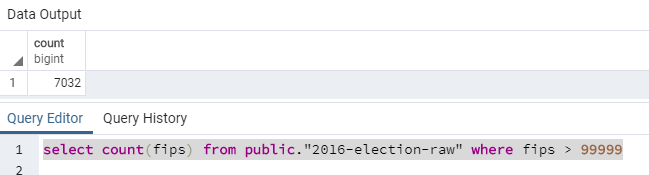
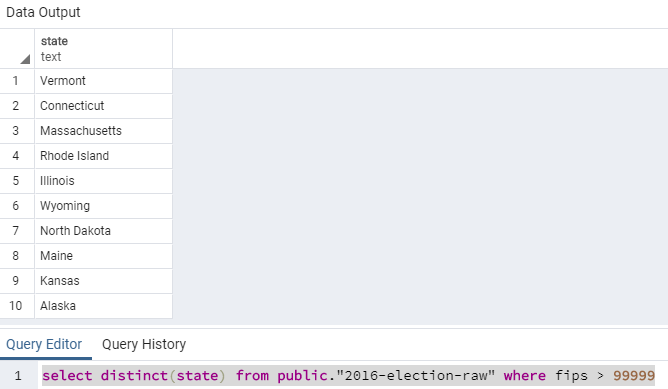
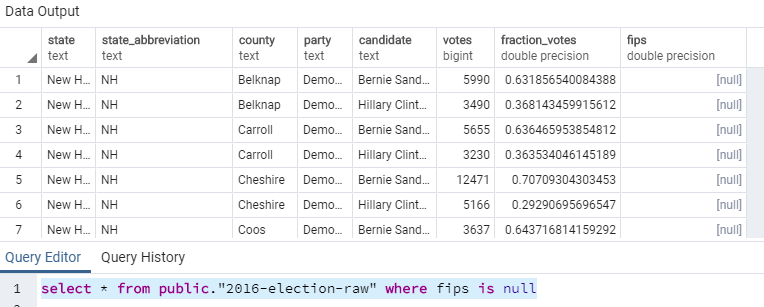
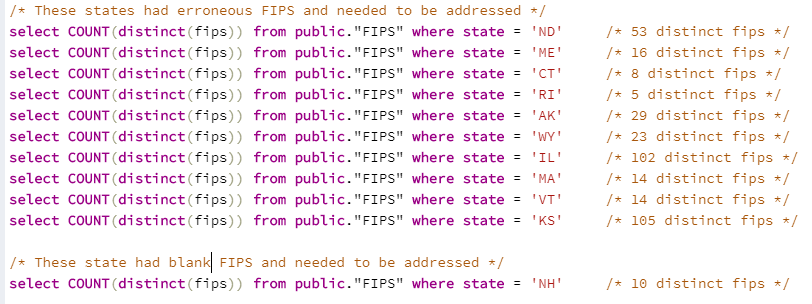
**2016 Primary Results | Data Sanitation**  
Data Source: <https://www.kaggle.com/benhamner/2016-us-election>  
  
**Note:** Several attempts were made to clean the data using SQLAlchemy, due to time limitations, that technology was abandoned in favor of a preferred technology of SQL.  
  
**Observations:** During reviewing of the data, we noticed that our key column FIPS county code ( Federal Information Processing Standards) that would be utilized was missing data and had erroneous data.   
FIPS are 5 positions and 7000+ records were > 99999. One state was blank.   
  
  
  
  
States where FIPS were erroneous.  




Using the FIPS table that would be our source to resolve the FIPS issues in the Elections table a brief summary was done below to see which FIPS would need to be cleaned up in our 2016 Elections table.

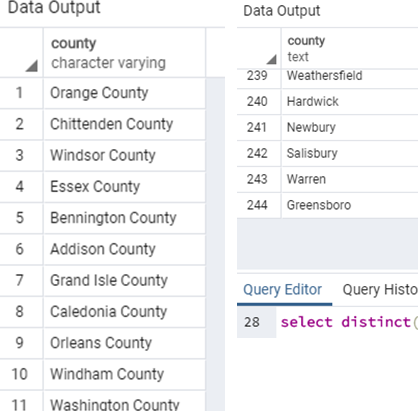


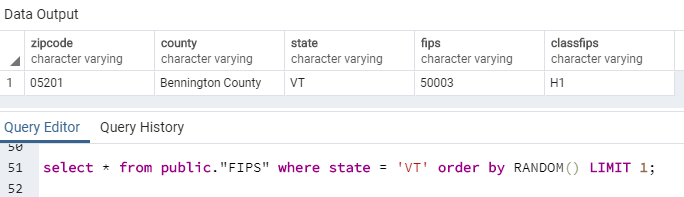
The only **true way** to resolve this would be to utilize the county column & state column in the FIPS table to select a distinct composite key in the elections table by county & state\_abbreviation column.

