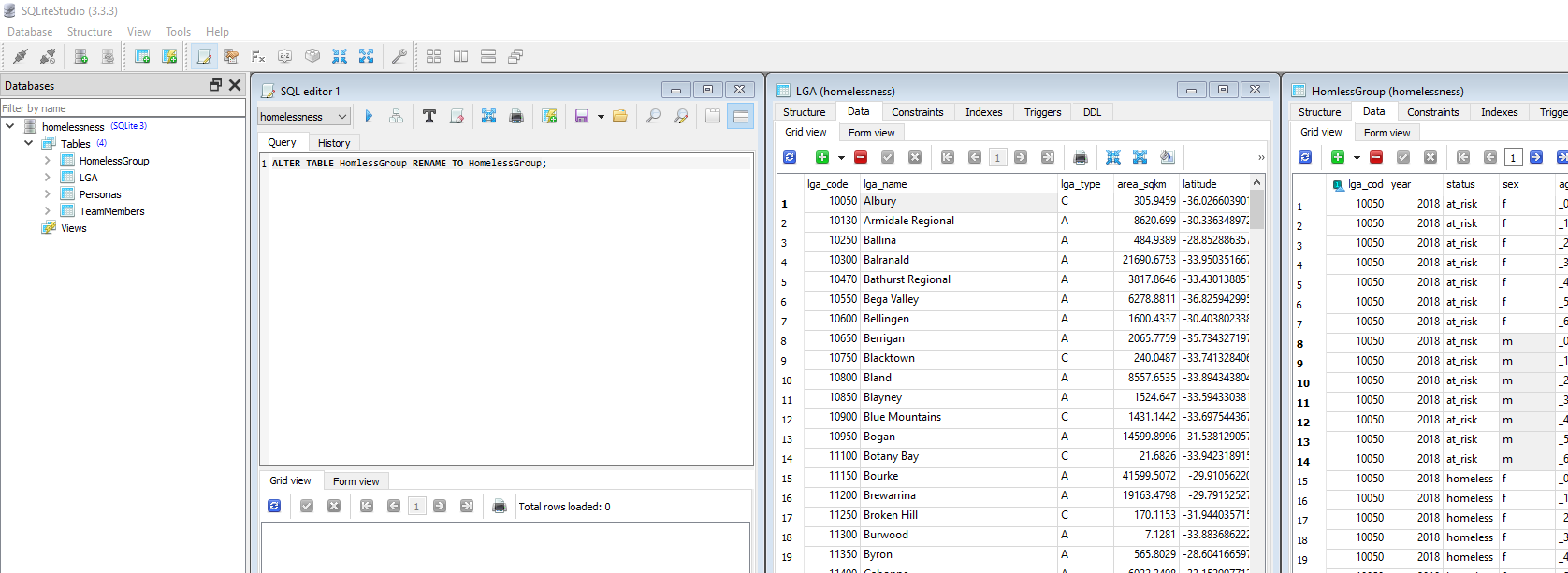
**SQL Queries Mastersheet: Project Group 69**

*Nathan Nguyen s3948938*



Renamed table to correct spelling

**Can add these definitions to the homepage as drop down lists.**

A person is considered to be homeless if they are living in any of the following circumstances:

- No shelter or improvised dwelling

- Short-term temporary accommodation

- House, Townhouse or Flat (couch surfing or with no tenure)

A person is considered to be at risk of homelessness if they are living in any of the following circumstances:

- Public or community housing (renter or rent free)

- Private or other housing (renter, rent-free or owner)

- Institutional settings

**Level 1: Green Tasks**

**Sub Task 1: Landing Page**

**3 Big Facts**

**Fact 1:** [\_\_\_\_\_\_ Australians are homeless]

SELECT SUM(count) AS TotalHomeless

FROM HomlessGroup

WHERE status = 'homeless';

**Fact 2:** [\_\_\_\_\_ Australians are at risk of being homeless]

SELECT SUM(count) AS TotalAtRisk

FROM HomelessGroup

WHERE status = 'at\_risk';

