**Database Homework 3**

**By: Ryan Riopelle**

**Cats**

**1) ## No place to put the index here because there is no where or join clause**

**2)**

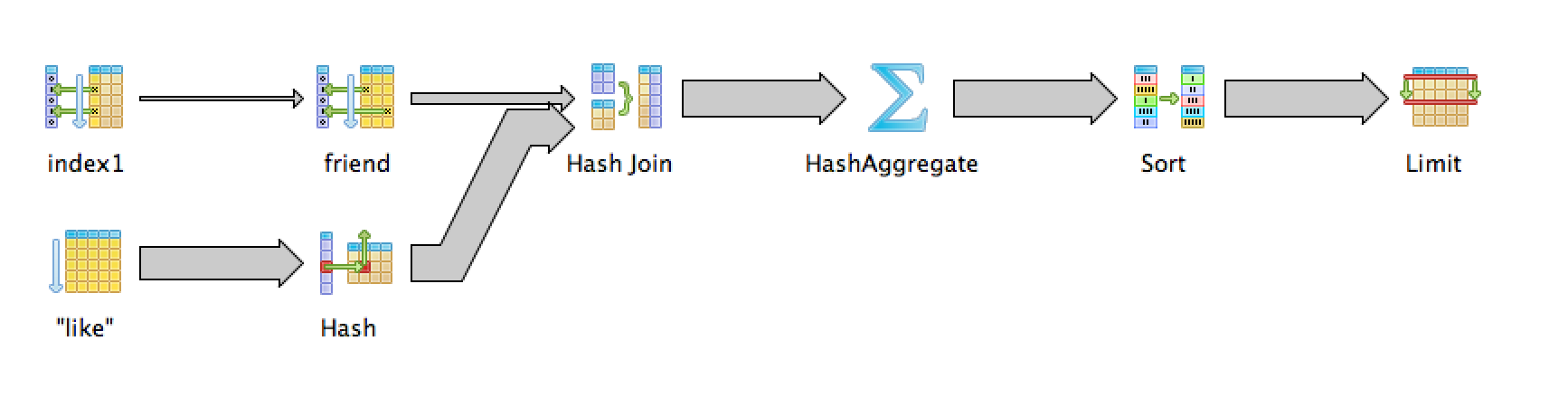
##Cost with no indexes created

"Limit (cost=107391.36..107391.39 rows=10 width=4)"

**# Reduced cost with Index on user\_id in the friends table**

CREATE INDEX index1 ON cats.friend(user\_id);

"Limit (cost=73619.53..73619.56 rows=10 width=4)"

