.

Delete character

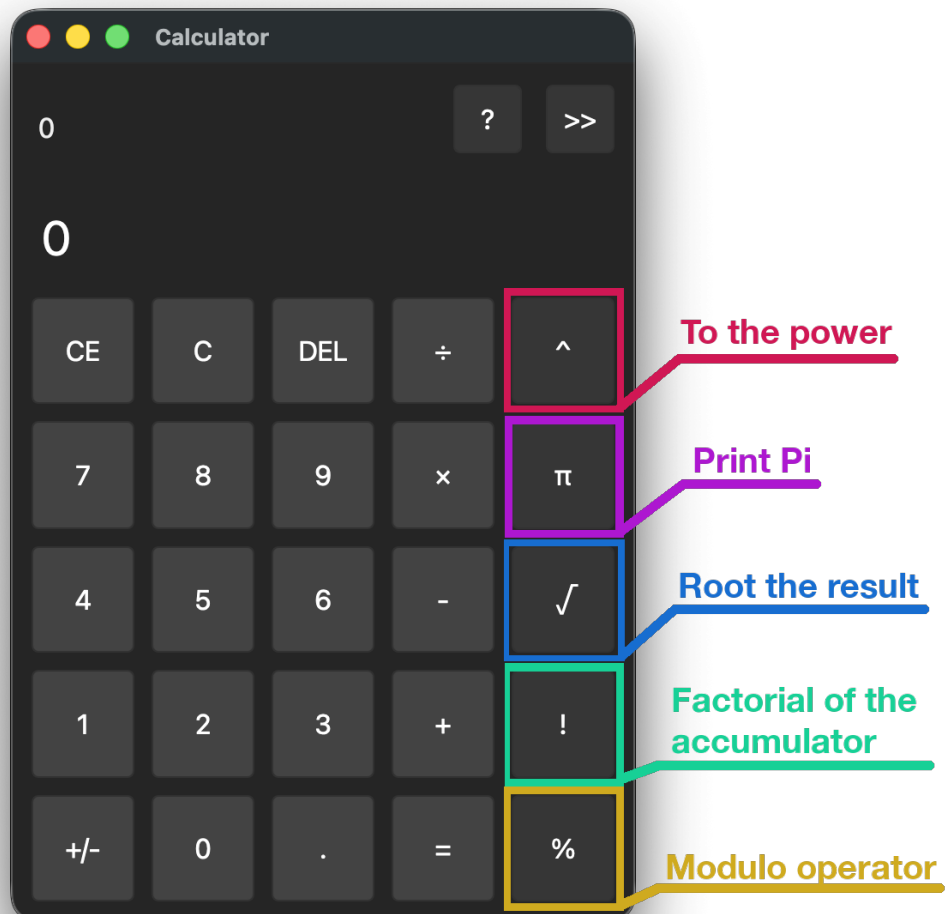
Delete button (**DEL**). Removes the last digit (or decimal point) you entered, allowing you to correct typing mistakes.

Toggle advanced mode

Toggle Extra Functions (**>>**). Press to show or hide the panel containing advanced mathematical functions (Power, Root, Modulo, Factorial, and Pi).

Advanced Layout

When clicking the toggle advance mode button, button will shrink to make room for new column with advance mathematical operations buttons.



To the power

Power button (^). Operator, that will output intermediate result to the power of accumulator.

Print Pi

Pi button (π). Replaces accumulator number with mathematical constant π . (it will leave operator and sign intact)

Root the result

Root button ($\sqrt{}$). It will root the intermediate result to the degree of accumulator. (for example: intermediate result is 4, accumulator is $\sqrt{2}$. Pressing = will result in 2.). For invalid numbers it return *NaN*.

Factorial of the accumulator

Factorial button (!). Takes accumulator and replaces it with it's factorial (if accumulator is 3, it will replace it with 6, it will leave operator and sign intact). If accumulator is a whole number, it will round it down first. If resulting factorial is too big, it will return *inf*.

