Day 5: File Handling & Errors

- ☐ Reading/writing files
- ☐ Context managers (with)
- ☐ Exceptions (try-except-finally)
- ☐ Custom exceptions

✅ 1. Reading and Writing Files

🔹 Writing to a File

```
with open("example.txt", "w") as f:
  f.write("Hello, this is Day 5.\n")
  f.write("You're learning file handling!")
```

🔹 Reading a File

```
with open("example.txt", "r") as f:
  content = f.read()
  print(content)
```

🔹 Appending to a File

```
with open("example.txt", "a") as f:
  f.write("\nAppended line.")
```

✅ 2. Reading Line by Line

```
with open("example.txt", "r") as f:
  for line in f:
    print(line.strip())
```

✅ 3. Exception Handling (Try/Except)

```
try:
  x = int(input("Enter a number: "))
  print(10 / x)
except ZeroDivisionError:
  print("You can't divide by zero.")
except ValueError:
  print("That's not a valid number.")
finally:
  print("Done.")
```

✅ 4. Custom Exceptions (Optional)

```
class TooShortNameError(Exception):
    pass


name = input("Enter name: ")
if len(name) < 3:
    raise TooShortNameError("Name is too short!")
```

✅ 5. Context Managers (with)

The with statement automatically closes the file:

```
with open("data.txt", "w") as file:
    file.write("Safe and clean file writing.")
```

🧠 Mini Exercises:

1. Write a program to write a list of 5 names to a file, one per line.
2. Read the same file and print only names that start with A.
3. Use try-except to catch file not found errors when reading a file.
4. Use a with block to safely read and print all lines in a file.