**Fact 3:** [Of those homeless, \_\_\_\_\_\_ are aged 60+]

SELECT SUM(count) AS TotalAtRisk

FROM HomelessGroup

WHERE age\_group = '\_60\_plus' AND status = 'homeless';

**Total Number of LGA’s:** [There are \_\_\_\_\_ LGAs in total]

SELECT COUNT(lga\_name) AS TotalLGAs

FROM LGA;

**Total Population:**

SELECT SUM(year18) AS TotalPopulation2018

FROM Population;

SELECT SUM(year16) AS TotalPopulation2016

FROM Population;

**Sub Task 2: Personas & Team Names**

1. **Personas: SQL outputs single rows (make into objects)**

**Mike Hawk: 3x attributes**

SELECT name, description, goals

FROM Personas

LIMIT 1;

**Phil McAvity: 3x attributes**

SELECT name, description, goals

FROM Personas

WHERE name = 'Phil McAvity';

1. **Team Names: Same thing**

**Nathan Nguyen: 2x attributes**

SELECT name, id, description

FROM TeamMembers

WHERE name = 'Nathan Nguyen';

**Michael Chatham: 2x attributes**

SELECT name, id, description

FROM TeamMembers

WHERE name = 'Michael Chatham;

**Level 2: Orange Tasks**

**Sub Task 1: Age & School Completion Ranking by LGA**

1. **Raw Total Values:** (choose any, change values based on options below)

SELECT SUM(count) AS RawTotal

FROM HomelessGroup

WHERE sex = '\_\_\_\_\_' AND status = '\_\_\_\_\_' AND age\_group = ‘\_\_\_\_\_’;

2x sex variables (m/f) \* 2x status variables (homeless/at\_risk) \* 7x age variables (\_0\_9/\_10\_19/\_20\_29/\_30\_39\_/\_40\_49/\_50\_59/\_60\_plus) = 28x different sql queries

1. **Proportional Values compared to the Total Population (as a percentage, CHANGE HIGHLIGHTED SECTION TO DESIRED FILTERS)**

SELECT Proportion AS TotalProportion

FROM (

SELECT \*,

ROUND( (count / year18) \* 10000, 2) AS Proportion

FROM (

SELECT Population.year18,

Population.lga\_name,

HomelessGroup.status,

HomelessGroup.sex,

HomelessGroup.age\_group,

HomelessGroup.count

FROM Population

INNER JOIN

HomelessGroup ON Population.lga\_code = HomelessGroup.lga\_code

)

)

WHERE lga\_name = 'Albury' AND

status = 'homeless' AND

sex = 'f' AND

age\_group = '\_0\_9';

1. **Ranking of LGA’s (by age or sex) from best to worse**

* **OUTPUTS TABLE RANKING ALL LGA’s BY NO. OF HOMELESS/ATRISK PEOPLE**

SELECT \*,

ROW\_NUMBER() OVER (ORDER BY count DESC) AS Ranking

FROM (

SELECT SUM(count) AS count,

lga\_name

FROM (

SELECT \*

FROM (

SELECT Population.lga\_name,

HomelessGroup.status,

HomelessGroup.sex,

HomelessGroup.age\_group,

HomelessGroup.count

FROM Population

INNER JOIN

HomelessGroup ON Population.lga\_code = HomelessGroup.lga\_code

)

WHERE sex = 'f' AND

status = 'homeless'

)

GROUP BY lga\_name

ORDER BY SUM(count) DESC

);

**GIGA-CHAD QUERY (ALL THREE QUERIES COMBINED)**

**TABLE SHOWING LGA RANKINGS BY PROPORTION (CHANGE GREEN IF YOU WANT TO RANK BY COUNT AND/OR ASCENDING, CHANGE YELLOW FOR USER VALUES)**

