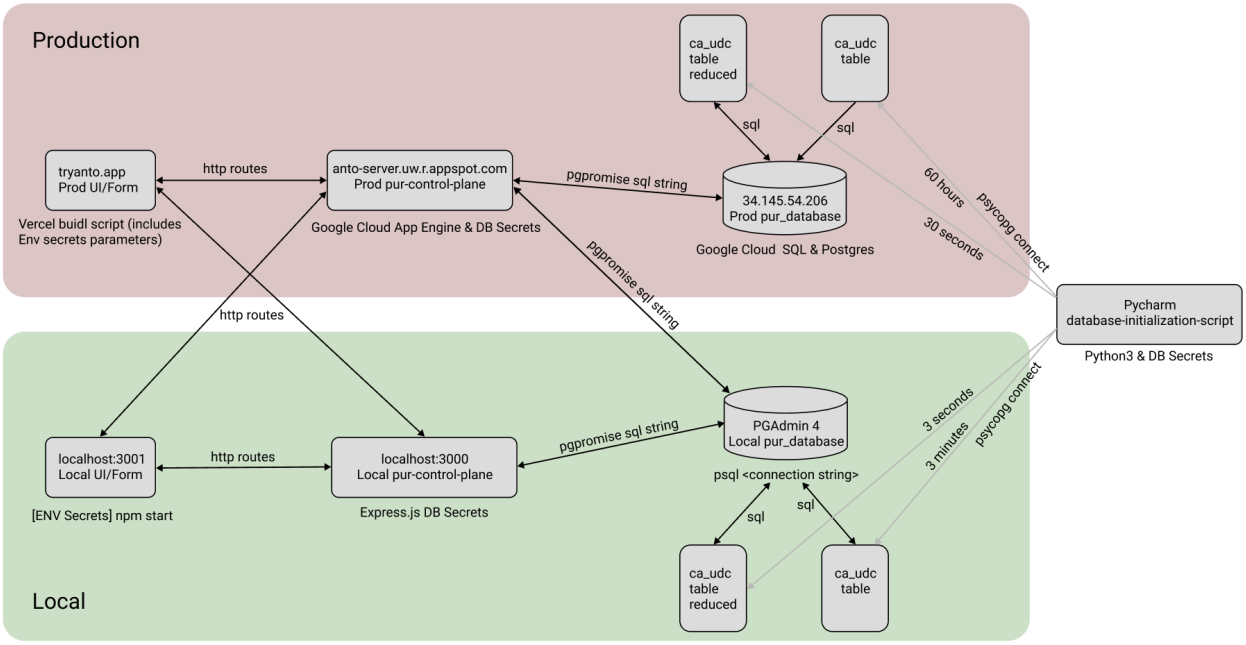
Anto Documentation

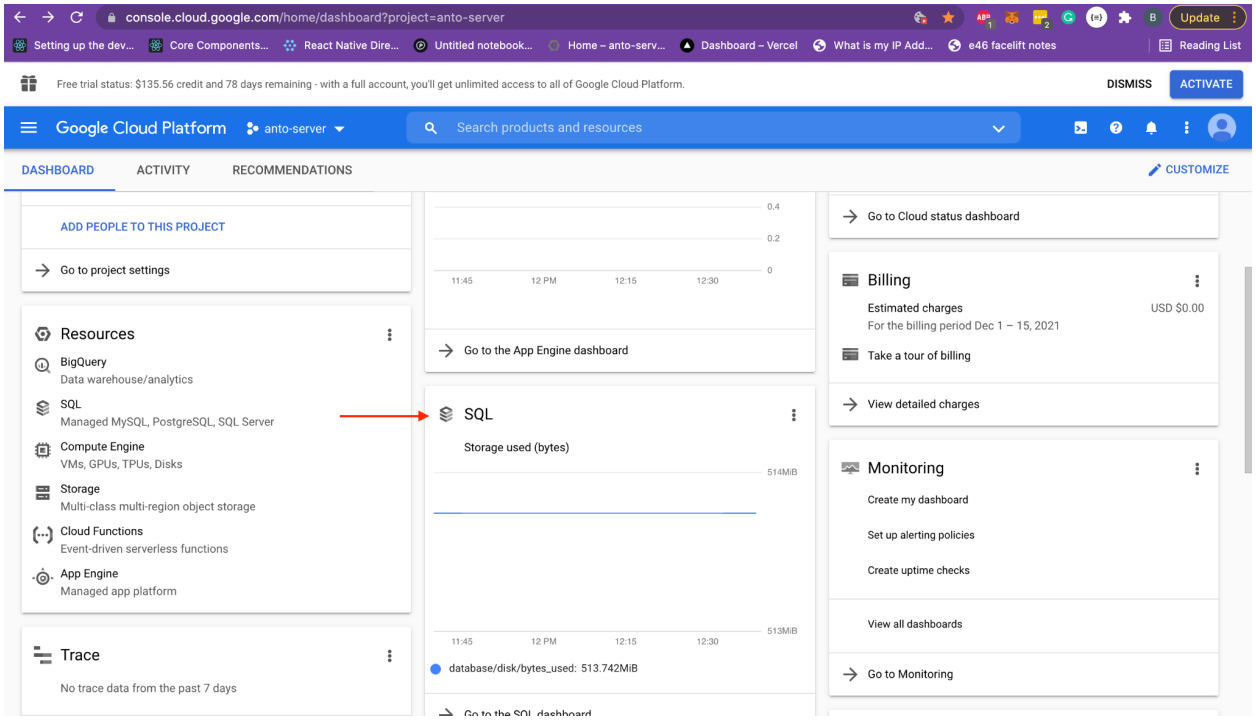
Anto Subsystems



