Final Project

Name: **Shivam Pabrekar**

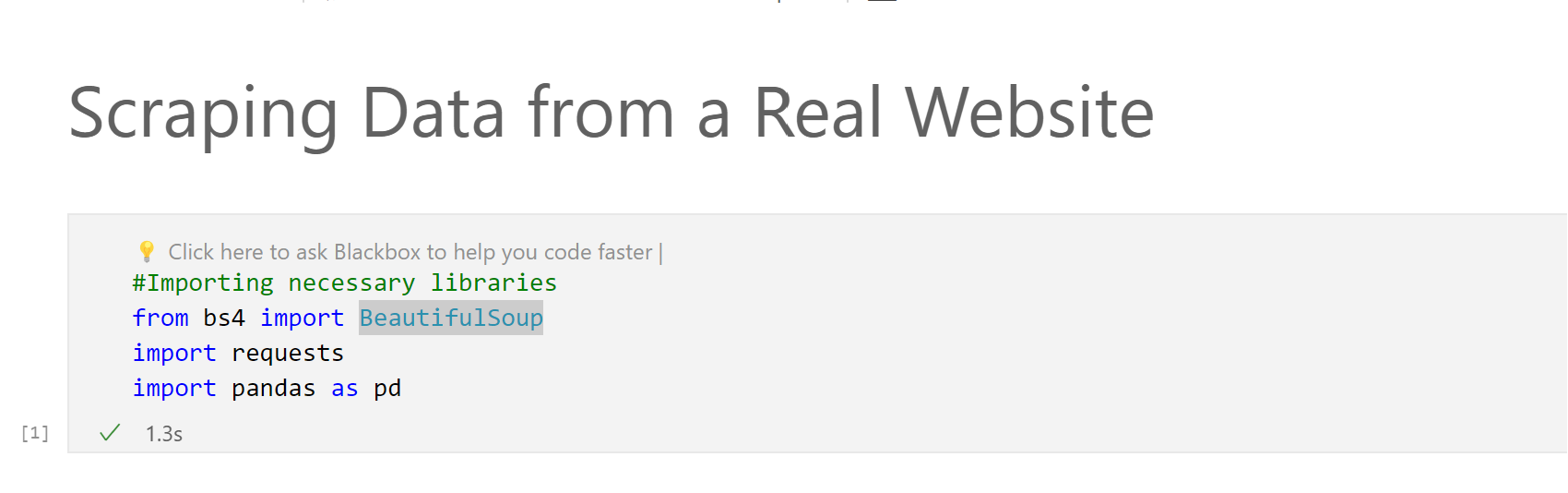
Internship: Python Programming

**Aim:** Work on Project name Web Scraper.

**Requirements:** PC/laptop, VS Code, Jupyter.

**Procedure:**

1. Importing necessary libraries:



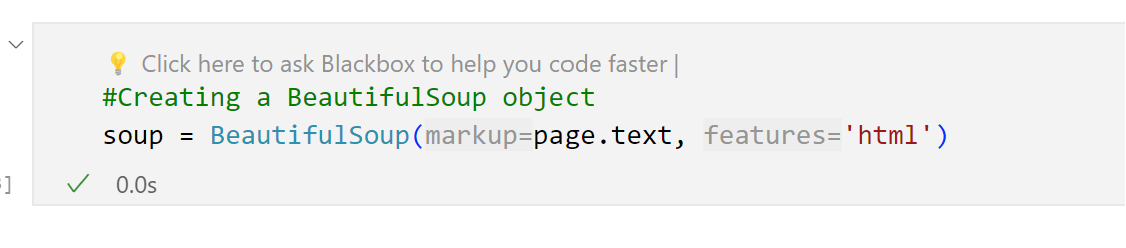
* **BeautifulSoup** is a library for pulling data out of HTML and XML files.
* **requests** is used for making HTTP requests to get the HTML content of a webpage.
* **pandas** is a data manipulation and analysis library.

