Introduction to Amazon Redshift

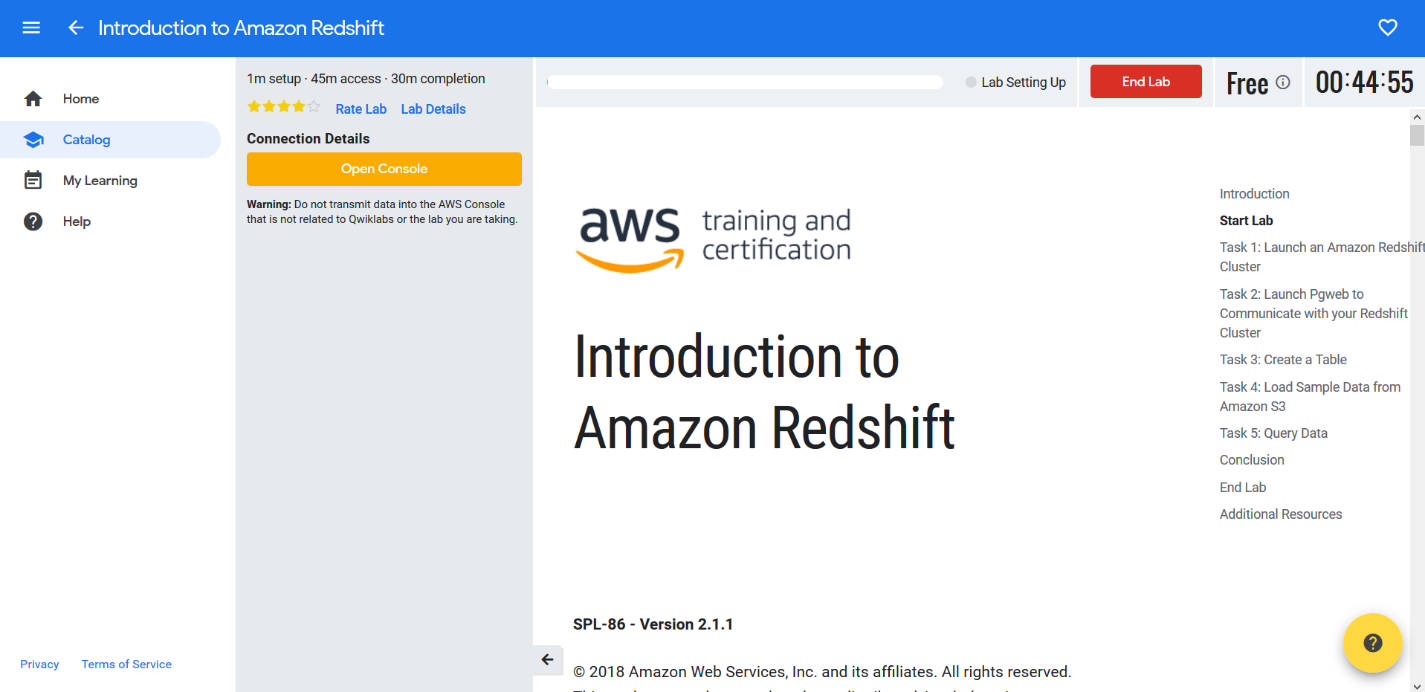
Topics Covered:

1. Launch an Amazon Redshift cluster
2. Connect SQL client to Redshift cluster
3. Load data from S3 bucket into Redshift cluster
4. Run queries against stored in amazon Redshift

Amazon Redshift is a fast, fully managed, petabyte-scale data warehouse service that makes it simple and cost-effective to efficiently analyze all your data using your existing business intelligence tools. It is optimized for datasets ranging from a few hundred gigabytes to a petabyte or more and costs less than $1,000 per terabyte per year, a tenth the cost of most traditional data warehousing solutions.

Lab:

