**FieldMapper Class**

**Overview**

* Purpose: Reads data from Excel or CSV files, determines its source (Rival IQ or Phantom Buster), maps fields accordingly, and uploads the mapped data to MongoDB.
* Key Features: Handles multiple file formats, identifies data sources, applies appropriate field mapping rules, and integrates with MongoDB.

**Class Methods**

**init(self, file\_path)**

* **Parameters:**
  + file\_path (str): Path to the input Excel or CSV file.
* **Description:**
  + Initializes the class instance and stores the file path.

**read\_file(self) -> pd.DataFrame**

* **Returns:**
  + pd.DataFrame: A Pandas DataFrame containing the data read from the input file.
* **Description:**
  + Uses Pandas to read the data from the specified file path, handling Excel or CSV formats.

**detect\_source(self, df) -> str**

* **Parameters:**
  + df (pd.DataFrame): Input DataFrame.
* **Returns:**
  + str: The identified data source ('Rival IQ' or 'Phantom Buster').
* **Raises:**
  + ValueError: If the source cannot be determined.
* **Description:**
  + Examines the DataFrame's columns to identify the source based on the presence of specific fields.

**map\_fields(self, df) -> pd.DataFrame**

* **Parameters:**
  + df (pd.DataFrame): Input DataFrame.
* **Returns:**
  + pd.DataFrame: A DataFrame with mapped fields.
* **Raises:**
  + RuntimeError: If an error occurs during field mapping.
* **Description:**
  + Applies field mapping rules based on the detected source, ensuring consistency and compatibility with MongoDB.

**upload\_to\_mongodb(self, database\_name, collection\_name) -> None**

* **Parameters:**
  + database\_name (str): Name of the MongoDB database.
  + collection\_name (str): Name of the MongoDB collection.
* **Raises:**
  + RuntimeError: If an error occurs during data upload.
* **Description:**
  + Connects to MongoDB and uploads the mapped data to the specified database and collection.

**Main Script**

**main()**

* **Description:**
  + Orchestrates the overall process:
    1. Reads the input file.
    2. Detects the source.
    3. Maps fields.
    4. Uploads data to MongoDB.

**Execution**

* Run the script: python your\_script\_name.py