1. Defining the URL and making an HTTP request:



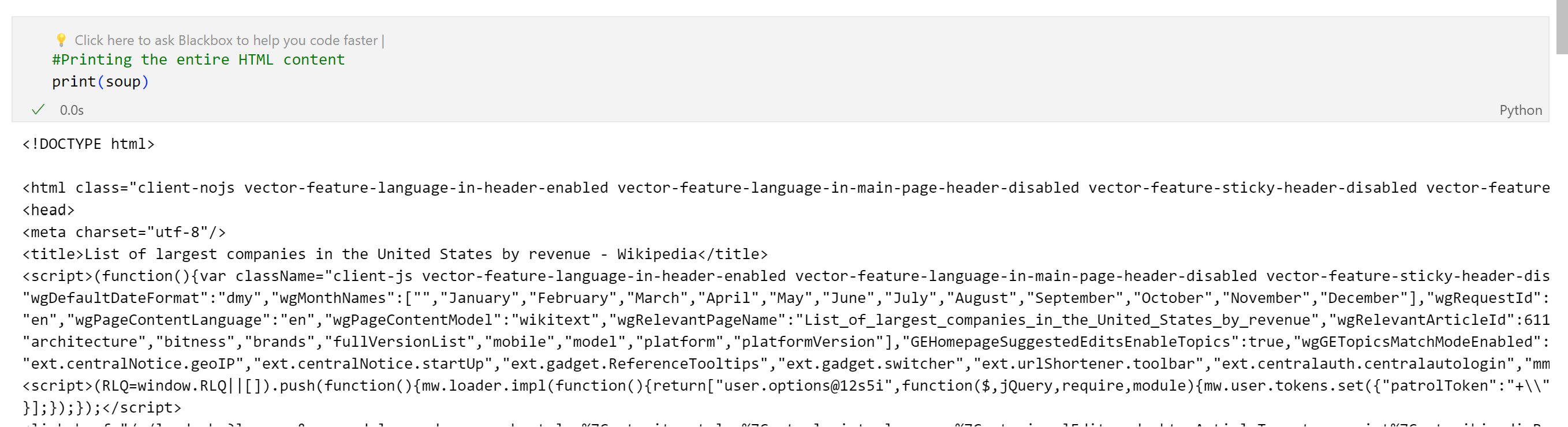
* **url** is the link to the Wikipedia page listing the largest companies in the United States by revenue.
* **requests.get(url)** sends an HTTP GET request to the specified URL and stores the HTML content in the **page** variable.

1. Creating a BeautifulSoup object:



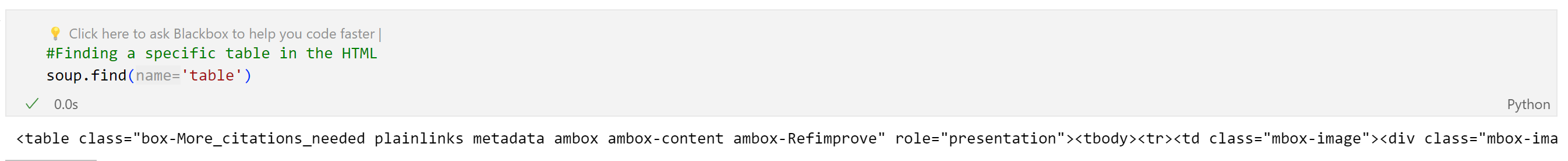
* **BeautifulSoup(page.text, 'html')** creates a BeautifulSoup object **(soup)** from the HTML content of the webpage.