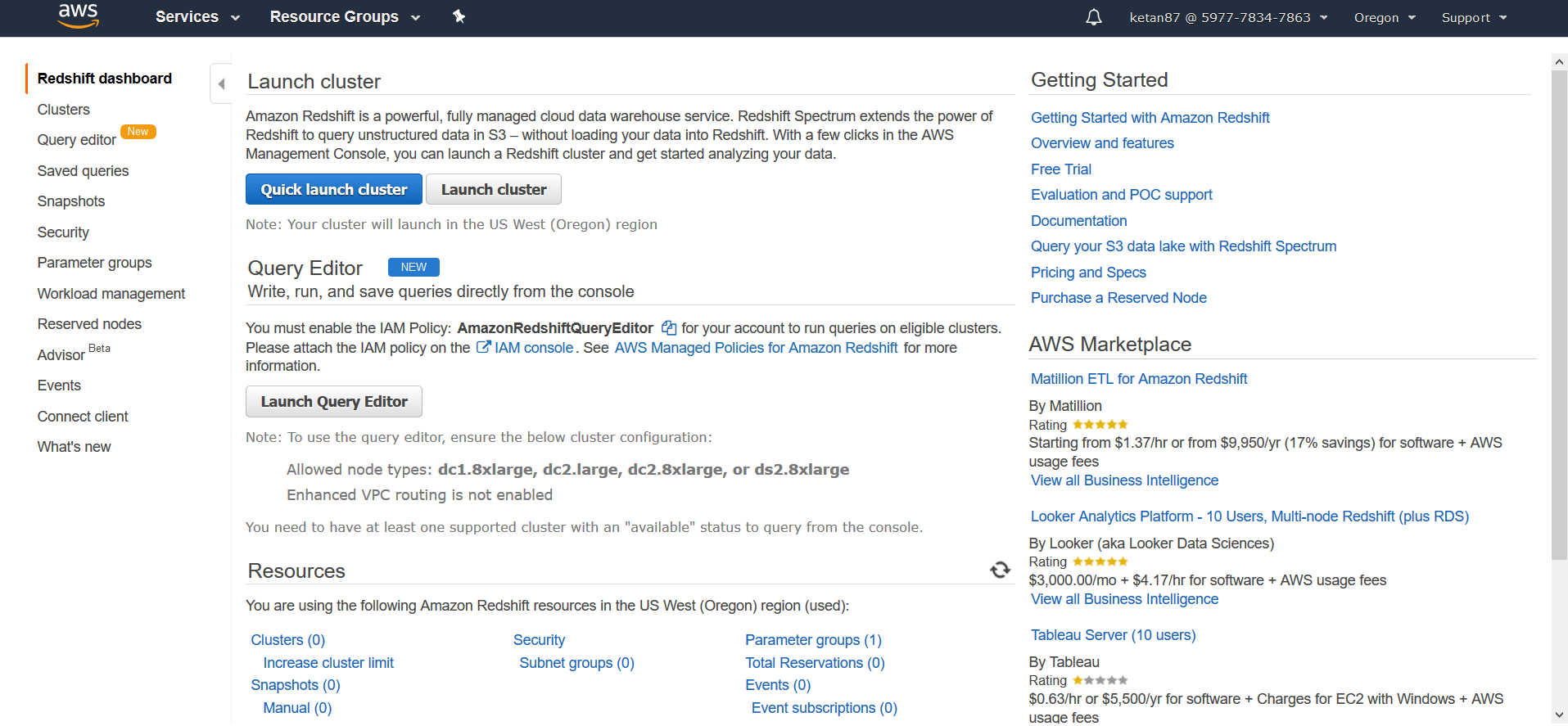
1. Start Lab

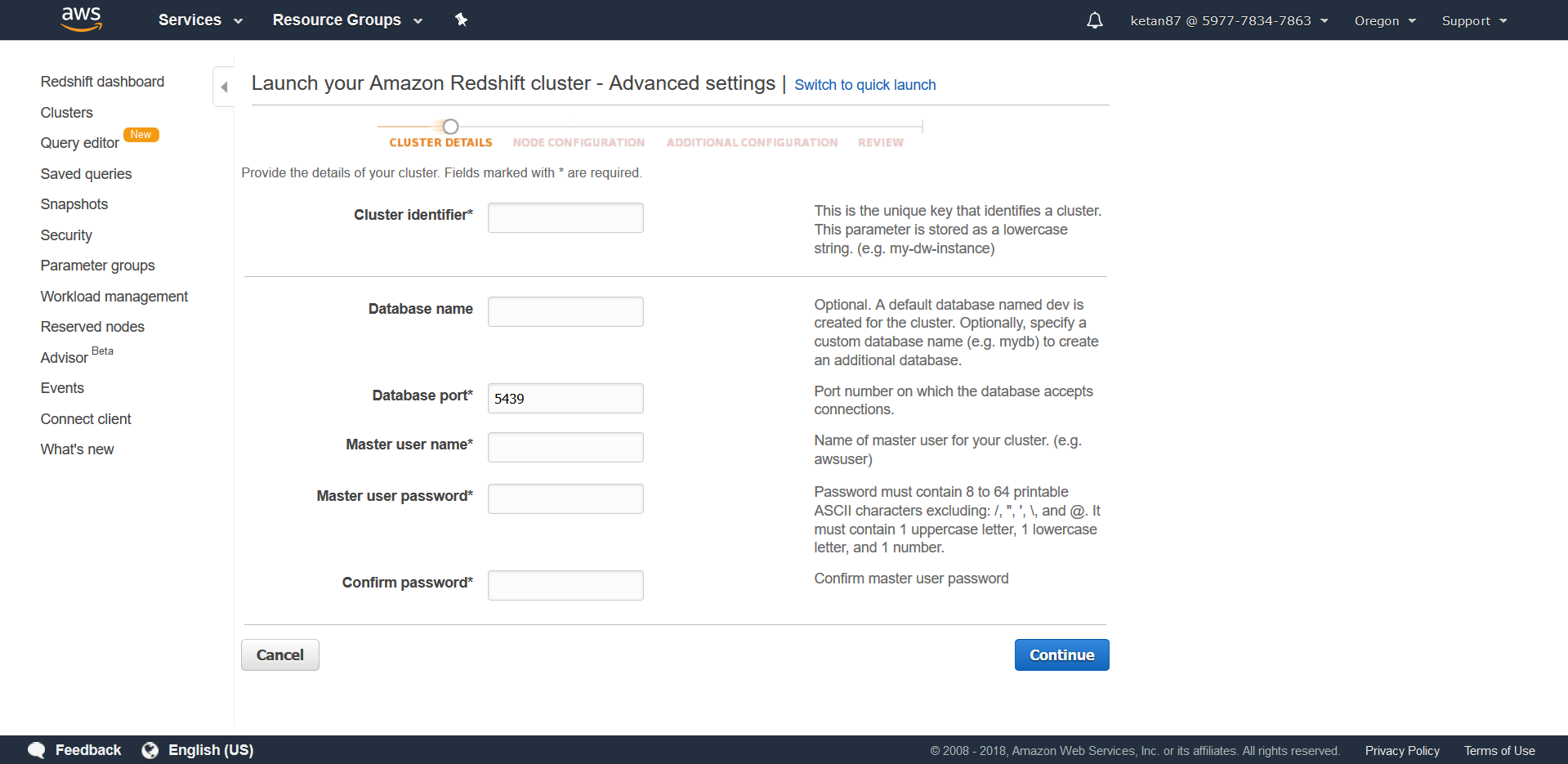


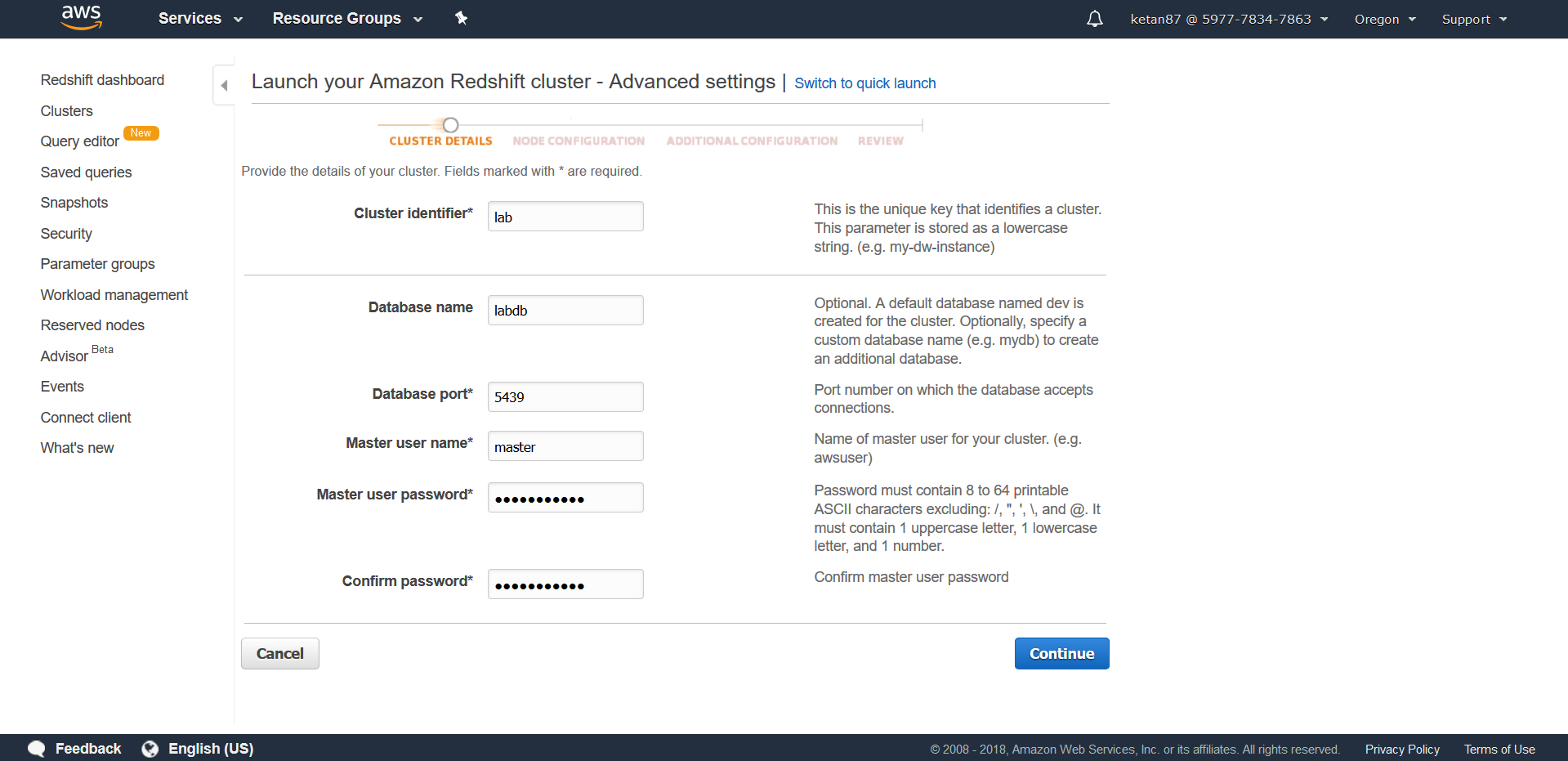
1. Launch an Amazon Redshift Cluster.

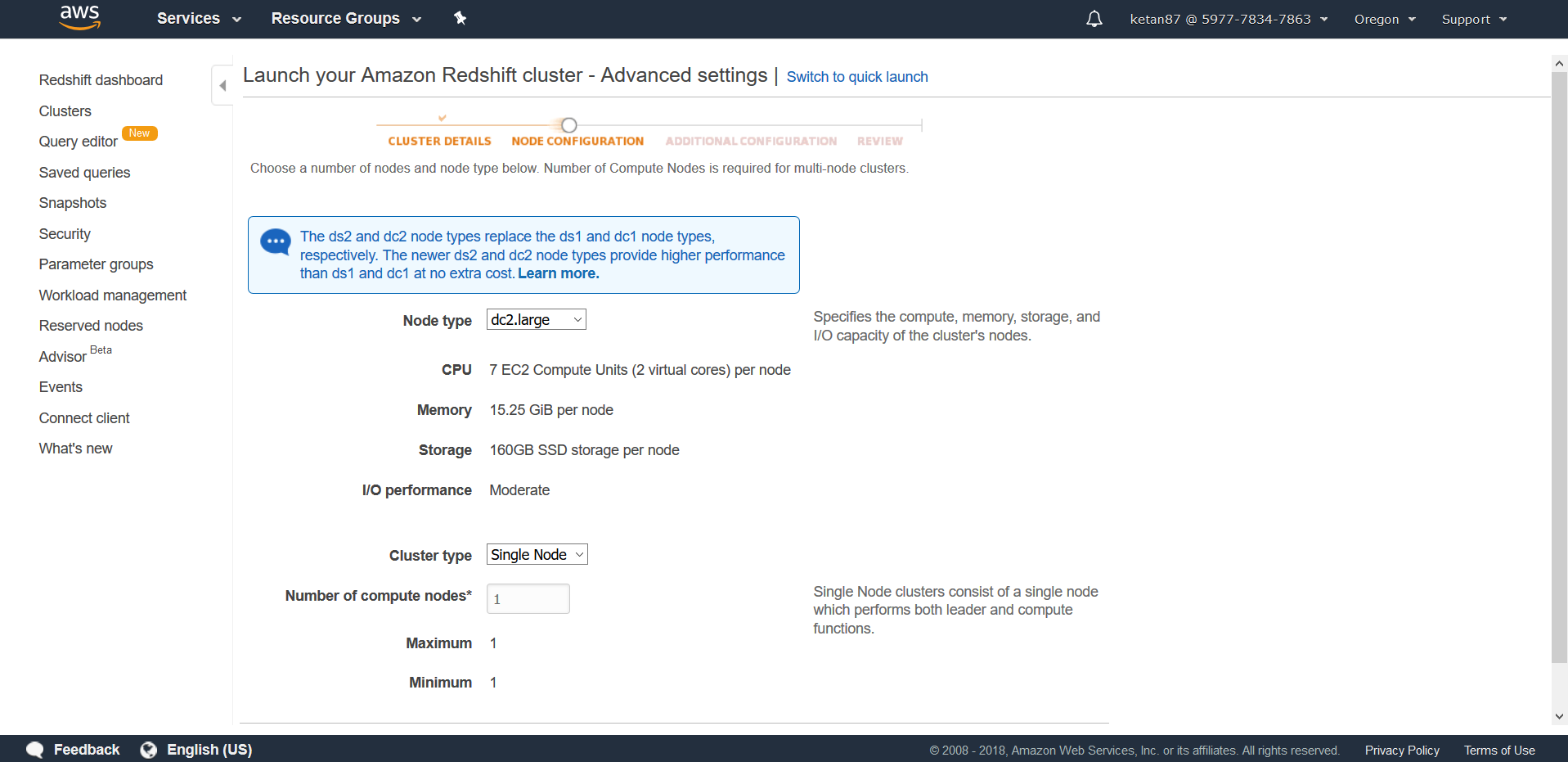
A cluster is a fully managed data warehouse that consists of a set of commute nodes. It runs on Amazon Redshift and contains one or more databases

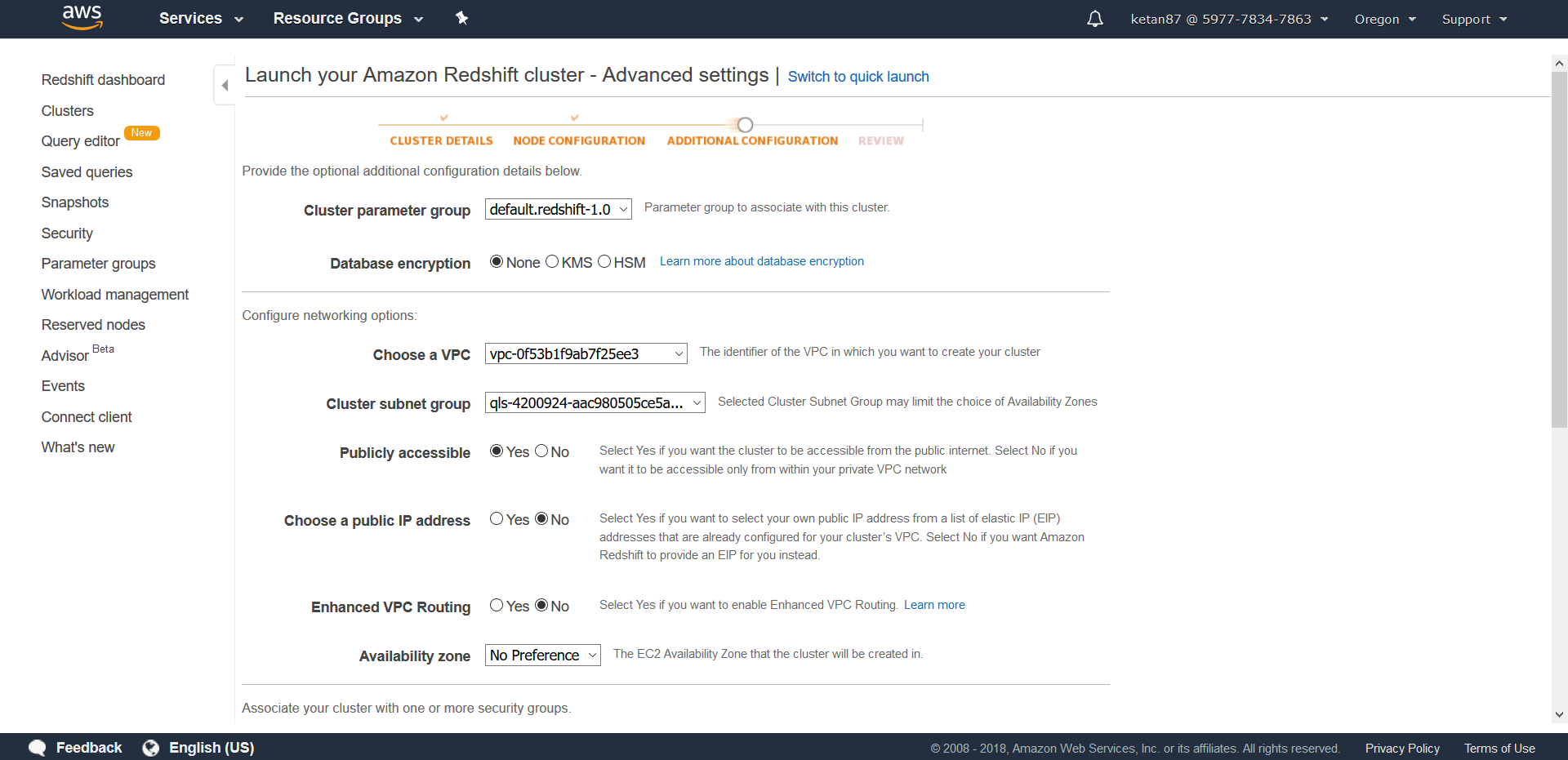
* 1. Launch a cluster in Redshift Cluster Creation Wizard
  2. Provide following details: Cluster identifier: “lab”, Database name: “labdb”, Database Port: “5439”, Master user name: “master”, Master user password: “Redshift123”, Confirm password: “Redshift123”
  3. Select next and keep everything as default.
  4. Review and Click Launch Cluster
  5. Make sure the cluster is up and running
  6. Review the Cluster properties

