

Fitbit API Data Extraction Project Summary:

1. Project Overview: This project facilitates the extraction of various health metrics from Fitbit devices:
 - ECG Table: Extracts electrocardiogram data
 - Daily Summary Table: Retrieves sleep, respiratory rate, and body temperature data
 - Minutes Table: Collects minute-by-minute step count, heart rate, SpO2, and heart rate variability data
2. Functionality:
 - Utilizes OAuth 2.0 for secure API authentication
 - Retrieves data from Fitbit's servers via API calls
 - Processes and exports data into structured CSV files
3. Data Type Distinctions:
 - ECG and Minutes data provide high-frequency, detailed information
 - Daily Summary offers aggregated daily metrics
4. Application Credentials: Users have two options for API access:
 - a. Utilize provided credentials:
 1. Client ID: 23PHMD
 2. Client Secret: 0ce3a7d7167a733fd9a3b5c23a3438f6
 3. Redirect URL: <http://127.0.0.1:8080/>
 - b. Create a personal Fitbit developer application:
 4. Visit <https://dev.fitbit.com/>
 5. Select "Manage" and click 'Register An App'
 6. Register or log in and create a new application
 7. Select "Personal" for OAuth 2.0 Application Type
 8. Set Callback URL to <http://127.0.0.1:8080/>
 9. Select "Save"

Edit the Application

Application Name *

aarathi - data collection

Description *

data collection

Application Website URL *

https://www.goodreads.com/

Organization *

myself

Organization Website URL *

https://www.goodreads.com/

Terms of Service URL *

https://www.goodreads.com/

Privacy Policy URL *

https://www.goodreads.com/

OAuth 2.0 Application Type *

☐ Server

☐ Client

☒ Personal

Redirect URL *

http://127.0.0.1:8080/

Default Access Type *

☐ Read & Write

☒ Read Only

Note: You can put in anything for Application Name, Description, Application Website URL, Organization, Organization Website URL, Terms of Service URL, Privacy Policy URL as long as it doesn't include the word 'fitbit'.

5. Execution Instructions:

- Install Python and required libraries (requests, pandas)
 - Insert appropriate Client ID and Client Secret into the scripts (if you're using our own developer app otherwise just start execution of script)
 - Execute required script via command line: `python script_name.py`
 - Input the desired date for data extraction
 - For initial authorization, use the following URL:
https://www.fitbit.com/oauth2/authorize?response_type=code&client_id=CLIENT_ID_HERE&redirect_uri=http%3A%2F%2F127.0.0.1%3A8080%2F&scope=oxygen_saturation%20electrocardiogram%20sleep%20respiratory_rate%20activity%20heartrate%20temperature%20settings
 - Replace CLIENT_ID_HERE with 23PHMD if using provided credentials, or your personal Client ID if using a self-created application
 - Input the resulting authorization code when prompted
6. Authorization Code:
- A temporary code Fitbit provides to allow the app to access your data
 - Appears as a long string of letters and numbers, typically ending with a "#"
 - Found in the URL after approving access
 - Example: After approval, you'll be redirected to a URL like:
<http://127.0.0.1:8080/?code=1234abcd5678efgh9012ijkl3456mnop#>
 - The authorization code here is: 1234abcd5678efgh9012ijkl3456mnop
 - When entering this into the script:
 1. Copy the authorization code
 2. Do not include anything that comes after "#" (including #)
 3. Paste it into the script when prompted
7. Output:
- The scripts generate CSV files containing the extracted Fitbit data
 - Output files are saved in the same directory as the executed script
8. Best Practices:
- Maintain the `fitbit_tokens.json` file in the script directory
 - Re-authorize if authentication errors occur