**API DATA FETCH WORKFLOW**

**Doc Status:** In progress  
**Doc Type:** Technical Documentation  
**Contributor:** Archit Singhania  
**Last Updated:** November 25, 2024

### Task 1: Cricket API Data Fetching Project Documentation

**Project Overview**

* **Development Environment:**
  + Python 3.12.3
  + Jupyter Notebook (IPython Kernel)
* **Data Source:** [RapidAPI.com](https://rapidapi.com/" \t "/private/var/folders/s8/7qnqk0pj3z30tkk5jp4py3b40000gp/T/com.kingsoft.wpsoffice.mac.global/wps-archits/x/_new) (Cricbuzz Cricket API)

**Task Description**  
Fetch live cricket match data and display details such as match summary, scores, and statistics. Also fetches and cleans recent commentary data for key events like wickets, boundaries, etc.

**Libraries and Imports**

* **Core Libraries:**
  + requests: For API interactions
  + json: For parsing JSON responses
  + pandas: For tabular data representation

**API Interaction**

* **Platform:** [RapidAPI.com](https://rapidapi.com/" \t "/private/var/folders/s8/7qnqk0pj3z30tkk5jp4py3b40000gp/T/com.kingsoft.wpsoffice.mac.global/wps-archits/x/_new)
* **API Name:** Cricbuzz Cricket API
* **Endpoint:** https://cricbuzz-cricket.p.rapidapi.com/mcenter/v1/{match\_id}
* **Authentication:** RapidAPI Key Authentication

**Code Logic and Workflow**

1. setup\_api() **function:**
   * Configures API endpoint and headers
   * Sets authentication with API key and host
2. fetch\_data() **function:**
   * Sends GET request to the Cricbuzz API
   * Handles request errors gracefully
   * Returns the JSON response
3. process\_data() **function:**
   * Extracts details like match summary, teams, and scores
   * Structures data into a pandas.DataFrame for better readability
4. display\_results() **function:**
   * Calls fetch\_data() and process\_data()
   * Displays data in tabular format using pandas

**Data Processing Highlights**

* Clean data extraction for match details
* Structured output for analysis
* Effective error handling for incomplete data

**Sample Data Output**

**Match Summary:**

* + Teams: India vs. Australia
  + Status: India won by 6 wickets
  + Scores: India 255/4 (47.3 overs), Australia 252/8 (50 overs)

**Recent Commentary:**

* Ball 47.3: "FOUR! Rohit Sharma finishes the match in style."
* Ball 47.2: "Single taken. India one run away from victory."

**Potential Improvements**

1. Add logging for API responses
2. Export match data to JSON or CSV
3. Introduce visualizations for match performance

### Task 2: LinkedIn Data Fetching Project Documentation

**Project Overview**

**Development Environment:**

* + Python 3.12.3
  + Jupyter Notebook

**Data Source:** [RapidAPI.com](https://rapidapi.com/" \t "/private/var/folders/s8/7qnqk0pj3z30tkk5jp4py3b40000gp/T/com.kingsoft.wpsoffice.mac.global/wps-archits/x/_new) (LinkedIn API)

**Task Description**  
Retrieve LinkedIn profile data, including user details, skills, and connections, for further analysis.

**Libraries and Imports**

**Core Libraries:**

* + requests: For API interactions
  + json: For parsing JSON responses

**API Interaction**

* **Platform:** [RapidAPI.com](https://rapidapi.com/" \t "/private/var/folders/s8/7qnqk0pj3z30tkk5jp4py3b40000gp/T/com.kingsoft.wpsoffice.mac.global/wps-archits/x/_new)
* **API Name:** LinkedIn Profile API
* **Endpoint:** https://linkedin-api.p.rapidapi.com/v1/user/profile/{id}
* **Authentication:** RapidAPI Key Authentication

**Code Logic and Workflow**

1. setup\_api() **function:**
   * Configures API endpoint and headers
   * Sets API key and host for authentication
2. fetch\_data() **function:**
   * Sends a GET request to the API
   * Handles errors and missing data
   * Returns JSON response
3. process\_data() **function:**
   * Extracts key profile details like name, headline, and skills
   * Structures data for CSV export
4. export\_to\_csv() **function:**
   * Saves extracted data to a CSV file for further use

**Data Processing Highlights**

* Authentication handled using API key
* Structured extraction of LinkedIn profile attributes

**Sample Data Output**

**User Profile:**

* + Name: John Doe
  + Headline: Senior Software Engineer
  + Skills: Python, Machine Learning, Cloud Computing

### Error Handling

1. **API Response Validation:**

* Catches invalid status codes.
* Logs missing or null fields.

1. **Data Handling:**

* Uses fallback values for missing attributes (e.g., "N/A").

**Potential Improvements**

1. Add pagination support for large datasets
2. Include visualizations of skills and endorsements
3. Add functionality for batch processing of multiple profiles

### **Task 3: Stock Market News Fetching Project Documentation**

**Project Overview**

**Development Environment:**

* + Python 3.12.3
  + Jupyter Notebook

**Data Source:** [RapidAPI.com](https://rapidapi.com/" \t "/private/var/folders/s8/7qnqk0pj3z30tkk5jp4py3b40000gp/T/com.kingsoft.wpsoffice.mac.global/wps-archits/x/_new) (Stock Market News API)

**Task Description**  
Retrieve and display the latest stock market news articles for specific tickers.

**Article Metadata:** Show publication dates and sources.

**Libraries and Imports**

* **Core Libraries:**
  + requests: For API interactions
  + json: For parsing JSON responses
  + rich.console and rich.table: For displaying tabular data in CLI

**API Interaction**

* **Platform:** [RapidAPI.com](https://rapidapi.com/" \t "/private/var/folders/s8/7qnqk0pj3z30tkk5jp4py3b40000gp/T/com.kingsoft.wpsoffice.mac.global/wps-archits/x/_new)
* **API Name:** Stock Market News API
* **Endpoint:** https://stock-news.p.rapidapi.com/v1/news/{ticker}
* **Authentication:** RapidAPI Key Authentication

**Code Logic and Workflow**

1. setup\_api() **function:**
   * Configures API endpoint and headers
   * Sets API key for authentication
2. fetch\_news() **function:**
   * Sends GET request to fetch news data
   * Handles errors and missing values
3. display\_results() **function:**
   * Formats and displays news articles using rich.table

**Data Processing Highlights**

* News headlines and publication dates displayed in styled tables
* Handles API limits and errors gracefully

**Sample Data Output**

**News Headlines:**

* + Article: "Tech Stocks on the Rise"
  + Date: November 24, 2024
  + Source: Bloomberg

**Potential Improvements**

1. Export articles to a CSV or JSON file
2. Add ticker filtering functionality
3. Include sentiment analysis of articles