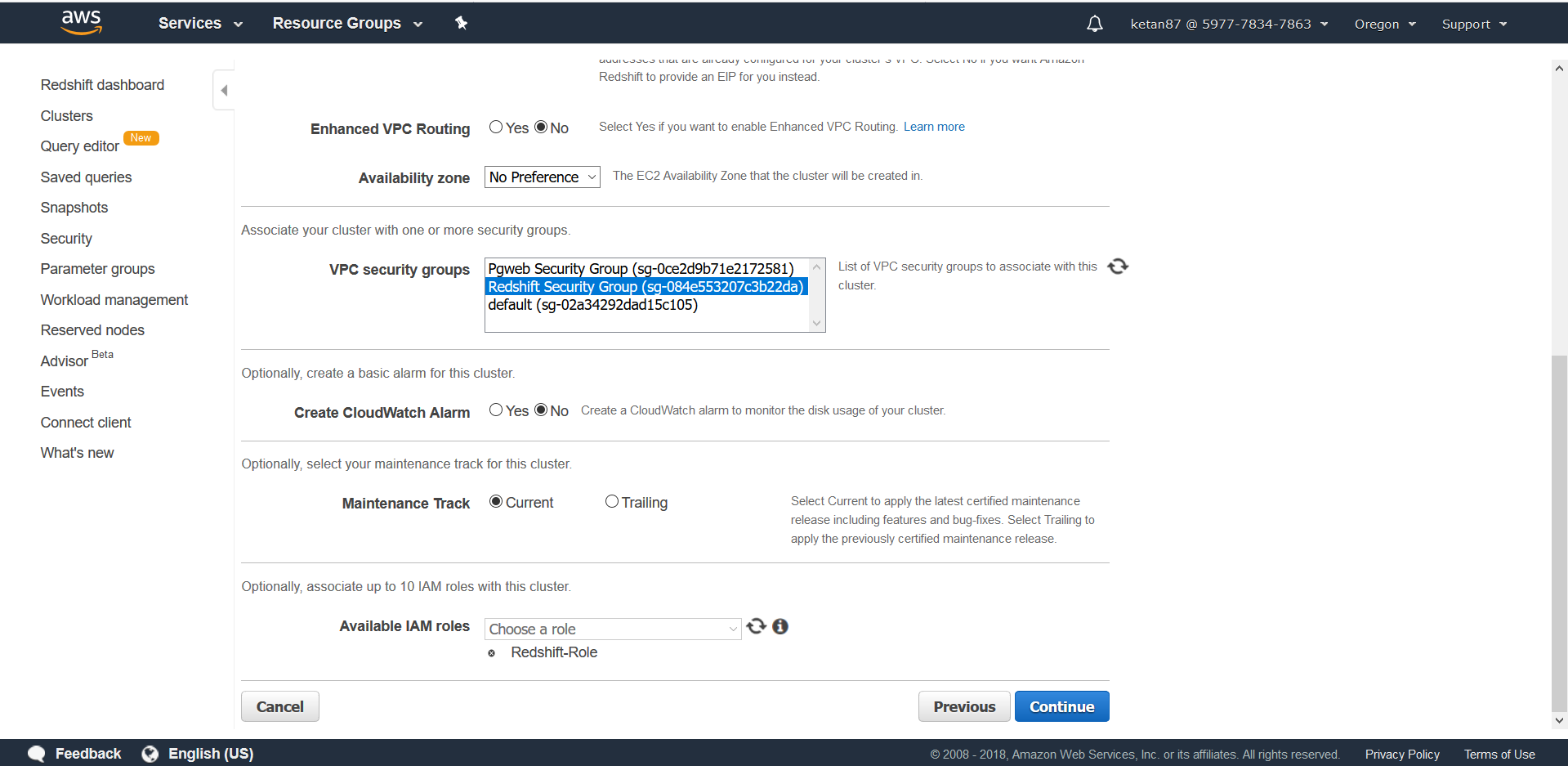


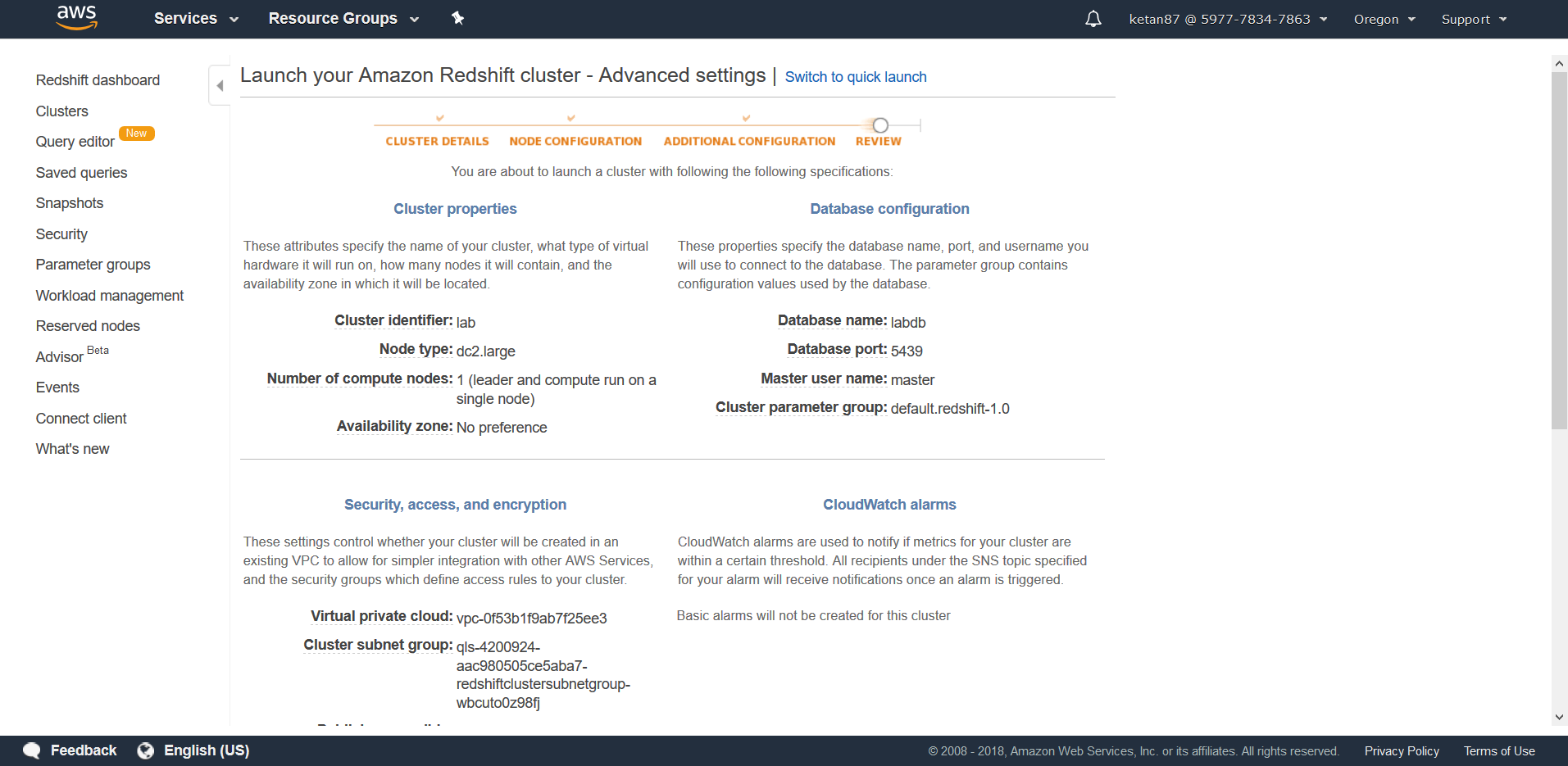


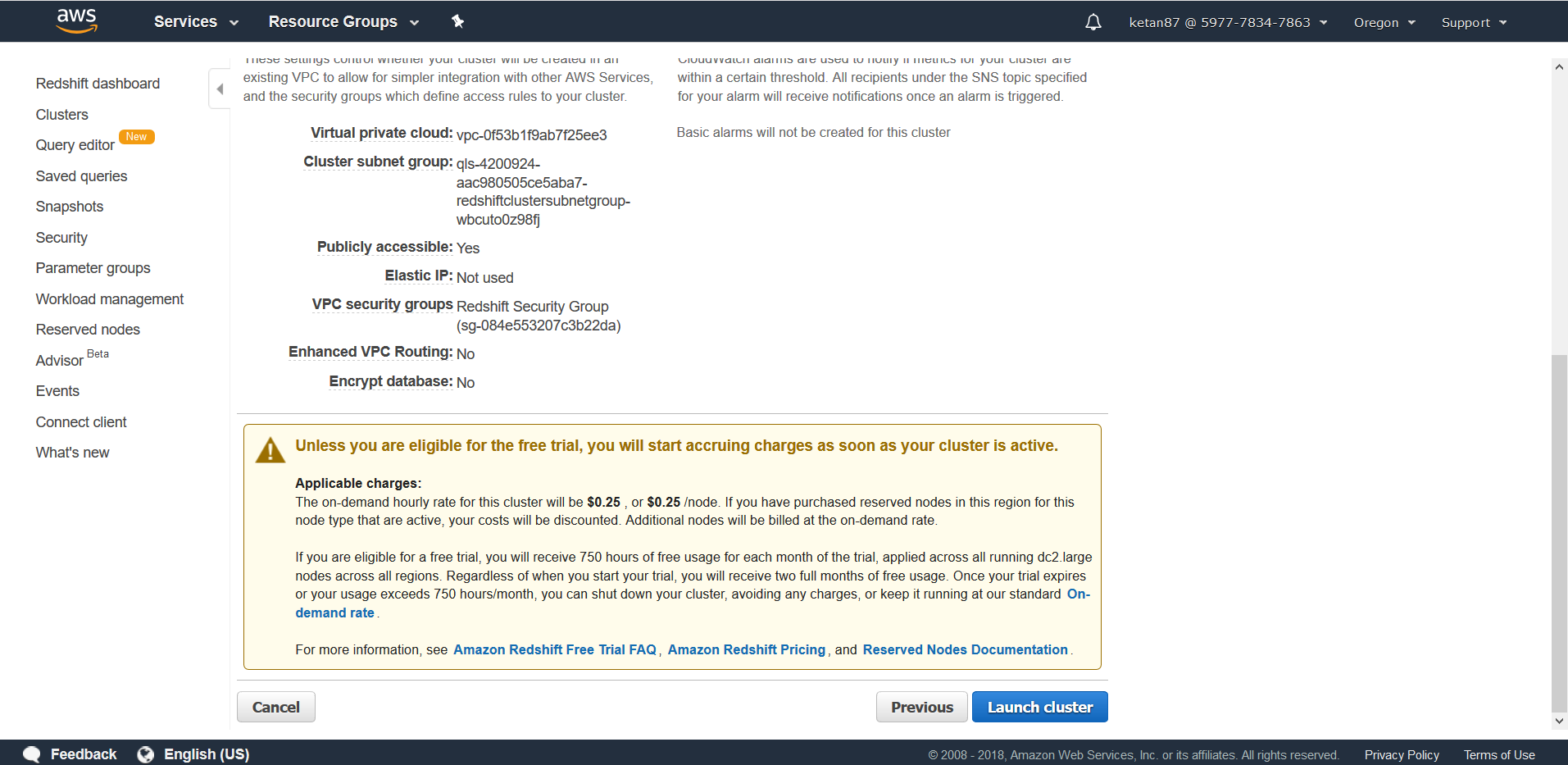


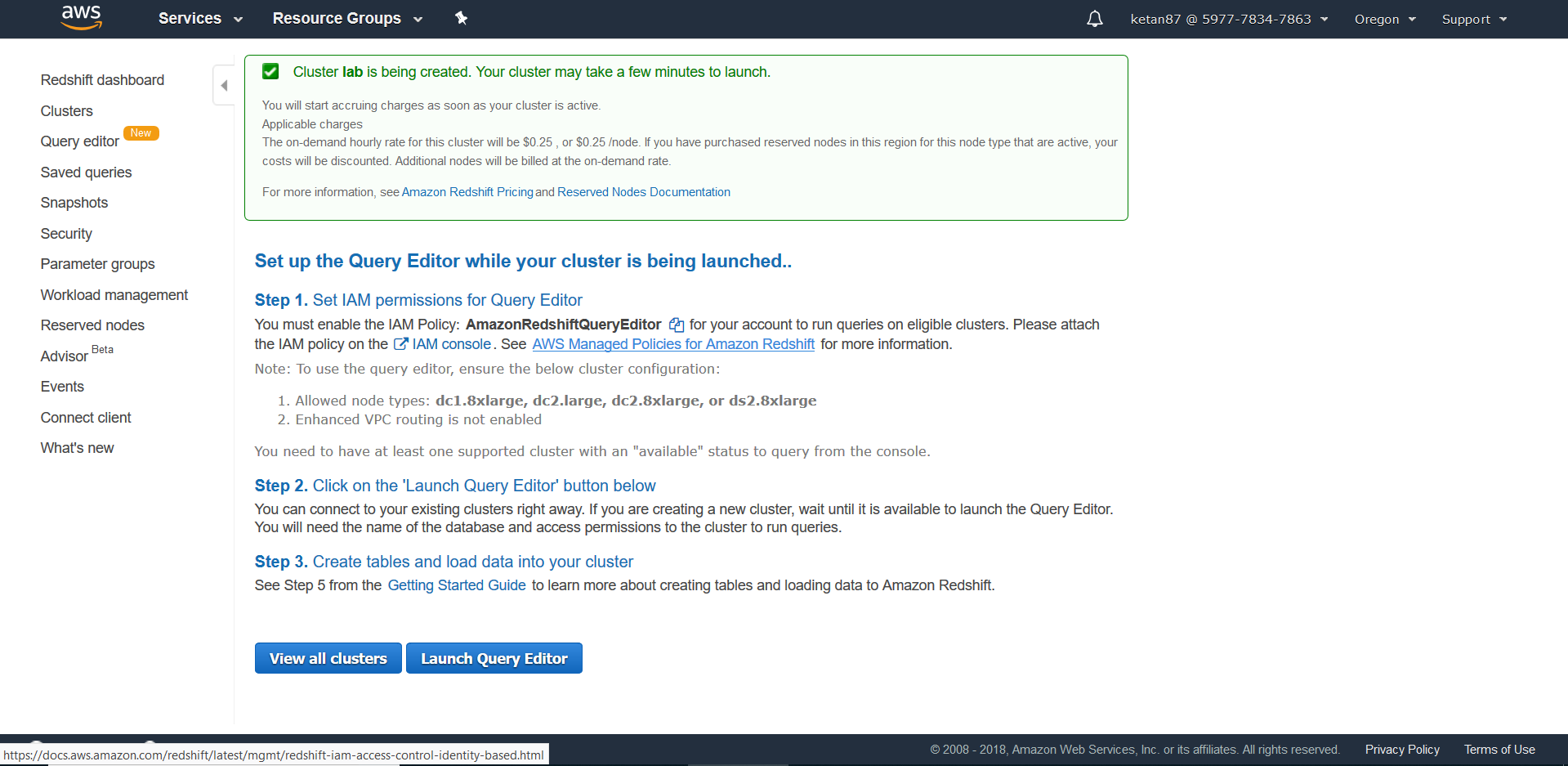


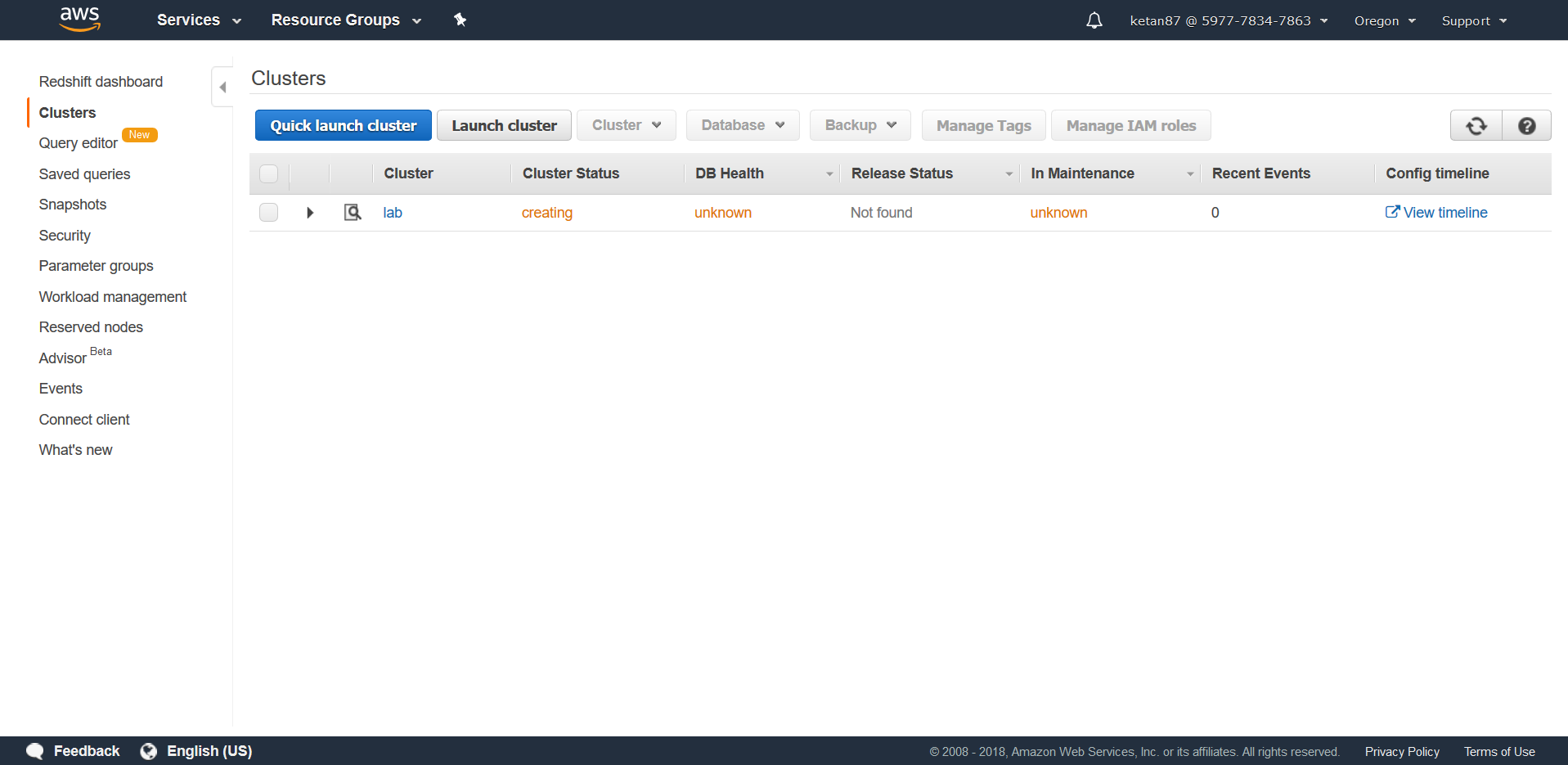


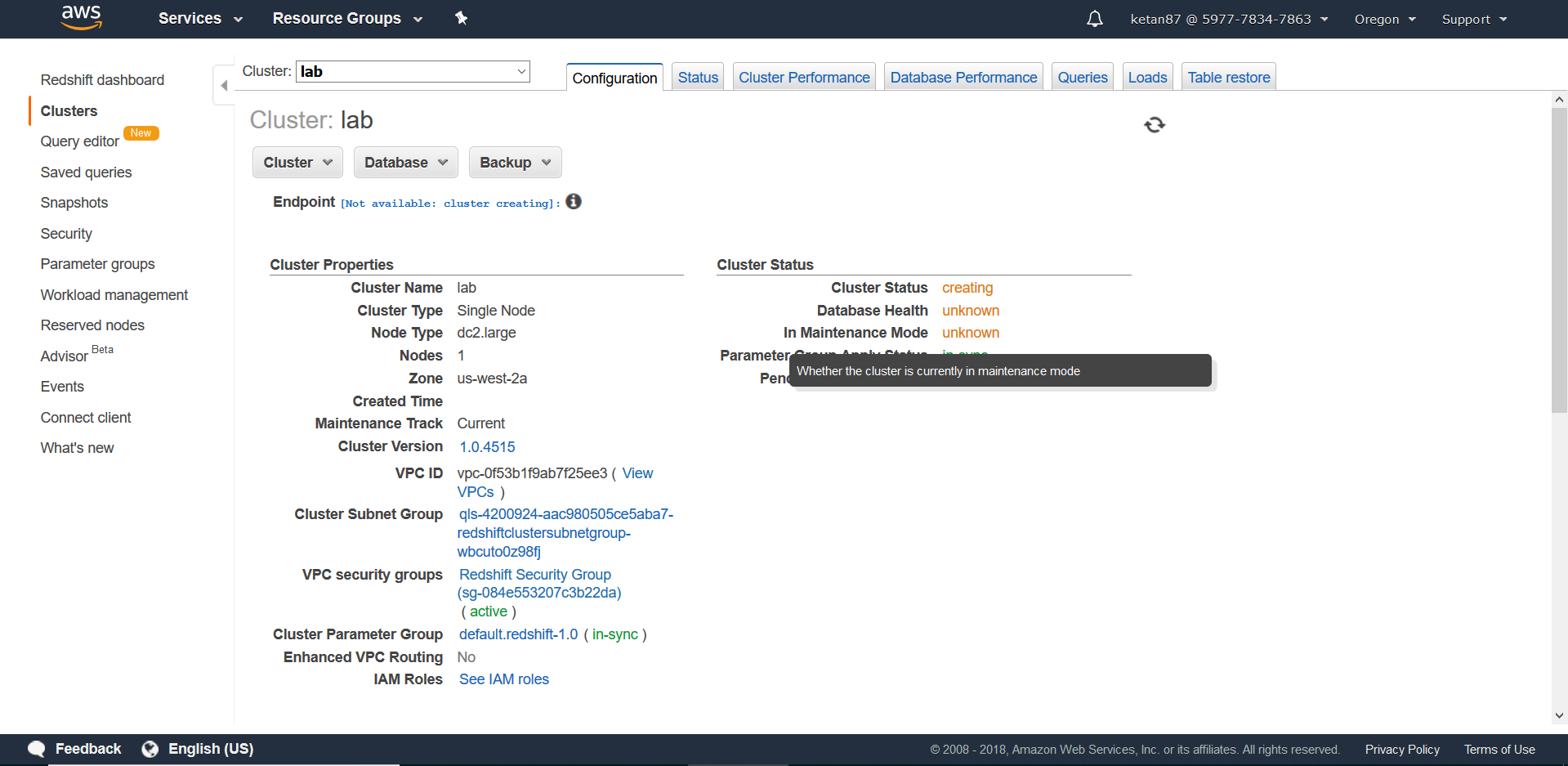




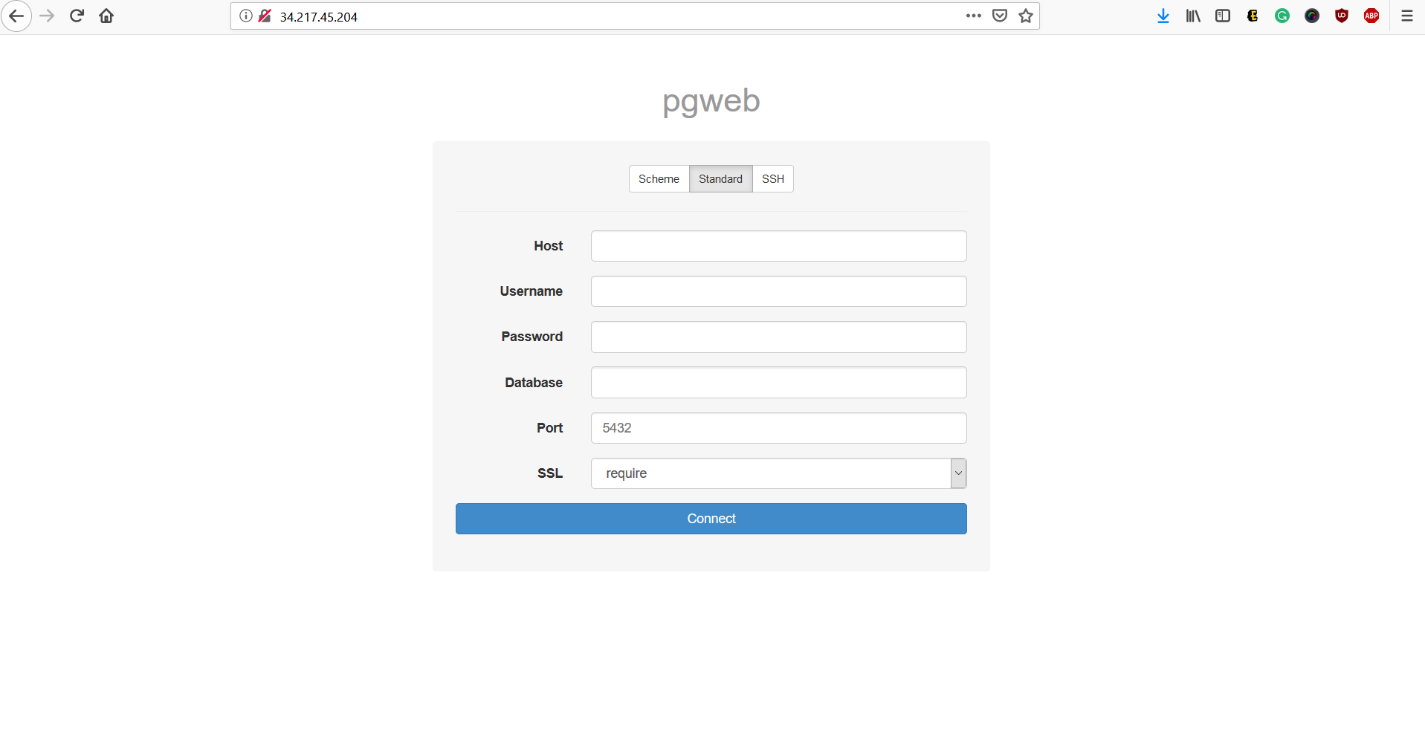


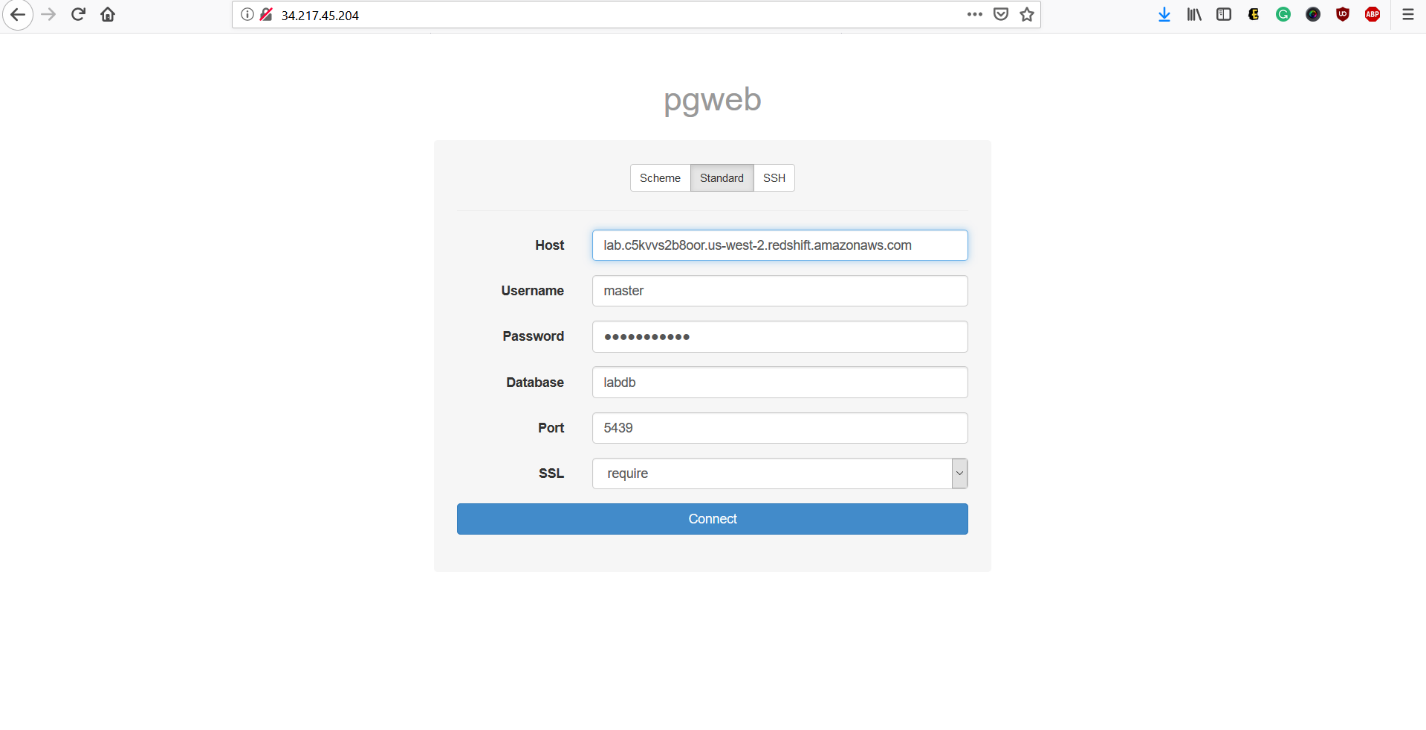






1. Launch pgweb to communicate with your redshift cluster
   1. Make note of pgweb IP and open it in a browser
   2. Make a note of your cluster – lab endpoint in the cluster property section
   3. Connect to the Redshift via SQL Client using the credentials entered while creating the Redshift cluster database