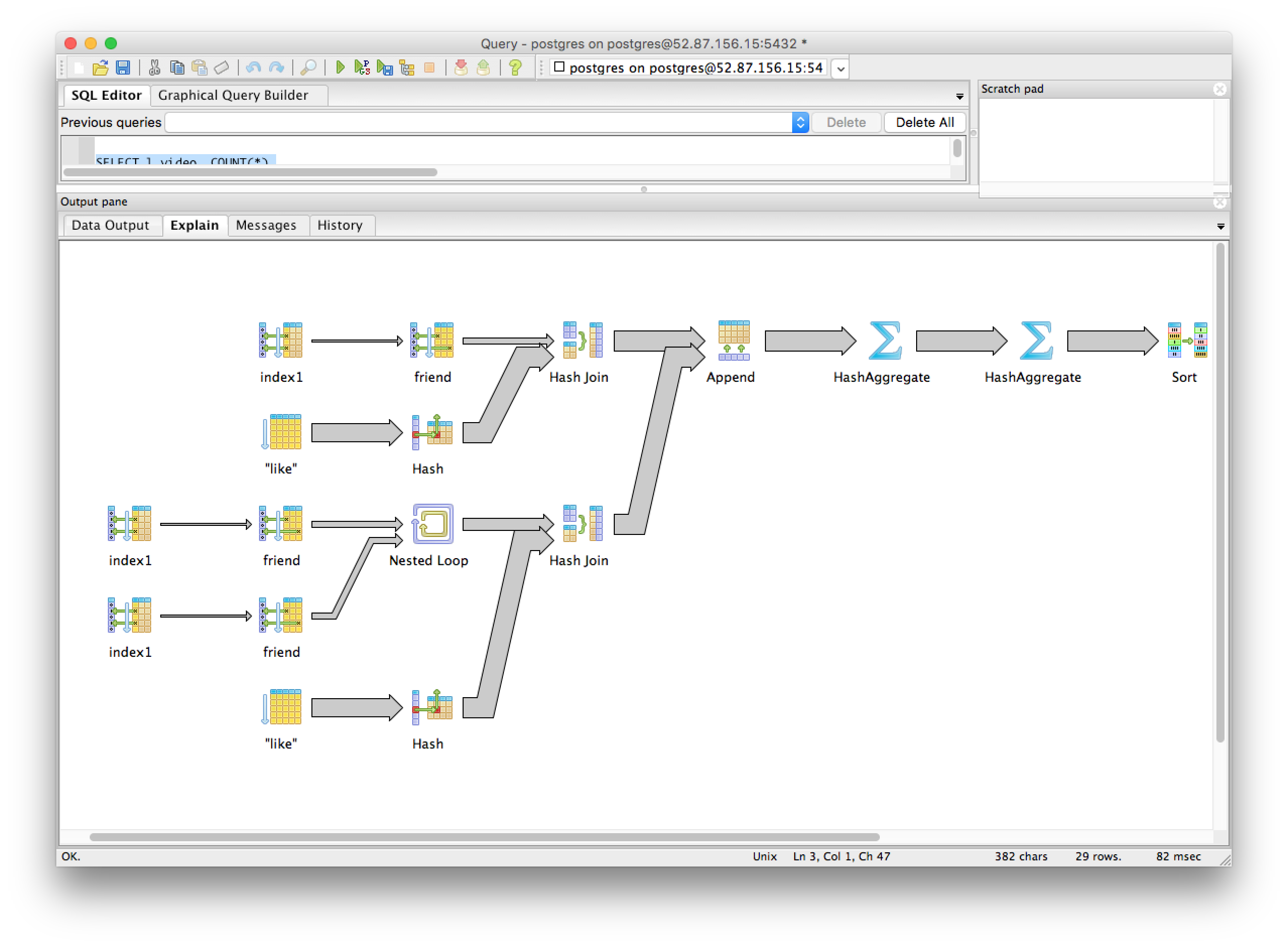
**## 3)**

##Cost with no indexes created

"Limit (cost=284830.76..284830.79 rows=10 width=4)"

## **Reduced Cost With Indexes Created on user\_id in the friends table**

"Limit (cost=152067.65..152067.68 rows=10 width=4)"



**## 4)**

##Cost with no indexes

"Limit (cost=214655.54..214655.57 rows=10 width=4)"

**## Cost was reduced with indexes on the like table for user ID**

CREATE INDEX index1 ON cats.like(user\_id);

"Limit (cost=186603.76..186603.79 rows=10 width=4)"

##Cost with indexes created on video\_id of the like table, this didn't show any improvements over the user\_id