1. Printing the entire HTML content:



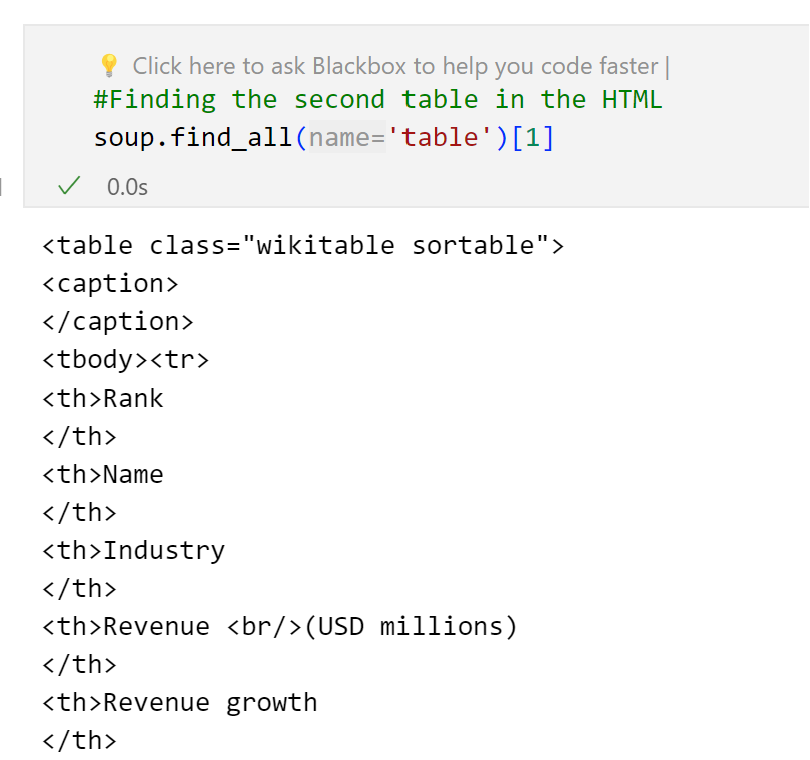
* This prints the entire HTML content of the webpage. It helps you inspect the structure of the HTML to locate elements you want to scrape.

1. Finding a specific table in the HTML:



* This finds the first **<table>** element in the HTML. It's a way to locate the table containing the data of interest.

1. Finding the second table in the HTML:



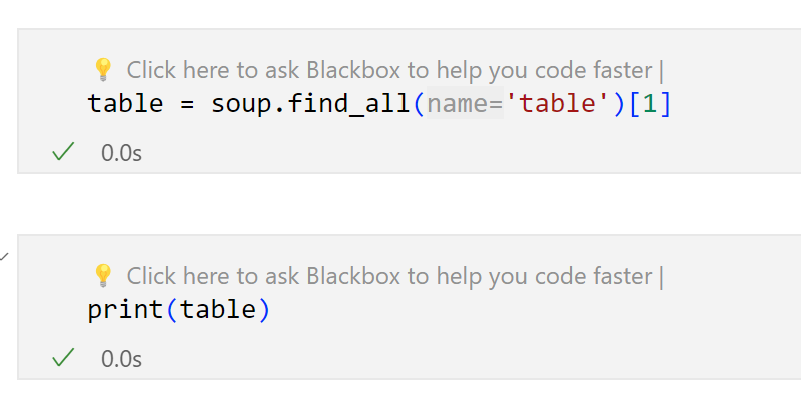
* This finds all **<table>** elements in the HTML and selects the second one (index 1).