1. Create a Table
   1. In the SQL script section in pgweb, paste the following query and execute

CREATE TABLE users (

userid INTEGER NOT NULL,

username CHAR(8),

firstname VARCHAR(30),

lastname VARCHAR(30),

city VARCHAR(30),

state CHAR(2),

email VARCHAR(100),

phone CHAR(14),

likesports BOOLEAN,

liketheatre BOOLEAN,

likeconcerts BOOLEAN,

likejazz BOOLEAN,

likeclassical BOOLEAN,

likeopera BOOLEAN,

likerock BOOLEAN,

likevegas BOOLEAN,

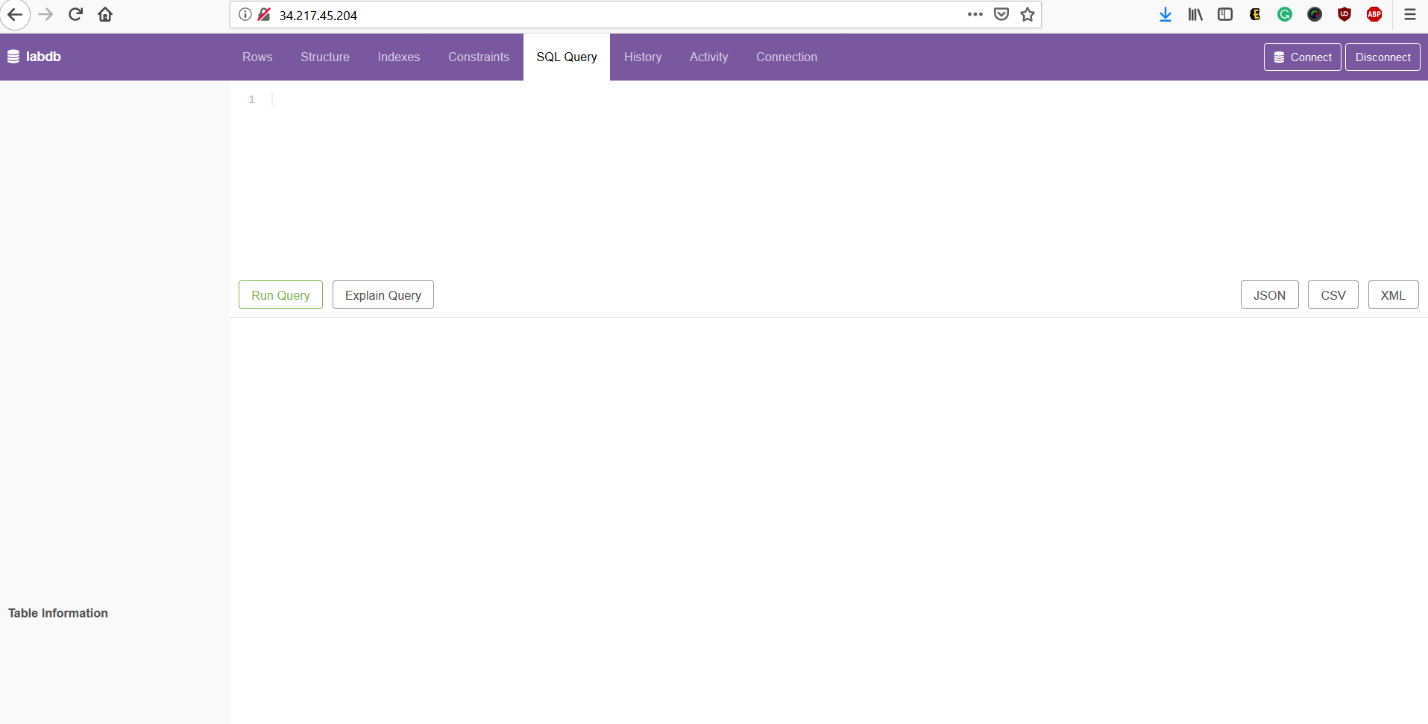
likebroadway BOOLEAN,

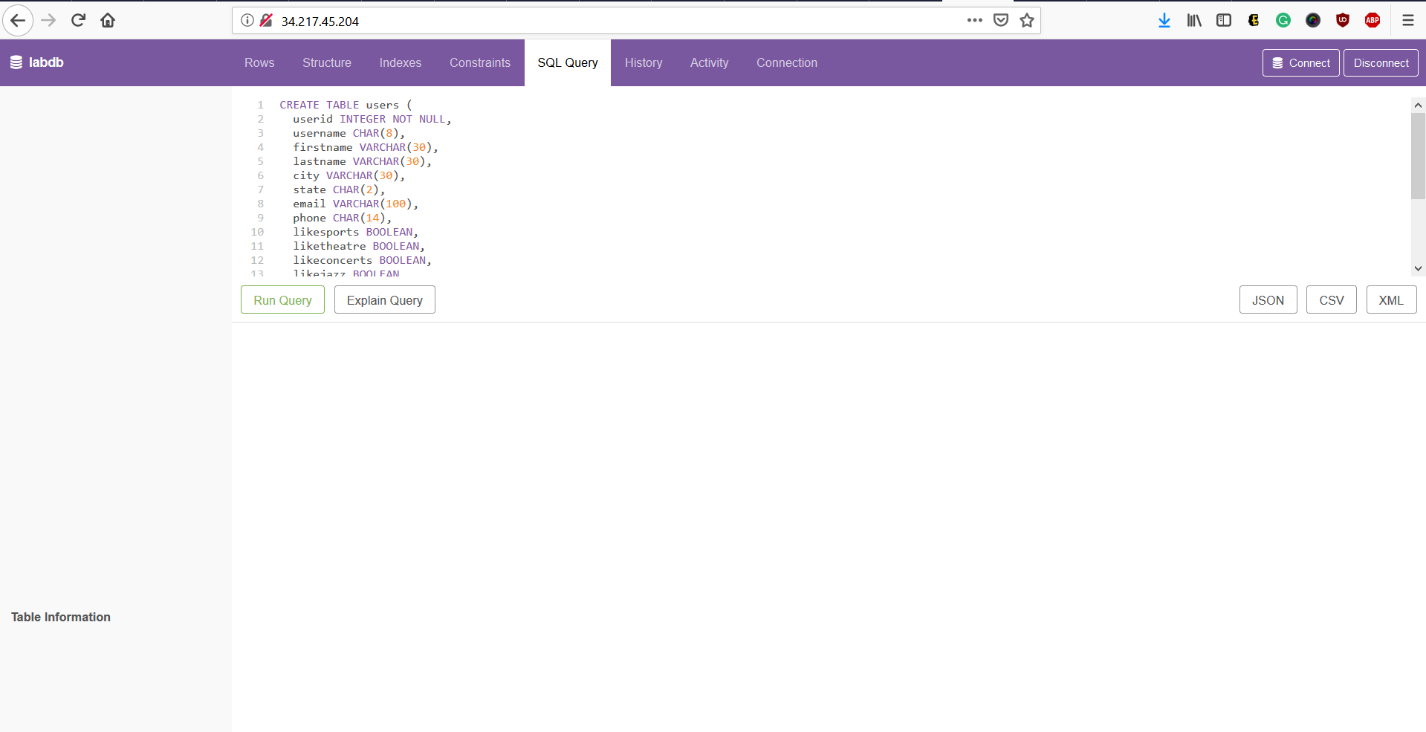
likemusicals BOOLEAN

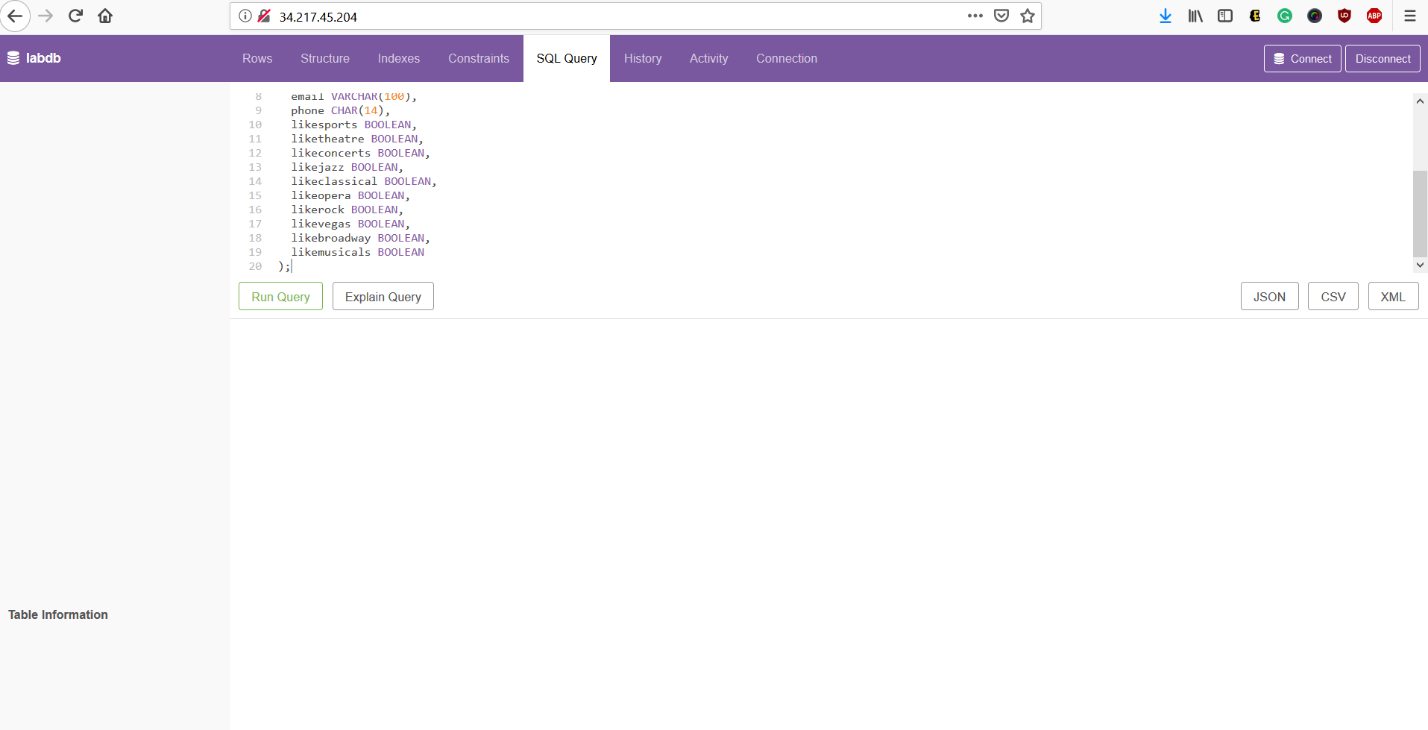
);

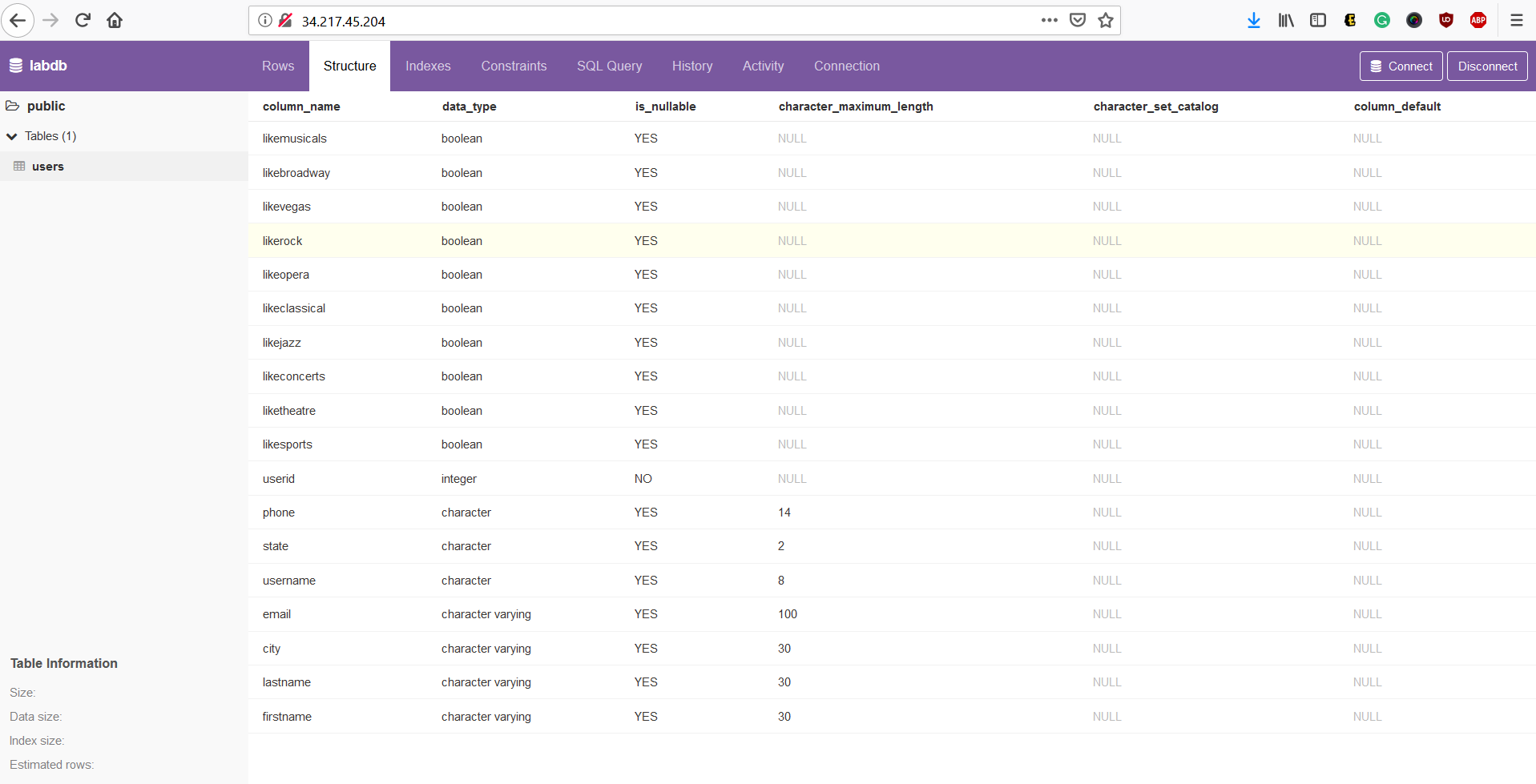
This creates a User Table with the attributes of it.