SELECT ranking,

lga\_name AS LGA,

Proportion,

count

FROM (

SELECT \*,

ROW\_NUMBER() OVER (ORDER BY Proportion DESC) AS ranking

FROM (

SELECT \*,

ROUND( (count / year18) \* 100, 3) AS Proportion

FROM (

SELECT Population.year18,

Population.lga\_name,

HomelessGroup.status,

HomelessGroup.sex,

HomelessGroup.age\_group,

HomelessGroup.count

FROM Population

INNER JOIN

HomelessGroup ON Population.lga\_code = HomelessGroup.lga\_code

)

WHERE status = 'homeless' AND

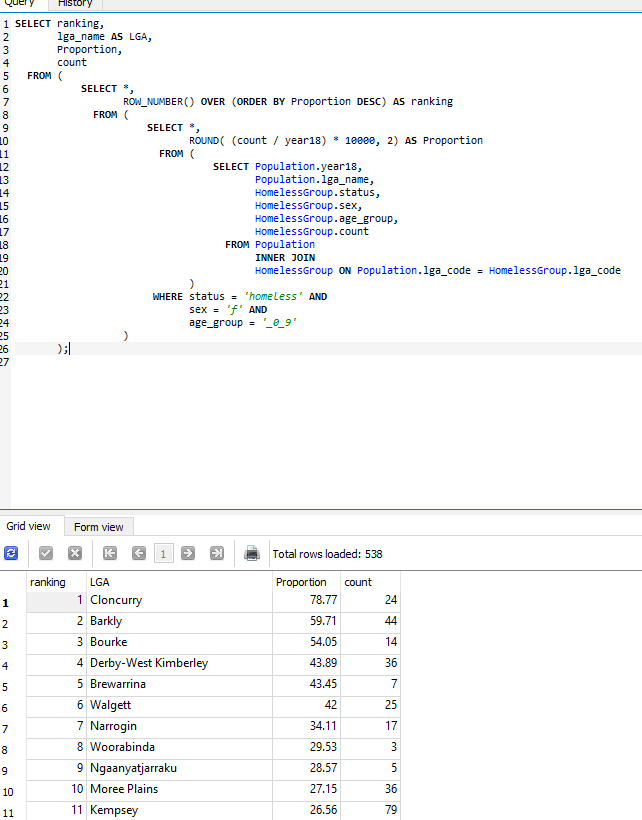
age\_group = '\_0\_9' AND

sex = 'f'

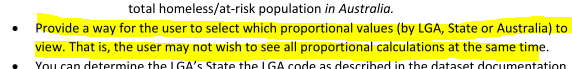
)

) GROUP BY LGA ORDER by ranking LIMIT 200

;



**Sub Task 2: Focused view of socioeconomic outcomes by lga**



***Change the green highlight to any of these three values to view proportional values accordingly: lgaProp, stateProp, ausProp***

SELECT lga\_name AS NAME,

state AS STATE,

lga\_type AS TYPE,

ROUND(area\_sqkm, 2) AS TOTALSQKM,

lga\_population AS TOTALPOP,

count AS TOTALNUM,

LgaProp

FROM (

SELECT \*,

Round( (count / lga\_population) \* 10000, 2) AS lgaProp,

Round( (count / state\_population) \* 10000, 2) AS stateProp,

Round( (count / 24980000.0) \* 10000, 2) AS ausProp

FROM (

SELECT \*

FROM (

SELECT LGA.lga\_name,

LGA.lga\_type,

LGA.area\_sqkm,

LGA.state,

LGA.lga\_population,

HomelessGroup.count,

HomelessGroup.Sex,

HomelessGroup.status,

HomelessGroup.age\_group,

StateTotalPop.state\_population

FROM LGA

INNER JOIN

HomelessGroup ON LGA.lga\_code = HomelessGroup.lga\_code

INNER JOIN

StateTotalPop ON LGA.state = StateTotalPop.state

)

