Python Developer Internship — Task 1 (Calculator CLI)

Reference: Task brief provided by course.

Hints / Mini Guide

1. Use functions for each operation (+, -, *, /)
2. Take user input using input()
3. Loop until user chooses to exit

Objective & Tools

Objective: Create a command-line calculator supporting basic operations.
Tools: Python 3.x, any text editor (VS Code recommended), terminal.
Deliverable: A Python script named calculator.py

Sample calculator.py (copy, modify, run)

```python
# calculator.py - Simple CLI calculator

def add(a, b):
    return a + b

def sub(a, b):
    return a - b

def mul(a, b):
    return a * b

def div(a, b):
    if b == 0:
        return "Error: Division by zero"
    return a / b

def get_number(prompt):
    while True:
        try:
            return float(input(prompt))
        except ValueError:
            print("Please enter a valid number.")

def main():
    print("Simple Calculator - enter 'q' at any time to quit")
    while True:
        print("\nOperations:")
        print("1) Add (+)")
        print("2) Subtract (-)")
        print("3) Multiply (*)")
        print("4) Divide (/)")
        print("5) Exit")
        choice = input("Choose an option (1-5): ").strip()
        if choice.lower() == 'q' or choice == '5':
            print("Goodbye!")
            break
        if choice not in ('1','2','3','4'):
            print("Invalid choice, try again.")
            continue
        a = get_number("Enter first number: ")
        b = get_number("Enter second number: ")

        if choice == '1':
            print("Result:", add(a,b))
        elif choice == '2':
            print("Result:", sub(a,b))
        elif choice == '3':
            print("Result:", mul(a,b))
        elif choice == '4':
            print("Result:", div(a,b))

if __name__ == '__main__':
    main()
```

How to run

```
1. Save the file as calculator.py
2. Open terminal and run:
   python3 calculator.py
3. Follow the on-screen menu. Press 5 or 'q' to exit.
```

Recommended repository structure

```
python-task-1/
■■ calculator.py
■■ README.md
■■ screenshots/
   ■■ run_terminal.png
```

README.md template (paste into your repo)

~~~
# Python Task 1 — Calculator CLI

**Student:** Tarun Kusumba
**Task:** Build a simple command-line calculator in Python.
**Deliverable:** calculator.py

## What I implemented
- Functions for add, subtract, multiply, and divide.
- Input validation for numeric input.
- Menu loop to continue until user exits.

## How to run
```
python3 calculator.py
```

## Files included
- `calculator.py` — main script
- `screenshots/` — optional run screenshots

## Notes
Replace this section with any extra features you added (e.g., history, power, modulo).
~~~

Interview Questions (from task)

1. What is normalization?
2. Explain primary vs foreign key.
3. What are constraints?
4. What is a surrogate key?
5. How do you avoid data redundancy?
6. What is ER diagram?
7. What are the types of relationships in DBMS?
8. Explain the purpose of AUTO_INCREMENT.
9. What is the default storage engine in MySQL?
10. What is a composite key?

Submission

1. Create a GitHub repository (e.g., python-task-1)
2. Add calculator.py, README.md, screenshots
3. Push and paste repository link into the submission form:
   https://forms.gle/8Gm83s53KbyXs3Ne9

Notes & Tips

- Validate inputs to avoid crashes.
- Add comments and a short README for reviewers.
- If you add extra features (history, power, modulo), mention them in README.
- Test edge cases such as division by zero and non-numeric input.

Sample Output Formats (for README)

```
Example Results Section:

Hosts/Results: (Replace with your findings)
- Script executed and menu displayed correctly.
- Sample run:
  > Choose an option (1-5): 1
  > Enter first number: 12
  > Enter second number: 8
  > Result: 20.0

Include terminal screenshots in /screenshots.
```