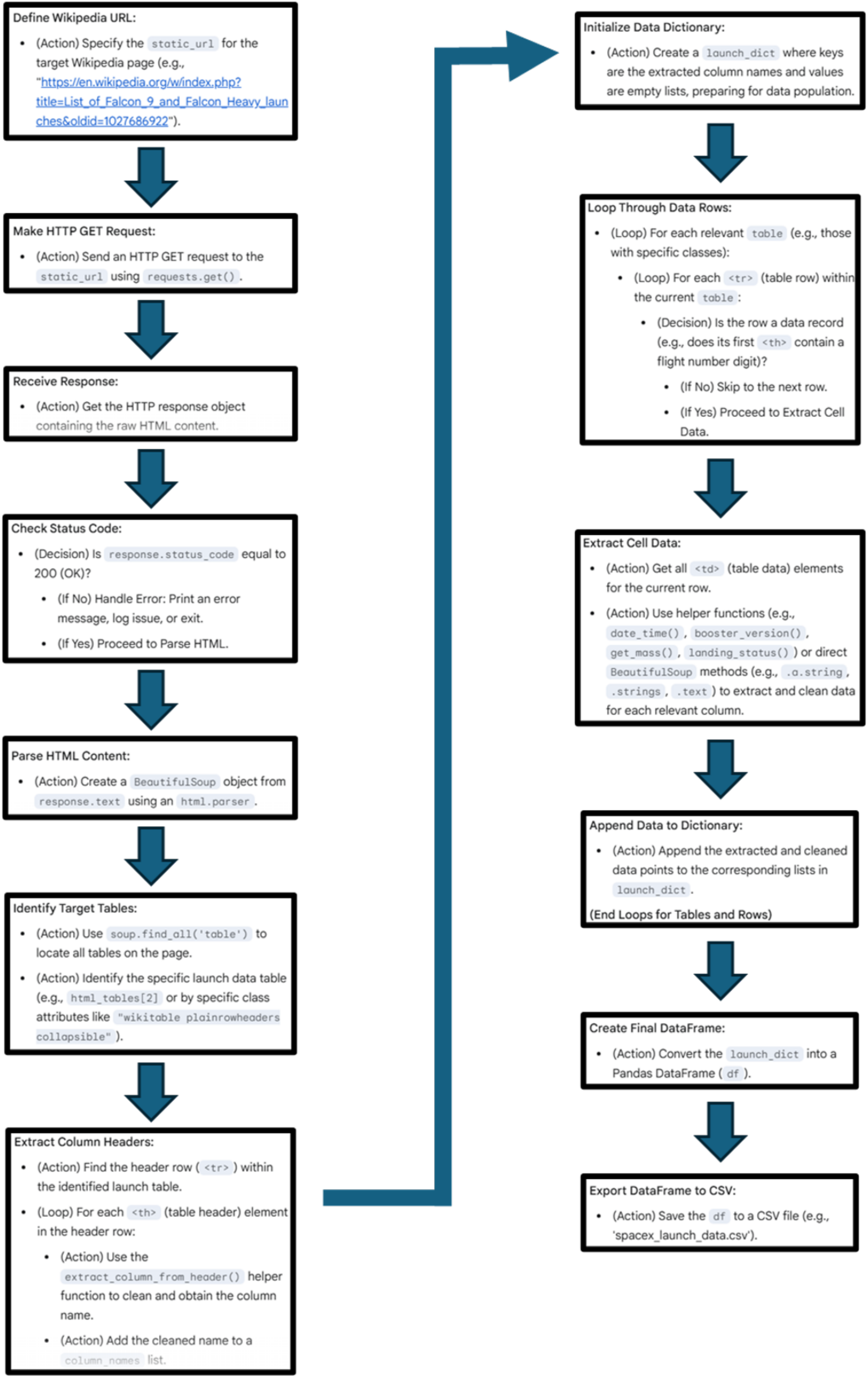

Define Wikipedia URL:

- (Action) Specify the `static_url` for the target Wikipedia page (e.g., "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922").

Make HTTP GET Request:

- (Action) Send an HTTP GET request to the `static_url` using `requests.get()`.

Receive Response:

- (Action) Get the HTTP response object containing the raw HTML content.

Check Status Code:

- (Decision) Is `response.status_code` equal to 200 (OK)?
  - (If No) Handle Error: Print an error message, log issue, or exit.
  - (If Yes) Proceed to Parse HTML.

Parse HTML Content:

- (Action) Create a `BeautifulSoup` object from `response.text` using an `html.parser`.

Identify Target Tables:

- (Action) Use `soup.find_all('table')` to locate all tables on the page.
- (Action) Identify the specific launch data table (e.g., `html_tables[2]` or by specific class attributes like `"wikitable plainrowheaders collapsible"`).

Extract Column Headers:

- (Action) Find the header row (`<tr>`) within the identified launch table.
- (Loop) For each `<th>` (table header) element in the header row:
  - (Action) Use the `extract_column_from_header()` helper function to clean and obtain the column name.
  - (Action) Add the cleaned name to a `column_names` list.

Initialize Data Dictionary:

- (Action) Create a `launch_dict` where keys are the extracted column names and values are empty lists, preparing for data population.

Loop Through Data Rows:

- (Loop) For each relevant `table` (e.g., those with specific classes):
  - (Loop) For each `<tr>` (table row) within the current `table`:
    - (Decision) Is the row a data record (e.g., does its first `<th>` contain a flight number digit)?
      - (If No) Skip to the next row.
      - (If Yes) Proceed to Extract Cell Data.

Extract Cell Data:

- (Action) Get all `<td>` (table data) elements for the current row.
- (Action) Use helper functions (e.g., `date_time()`, `booster_version()`, `get_mass()`, `landing_status()`) or direct `BeautifulSoup` methods (e.g., `.a.string`, `.strings`, `.text`) to extract and clean data for each relevant column.

Append Data to Dictionary:

- (Action) Append the extracted and cleaned data points to the corresponding lists in `launch_dict`.

(End Loops for Tables and Rows)

Create Final DataFrame:

- (Action) Convert the `launch_dict` into a Pandas DataFrame (`df`).

Export DataFrame to CSV:

- (Action) Save the `df` to a CSV file (e.g., 'spacex_launch_data.csv').