**Simple web Application**

Language Used: HTML, Bootstrap, Pandas, Flask

**Import necessary modules:**

Flask, render\_template, and request from the flask module.

pandas as pd for data manipulation).

csv for reading and writing CSV files.

**Create a new Flask web application:**

app = Flask(\_\_name\_\_,template\_folder= 'template') creates a new Flask web application and sets the template folder to 'template' which contains your HTML files.

**Define the route for the index page (/):**

@app.route('/') defines the URL for the index page of your web application.

def index(): is the function that's called when a user navigates to the index page.

The function reads the data from the 'pod-data.csv' file and converts it into a list of dictionaries.

It then renders the 'task.html' template and passes the data to it.

**Define the route for the search functionality:**

@app.route('/search', methods=['POST']) defines the URL for the search feature of your web application and allows POST requests.

def search(): is the function that's called when a user submits the search form.

It retrieves the search query from the submitted form.

It reads the data from the 'pod-data.csv' file, similar to the index function.

It then filters the data to include only the rows where the 'Customer' column contains the search query.

Finally, it renders the 'task.html' template with the filtered data.

**Run the application:**

if \_\_name\_\_ == '\_\_main\_\_': app.run(debug=True) runs the application in debug mode if the script is run directly (not imported as a module). Debug mode will provide more detailed error messages, and it will automatically reload the application whenever a change is made to the code.

So, this script creates a simple web application that displays data from a CSV file and allows users to search the data by 'Customer' field. It uses Flask for the web application and the Jinja2 templating engine for generating the HTML.

**Searching Using Job Type ‘Delivery’**

A screenshot of a data

Description automatically generated with medium confidence

**Searching Using Customer: E.g.: ‘Fox’**

A screenshot of a computer

Description automatically generated with medium confidence