WHERE lga\_name = 'Brisbane' AND

status = 'homeless' AND

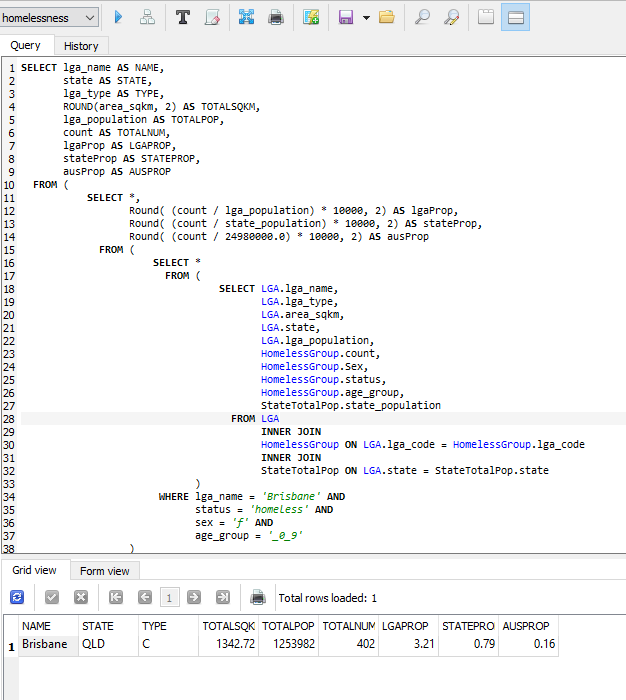
sex = 'f' AND

age\_group = '\_0\_9'

)

) LIMIT 200

;



**Level 3: Red Tasks**

**Sub Task 1: Comparing Homelessness to Other Factors**

Change yellow highlight to show what you want (m/f or SA/VIC/NT/etc or \_0\_9/etcetc << add + inbetween to add more age ranges)

(income filters kinda self explanatory just chuck your range and put BETWEEN between your min max numbers)

Change to which column you want to sort by

Code below assumes that user has inputted checked ‘all’ for everything

SELECT \*

FROM (

SELECT \*

FROM (

SELECT LGA,

state AS Region,

RatioTotal,

RatioGender,

RatioAge,

Income.median\_age AS MedAge,

Income.median\_mortgage\_repay\_monthly AS MedMortgage,

Income.median\_rent\_weekly AS MedRent,

Income.median\_household\_weekly\_income AS MedIncome

FROM (

SELECT lga\_name AS LGA,

homCount AS RatioTotal,

(m + f) AS RatioGender,

(\_0\_9 + \_10\_19 + \_20\_29 + \_30\_39 + \_40\_49 + \_50\_59 + \_60\_plus) AS RatioAge,

state

FROM (

SELECT \*,

ROUND( (total / year18) \* 100, 2) AS homCount,

ROUND( (f\_ / year18) \* 100, 2) AS f,

ROUND( (m\_ / year18) \* 100, 2) AS m,

ROUND( (\_0\_9\_ / year18) \* 100, 2) AS \_0\_9,

ROUND( (\_10\_19\_ / year18) \* 100, 2) AS \_10\_19,

ROUND( (\_20\_29\_ / year18) \* 100, 2) AS \_20\_29,

ROUND( (\_30\_39\_ / year18) \* 100, 2) AS \_30\_39,

ROUND( (\_40\_49\_ / year18) \* 100, 2) AS \_40\_49,

ROUND( (\_50\_59\_ / year18) \* 100, 2) AS \_50\_59,

ROUND( (\_60\_plus\_ / year18), 2) AS \_60\_plus,

state

FROM (

SELECT homelessCount.\*,

Population.year18,

Population.lga\_name,

Population.state

FROM homelessCount

INNER JOIN

Population ON homelessCount.lga\_code = Population.lga\_code

)

)

)

INNER JOIN

Income ON LGA = Income.lga\_name

ORDER BY LGA

)