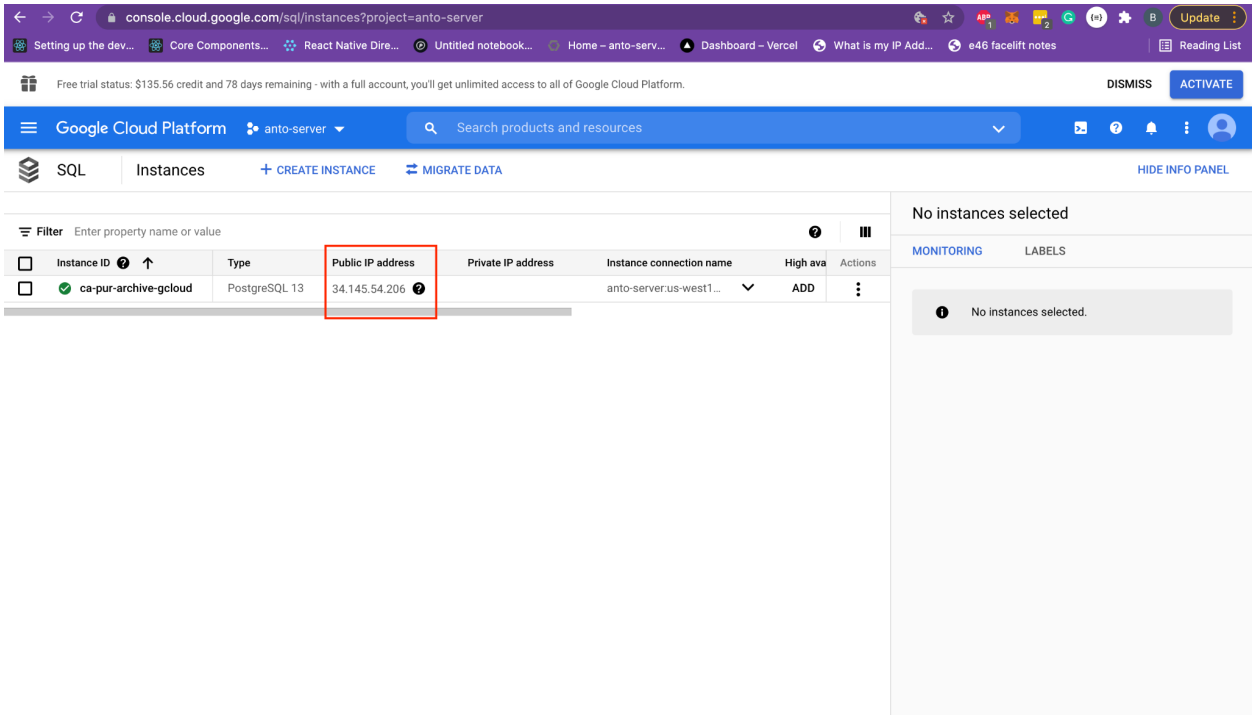
Each subsystem should have its documentation described from the first file to start-up and commits.