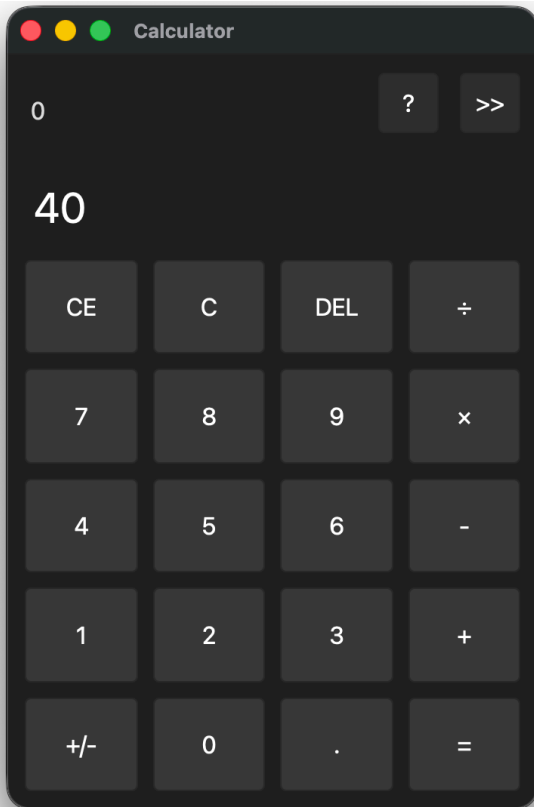
Modulo operator

Modulo button (%). Calculates the remainder after division. Returns *NaN* if dividing be zero.

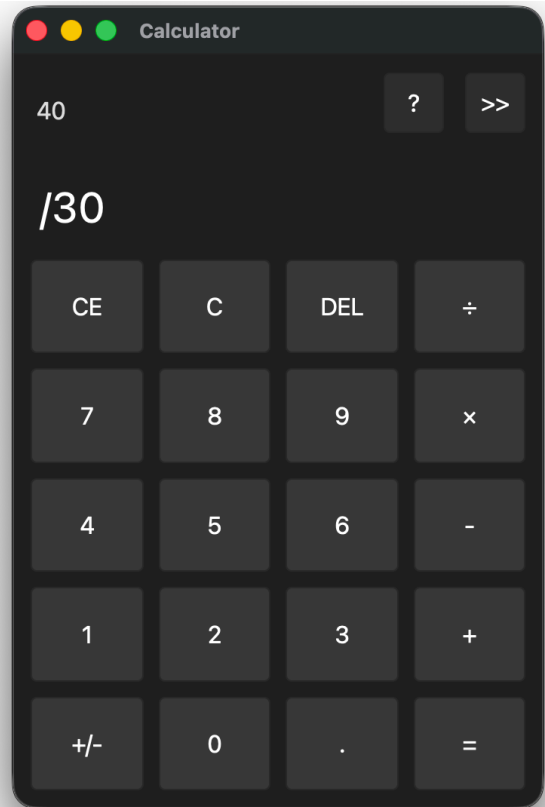
Sample simple workflow

Calculating $40 / -30$.