WHERE region IN ('VIC', 'NSW') AND

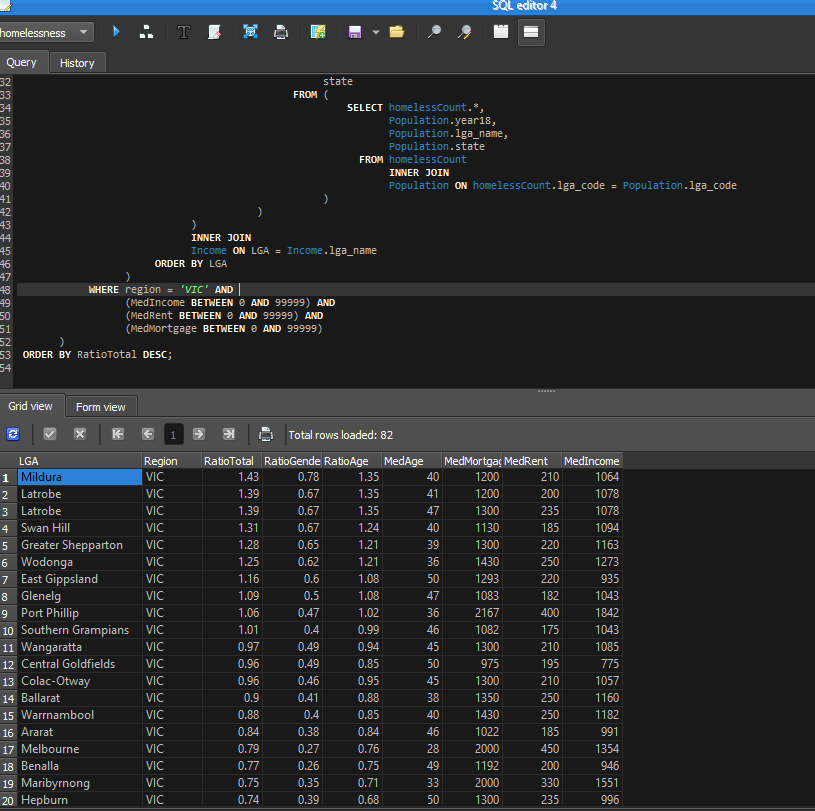
(MedIncome BETWEEN 0 AND 99999) AND

(MedRent BETWEEN 0 AND 99999) AND

(MedMortgage BETWEEN 0 AND 99999)

)

ORDER BY RatioTotal DESC LIMIT 200;