Connecting to the Production Database

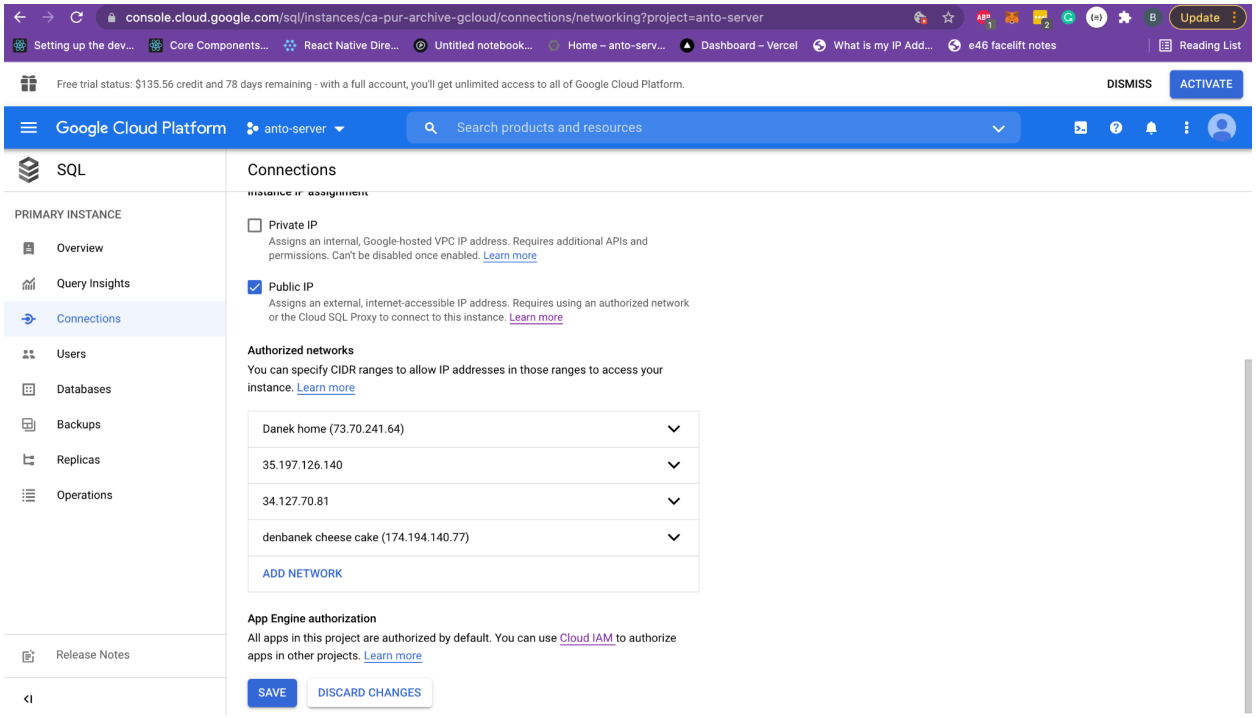
- Log into Google Cloud Platform
- Go to the Anto Project
- Find the “SQL” section and select “go to the SQL dashboard”



