Milestone 3
Team: HubLyfe

Questions with select statements:

#1. When a user moves to a city they may want to know where most renters live, how many people live in that neighborhood, and what the average rent is of that neighborhood.

```
SELECT Demographic.NeighborhoodName, Demographic.Population,
       (Demographic.RenterOccupiedUnits/Demographic.OccupiedHousingUnits)*100  AS '%
      Renter occupied',  Rent.Price
FROM Demographic
INNER JOIN Rent
ON Rent.NeighborhoodName = Demographic.NeighborhoodName
WHERE Rent.OccupancyType = 'All rentals'
ORDER BY((Demographic.RenterOccupiedUnits/Demographic.OwnerOccupiedUnits)*100) DESC
LIMIT 1;
```

#2. If a family is moving to a new area and they have young kids, they would want to move to a neighborhood with the most recently built elementary school.

```
SELECT PublicSchool.NeighborhoodName, PublicSchool.SchoolName,SchoolType.SchoolTypology
FROM PublicSchool
LEFT OUTER JOIN SchoolType
ON PublicSchool.SchoolId = SchoolType.SchoolId
HAVING SchoolType.SchoolTypology = 'Elementary School'
ORDER BY (PublicSchool.YearBuilt) DESC LIMIT 1;
```

#3.If a user values how much they pay in rent based on how many restaurants are nearby, they can use the ratio of average rent to number of neighborhoods as a good metric. (i.e. A ratio closer to 1 (or below 1) means there are as many or more restaurants in the area in relation to rent paid)

```
SELECT RESTANDRENT.Neighborhood,RESTANDRENT.AverageRent/RESTANDRENT.NumberOfRestaurants
       AS AverageRentPerRestaurant
FROM (
     SELECT RESTAURANTS.NeighborhoodName AS Neighborhood,
           RESTAURANTS.NumberOfRestaurants AS NumberOfRestaurants , AVG(Rent.Price) AS
           AverageRent
     FROM (
          SELECT ZipCode.NeighborhoodName AS NeighborhoodName, COUNT(*) AS
                 NumberOfRestaurants
          FROM Restaurant
          LEFT OUTER JOIN ZipCode
          ON Restaurant.Zip = ZipCode.Zip
          GROUP BY ZipCode.NeighborhoodName) AS RESTAURANTS
     INNER JOIN Rent
     ON RESTAURANTS.NeighborhoodName = Rent.NeighborhoodName
     GROUP BY RESTAURANTS.NeighborhoodName) AS RESTANDRENT
GROUP BY RESTANDRENT.Neighborhood
ORDER BY AverageRentPerRestaurant;
```

#4. A young professional would want to know which neighborhood has the most people their age and are also college educated so that they are like minded.

```
SELECT Demographic.NeighborhoodName, Demographic.Population AS 'Total Population',
        AgeData.AgePercentage AS 'Population of User''s Age',
        EducationalAttainment.EdPopulation AS'Population of degree holders'
FROM Demographic
LEFT OUTER JOIN AgeData
ON Demographic.DemographicId = AgeData.DemographicId
LEFT OUTER JOIN EducationalAttainment
ON Demographic.DemographicId = EducationalAttainment.DemographicId
WHERE EdType = 'some college or Associate''s Degree' AND
        AgeData.AgeRange = (
           SELECT
                 CASE
                 WHEN (User.Age REGEXP '^([0-9])$') THEN '0-9 years'
                 WHEN (User.Age REGEXP '^(1[0-9])$') THEN '10-19 years'
                 WHEN (User.Age REGEXP '^(2[0-9]|3[0-4])$') THEN '20-34 years'
                 WHEN (User.Age REGEXP '^(3[5-9]|4[0-9]|5[0-4])$') THEN '35-54 years'
                 WHEN (User.Age REGEXP '^(5[5-9]|6[0-4])$') THEN '55-64 years'
                 ELSE '65 years and over'
                 END AS AgeRange
           FROM User
           WHERE User.UserName = 'sheela27')
ORDER BY AgeData.AgePercentage DESC, EdPopulation DESC LIMIT 1;
```

#5. A teacher is moving to boston and wants to know the salary, and neighborhoods that this job is offered.

```
SELECT Salary.NeighborhoodName AS 'Neighborhood with  Average Teacher Salary > Total
        Average Teacher Salary' , AvgTeacherSalaryPerNeighborhood
FROM (
      SELECT DISTINCT ZipCode.NeighborhoodName, AVG(Salary ) AS
             AvgTeacherSalaryPerNeighborhood,  (SELECT AVG(Salary) AS 'Average Teacher
             Salary'FROM JobDetail WHERE JobTitle like 'Teacher%') AS AvgTeacherSalary
      FROM JobDetail
LEFT OUTER JOIN  ZipCode
ON JobDetail.Zip = ZipCode.Zip
WHERE JobTitle like 'Teacher%'
GROUP BY ZipCode.NeighborhoodName) AS Salary
WHERE AvgTeacherSalaryPerNeighborhood > AvgTeacherSalary
ORDER BY AvgTeacherSalaryPerNeighborhood DESC;
```