**Sub Task 2: Change in Homelessness over time**

TEMPORARY BETA QUERY – ONLY FILTERS BY STATE NOT BY AGE OR SEX

SELECT \*,

ROUND( (totalHom / totalRisk), 2) AS ratioHomRisk

FROM (

SELECT lga\_code AS code,

lga\_name AS LGA,

state,

(hom18 - hom16) AS totalHom,

ROUND( (hom16 / hom18), 2) AS ratioHom,

(risk18 - risk16) AS totalRisk,

ROUND( (risk16 / risk18), 2) AS ratioRisk,

(year18 - year16) AS totalYear,

ROUND( (year16 / year18), 2) AS ratioYear

FROM (

SELECT \*

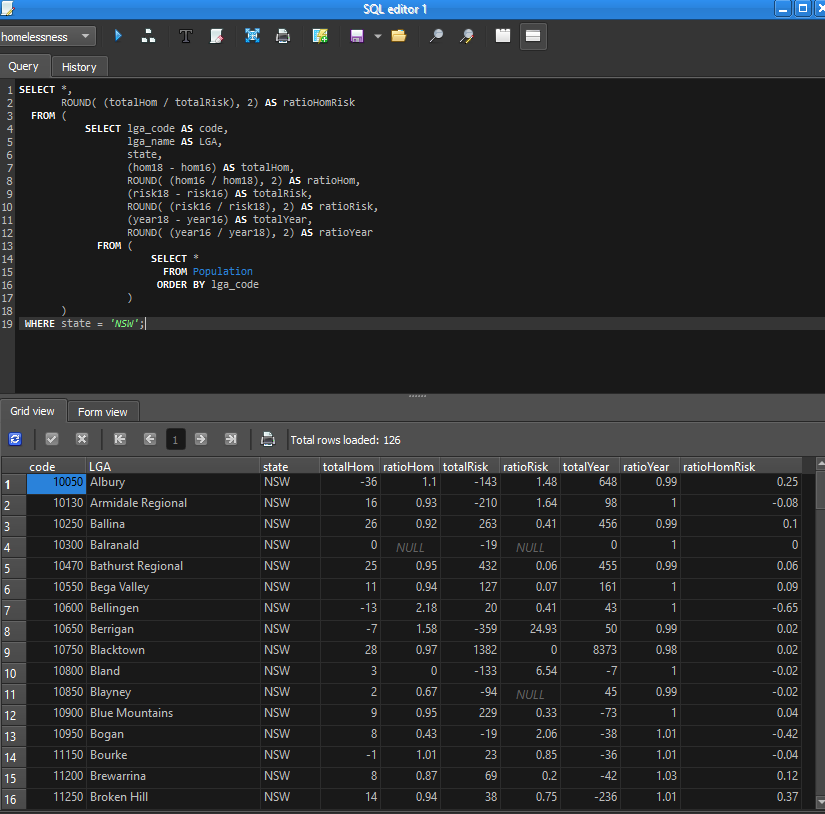
FROM Population

ORDER BY lga\_code

)

)

WHERE state = 'NSW';



**HUMUNGANOID GIGA SUPLEX CHAD FINAL FORM QUERY!!!**

**So this ones kinda whack. The filters available are: sex, age and region. Homeless and At-Risk are always on the table.**

**For sex and age, you have to change them once for each of the four columns. IE make them the same for each of the four WHERE columns.**

**For region, you can change that at the bottom in the WHERE clause. Best of luck.**

SELECT \*,

ROUND( (homIncrease / riskIncrease), 4) AS ratioHomtoRisk

FROM (

SELECT code,

LGA,

state,

(homCount18 - homCount16) AS homIncrease,

ROUND( (homCount16 / homCount18), 2) AS homRatio,

(riskCount18 - riskCount16) AS riskIncrease,

ROUND( (riskCount16 / riskCount18), 2) AS riskRatio,

(year18 - year16) AS yearIncrease,

ROUND( (year16 / year18), 2) AS yearRatio

FROM (

SELECT lga\_code AS code,

lga\_name AS LGA,

state,

year16,

year18,

count16 AS homCount16,

count18 AS homCount18,

riskCount16,

riskCount18

FROM (

SELECT \*,

Population.lga\_name,

Population.state,

Population.year16,

Population.year18

FROM (

SELECT \*

FROM (

SELECT \*

FROM (

SELECT \*

FROM (

SELECT \*,

lga\_code AS code18,

sum(count) AS count18

FROM HomelessGroup

WHERE status = 'homeless' AND

sex = 'f' AND

age\_group = '\_0\_9'

GROUP BY lga\_code

)

INNER JOIN

(

SELECT \*,

lga\_code AS code16,

sum(count) AS count16

FROM HomelessGroup2

WHERE status = 'homeless' AND

sex = 'f' AND

age\_group = '\_0\_9'

GROUP BY lga\_code

)

ON code18 = code16

)

INNER JOIN

(

SELECT \*,

lga\_code AS riskCode18,

sum(count) AS riskCount18

FROM HomelessGroup

WHERE status = 'at\_risk' AND

sex = 'f' AND

age\_group = '\_0\_9'

GROUP BY riskCode18

)

ON riskCode18 = code16

)

INNER JOIN

(

SELECT \*,

lga\_code AS riskCode16,

sum(count) AS riskCount16

FROM HomelessGroup2

WHERE status = 'at\_risk' AND

sex = 'f' AND

age\_group = '\_0\_9'

GROUP BY riskCode16

)

ON riskCode16 = code16

)

INNER JOIN

Population ON code18 = Population.lga\_code

)

)

)

WHERE state IN ('VIC')

LIMIT 200;

