

Airtable to Latimer Core Data Extractor Documentation

Author: Matt Woodburn

Version: 1.0

Date: 2024-08-28

Script URL:

https://colab.research.google.com/drive/1cfWJd_iTUtpqj6ErwOvHrqrQf0aEHUgY?usp=sharing

This document describes an experimental data extractor that is intended to provide users with a no-code tool for extracting data from [Latimer Core](#) (LtC) conformant [Airtable](#) bases in JSON and CSV formats. The aims of the script are:

- to explore and demonstrate how LtC data might be extracted from a source system/database in which they're stored (in this case, Airtable) and converted into formats that are suitable for sharing with other people (e.g. CSV) and/or other systems (e.g. JSON)
- to allow people to explore what LtC datasets that they work on in Airtable sandboxes look like in these other formats

The extractor is currently in the form of Python code run in a Jupyter Notebook, and shared using Google Colab. This enables anyone (with a Google account) to view and run the script against their Airtable bases, and to take their own copies if they want to try out any edits to the script itself.

Notes:

- To run the script in Google Colab, you will need to log in to a Google account.
- While you can make edits directly to the script, these will be temporary, meaning that they:
 - won't be visible to other users
 - won't change the original script (so can be done safely without breaking anything for anyone else)
 - won't be saved, so if your session ends or you close the browser page/tab they'll be lost

- You can however take a copy of the original script for editing using 'File'->'Save a copy in Drive' from the top menu bar. If you would rather run the script locally, you can download the Jupyter Notebook using 'File'->'Download'->'Download .ipynb' from the top menu. However, please note that the last cell in the notebook would need to be modified to save files locally rather than download them.
- An Airtable free account limits the number of API calls to 1,000 per month. This roughly equates to 30 to 40 runs of the script.

Caveat:

The Jupyter notebook is an experimental script originally developed for the [TDWG 2024 Latimer Core Workshop](#), and is an initial exploratory piece of work rather than a robust and sustainable tool. For more information on setting up LtC-compliant Airtable bases as the source for the tool, please read the [Latimer Core Workshop orientation document](#). The output formats of the script are also experimental, and should be treated as demonstrative rather than definitive.

Preparation

Creating an Airtable API bearer token

1. In Airtable, click on your user icon in the top left hand corner, and select "Builder hub".
2. In the menu on the left of the screen, click "Personal access tokens".
3. Towards the top right of the page, click "+ Create new token".
4. Give the token a name (e.g. "LtC Colab script access").
5. Under the "Scopes" section, click "Add a scope", and select "data.records:read".
6. Under the "Access" section, click "Add a base". Select "TDWG: Latimer Core Digitisation Example".
7. Click "Add a base" again and select "TDWG: Latimer Core Template".
8. Click "Create token". You will get a pop-up with the long token string - copy the full string and store it somewhere accessible, as it's not possible to view it after this point.