* 1. The structure of the table can be seen by clicking the user table and selecting structure









1. Loading Sample Data from Amazon S3 to Redshift cluster. Redshift can import data from Amazon S3 through various file formats
   1. In SQL Query tab, run the following query,

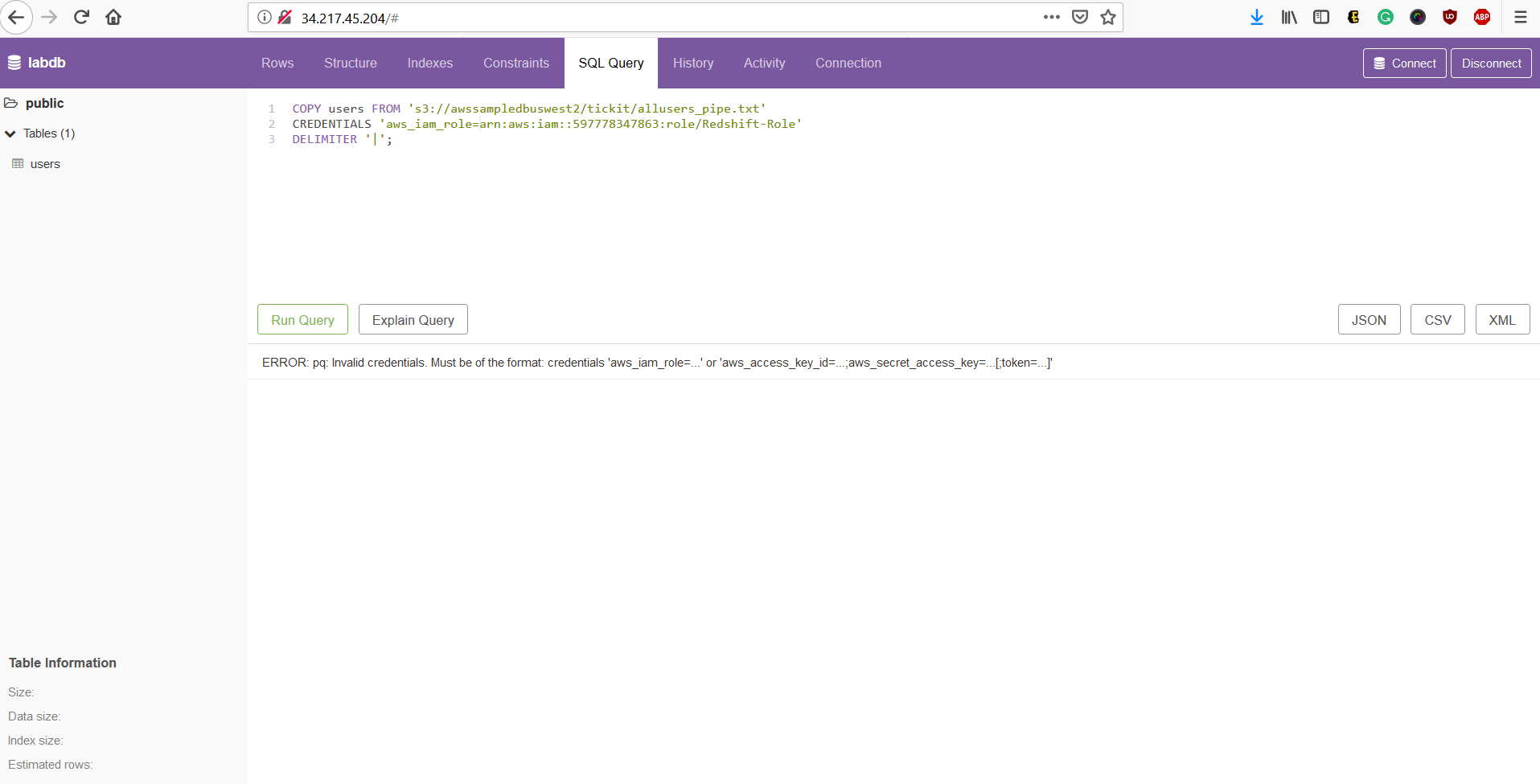
COPY users FROM 's3://awssampledbuswest2/tickit/allusers\_pipe.txt'

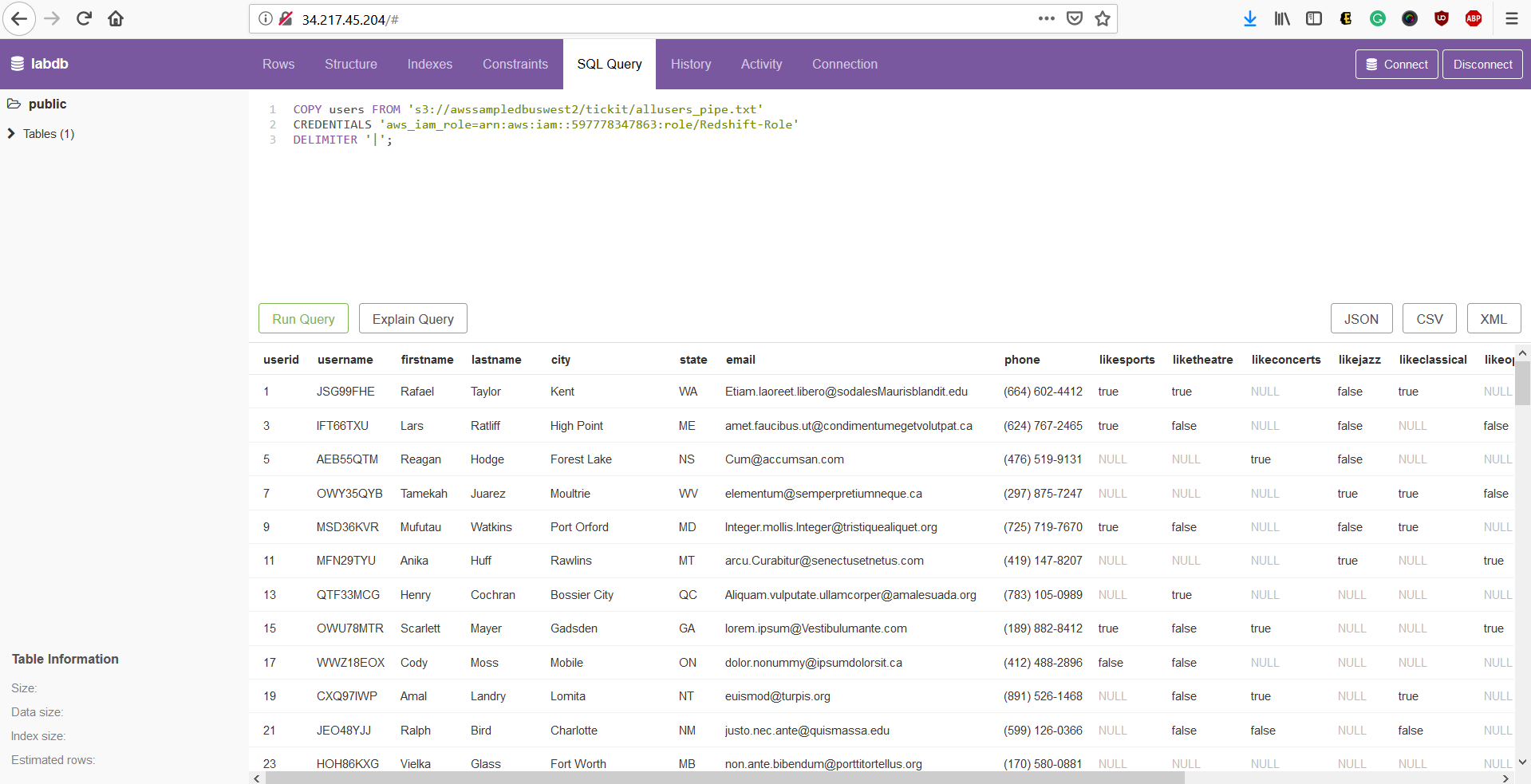
CREDENTIALS 'aws\_iam\_role=arn:aws:s3:::mybucket300987'

DELIMITER '|';

We are passing IAM role which gives the Amazon Redshift to access and pull data from the Amazon S3 bucket

* 1. Post run, there will be around 49,990 rows added to the User Table which can be seen in the rows





1. Query Data. Once data is ingested into Amazon Redshift, you can run query as per your requirement
   1. To compute the number of rows:

SELECT COUNT(\*) FROM users;

* 1. To display users in Ohio who like sports but not opera and sort them as per name:

SELECT userid, firstname, lastname, city, state

FROM users

WHERE likesports AND NOT likeopera AND state = 'OH'

ORDER BY firstname;

* 1. To display top 10 cities who like Jazz:

SELECT

city,

COUNT(\*) AS count

FROM users

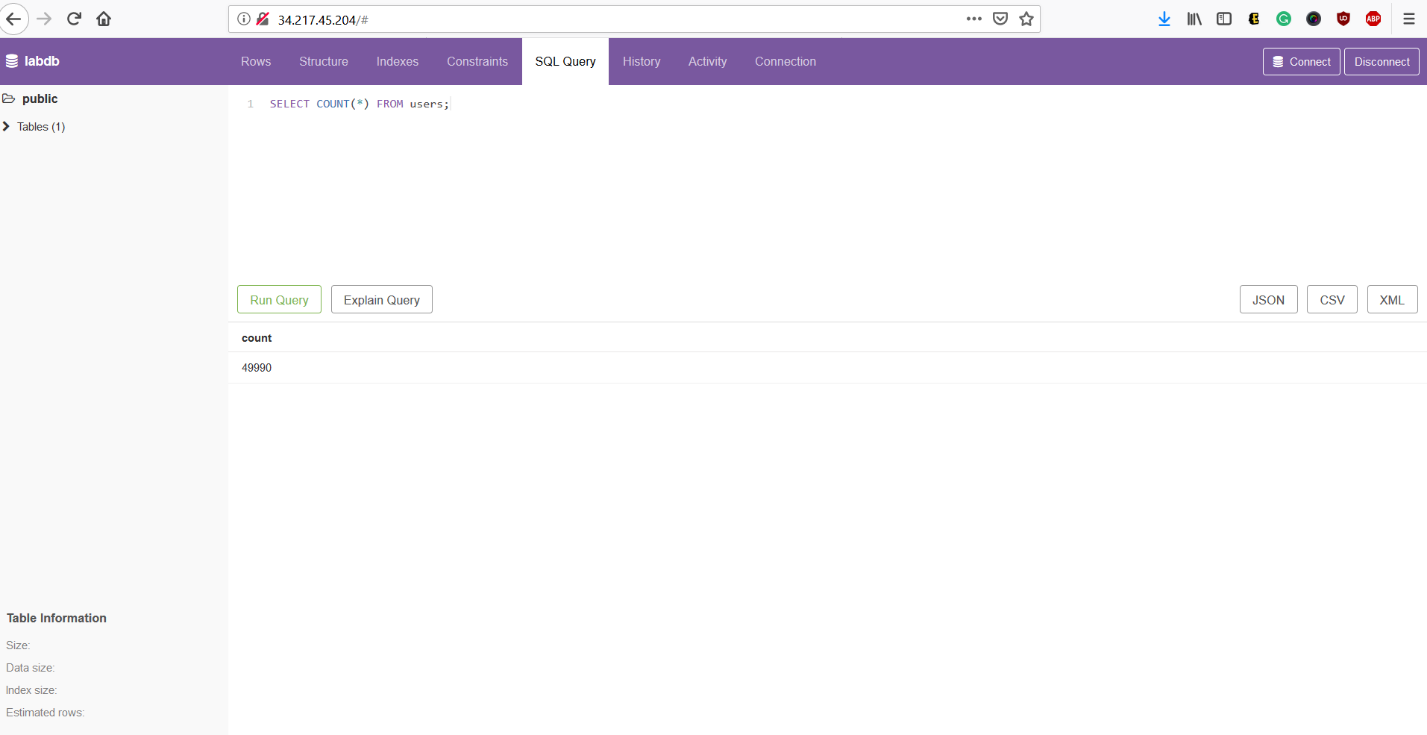
WHERE likejazz

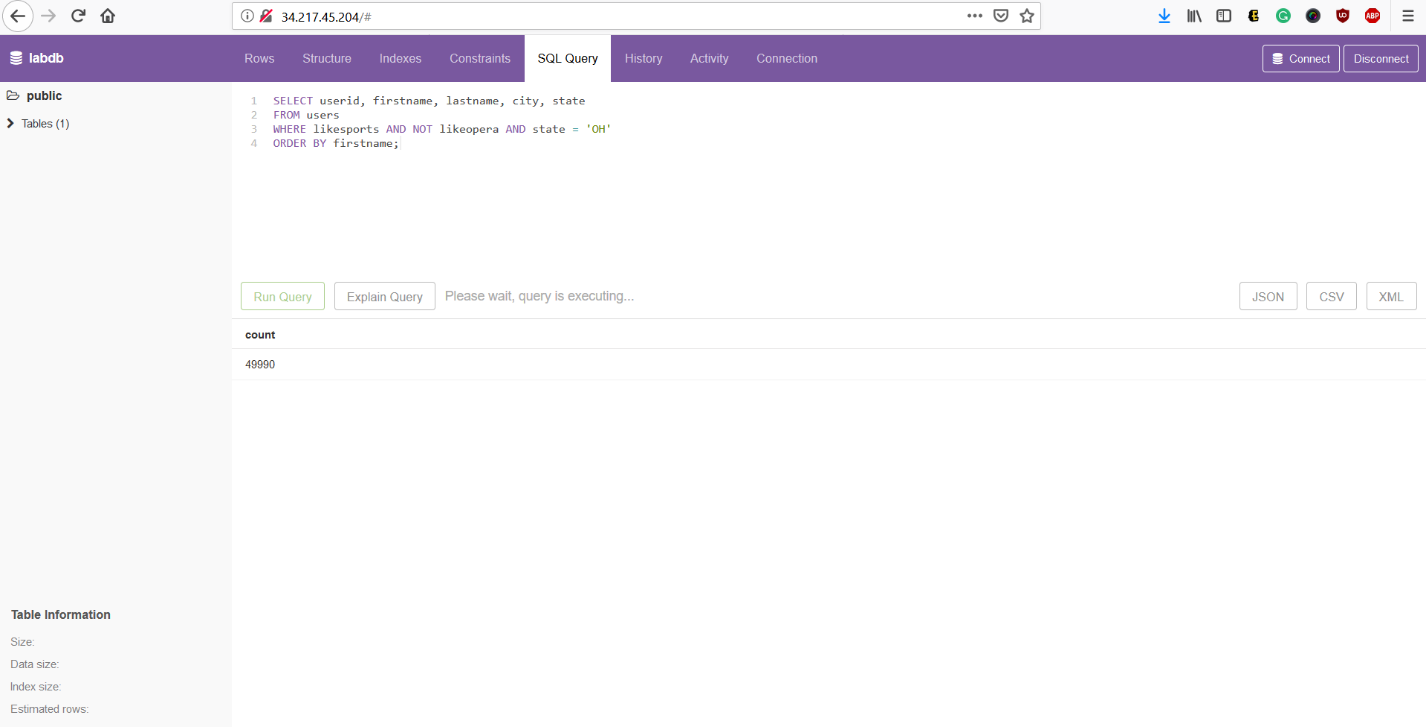
GROUP BY city

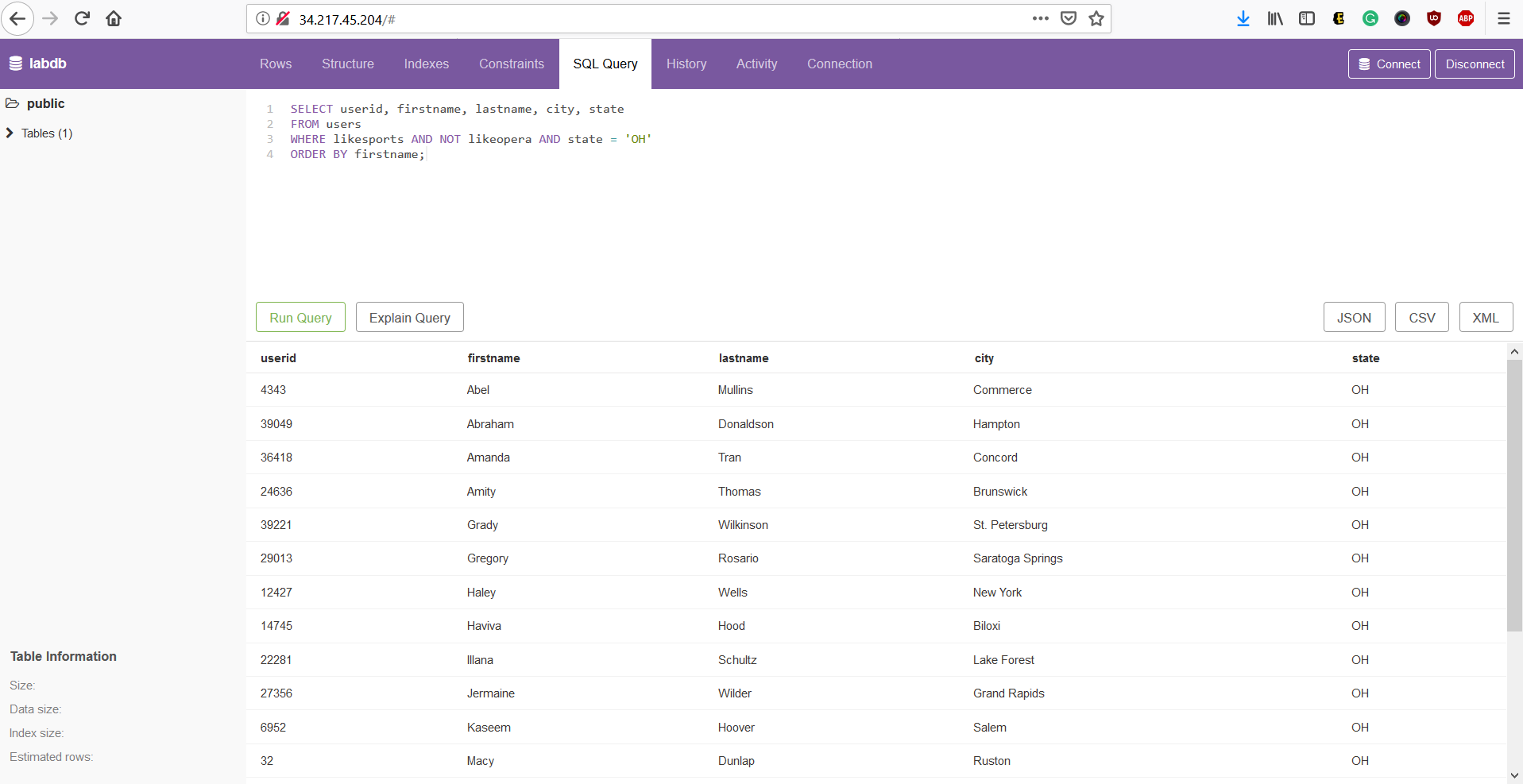
ORDER BY count DESC

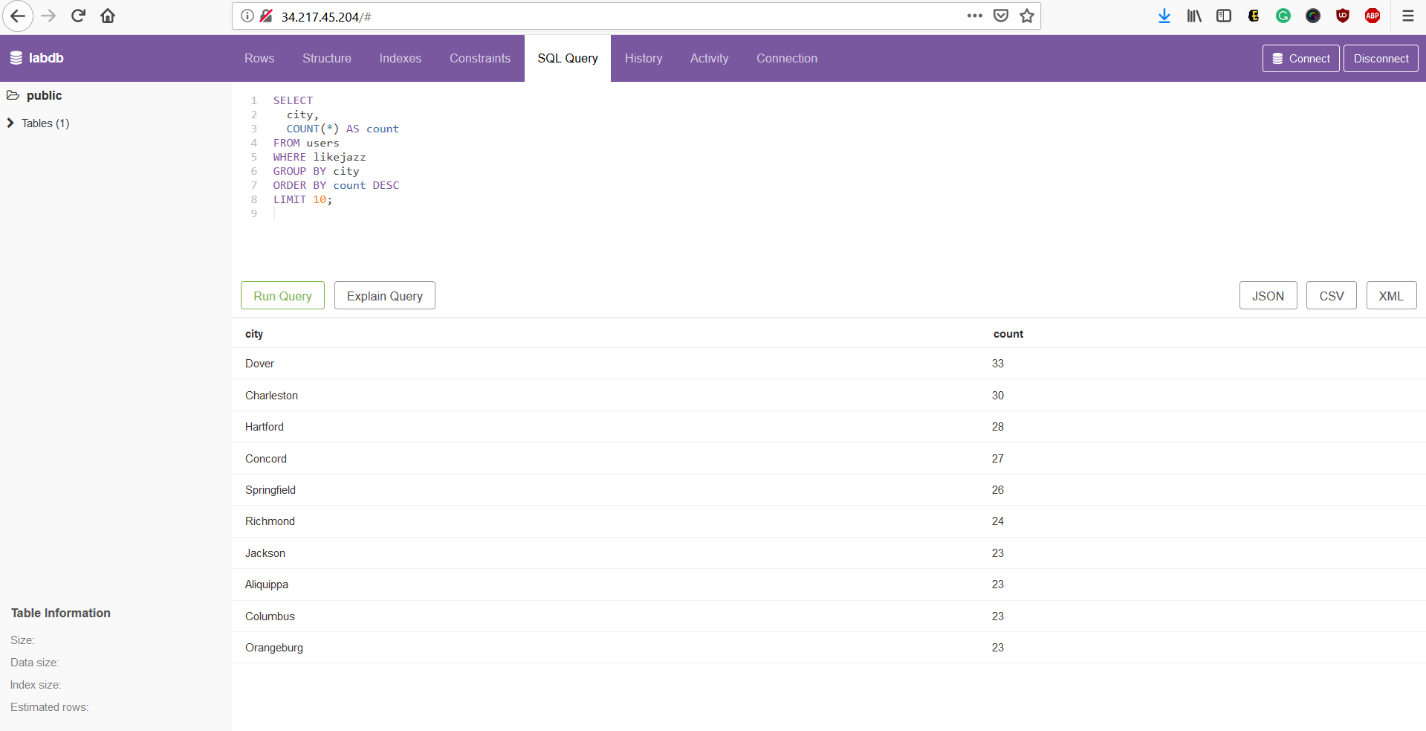
LIMIT 10;

* 1. One can run any query as per their requirement









1. End Lab

Conclusion:

We learned how to:

1. Launch an Amazon Redshift cluster
2. Connect SQL client to Redshift cluster
3. Load data into Redshift cluster
4. Execute queries against data stored in the cluster

Total Time: 1 hour 10 minute