In Class Activity 8 Adnan Inusah

Activity 0: Open AWS Account

A screenshot of a computer

Description automatically generated

Activity 1: Creating tables

A screenshot of a computer

Description automatically generated

Activity 2: GeoSQL

SELECT ST\_POINT(-93.610065, 41.948637) AS point\_feature

LIMIT 1;

SELECT ST\_DISTANCE(

ST\_POINT(-93.610065, 41.948637),

ST\_POINT(-93.610229, 41.997339)

) AS distance\_btn\_points

LIMIT 1;

WITH points AS (

SELECT

ST\_POINT(-93.610065, 41.948637) AS point\_1,

ST\_POINT(-93.610229, 41.997339) AS point\_2,

ST\_POINT(-93.619738, 41.972017) AS point\_3

)

SELECT ST\_DISTANCE(

ST\_LINESTRING(ARRAY[point\_1, point\_2]),

point\_3)

AS distance\_to\_line

FROM points

LIMIT 1;

WITH points AS (

SELECT

ST\_POINT(-93.610065, 41.948637) AS point\_1,

ST\_POINT(-93.610229, 41.997339) AS point\_2,

ST\_POINT(-93.619738, 41.972017) AS point\_3

)

SELECT ST\_DISTANCE(

ST\_LINESTRING(ARRAY[point\_1, point\_2]),

point\_3) \* 364567

AS distance\_to\_line\_in\_feet

FROM points

LIMIT 1;

WITH buffer\_area AS (

SELECT ST\_BUFFER(ST\_POINT(-94.295980, 41.502954), 0.05) AS buffer\_area

)

SELECT \*,

ST\_POINT(longitude, latitude) AS point\_geometry

FROM adnan\_gps\_data;

WHERE

ST\_CONTAINS(buffer\_area, ST\_POINT(longitude, latitude))

A screenshot of a computer

Description automatically generated

Activity 3: Data conflation through spatial join

WITH line\_segments AS (

SELECT

route\_id,

ST\_LINESTRING(ARRAY[

ST\_POINT(segment\_start\_longitude, segment\_start\_latitude),

ST\_POINT(segment\_end\_longitude, segment\_end\_latitude)

]) AS route\_line

FROM adnan\_lrs\_data

WHERE NOT (segment\_start\_latitude = segment\_end\_latitude AND segment\_start\_longitude = segment\_end\_longitude)

),

point\_to\_line\_distances AS (

SELECT

b.datapointid,

b.journeyid,

b.latitude,

b.longitude,

a.route\_id,

ST\_DISTANCE(ST\_POINT(b.longitude, b.latitude), a.route\_line) AS distance\_to\_route

FROM (

SELECT \*

FROM adnan\_gps\_data

WHERE hour = 1 AND day = 2

) AS b

CROSS JOIN line\_segments AS a

)

SELECT

datapointid,

journeyid,

latitude,

longitude,

route\_id,

distance\_to\_route

FROM (

SELECT

datapointid,

journeyid,

latitude,

longitude,

route\_id,

distance\_to\_route,

ROW\_NUMBER() OVER (PARTITION BY datapointid ORDER BY distance\_to\_route) AS row\_num

FROM point\_to\_line\_distances

) AS ranked\_distances

WHERE row\_num = 1