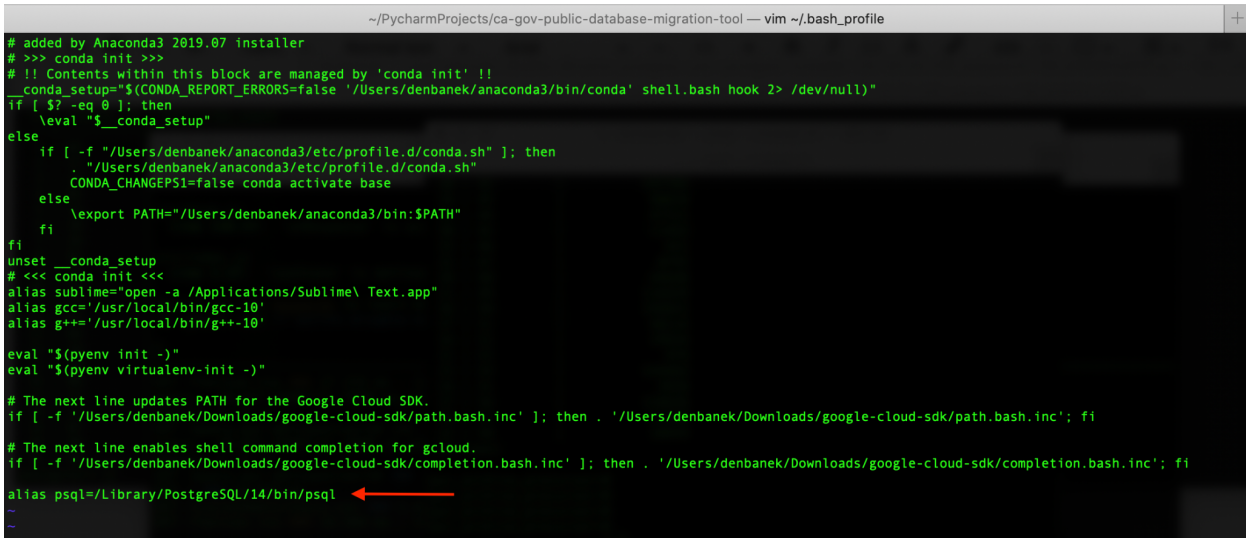
- Note the public ip address for the DB. It should be reserved and immutable for this service to run (we will use this field to connect to the DB remotely in the future)



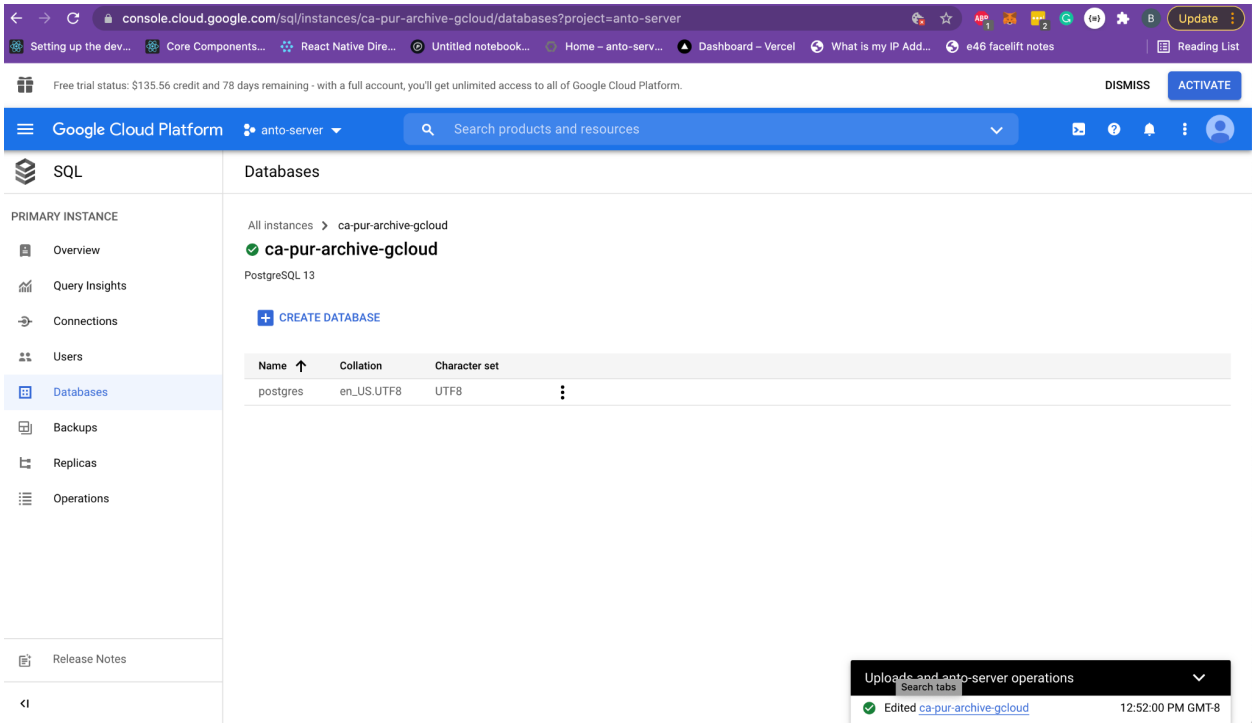
- Select the actions (horizontal ellipses …) button and select edit
- Select the “Connections” section on the left column. Scroll until you see “Authorized Networks”



