# DROID Baseline test

The purpose of the following test is to establish that you have set up your instance of DROID in a way that will generate consistent results with others.

By following the instructions below you will be guided through the process of getting a test set of files, setting up your instance of DROID, running the files through DROID, and finally comparing the results with a baseline set of results.

## Preconditions

To work through this example set, you first need to ensure you have installed the following applications onto your machine.

1. Python (v2.7 or v3.4) (<https://www.python.org/downloads/>)
2. DROID <https://github.com/digital-preservation/droid>
   1. Its worth noting that in our experience it’s not wise to run different version of DROID on one machine unless you know what you’re doing. DROID uses some storage locations outside its own folder structure, and this can lead to some confusing results if you’re 100% sure what’s going on.

For this test it is suggested that you install and use a slightly older version of DROID. v6.1.5 <http://nationalarchives.gov.uk/documents/information-management/droid-binary-6.1.5-bin.zip>

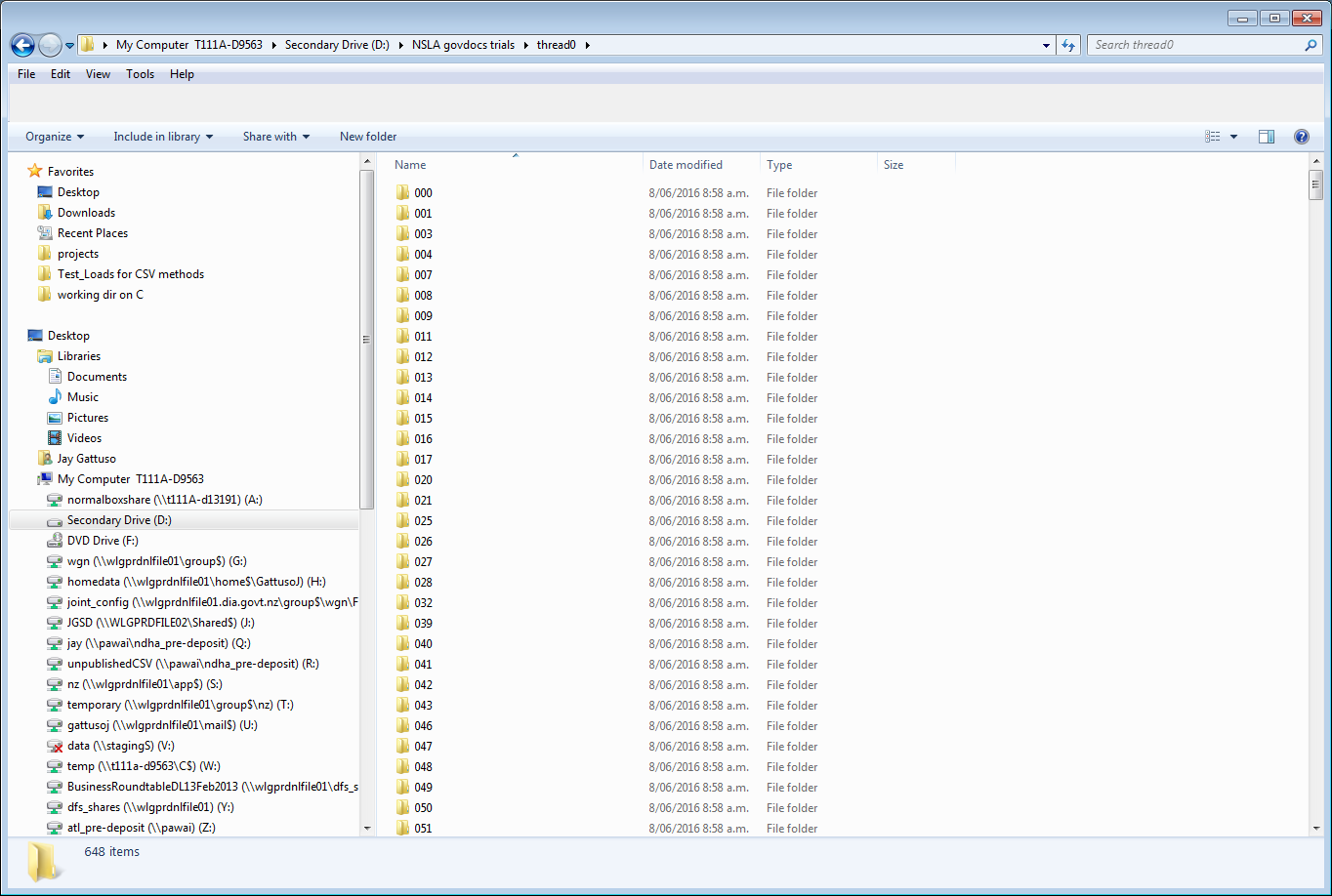
You might also want to install a dev environment for handling scripts. Sublime Text is very good and well used <https://www.sublimetext.com/3>