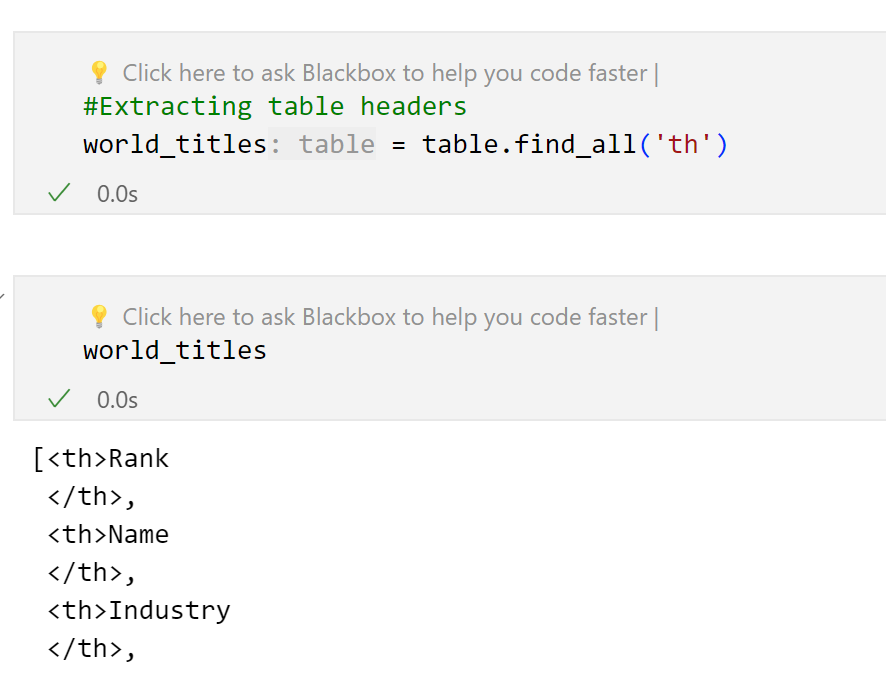
1. Finding a table with a specific class:



* This finds a **<table>** element with the specified class **(wikitable sortable).**

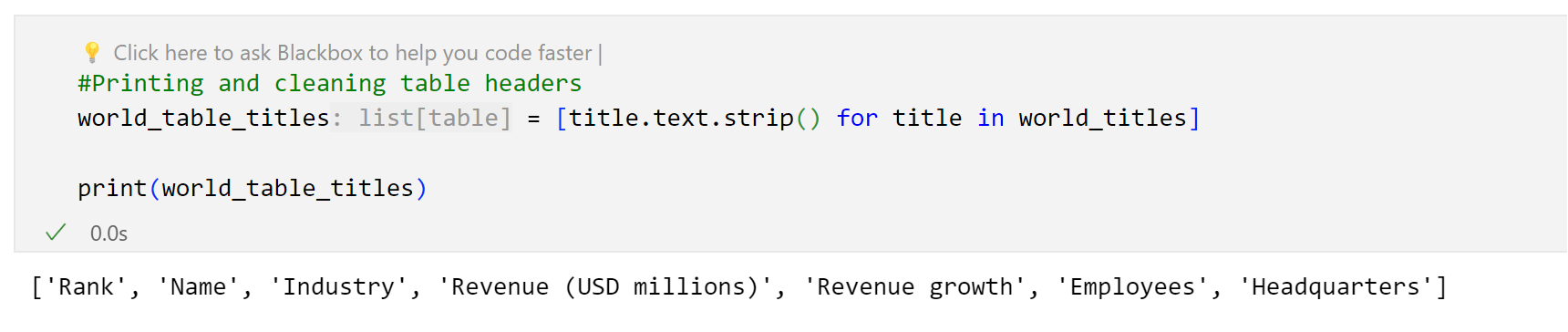
1. Extracting table headers:





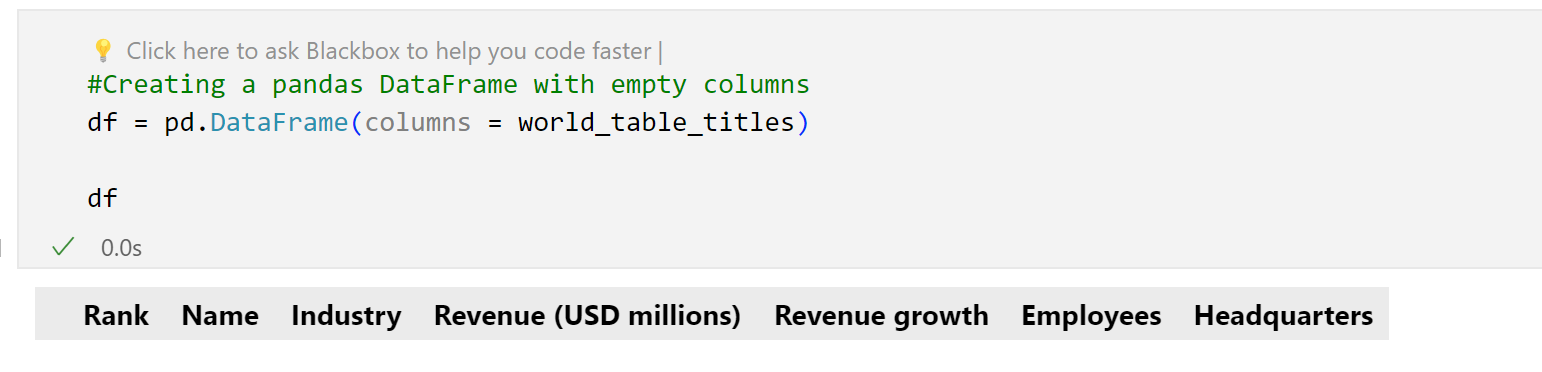
* This finds all **<th>** (table header) elements within the selected table.