- Add your IP address to the whitelist.
    - **Do not add public networks like coffee shops or airports to this field. Anybody on this network with the credentials would have permission to augment our database if this were the case.**
    - If you need to figure out the network you're on visit: http://ipv4.whatismyv6.com/
- Hit save.
    - Now that inbound connections from your IP address are permitted, all you need is the credentials to connect to the database remotely.
- Make sure you have the `psql` bash command available.
    - it was kind of a pain to get this to work, you need to download it
    - I went to this site: https://www.postgresql.org/download/macosx/ and downloaded the installer, which gave me the binary
    - of course the binary wasn't added to my $PATH in bash, so I just made an alias in my bash profile
    - On your terminal/linux command line edit `~/.bash_profile` and add this as the last line `alias psql=/Library/PostgreSQL/14/bin/psql`
    -



- Using the psql command will open a connection with the database where you can execute SQL in the context of the publicly available database. The database parameters are visible on the Google Cloud Platform dashboard and settings. Note that a GCP SQL instance may have multiple databases, and you are only remotely connecting to a single one.



- `psql "sslmode=disable dbname=postgres user=postgres hostaddr=34.145.54.206 password=<KEY>"`
  - We already noted the hostaddr of the database, it's the public IP of the DB
  - dbname is visible on the "Databases" tab of the SQL instance settings (see the above screenshot)

    - The password (<KEY>) is a string only visible during the configuration of the database, and I've written it down for future use (but not here, somewhere very safe).
- If this command runs successfully, you should see something like this.

```
vodnik:ca-gov-public-database-migration-tool denbanek$ psql "sslmode=disable d
bname=postgres user=postgres hostaddr=34.145.54.206 password=1GEJbNOKlmDFjFsp"
psql (14.1, server 13.4)
Type "help" for help.

postgres=>
```

- You can now execute SQL in the context of the postgres database on the ca-pur-archive-gcloud Google Cloud Platform SQL instance.

Useful Commands on the sql instance

\dt	list all tables
SELECT COUNT(*) FROM <TABLE>	count the number of rows in the table
DROP TABLE <TABLE>	delete a table (probably don't do this for fun, could be costly)

Populating the Database Tables:

Some Tables we keep for running Anto. Here are some screenshots of what they hold.

```
postgres=> \dt
                List of relations
 Schema |          Name          | Type  |  Owner
--------+------------------------+-------+----------
 public | ca_reduced_udc         | table | postgres
 public | ca_udc                 | table | postgres
 public | ca_yearly_reduced_udc  | table | postgres
(3 rows)
```