#6. Based on average age percentage in a neighborhood of interest, the user can find how many schools are available for that age range.

```
SELECT temp1.NeighborhoodName, temp1.AgeRange, temp1.Counts AS 'Number of Schools for
Age Range', temp2.AgePercentage AS 'Age Population'
FROM (
     SELECT NeighborhoodName, '0-9 years' AS AgeRange, Case1 AS 'Counts'
     FROM (
           SELECT NeighborhoodName,
           COUNT(CASE WHEN SchoolTypology IN ('Elementary School','Early
                Learning','k-8') THEN 1                     END) AS 'Case1',
           COUNT(CASE WHEN SchoolTypology IN ('Middle School', 'High School', 'k-8')
                THEN 1 END) AS 'Case2'
           FROM PublicSchool
           INNER JOIN SchoolType ON SchoolType.SchoolId = PublicSchool.SchoolId
           GROUP BY  NeighborhoodName) AS table1

    UNION ALL

    SELECT NeighborhoodName, '10-19 years' AS AgeRange, Case2 AS 'Number of Schools'
    FROM(
           SELECT NeighborhoodName,
           COUNT(CASE WHEN SchoolTypology IN ('Elementary School', 'Early Learning',
                'k-8') THEN 1 END) AS 'Case1',
           COUNT(CASE WHEN SchoolTypology IN ('Middle School', 'High School', 'k-8')
                THEN 1 END) AS 'Case2'
           FROM PublicSchool
           INNER JOIN SchoolType ON SchoolType.SchoolId = PublicSchool.SchoolId
           GROUP BY  NeighborhoodName) AS table2
           ) AS temp1
    INNER JOIN

    (SELECT NeighborhoodName, AgeRange, AgePercentage
     FROM Demographic
     LEFT JOIN AgeData ON AgeData.DemographicId = Demographic.DemographicId
    ) AS temp2

ON temp1.NeighborhoodName = temp2.NeighborhoodName AND temp1.AgeRange = temp2.AgeRange
ORDER BY temp1.NeighborhoodName;
```

#7. Busy users who depend on take out food for most of their meals would like to know top 5 neighborhoods have the most take out restaurants.

```
SELECT ZipCode.NeighborhoodName,Restaurant.RestaurantType, Restaurant.LicenseStatus,
       Count(*) AS 'Total Take Out Restaurants'
FROM Restaurant
LEFT OUTER JOIN ZipCode
ON Restaurant.Zip = ZipCode.Zip
GROUP BY ZipCode.NeighborhoodName, Restaurant.RestaurantType, Restaurant.LicenseStatus
HAVING Restaurant.RestaurantType = 'Eating & Drinking w/ Take Out' AND
Restaurant.LicenseStatus = 'Active'
ORDER BY  Count(*) DESC, ZipCode.NeighborhoodName
LIMIT 5;
```

#8. A rich single person wants to move to the most expensive neighborhood and live in a studio or 1 bedroom with the highest rent.

```
SELECT * FROM (
     (SELECT NeighborhoodName, OccupancyType, Price
      FROM Rent
      WHERE OccupancyType in ('1 Bed')
      ORDER BY Price DESC LIMIT 1)
     UNION
     (SELECT NeighborhoodName, OccupancyType, Price
      FROM Rent
     WHERE OccupancyType in ('Studio')
     ORDER BY Price DESC LIMIT 1)
)AS ExpensiveAccomodation;
```

#9. A user trying to switch careers to something more popular may want to know the average salary of every department in the different neighborhoods in Boston so he can choose where to move to.

```
SELECT ZipCode.NeighborhoodName AS Neighborhood, SALARY.DepartmentName AS Department,
       SALARY.MONEY
FROM ZipCode
LEFT OUTER JOIN
(SELECT DepartmentName, Zip, AVG(Salary) AS MONEY
 FROM JobDetail
 GROUP BY DepartmentName, Salary, Zip) AS SALARY
ON ZipCode.Zip = SALARY.Zip
GROUP BY ZipCode.NeighborhoodName, SALARY.MONEY, SALARY.DepartmentName
ORDER BY ZipCode.NeighborhoodName, SALARY.MONEY DESC;
```

#10. A foreign born retired (65+) hispanic couple wants to move to a neighborhood with similar people to be able to make more friends. They want to know Age percentage, foreign born percentage (foreign born/pop), for every neighborhood containing hispanic type.

```
SELECT NeighborhoodName, Population, ROUND( (ForiegnBorn/Population)*100) AS '% Foriegn
Born',ROUND((EthnicityPopulation/Population)*100) AS '% Hispanic',
ROUND((AgePercentage/Population)*100) AS '% 65+'
FROM Demographic
LEFT OUTER JOIN Ethnicity ON Demographic.DemographicId = Ethnicity.DemographicId
LEFT OUTER JOIN AgeData ON Demographic.DemographicId = AgeData.DemographicId
WHERE EthnicityType = 'Hispanic' AND AgeData.AgeRange = '65 years and over'
ORDER BY (AgePercentage/Population)*100 DESC,(EthnicityPopulation/Population)*100 DESC,
(ForiegnBorn/Population)*100 DESC;
```

Changes to relational model:
1. The only change was adding a column ‘Age’ to User table to answer question 4