## Getting some data

Next you will need to grab the test set of file. They are available from here:

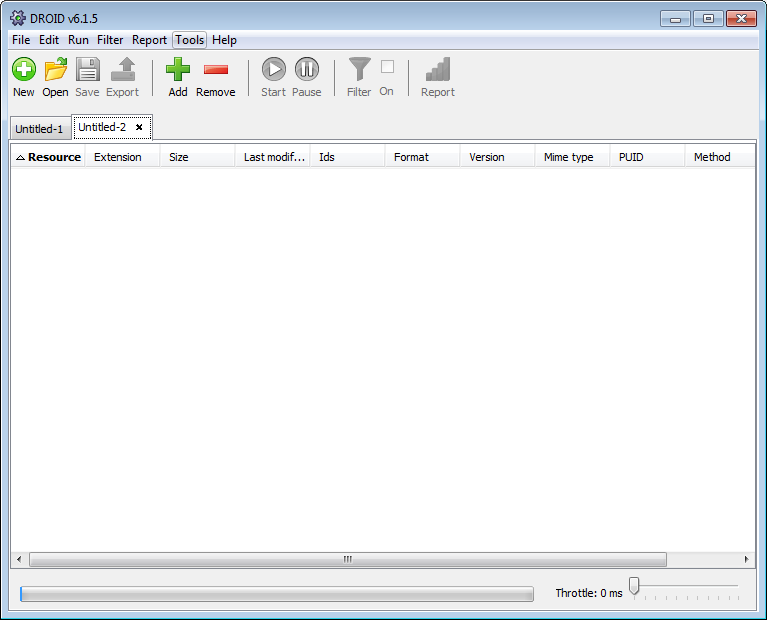
<http://digitalcorpora.org/corp/files/govdocs1/threads/thread0.zip>

Download the zip file to an appropriate location on your machine, and unpack the zip file into its own folder. You should see something like this:-



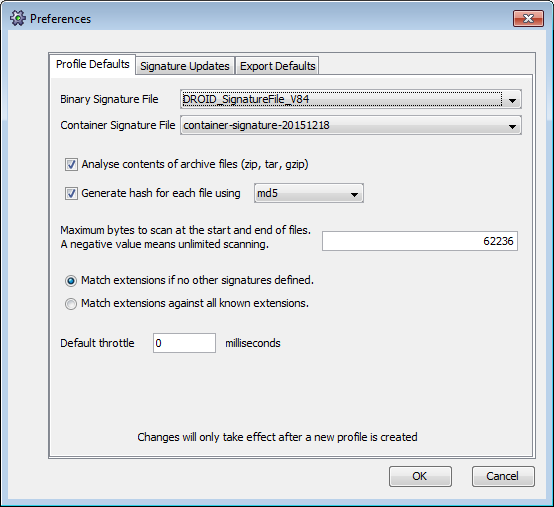
## Set up DROID

Next you need to set your DROID to behave in the same way as the baseline. Open the preferences menu in “tools”:-



And make sure your DROID settings are:-

* Max Byte Scan = 65536
* Binary Sig: v84
* Container Sig: v20151218
  + You may not have this container file in your install, if not, you’ll need to grab it from the PRONOM website, and copy it to the C:\Users\user \.droid6\container\_sigs folder. This folder might have a slightly different location – search for folders (including hidden) called “.droid” if you have trouble locating it. Once in the folder, the DROID Container Signature File list should automatically update with your new file.
* “Match extensions if no other sigs defined”
* Generate hash for each file using MD5