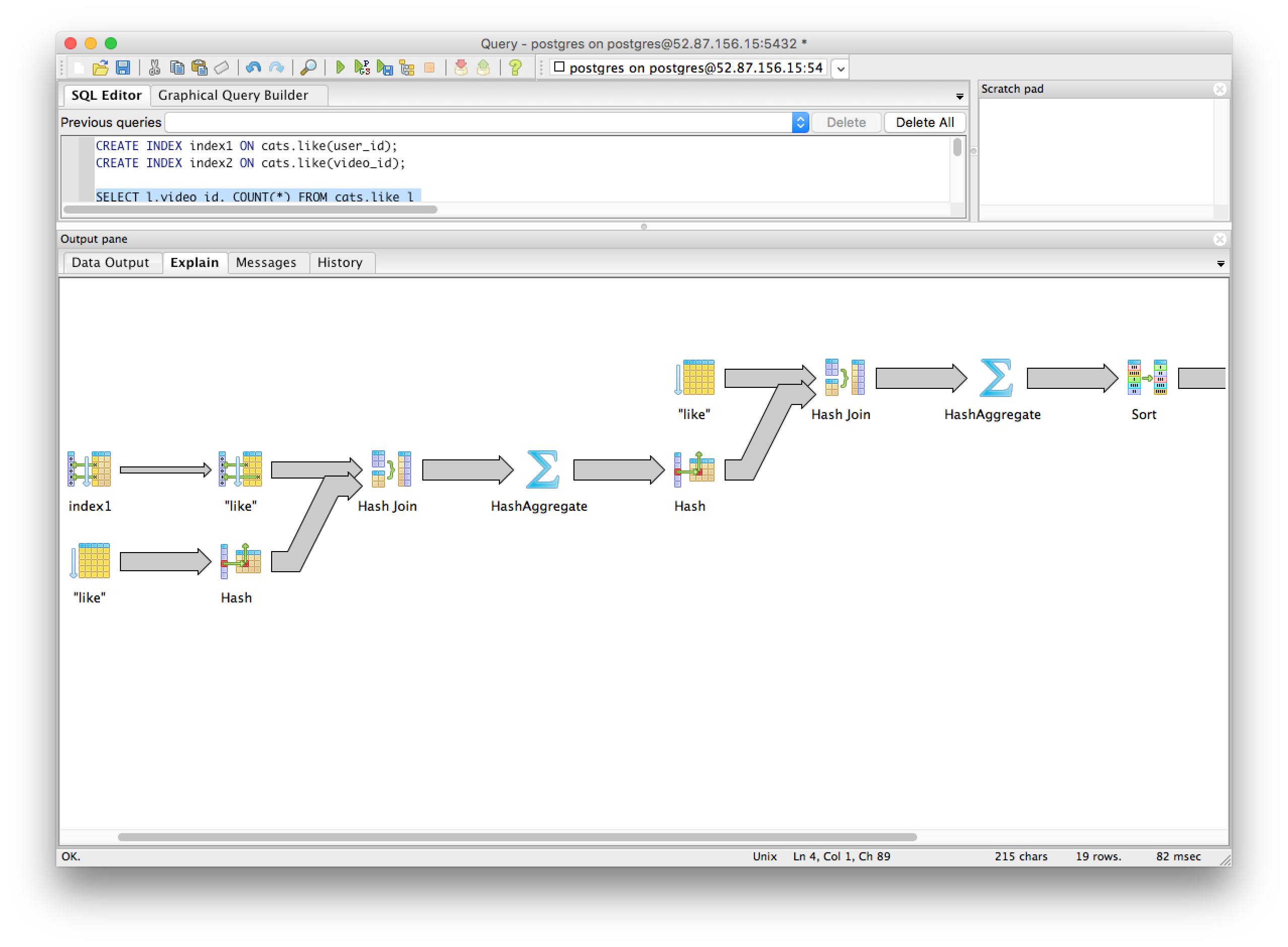
CREATE INDEX index2 ON cats.like(video\_id);

## cost with both

CREATE INDEX index1 ON cats.like(user\_id);

CREATE INDEX index2 ON cats.like(video\_id);

"Limit (cost=186603.76..186603.79 rows=10 width=4)"



**## 5)**

## Cost of queries with no indexes

"Limit (cost=256169.77..256169.80 rows=10 width=12)"

**## reduced cost of queries with indexes on the like\_user**

CREATE INDEX index1 ON cats.like(user\_id);

"Limit (cost=228117.99..228118.02 rows=10 width=12)"

