Gaoyuan Liang
H00353899

1) Highest Flickr photo density [3 marks]

Which cell has the highest Flickr photo density? Give the result as the **centroid point coordinate** in the British National Grid Coordinate system (i.e. the same as the points are stored in). Include the SQL you used to determine the result.

SQL:

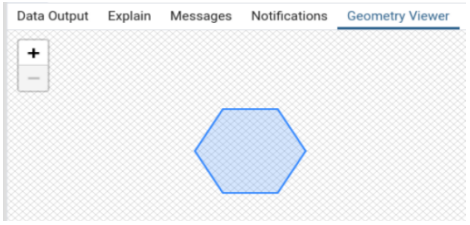
Query Editor | Query History

```
1  CREATE TABLE flickrgrid AS
2  SELECT COUNT(*) as count,b.geom
3  FROM flickr_edin a
4  JOIN grid100m b ON ST_WITHIN(a.geom,b.geom)
5  GROUP BY b.geom;
6
7  select geom, st_astext(geom) as geom_text
8  from flickrgrid
9  order by count desc
10 limit 1;
11
```

Result:

Data Output | Explain | Messages | Notifications | Geometry Viewer

	geom geometry	geom_text text
1	0103000020346C0000010000000700000003...	POLYGON((326257.691453624 672650,326286.55896708...



2) Route lengths [3 marks]

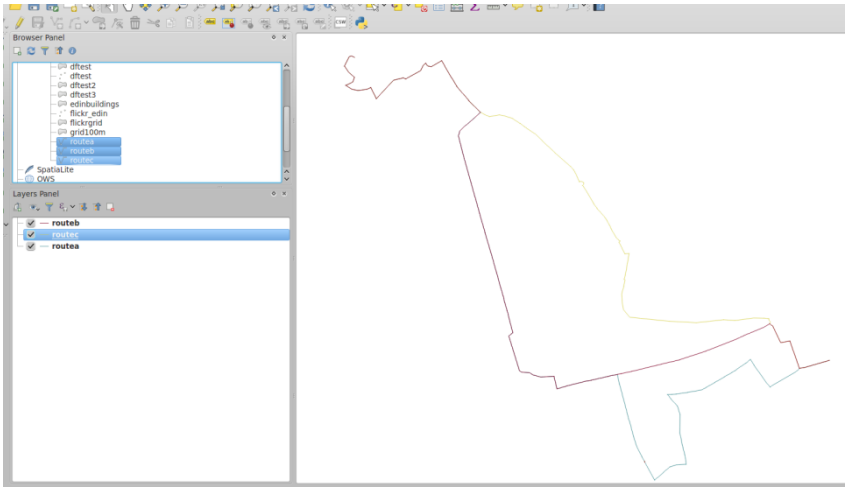
Load the following datasets from www.macs.hw.ac.uk/~pb56/f21df into your PostgreSQL database

- routeb.sql
- routec.sql

(you should have already loaded routea.sql during the lab)
How long is each of the routes - routea, routeb, routec - in metres? (show the SQL you used)

SQL:
Data loading:

```
macslinux:~$ psql -h localhost -U hw -d hw -W -q -f df_routea.sql
Password:
macslinux:~$ psql -h localhost -U hw -d hw -W -q -f df_routeb.sql
Password:
macslinux:~$ psql -h localhost -U hw -d hw -W -q -f df_routec.sql
Password:
macslinux:~$
```



How long is each of the routes:

Query Editor | Query History

```
1  drop table if exists route_line_length;
2
3  create table route_line_length as
4  select geom, st_length(geom) as line_length, 'routea' as route_name  from routea
5  union all
6  select geom, st_length(geom) as line_length, 'routeb' as route_name  from routeb
7  union all
8  select geom, st_length(geom) as line_length, 'routec' as route_name  from routec;
9
10 drop table if exists route_length;
11
12 create table route_length as
13 select route_name, sum(line_length) as route_length
14 from route_line_length
15 group by route_name;
16
17 select * from route_length;
```

Results:

Data Output | Explain | Messages | Notific

	route_name text	route_length double precision
1	routea	5553.43176838058
2	routeb	4687.01037328406
3	routec	4064.18915979505

3) Route popularity [3 marks]

Use the flickr photo data to show the popularity of each route according to how many photos have been taken along the route within a 25m buffer. You should normalise this by route length (i.e. photos per metre).
Give the SQL used and show your results in order from most popular route to least popular route.
(Preferably do this as a single SQL statement.)