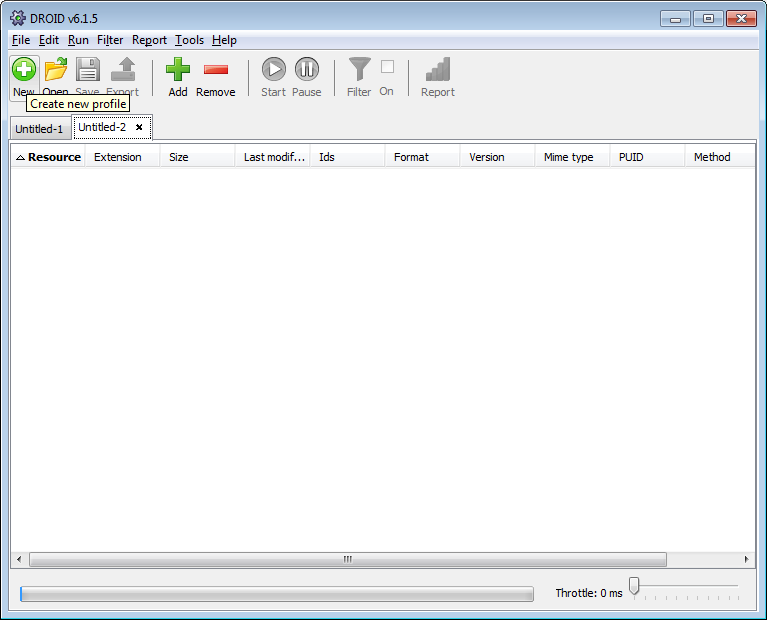


If you need to get copies of signature files you can find them on the PRONOM website:

<http://www.nationalarchives.gov.uk/aboutapps/pronom/droid-signature-files.htm>

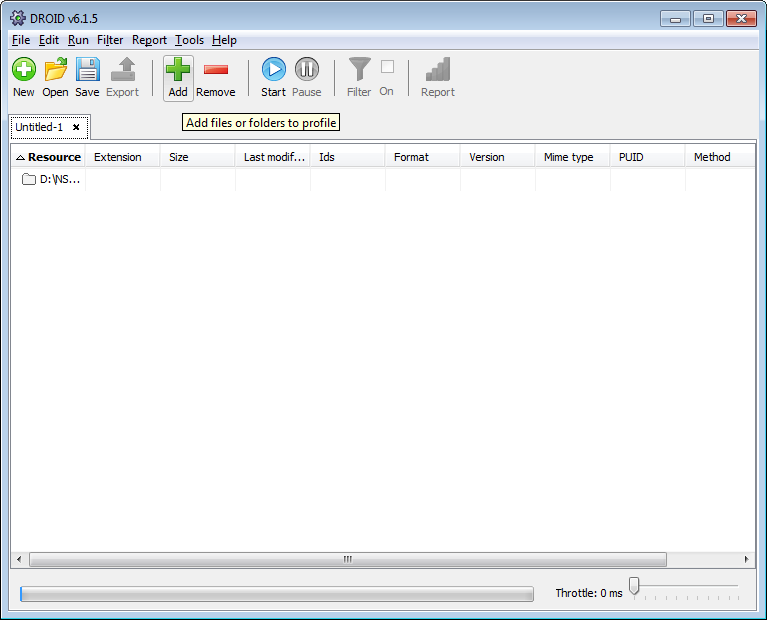
When you are happy they are the same, click OK.