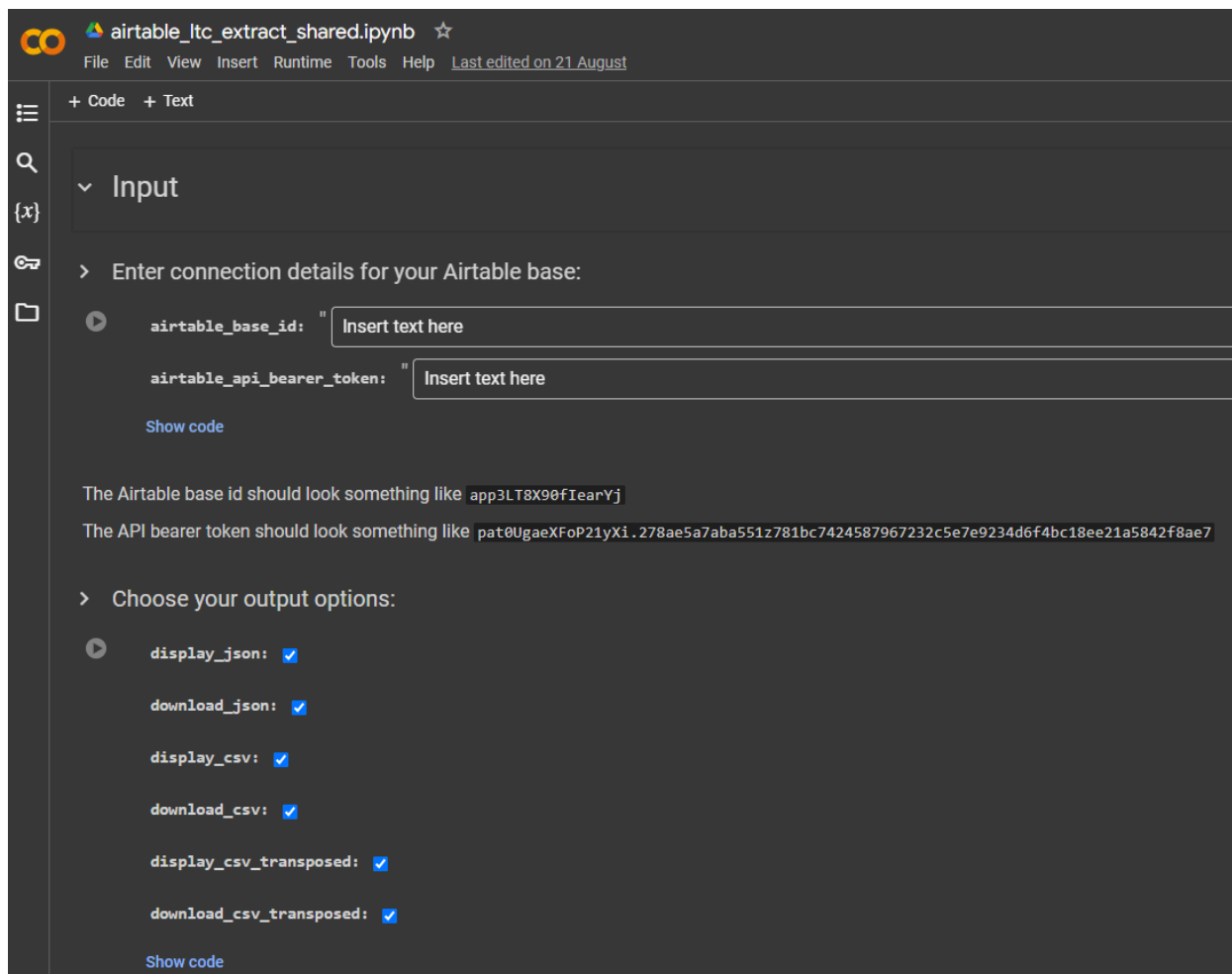
Getting your Airtable base IDs

1. Click "Back to home" towards the top right of the screen.
2. From the list of bases, click into the "TDWG: Latimer Core Digitisation Example" base.

3. In the URL at the top of your browser, the base ID comes directly after “airtable.com”, surrounded by slashes and starting with “app” - for example, “apphqjDuC81ffdAiT”. Copy and paste the ID string to somewhere where you can access it.
4. Repeat the process for the “TDWG: Latimer Core Template” base.

Running the script

1. Open the [Google Colab script](#).



2. In the 'Input' section:
 - paste the relevant Airtable base id into the *airtable_base_id* field
 - paste your API bearer token into the *airtable_api_bearer_token* field

3. In the 'Choose your output options' section, select the formats in which you want to view and/or download the data (json, csv, transposed csv) using the checkboxes. The 'display' options will show the outputs in the 'View results' section further down the page, while the 'download' options will download those outputs as files.
4. In the top menu, click 'Runtime' and choose 'Run all'. Note: the first time you run the script, you may get a warning that the notebook was not authored by Google. If you're happy to proceed, click 'Run anyway'.
5. The script may take a few minutes to run, depending on the amount of data in the Airtable base.

In-progress steps will display spinning icons. When complete, they will display a green check-mark.

Once the script has completed, your browser should download the output files if you've selected download options.

Note: the first time you try to download multiple files, you may get an alert that the page wants to download multiple files. Click 'Allow' (or similar, depending on your browser) to provide permission.

6. The downloaded files should be visible in your standard browser downloads folder, and can be opened in a text editor or other appropriate software tool.