SQL:

Query Editor | Query History

```
select route_length.route_name,
route_photo_count.photo_count/route_length.route_length popularity,
route_length.route_length,
route_photo_count.photo_count
from route_length
join (
    select route_name, count(*) as photo_count
    from (
        SELECT flickr_edin.*, route.route_name
        FROM flickr_edin
        JOIN route_line_length as route ON
        ST_INTERSECTS(flickr_edin.geom,ST_BUFFER(route.geom,25))
    ) photo_over_route
    group by route_name
) route_photo_count
on route_length.route_name = route_photo_count.route_name
order by route_photo_count.photo_count/route_length.route_length desc;
```

Results:

Data Output | Explain | Messages | Notifications

	route_name text	popularity double precision	route_length double precision	photo_count bigint
1	routeb	2.33517725123598	4687.01037328406	10945
2	routea	2.01677097461954	5553.43176838058	11200
3	routec	0.535162098633638	4064.18915979505	2175

4) Temporal Patterns [3 marks]

Make a new table and add a point at Nelson Monument on Calton Hill which is British National Grid
coordinates (326253.33,674110.63).

TIPS:

To set the coordinate system (spatial reference ID - SRID) use the EPSG code 27700 for British National Grid.

Check the online help for ST_SETSRID( ) – e.g. http://postgis.org/docs/ST_SetSRID.html

How many photos have been taken within 200m of this location for each day of the week? (Show your SQL for all steps)

SQL:

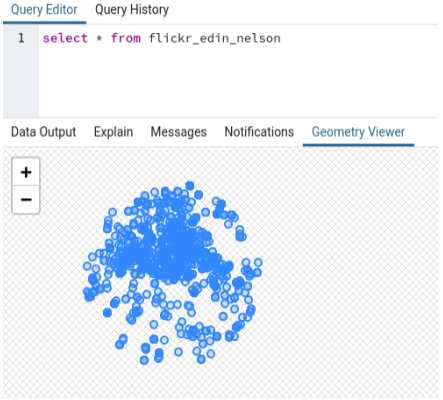
Query Editor | Query History

```sql
drop table if exists Nelson_Monument;
CREATE TABLE Nelson_Monument (id serial,geom geometry);
INSERT INTO Nelson_Monument (geom) values (ST_SetSRID(ST_Point(326253.33,674110.63),27700));

drop table if exists flickr_edin_nelson;
create table flickr_edin_nelson as
select flickr_edin.*
from flickr_edin
join Nelson_Monument
on ST_INTERSECTS(flickr_edin.geom,ST_BUFFER(Nelson_Monument.geom,200));

select dow, count(*) as photo_count
from (
    select flickr_edin_nelson.*,
    EXTRACT('dow' FROM date_taken) as dow
    from flickr_edin_nelson
) as flickr_edin_nelson_dow
group by dow
order by dow;
```

Results:

Data Output | Explain | Messages | Notificatio

	dow double precision	photo_count bigint
1	0	430
2	1	148
3	2	203
4	3	101
5	4	419
6	5	562
7	6	331



5) Data Aggregation [3 marks]

Use the supplied hexagon grid to calculate the most popular locations for photos where the tag includes each of the following terms – giving the centroid of the most popular cell in each case (i.e. 4 queries and outputs).

- Castle
- Calton Hill
- Royal Mile
- Meadows

(For each term show your SQL and the cell centroid with the highest count.)

SQL and results

- Castle

hw/hw@localhost

Query Editor | Query History

```
1  select
2  st_astext(st_centroid(cell_geom)) as cell_centroid,
3  st_astext(cell_geom) as most_popular_cell,
4  count(*) as photo_count
5  from (
6      select flickr_edin.*,
7      grid100m.geom as cell_geom
8      from flickr_edin
9      join grid100m on ST_WITHIN(flickr_edin.geom,grid100m.geom)
10     where flickr_edin.usertags ILIKE '%Castle%'
11 ) as grid100m_term
12 group by cell_geom
13 order by count(*) desc
14 limit 1;
```

Data Output | Explain | Messages | Notifications | Geometry Viewer

	cell_centroid text	most_popular_cell text	photo_count bigint
1	POINT(325189.593455623 673500)	POLYGON((325131.858428704 673...	942