If you have had to change anything in the preferences, create a new profile:-

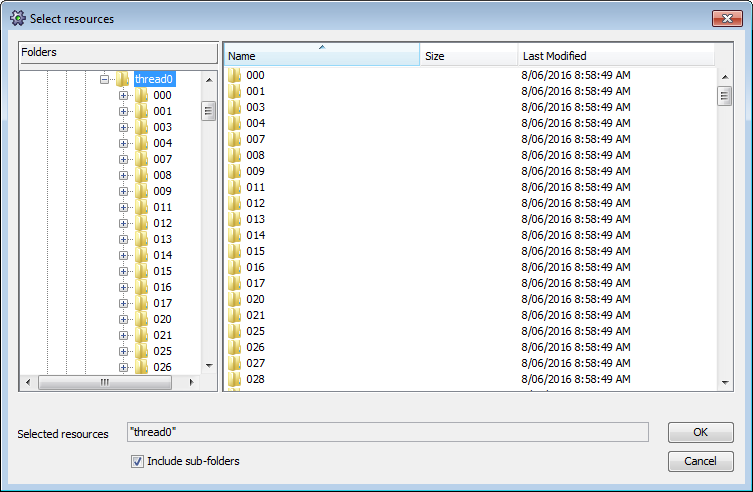


## Run the files through DROID

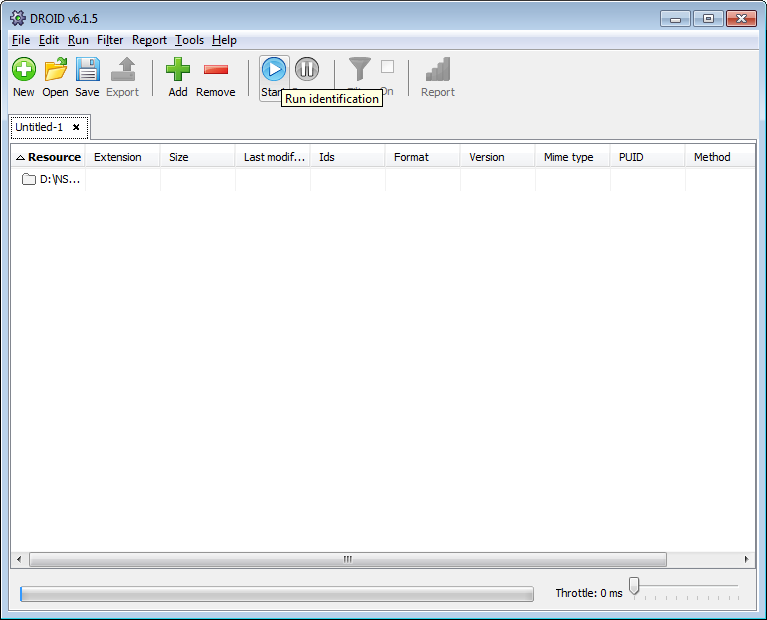
Add files to the DROID profile:-



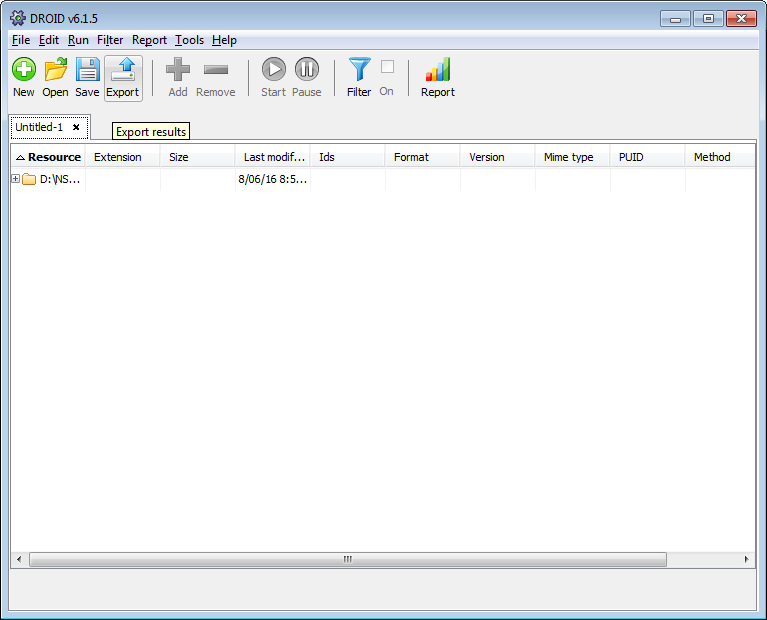
Point to unpacked folder at the top level:-



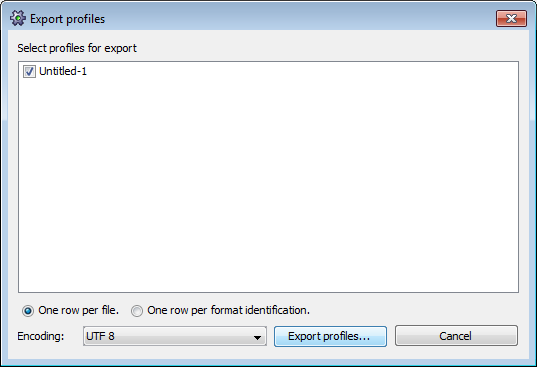
Run DROID:-



When complete, export results:-



Pick your “profile”, set to one row per file. Click Export profiles:-



(If you needed to create a new profile, make sure you select the correct profile here as there will multiple options available to you).

