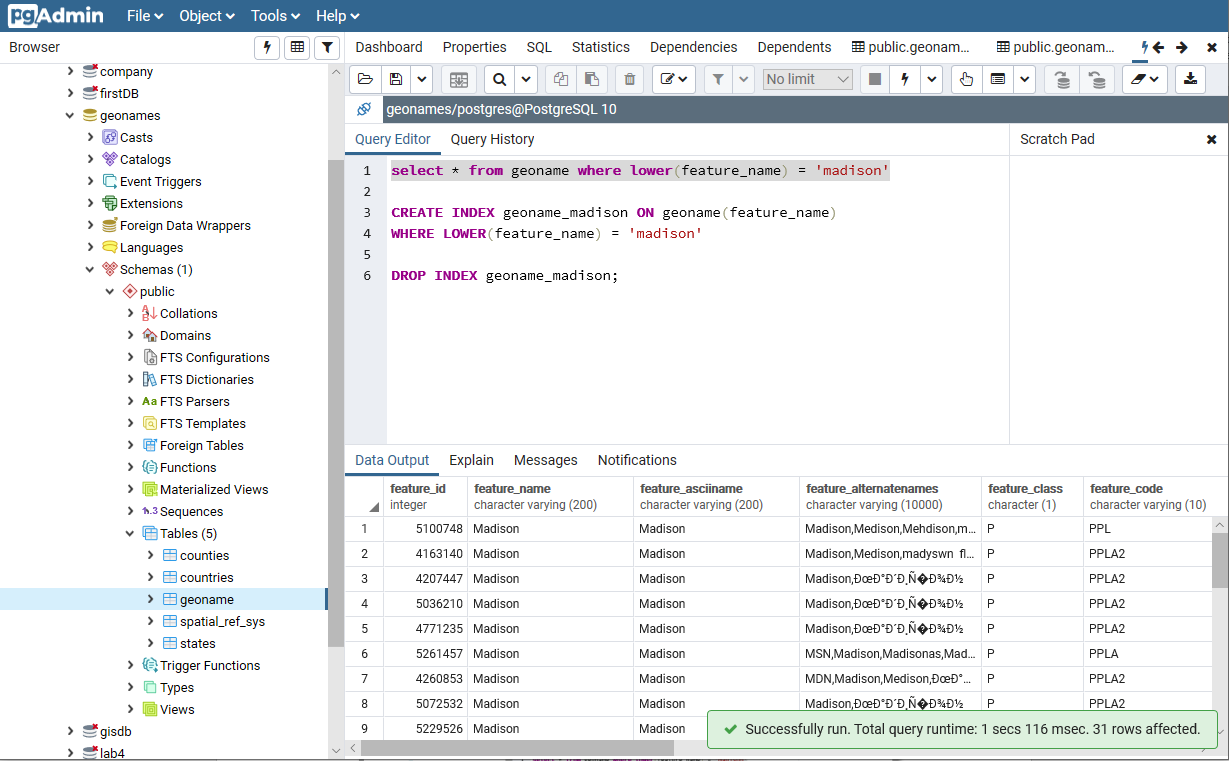
GEOG 574: Lab 9 – PostGIS Tutorial III: Spatial Indexing and Optimization