- Calton Hill

Query Editor | Query History

```
1  select
2  st_astext(st_centroid(cell_geom)) as cell_centroid,
3  st_astext(cell_geom) as most_popular_cell,
4  count(*) as photo_count
5  from (
6      select flickr_edin.*,
7      grid100m.geom as cell_geom
8      from flickr_edin
9      join grid100m on ST_WITHIN(flickr_edin.geom,grid100m.geom)
10     where flickr_edin.usertags ILIKE '%Calton Hill%'
11 ) as grid100m_term
12 group by cell_geom
13 order by count(*) desc
14 limit 1;
```

Data Output | Explain | Messages | Notifications | Geometry Viewer

	cell_centroid text	most_popular_cell text	photo_count bigint
1	POINT(326228.823940164 674200)	POLYGON((326171.088...	365

- Royal Mile

Query Editor | Query History

```
select
st_astext(st_centroid(cell_geom)) as cell_centroid,
st_astext(cell_geom) as most_popular_cell,
count(*) as photo_count
from (
    select flickr_edin.*,
    grid100m.geom as cell_geom
    from flickr_edin
    join grid100m on ST_WITHIN(flickr_edin.geom,grid100m.geom)
    where flickr_edin.usertags ILIKE '%Royal Mile%'
) as grid100m_term
group by cell_geom
order by count(*) desc
limit 1;
```

Data Output | Explain | Messages | Notifications | Geometry Viewer

	cell_centroid text	most_popular_cell text	photo_count bigint
1	POINT(325795.811238272 673650)	POLYGON((325738.076...	460

- Meadows

Query Editor | Query History

```
select
st_astext(st_centroid(cell_geom)) as cell_centroid,
st_astext(cell_geom) as most_popular_cell,
count(*) as photo_count
from (
    select flickr_edin.*,
    grid100m.geom as cell_geom
    from flickr_edin
    join grid100m on ST_WITHIN(flickr_edin.geom,grid100m.geom)
    where flickr_edin.usertags ILIKE '%Meadows%'
) as grid100m_term
group by cell_geom
order by count(*) desc
limit 1;
```

Data Output | Explain | Messages | Notifications | Geometry Viewer

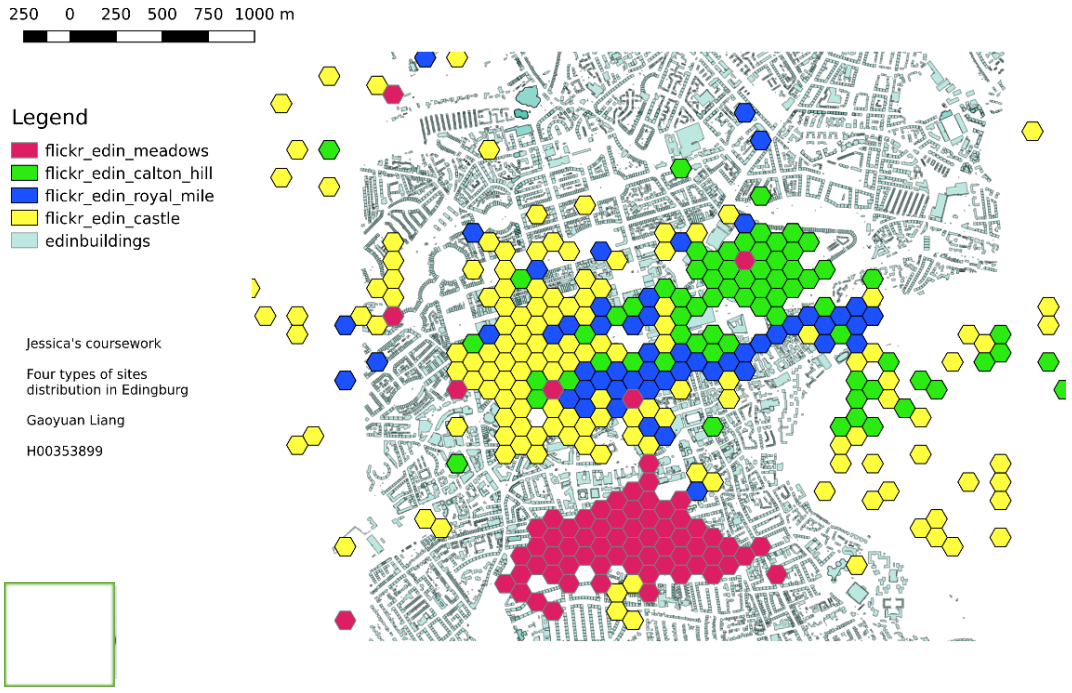
	cell_centroid text	most_popular_cell text	photo_count bigint
1	POINT(325795.811238272 672750)	POLYGON((325738.076...	326