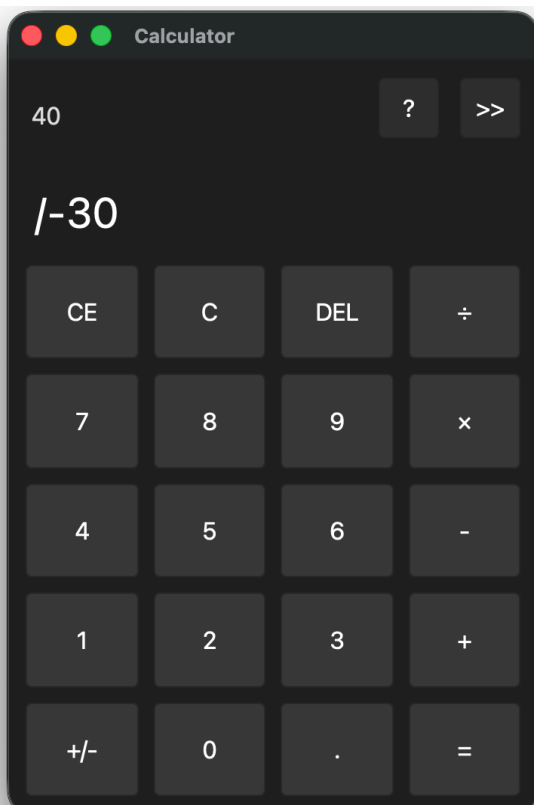
Typed 40 on the keypad



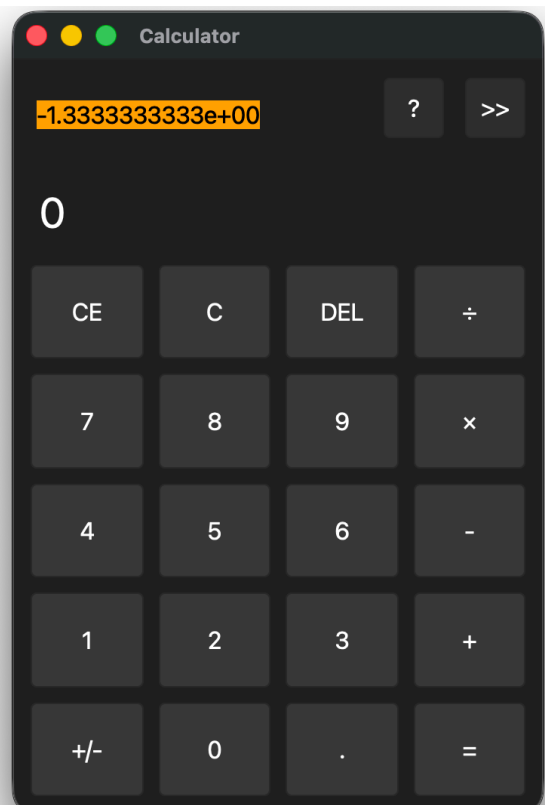
Used division operator and typed 30



Toggled the sign



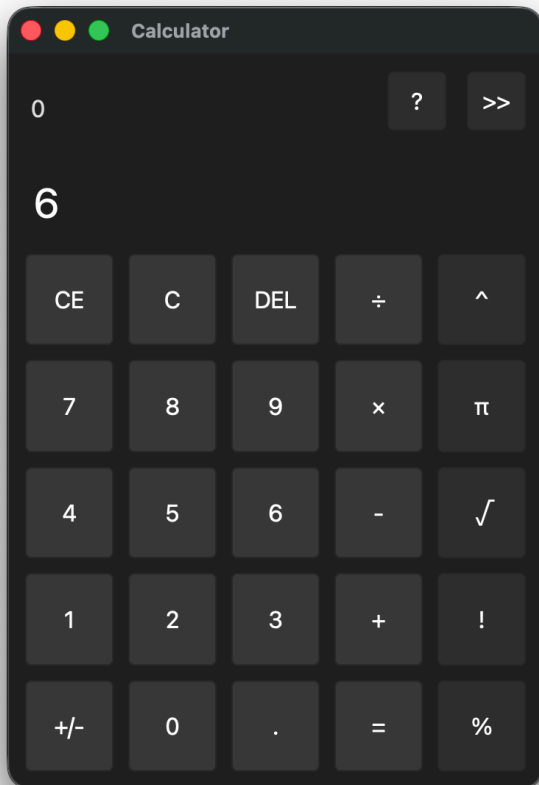
Pressed the result button



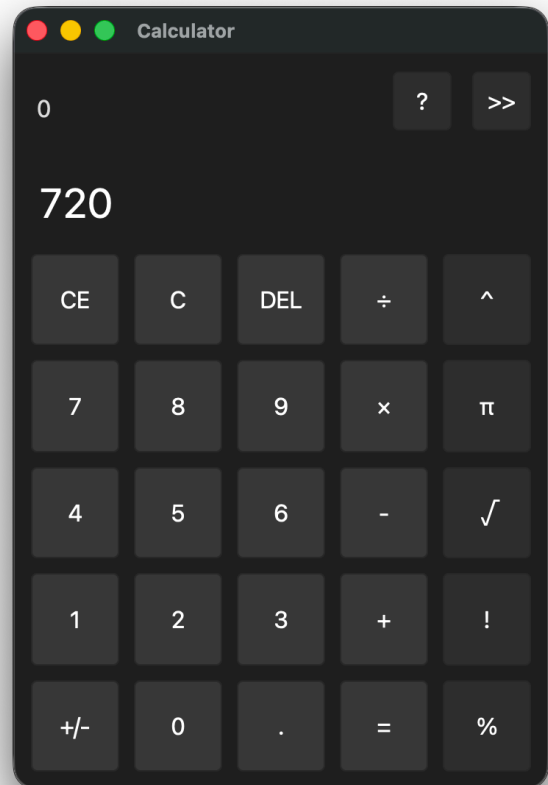
Sample advanced workflow

Calculating $\sqrt{(6!)}$.