Nikki Jackson-Hebson

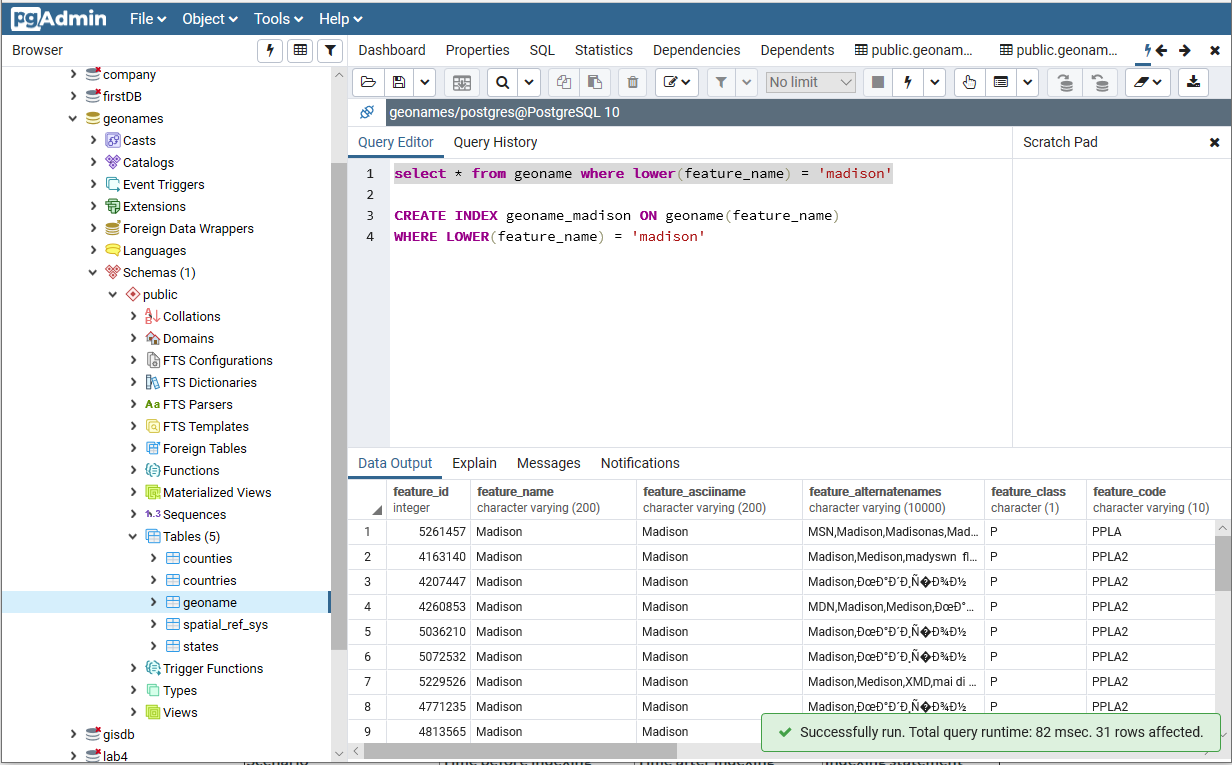
Command used to restore Lab9.backup file:

pg\_restore -h localhost -p 5432 -U postgres -d geonames lab9.backup

1. The total time for the query to run **without indexing is 1 second 115 milliseconds**.



The total time for the query to run **with indexing is 82 milliseconds.**



The **performance improvement factor is 13.59 or 14 rounded**.

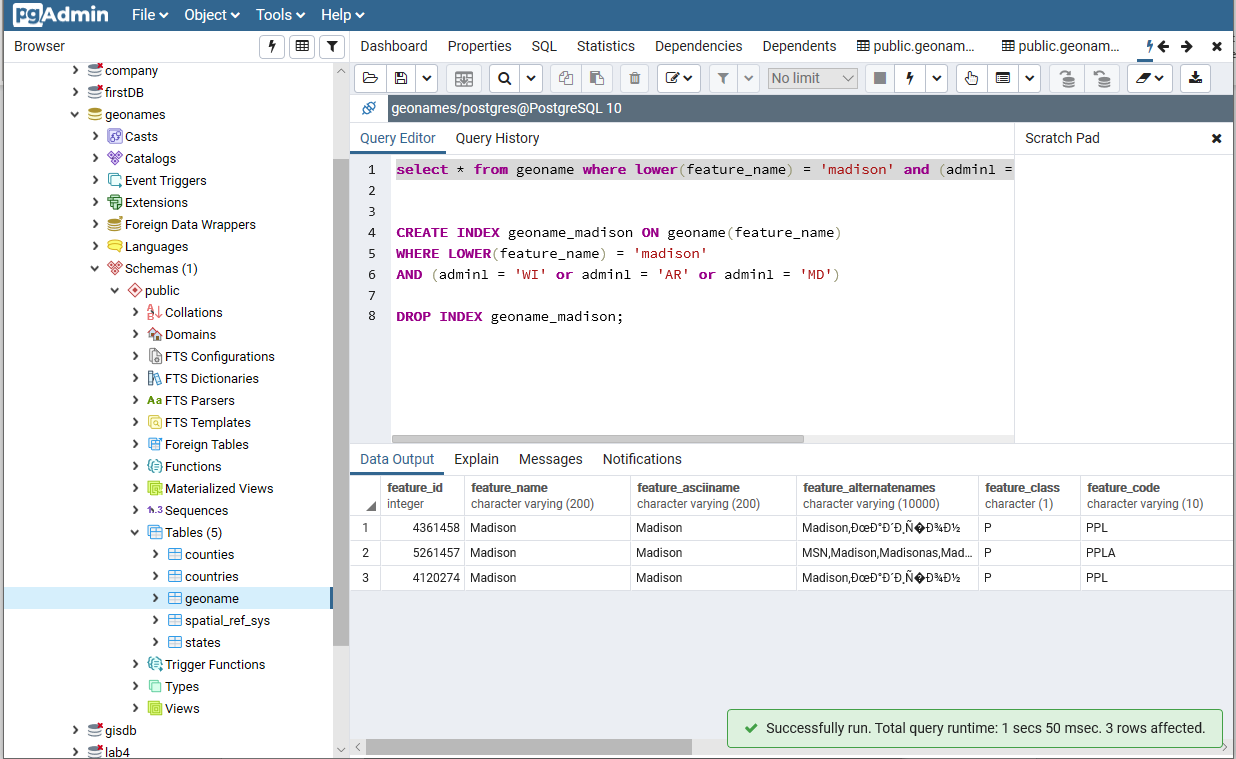
The index creation statement used:

**CREATE INDEX geoname\_madison**

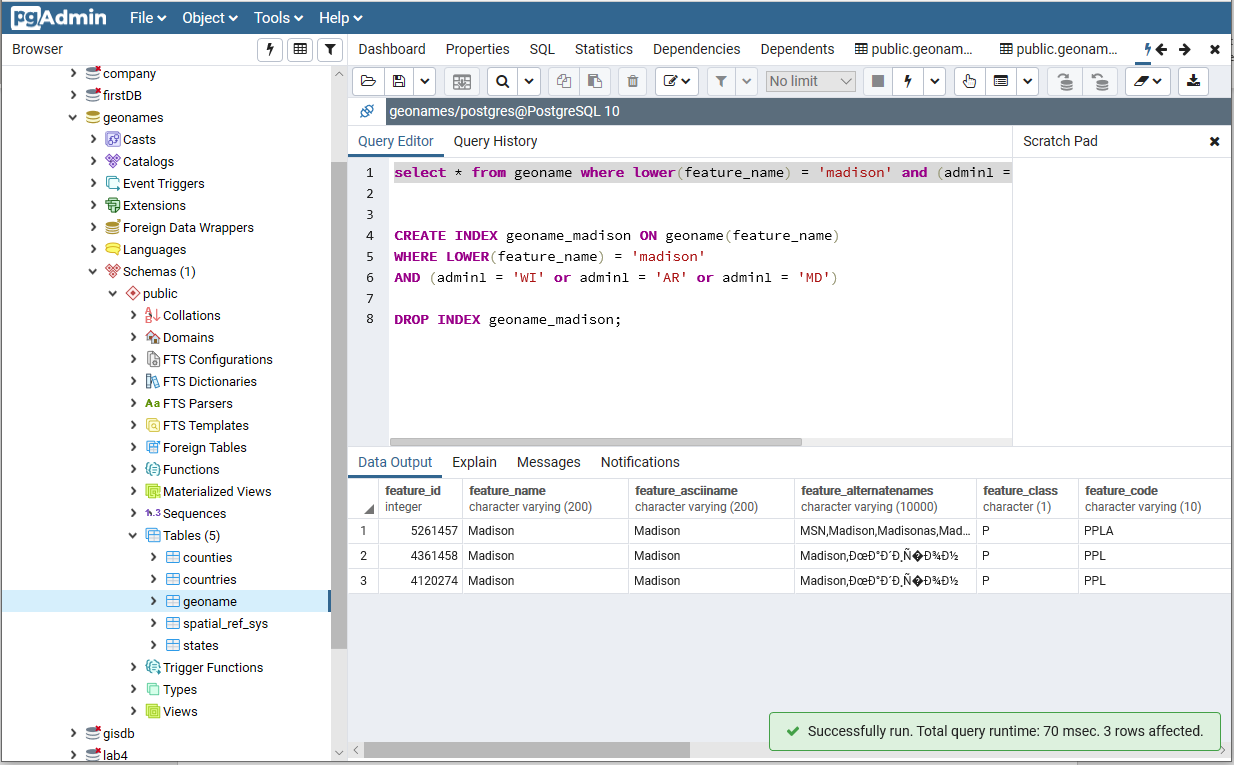
**ON geoname(feature\_name)**

**WHERE LOWER(feature\_name) = 'madison';**

1. The total time for the query to run **without indexing is 1 second 50 milliseconds.**



The total time for the query to run **with indexing is 70 milliseconds.**



The **performance improvement factor is 15**.

The index creation statement used:

**CREATE INDEX geoname\_madison ON geoname(feature\_name)**

**WHERE LOWER(feature\_name) = 'madison'**

**AND (admin1 = 'WI' or admin1 = 'AR' or admin1 = 'MD')**

**Yes –** there is performance improvement with the index creation statement used.The increase in performance is **due to reducing the index size. The smaller index takes less storage, easier to maintain, and faster to scan.**

1. The scenario that had the **best performance is the geoname and states scenario**. This scenario had the best performance due creating indexes on both the geoname and states layers. This allowed for faster searches with the creation and implementation of multiple indexes on multiple layers.