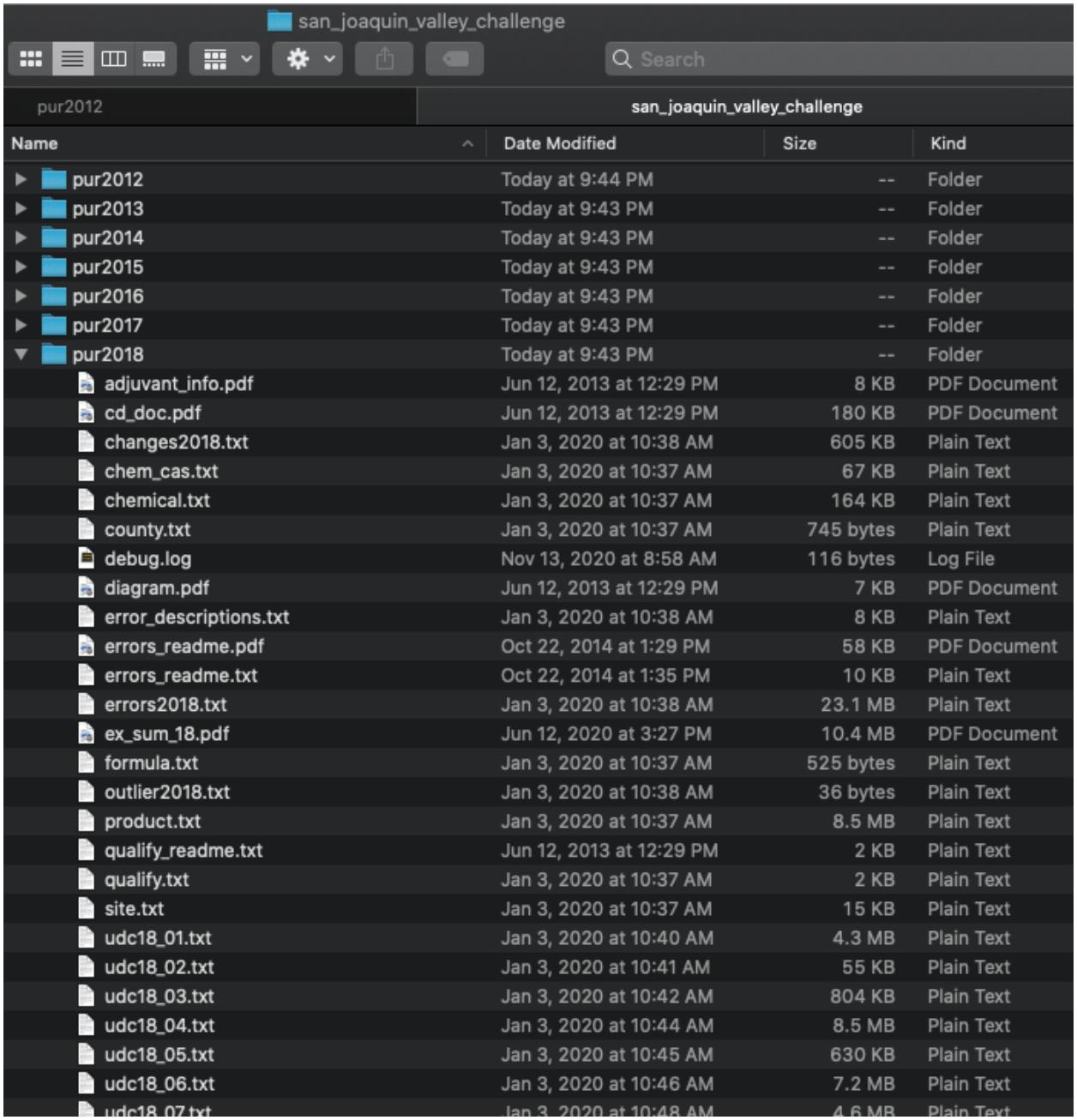
```
postgres=> SELECT count(*) FROM ca_reduced_udc;
 count
-------
    58
(1 row)

postgres=> SELECT * FROM ca_reduced_udc;
 id | county_cd | pesticide_count
----+-----------+-----------------
  1 | 03        |            6181
  2 | 17        |           15375
  3 | 16        |           96035
  4 | 02        |             496
  5 | 14        |             958
  6 | 28        |          110284
  7 | 29        |            5013
  8 | 01        |           36453
  9 | 15        |          228482
 10 | 39        |          197312
 11 | 11        |           47827
```

```
postgres=> SELECT count(*) FROM ca_yearly_reduced_udc;
 count
-------
    58
(1 row)

postgres=> SELECT * FROM ca_yearly_reduced_udc;
 id | county_cd | count_2015 | count_2012 | count_2013 | count_2014 | count_2016 | count_2018 | count_2017
----+-----------+------------+------------+------------+------------+------------+------------+------------
  1 | 1         |      39418 |      35773 |      36160 |      37982 |      40794 |      36453 |      38912
  2 | 2         |        380 |        297 |        325 |        314 |        343 |        496 |        347
  3 | 3         |       5852 |       4897 |       5708 |       5660 |       6108 |       6181 |       6091
  4 | 4         |      58341 |      49916 |      51055 |      53673 |      57199 |      59601 |      61182
  5 | 5         |       4986 |       4308 |       3960 |       4393 |       5207 |       5191 |       5055
  6 | 6         |      44602 |      36856 |      41029 |      39958 |      48093 |      48274 |      43939
  7 | 7         |      34540 |      31438 |      32412 |      34619 |      37784 |      37923 |      36840
  8 | 8         |       3278 |       3519 |       3408 |       1281 |       4171 |       4426 |       3917
  9 | 9         |      15875 |      15257 |      15833 |      15638 |      16566 |      16240 |      15444
 10 | 10        |     356073 |     326732 |     338975 |     334819 |     375947 |     408550 |     378789
 11 | 11        |      38348 |      32815 |      35185 |      35826 |      44244 |      47827 |      47185
```

```
postgres=> SELECT count(*) FROM ca_udc;
  count
---------
 1458353
(1 row)

postgres=> SELECT * FROM ca_udc WHERE id = 1;
 id | prodno | chem_code | lbs_chm_used | applic_dt  | county_cd | township
----+--------+-----------+--------------+------------+-----------+----------
  1 |   7583 |        34 |     0.055962 | 05/01/2018 | 03        |
(1 row)
```

Source Data File Structure:

The database population script relies on file naming patterns and file structure, here is an example of a structure that was used to build the `ca_yearly_reduced_udc` table.


san_joaquin_valley_challenge
Search
pur2012
san_joaquin_valley_challenge
Name
Date Modified
Size
Kind
pur2012 Today at 9:44 PM -- Folder
pur2013 Today at 9:43 PM -- Folder
pur2014 Today at 9:43 PM -- Folder
pur2015 Today at 9:43 PM -- Folder
pur2016 Today at 9:43 PM -- Folder
pur2017 Today at 9:43 PM -- Folder
pur2018 Today at 9:43 PM -- Folder
adjuvant_info.pdf Jun 12, 2013 at 12:29 PM 8 KB PDF Document
cd_doc.pdf Jun 12, 2013 at 12:29 PM 180 KB PDF Document
changes2018.txt Jan 3, 2020 at 10:38 AM 605 KB Plain Text
chem_cas.txt Jan 3, 2020 at 10:37 AM 67 KB Plain Text
chemical.txt Jan 3, 2020 at 10:37 AM 164 KB Plain Text
county.txt Jan 3, 2020 at 10:37 AM 745 bytes Plain Text
debug.log Nov 13, 2020 at 8:58 AM 116 bytes Log File
diagram.pdf Jun 12, 2013 at 12:29 PM 7 KB PDF Document
error_descriptions.txt Jan 3, 2020 at 10:38 AM 8 KB Plain Text
errors_readme.pdf Oct 22, 2014 at 1:29 PM 58 KB PDF Document
errors_readme.txt Oct 22, 2014 at 1:35 PM 10 KB Plain Text
errors2018.txt Jan 3, 2020 at 10:38 AM 23.1 MB Plain Text
ex_sum_18.pdf Jun 12, 2020 at 3:27 PM 10.4 MB PDF Document
formula.txt Jan 3, 2020 at 10:37 AM 525 bytes Plain Text
outlier2018.txt Jan 3, 2020 at 10:38 AM 36 bytes Plain Text
product.txt Jan 3, 2020 at 10:37 AM 8.5 MB Plain Text
qualify_readme.txt Jun 12, 2013 at 12:29 PM 2 KB Plain Text
qualify.txt Jan 3, 2020 at 10:37 AM 2 KB Plain Text
site.txt Jan 3, 2020 at 10:37 AM 15 KB Plain Text
udc18_01.txt Jan 3, 2020 at 10:40 AM 4.3 MB Plain Text
udc18_02.txt Jan 3, 2020 at 10:41 AM 55 KB Plain Text
udc18_03.txt Jan 3, 2020 at 10:42 AM 804 KB Plain Text
udc18_04.txt Jan 3, 2020 at 10:44 AM 8.5 MB Plain Text
udc18_05.txt Jan 3, 2020 at 10:45 AM 630 KB Plain Text
udc18_06.txt Jan 3, 2020 at 10:46 AM 7.2 MB Plain Text
udc18_07.txt Jan 3, 2020 at 10:48 AM 4.6 MB Plain Text