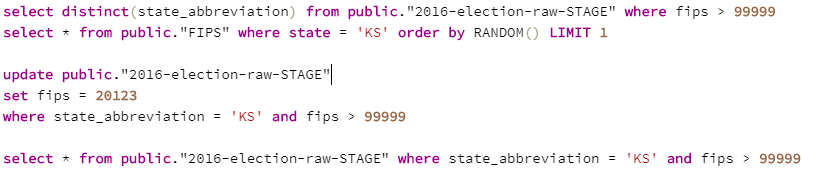
Vermont has over **240 distinct counties** in the **elections primary data** that has erroneous FIPS alone.   
The FIPS table **correctly has 14 distinct counties** for Vermont.

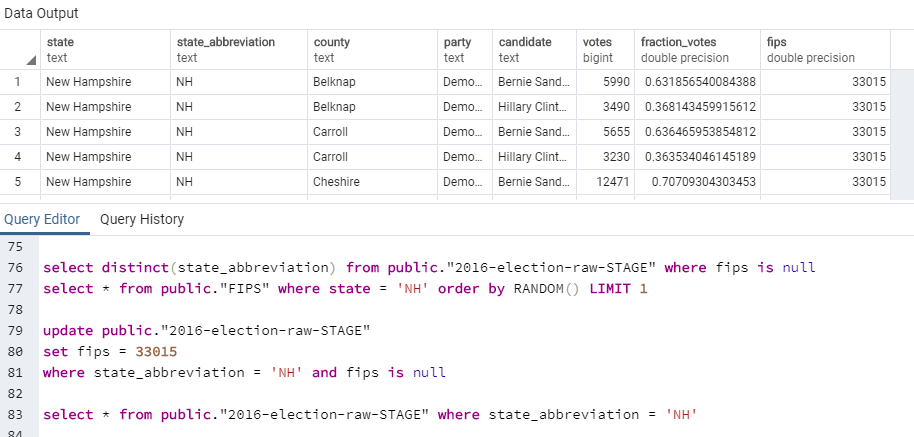


The county column in the FIPS table and the county column elections table would have to be synced up as they also do not match, **Essex** in one table and say **Essex County** in another.



**So,** for the sake of time, we decided to add a random FIPS by state from the FIPS table to the 2016 Elections Table for the entries of FIPS > 9999  
  


Selecting the erroneous FIPS, selecting a random FIPS from the FIPS table, updating the FIPS in the Elections table then selecting to confirm update.   
  


Then cleaning up the blank FIPS in the elections table for New Hampshire.  
  


select distinct(state) from public."2016-election-raw" where fips > 99999

select distinct(state\_abbreviation) from public."2016-election-raw" where fips > 99999

select distinct(fips) from public."FIPS" where state IN (select distinct(state\_abbreviation) from public."2016-election-raw" where fips > 99999)

/\* These states had erroneous FIPS and needed to be addressed \*/

select COUNT(distinct(fips)) from public."FIPS" where state = 'ND' /\* 53 distinct fips \*/

select COUNT(distinct(fips)) from public."FIPS" where state = 'ME' /\* 16 distinct fips \*/

select COUNT(distinct(fips)) from public."FIPS" where state = 'CT' /\* 8 distinct fips \*/

select COUNT(distinct(fips)) from public."FIPS" where state = 'RI' /\* 5 distinct fips \*/

select COUNT(distinct(fips)) from public."FIPS" where state = 'AK' /\* 29 distinct fips \*/

select COUNT(distinct(fips)) from public."FIPS" where state = 'WY' /\* 23 distinct fips \*/

select COUNT(distinct(fips)) from public."FIPS" where state = 'IL' /\* 102 distinct fips \*/

select COUNT(distinct(fips)) from public."FIPS" where state = 'MA' /\* 14 distinct fips \*/

select COUNT(distinct(fips)) from public."FIPS" where state = 'VT' /\* 14 distinct fips \*/

select COUNT(distinct(fips)) from public."FIPS" where state = 'KS' /\* 105 distinct fips \*/

/\* These state had blank FIPS and needed to be addressed \*/

select COUNT(distinct(fips)) from public."FIPS" where state = 'NH' /\* 10 distinct fips \*/

select \* from public."tax"

select \* from public."2016-election-raw" where fips is null

select \* from public."FIPS"

select distinct(county) from public."2016-election-raw" where state\_abbreviation = 'VT' and fips > 99999

select \* from public."FIPS" where state IN (select distinct(county) from public."2016-election-raw" where state\_abbreviation = 'VT' and fips > 99999)

select \* from public."FIPS" where state = 'VT' and county = 'Sprinfield County'

update