1. Printing and cleaning table headers:



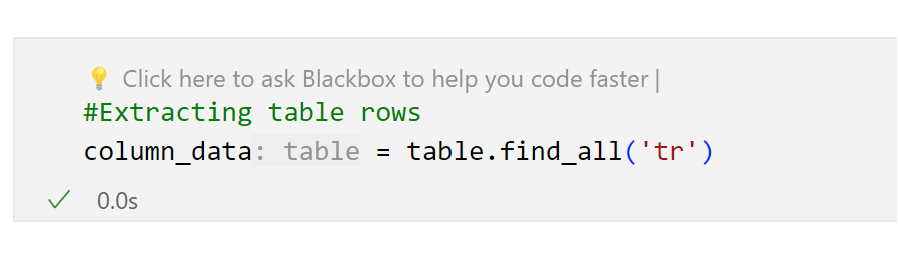
* This extracts the text content of each table header, removes leading and trailing whitespaces, and prints the cleaned table headers.

1. Creating a pandas DataFrame with empty columns:



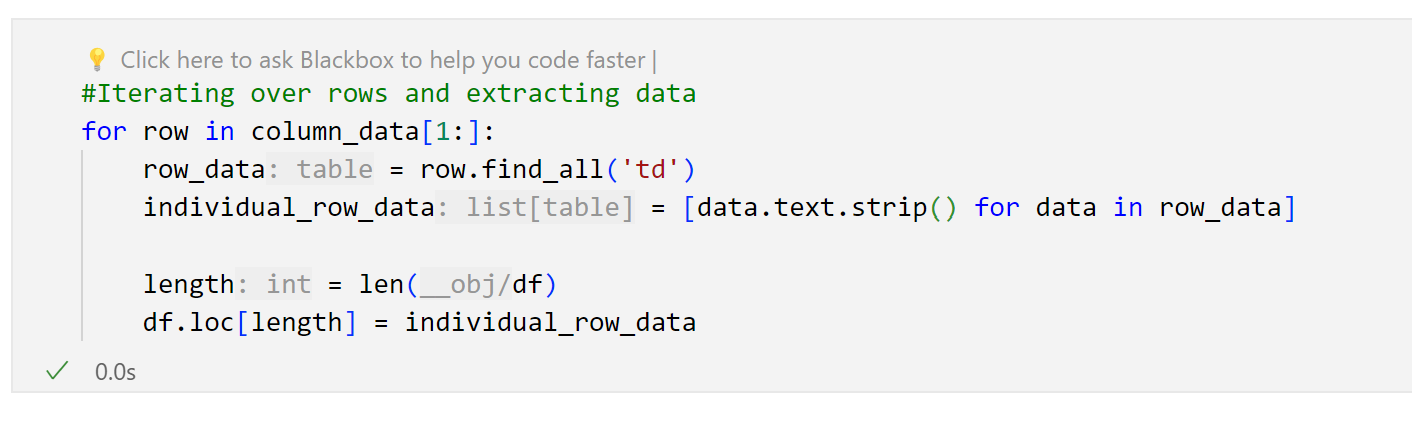
* This initializes an empty pandas DataFrame with columns named after the cleaned table headers.