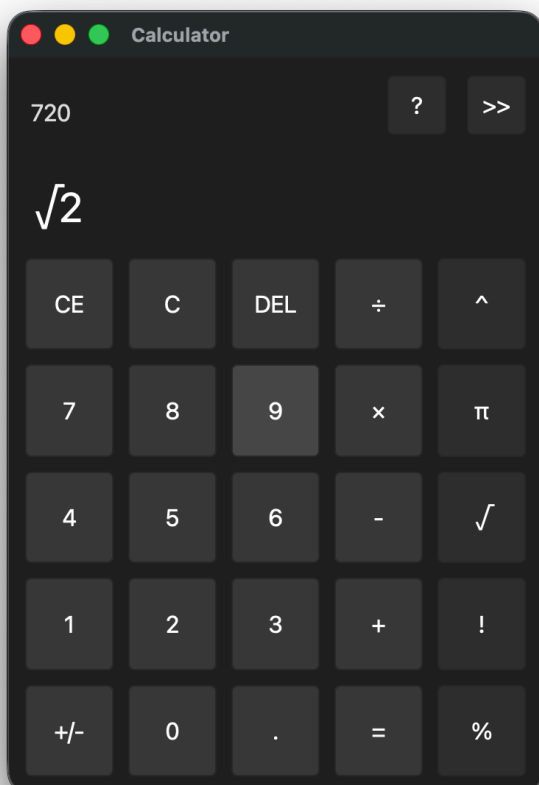
Typed 6 on the keypad



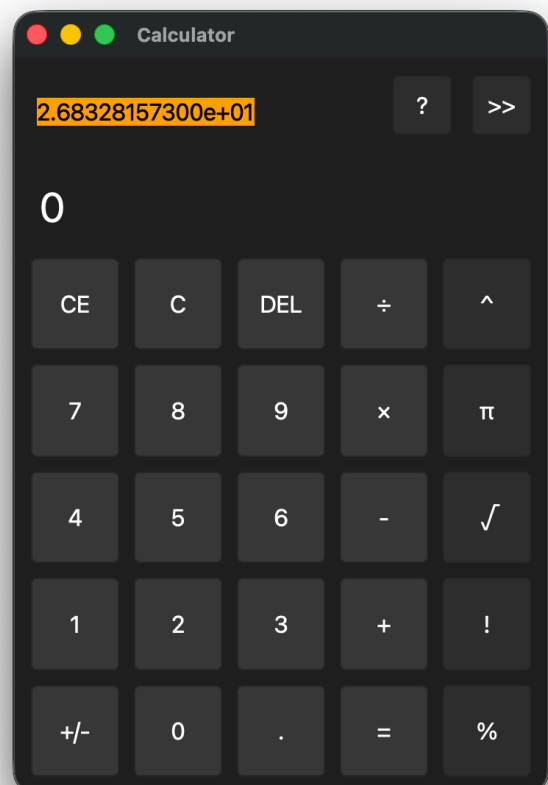
Pressed the ! button



Used the root operator and typed 2



Pressed the result button



Conclusion

Making the Most of Your Calculator

The application can be also entirely controlled with the keyboard. (numbers, operations, and special keys like Enter for equals, Backspace for delete). Pay attention to the orange highlight after pressing equals—it's your signal that the next input starts fresh.

Troubleshooting Tips

Invalid operations (like dividing by zero or taking the root of a negative number) return *NaN* (Not a Number). Very large results automatically switch to scientific notation to keep the display readable. Large factorials (> 9999) return *inf* to preserve system resources. Use CE (Clear Entry) to correct just the current input, or C (Clear All) to start completely fresh.

Getting Help

If you need additional guidance while using the calculator:

- Press the ? button (Manual) to open the complete user manual
- Press F1 on your keyboard for quick access to help
- Refer back to the Button Guide section if you forget what a particular button does