Jorge Andrés Díaz Naranjo
Big Data

Actividad Athena

1. Visualizar los datos del dataset Amazon Customer Review DataSet (arn:aws:s3:::amazon-reviews-pds) usando AWS Athena

- Crear una base de datos en AWS Athena

```
CREATE DATABASE reviews
```

- Crear un tabla que permita acceder a los datos del bucket que están en formato parquet

```
CREATE EXTERNAL TABLE IF NOT EXISTS reviews.reviews (
  marketplace string,
  customer_id string,
  review_id string,
  product_id string,
  product_parent string,
  product_title string,
  star_rating int,
  helpful_votes int,
  total_votes int,
  vine string,
  verfied_purchase string,
  review_headline string,
  review_body string,
  review_date date,
  year int
) PARTITIONED BY (
  product_category string
)
ROW FORMAT SERDE
'org.apache.hadoop.hive.ql.io.parquet.serde.ParquetHiveSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://amazon-reviews-pds/parquet/'
```

```
TBLPROPERTIES ('has_encrypted_data'='false');

ALTER TABLE reviews ADD PARTITION (product_category='Books')
ALTER TABLE reviews ADD PARTITION (product_category='Baby')
ALTER TABLE reviews ADD PARTITION (product_category='Kitchen')
```

2. **Realizar las siguientes consultas**

// Listar los comentarios de los productos de la categoría Baby

```
SELECT review_body FROM reviews WHERE product_category LIKE
'Baby'
```

// Calificación promedio por categoría para las categorías Baby y Books

```
SELECT avg(star_rating)
FROM reviews
WHERE product_category LIKE 'Baby'
        OR product_category LIKE 'Books'
```

// Libros con mejor calificación (con número de comentarios)

```
SELECT product_title,
         rating_avg,
         reviews_count
FROM
    (SELECT product_title,
         avg(star_rating) AS rating_avg,
         count(review_id) AS reviews_count
    FROM reviews
    WHERE product_category LIKE 'Books'
    GROUP BY  product_title)
WHERE rating_avg =
    (SELECT max(rating_avg)
    FROM
        (SELECT avg(star_rating) AS rating_avg
        FROM reviews
        WHERE product_category LIKE 'Books'
        GROUP BY  product_title))
```

// Productos de cocina cuyo calificación promedio están por debajo del promedio total de productos (kitchen, books & baby)

```
SELECT product_id, product_title, rating_avg
FROM   (
                SELECT   product_id,
                         product_title,
                         avg(star_rating) AS rating_avg
                FROM     reviews
                WHERE product_category LIKE 'Kitchen' GROUP
BY(product_id, product_title)
       )
WHERE  rating_avg <
       (
              SELECT avg(star_rating)
              FROM   reviews
              WHERE  product_category LIKE 'Kitchen'
              OR     product_category LIKE 'Books'
              OR     product_category LIKE 'Baby')
```

3. **Visualizar los datos del dataset open street maps dataset(arn:aws:s3:::osm-pds) usando AWS Athena**

https://aws.amazon.com/blogs/big-data/querying-openstreetmap-with-amazon-athena/

// Número de hospitales en la zona de suba

```
SELECT count(id) from planet
WHERE type = 'node'
  AND tags['amenity'] IN ('hospital')
  AND lon BETWEEN -74.114024 AND -74.075910
  AND lat BETWEEN 4.726262 AND 4.765376
```

// Distancia entre Portal Eldorado y la alcaldía de Bogotá

```
SELECT (T1 * 100) AS km_distance FROM (SELECT ST_DISTANCE(
    (SELECT ST_POINT(lon, lat) FROM planet WHERE type = 'node'
AND tags['name'] LIKE 'Portal Eldorado%'),
    (SELECT ST_POINT(lon, lat) FROM planet WHERE type = 'node'
AND tags['name'] LIKE 'Alcaldía Mayor de Bogotá')
    ) AS T1)
```

// Supermercado con mayor número de tiendas de Usaquén

```
SELECT supermarket_name,
         count
FROM
    (SELECT tags['name'] AS supermarket_name,
count(tags['name']) AS count
    FROM planet
    WHERE type = 'node'
            AND tags['shop'] IN ('supermarket')
            AND lon
        BETWEEN -74.076420
            AND -74.051185
            AND lat
        BETWEEN 4.732693
            AND 4.758284
    GROUP BY  tags['name'])
JOIN
    (SELECT max(count) AS max_count
    FROM
        (SELECT tags['name'], count(tags['name']) AS count
        FROM planet
        WHERE type = 'node'
                AND tags['shop'] IN ('supermarket')
                AND lon
            BETWEEN -74.076420
                AND -74.051185
                AND lat
            BETWEEN 4.732693
                AND 4.758284
        GROUP BY  tags['name']))
        ON count = max_count
```

// Supermercado con mayor número de tiendas en el sector de Kennedy

```
SELECT supermarket_name,
         count
FROM
    (SELECT tags['name'] AS supermarket_name,
count(tags['name']) AS count
    FROM planet
    WHERE type = 'node'
            AND tags['shop'] IN ('supermarket')
            AND lon
        BETWEEN -74.171508
            AND -74.133583
            AND lat
        BETWEEN 4.586473
            AND 4.675474
    GROUP BY  tags['name'])
JOIN
    (SELECT max(count) AS max_count
    FROM
        (SELECT tags['name'], count(tags['name']) AS count
        FROM planet
        WHERE type = 'node'
                AND tags['shop'] IN ('supermarket')
                AND lon
            BETWEEN -74.171508
                AND -74.133583
                AND lat
            BETWEEN 4.586473
                AND 4.675474
        GROUP BY  tags['name']))
        ON count = max_count
```