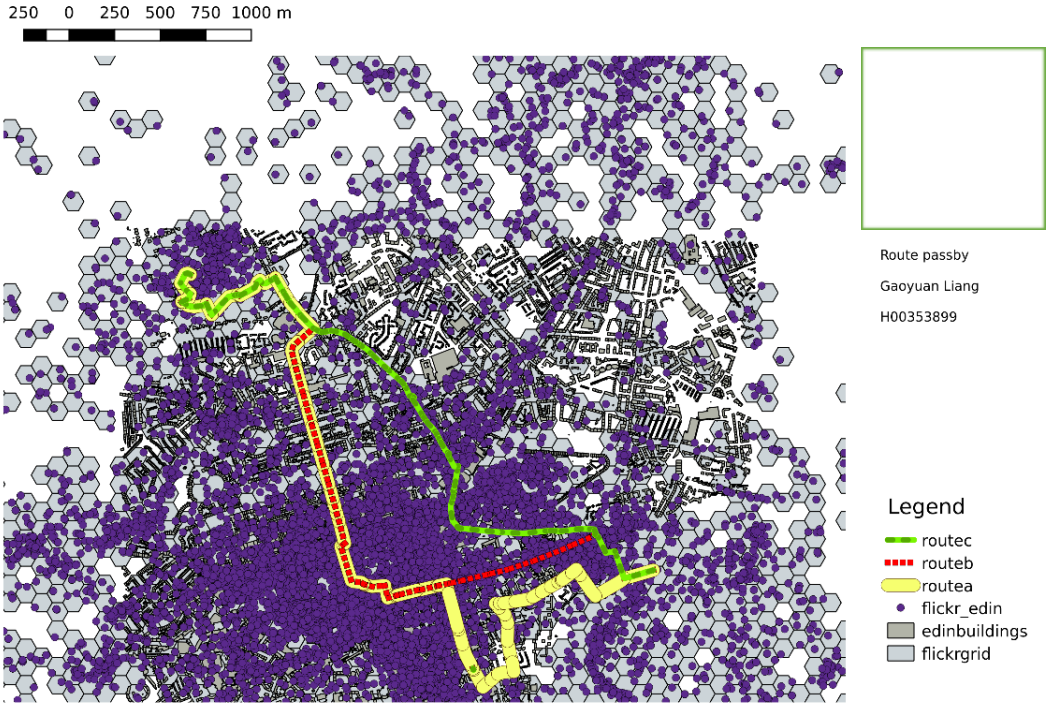
6) Visualization [20 marks]

Make an interesting visualisation of the Flickr data - this should include a map using QGIS to show the spatial pattern of the term across Edinburgh and may include other visualisations or summaries you think are useful.

For example you could focus on 1 of the terms from task 5. You could look to filter the data based on the dates, or userID, and include charts to show temporal patterns.

- Remember to check the lecture slides for tips about map design
- Treat the output as if it is to be displayed stand alone at A4 size (i.e. small poster)





7) Write up [15 marks]

Give an overview of your poster (from task 6) and explain the decisions you made regarding its design, and
what you wanted it to show/why. Highlight any issues and how you might be able to improve your analysis of the Flickr dataset.

My write up:

I made two maps from flikr data. The first map is to show the distribution of the cells of four terms -- Castle, Calton Hill, Royal Mile and Meadows. Firstly I build four tables from the flikr data by mapping the geom of photos to the cells, and then tagging the cells with the terms. Then I load the four tables to the viewer as four layers. Each term has a unique color. The purpose of doing so is to show the location of different terms. Moreover, to locate these terms more accurately, I also load and show the building layer as the back ground, so that the viewer can understand the nearby building.

My second map is a route map. I firstly load the three routes to the viewer and then show them by different shapes and color. The building layer is also given as a background. Meanwhile, the photo sites are also given as background to show the popularity of the routes.