Label your export file accordingly and save. For example, call your file my\_results.csv

## Comparing results with Python

Once you have saved the log file you will need to clean it of information that will complicate the comparison step. You can do this by running the file DROID\_export\_cleaner.py. This script will:

* Remove the “Folder” type rows – these are not needed for this test.
* Remove the first few data elements of each row – these will be specific to your files and will make the comparison difficult.
* Sort the results into alphanumerical order – this will allow us to do a row-by-row comparison
* Save a new version of the cleaned results – this will be used to compare against the baseline.csv

In this script you will need to change line number 6 which currently reads:

input\_csv\_filename = r"thread0.csv"

Set it to the name of your csv file and move your csv file into the folder that contains the python scripts, or set it to full path of your csv file

e.g:

input\_csv\_filename = r"my\_results.csv"

or

input\_csv\_filename = r" C:\Users\user\Desktop\my\_results.csv"

Save the file once you’ve made the changes, and run the script.

This script when complete will create a new file called cleaned\_csv.csv [[1]](#footnote-1)

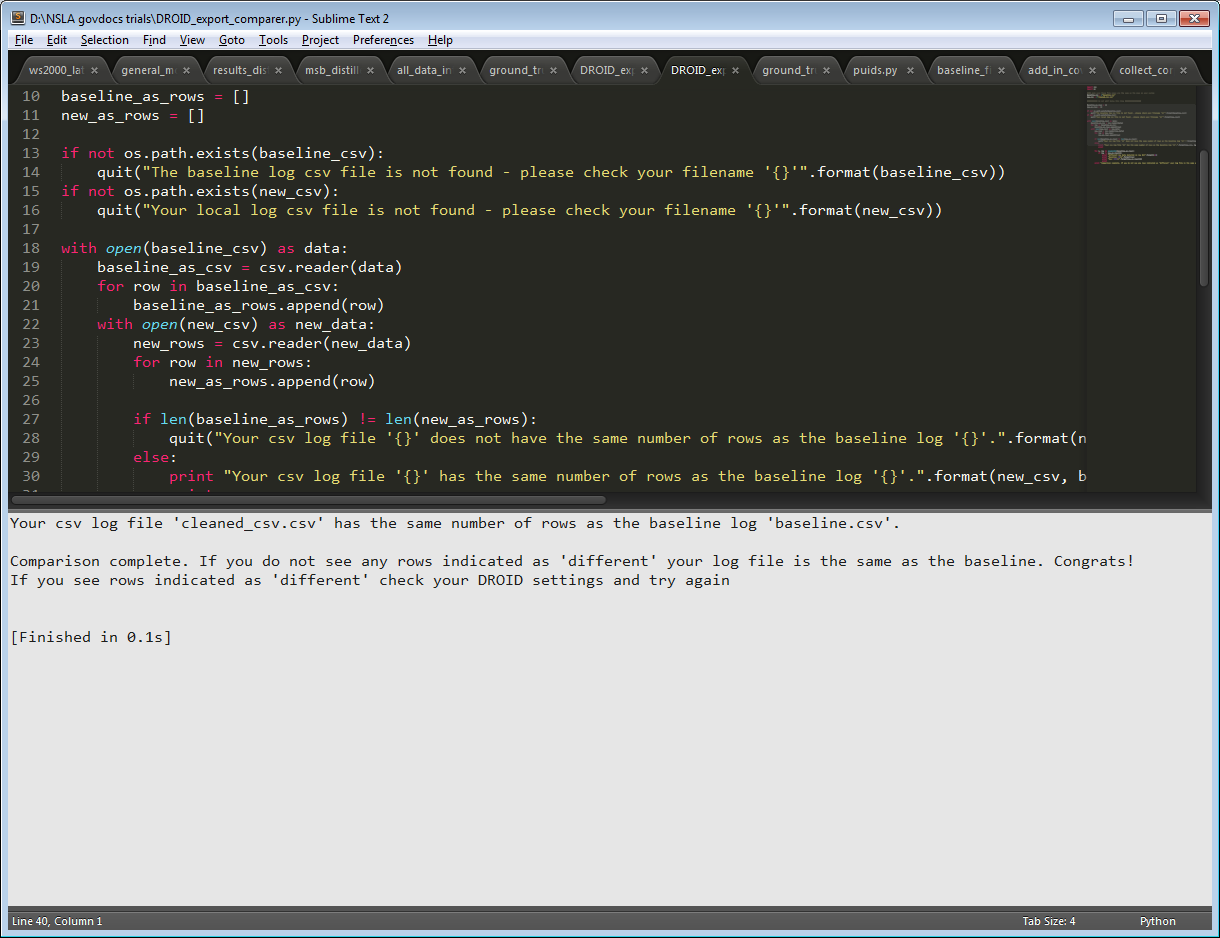
Run the python script “DROID\_export\_cleaner.py” either from command line (python “DROID\_export\_cleaner.py” or in your dev environment (ctrl+b in sublime text)

The next step is to compare your results with the baseline results.

