

Day 6: File Handling & Error Management

File handling and error management are essential aspects of programming that allow programs to interact with external data and handle unexpected situations safely.

1. File Operations

Files are used to store data permanently. Programs can perform various operations on files, such as reading, writing, and closing them.

Key File Operations:

- **Open:** Opens a file for reading, writing, or appending.
- **Read:** Retrieves data from a file.
- **Write:** Adds data to a file.
- **Close:** Closes the file and releases system resources.

Proper file management ensures data integrity and prevents memory leaks.

2. File Modes

File modes determine how a file is accessed. Common file modes include:

- '**r**' (**read**): Opens a file for reading.
- '**w**' (**write**): Opens a file for writing. Existing content is overwritten.
- '**a**' (**append**): Adds data to the end of the file without removing existing content.
- '**r+**' (**read and write**): Allows both reading and writing to a file.
- '**b**' (**binary**): Opens a file in binary mode for non-text data like images or audio.

3. Context Managers

Context managers provide a **safe way to handle files**.

- Using the **with statement**, files are automatically closed after operations are completed, even if an error occurs.

- This ensures better resource management and prevents file corruption.

4. JSON Handling

JSON (JavaScript Object Notation) is a standard format for exchanging data.

Key JSON Operations:

- **Parsing JSON from files:** Reads JSON data from a file and converts it into a usable data structure.
- **Writing JSON to files:** Converts data into JSON format and saves it to a file.
- **Parsing JSON strings:** Converts a JSON-formatted string into a program-friendly data structure.
- **Generating JSON strings:** Converts data structures into a JSON-formatted string for storage or transmission.

JSON is widely used for configuration files, APIs, and data interchange.

5. CSV Files

CSV (Comma-Separated Values) files are used to store tabular data.

Key Points:

- CSV files store data in rows and columns using a delimiter (usually a comma).
- Programs can read and write CSV files using specialized tools or libraries.
- CSV files are simple, lightweight, and compatible with spreadsheets and databases.

Proper CSV handling ensures accurate data import/export between systems.

6. Exception Handling

Exceptions are runtime errors that disrupt normal program flow.

Key Concepts:

- **try block:** Contains code that might raise an exception.
- **except block:** Handles the exception if it occurs.
- **finally block:** Executes code regardless of whether an exception occurred, often used for cleanup.
- **raise statement:** Manually triggers an exception when certain conditions are met.

Exception handling improves program reliability and prevents crashes.

7. Custom Exceptions

Sometimes built-in exceptions are insufficient. Programs can define **custom exceptions** for specific error conditions.

Key Points:

- Custom exceptions are defined as new classes derived from the base exception class.
- They allow precise error handling tailored to the program's requirements.
- Using custom exceptions enhances code clarity and maintainability.