🕑 Hour 2: Variables, Types, and the Hacker Calculator

🎯 Objectives

- Learn how to store and manipulate data
- Understand basic types and conversions
- Practice arithmetic operations
- Explore dynamic code execution and its risks

🧱 Topics Covered

- `input()`, `int()`, `float()`, `str()`
- Variable assignment
- Arithmetic operators (`+`, `-`, `*`, `/`)
- `type()` inspection
- `eval()` and its dangers

🧪 Activities

1. Simple Calculator

```
a = float(input("Enter first number: "))
b = float(input("Enter second number: "))
print("Sum:", a + b)
print("Difference:", a - b)
print("Product:", a * b)
print("Quotient:", a / b)
```

✅ *Checkpoint:* What happens if you enter a word instead of a number? How could you prevent that?

2. Eval-Based Expression Runner

```
expr = input("Enter a math expression: ")
result = eval(expr)
print("Result:", result)
```

✅ *Checkpoint:* What does `eval()` do? Try entering `3 + 4`, then `__import__('os').system('echo HACKED')`. What's the risk?

3. Log File With a Mysterious Line

```
print("2025-10-24 23:51:02 INFO User login: eve")
print("2025-10-24 23:51:11 INFO User eve opening email\033[30;40mDEBUG
Injected payload: -- pretend eve runs a dangerous command here -- \033[0m")
print("2025-10-24 23:51:15 INFO Session terminated")
```

✅ *Checkpoint:* At your job, you find these three log messages for a session by a user named eve.

- Can you figure out what all of those strange characters at the beginning and end of her log message are for?
- If this log were printed in the way Eve expects, critical information would be hidden. How do we make sure that hackers can't use tricks like this to fool us?