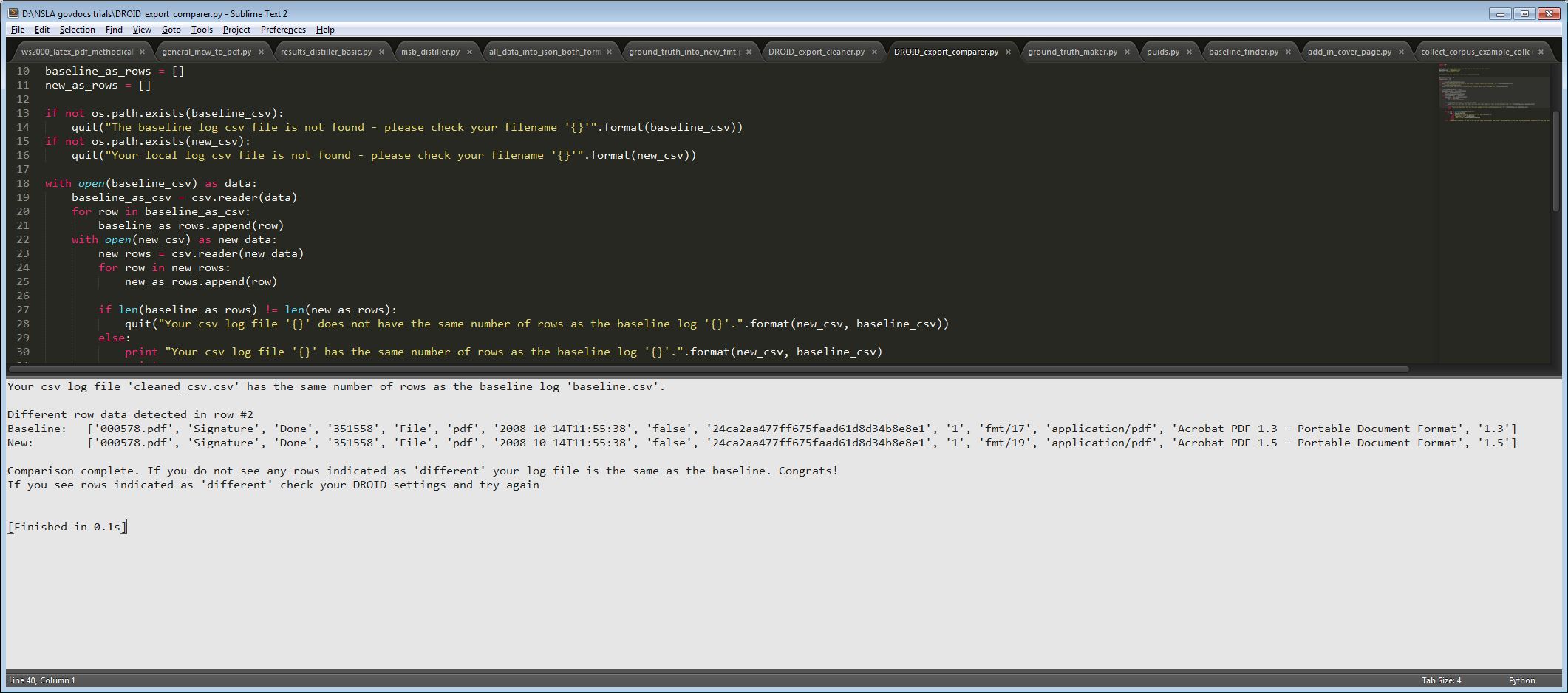
Make sure the filenames in “DROID\_export\_comparer.py” match your files, and run the comparison between the two cleaned log files.

Take a note of any information that is posted to the terminal – this will tell you how the comparison task went.

If you get no errors or problems, you should see something like this in your terminal window:-



If you do have some differences in your rows, you should see a message like this:-



This script should also inform you if it can’t find the required csv files.

1. If required, you can change this file name on line number 7

   output\_csv\_filename = "cleaned\_csv.csv"

   However, you will also need to change line 6 in the comparer script to match your new file name:

   new\_csv = r"cleaned\_csv.csv” [↑](#footnote-ref-1)