* **CSV Handling Problems**
* **Modify Existing CSV File**: Write a script to read **example.csv**, add a new column named "Email", and populate it with email addresses for each person. Then, save the modified data back into a new CSV file named **updated\_example.csv**.
* **Filter and Save Specific Data**: Create a script to read **pandas\_example.csv**, filter out rows where the age is above 25, and save these rows into a new CSV file named **age\_filtered.csv**.
* **Merge Two CSV Files**: Assume you have another CSV file **additional\_data.csv** with columns "SN", "Hobby". Write a script to merge this file with **example.csv** based on the "SN" column and save it as **merged\_data.csv**.
* **CSV to JSON Converter**: Write a program that converts **pandas\_example.csv** into a JSON file, where each row is a JSON object. Save this as **data.json**.
* **Handling Custom Delimiter**: Write a script that reads a CSV file **custom\_delimiter.csv** which uses a semicolon (;) as a delimiter, and print out its contents in a tabular format.
* **JSON Handling Problems**
* **Update JSON File**: Write a script to read **data.json**, add a new key-value pair **"occupation": "Engineer"** to the existing data, and save the updated data back to the file.
* **JSON to CSV Converter**: Convert the contents of **data.json** to a CSV format and save it as **data\_from\_json.csv**.
* **Filtering JSON Data**: Write a script to read **data.json**, filter out entries where **"hasChildren"** is **False**, and print the filtered data.
* **Complex JSON Structure**: Create a JSON file **complex\_data.json** that contains an array of objects, each with nested objects and arrays. Write a script to read and print each element of this complex structure.
* **Custom Object to JSON**: Extend the **User** class with a new attribute **hobbies** (a list of strings). Modify the **encode\_user** function to include this new attribute when converting to JSON.