

Problem 1: Crime Case Management System

Scenario:

You are creating a program for a police department to manage crime cases. The data for each case is stored in a file named `cases.json` in dictionary format.

Initial File Format (`cases.json`):

```
{
  "C101": {"Title": "Bank Robbery", "Suspects": ["John Doe", "Jane Smith"], "Status": "Open"},
  "C102": {"Title": "Hacking Incident", "Suspects": ["Alice Brown"], "Status": "Closed"},
  "C103": {"Title": "Hit and Run", "Suspects": [], "Status": "Open"}
}
```

Instructions:

Write a Python program that allows the user to:

1. View all cases:

Display all cases with details in a readable format.

Example:

vbnet

Copy code

Case ID: C101

Title: Bank Robbery

Suspects: John Doe, Jane Smith

Status: Open

○

2. Add a new case:

- Prompt the user to input a **Case ID**, **Title**, **Suspects** (comma-separated), and **Status**.
- Add the new case to the `cases.json` file.

3. Update the status of a case:

- Ask the user for a **Case ID** and the new status (**Open**, **Under Investigation**, or **Closed**).
- Update the status in the file.

4. Add suspects to a case:

- Allow the user to add one or more suspects to a specific case.
- Ensure duplicates are not added to the suspect list.

5. Search for open cases:

- Display all cases with a status of **Open**.

6. Delete a case:

- Allow the user to input a **Case ID** and remove it from the file.

7. Find cases involving a specific suspect:

- Ask the user to input a suspect's name and display all cases where that suspect is mentioned.

Bonus Requirements:

- Use Python's `json` module for file handling.
- Handle edge cases like invalid `Case IDs` or missing files gracefully.

Sample Output:

Choose an option:

1. View all cases
2. Add a new case
3. Update case status
4. Add suspects to a case
5. Search open cases
6. Delete a case
7. Find cases by suspect

Enter your choice: 7

Enter suspect name: John Doe

Cases involving John Doe:

1. Bank Robbery (C101)