1. Extracting table rows:



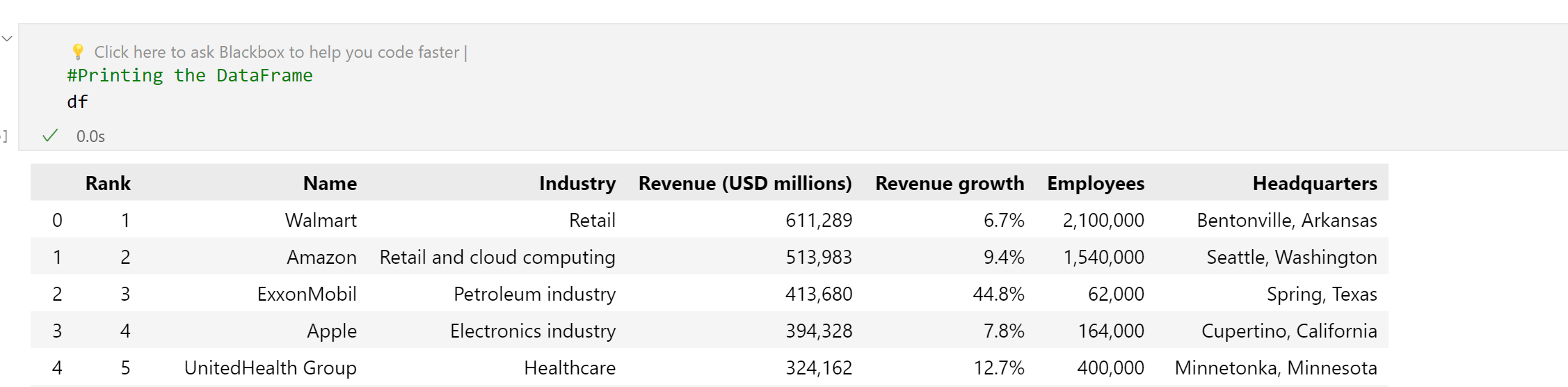
* This finds all **<tr>** (table row) elements within the selected table.

1. Iterating over rows and extracting data:



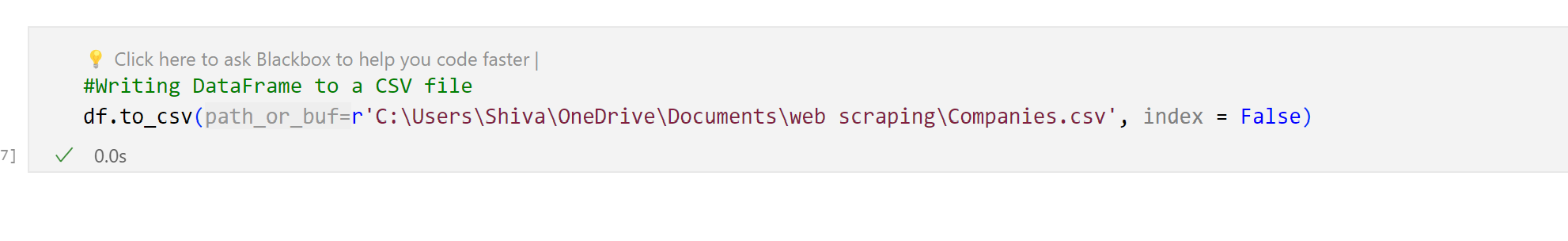
* This iterates over each table row, extracts the text content of each **<td>** (table data) element, removes leading and trailing whitespaces, and adds the data to the pandas DataFrame.

1. Printing the DataFrame:



* This prints the final pandas DataFrame containing the scraped data.

1. Writing DataFrame to a CSV file:



* This writes the DataFrame to a CSV file at the specified file path. The **index=False** parameter prevents writing row indices to the CSV file.

Output:

