Homework 2: SQL

Q1 [5 pt] write a SQL query to find name of all the business in the 'fast food' category

```sql
SELECT Name
FROM
  Business
WHERE
  Category = 'fast food';
```

Q2 [10 pt] write a SQL query to find the top 3 users based on total number of fans

```sql
SELECT
  ID,
  NAME,
  tot_cred
FROM
  student
ORDER BY
  tot_cred DESC
  LIMIT 3;
```

Q3 [10 pt] write a SQL query to find the number of users registered after 2020

```
SELECT
  COUNT(*)
FROM
  User
WHERE
  YEAR ( Yelping_since ) > 2020;
```

Q4 [10 pt] write a SQL query to list the cities with the most reviews in descending order

```
SELECT
  City
FROM
  Business
ORDER BY
  Review_count DESC
```

Q5 [10 pt] write SQL to update stars and review_counts for business named " yami sushi"

```
UPDATE Business
SET Stars = 4.5, Review_count = 3000
WHERE
  NAME = 'yami sushi';
```

Q6 [10 pt] write a SQL query to find the distribution of star ratings to the business in the Pittsburgh (return stars and corresponding counts)

```
SELECT
  stars,
  count(*)
FROM
  Business
GROUP BY
  stars;
```

Q7 [10 pt] for the restaurant named “yami sushi” write a SQL query to find if there are more reviews with the word ‘love’ or with the word ‘hate’

```
SELECT love_review, hate_review,
CASE
  WHEN love_review > hate_review THEN 'More love review'
  WHEN hate_review > love_review THEN 'More hate review'
  ELSE 'Equal'
END AS Result
FROM (SELECT count(*) as love_review FROM Business, Review WHERE
Review.Business_id =       Business.ID AND Business.Name = 'yami
sushi' AND Review.Text LIKE '%love%') as t1,
    (SELECT count(*) as hate_review FROM Business, Review WHERE
Review.Business_id = Business.ID AND Business.Name = 'yami sushi' AND
Review.Text LIKE '%hate%') as t2;
```

Q8 [10 pt] for each user, calculate average star, average number of “useful”, “funny” and “cool” for all the reviews this user wrote

```
SELECT User.ID, User.Name, AVG(Stars), AVG(Useful), AVG(Funny),
AVG(Cool)
FROM User, Review
WHERE User.ID = Review.User_id
GROUP BY User.ID, User.Name;
```

Q9 [10 pt] Find the user who has the largest number of fans, return the name and amount for the top 5 businesses he tipped

```
SELECT User.Name, Tip.amount
FROM User, Tip, (SELECT MAX(Fans) as max_fans From User) as t1
WHERE User.ID = Tip.User_id AND User.Fans = t1.max_fans
ORDER BY Tip.amount DESC
LIMIT 5;
```

Q10 [15 pt] Group business based on the one that are open and the ones that are closed, what differences you can find between those two groups (you can use different SQL to answer different sub questions)

1. ## Are there more reviews written for the business that are still open

```
SELECT open_review, close_review,
CASE
  WHEN open_review > close_review THEN 'More reviews with open
business'
  WHEN close_review > open_review THEN 'More reviews with
closed business'
  ELSE 'Equal'
END AS Result
FROM (SELECT count(*) as open_review FROM Business WHERE
Is_open = 1) as t1,
    (SELECT count(*) as close_review FROM Business WHERE
Is_open = 0) as t2;
```

2. ## Are the average star rating higher for business that are open than business that are closed

```
SELECT open_stars, close_stars,
CASE
  WHEN open_stars > close_stars THEN 'open business average
stars higher'
  WHEN close_stars > open_stars THEN 'closed business average
stars higher'
  ELSE 'Equal'
END AS Result
FROM (SELECT AVG(Stars) as open_stars FROM Business WHERE
Is_open = 1) as t1,
    (SELECT AVG(Stars) as close_stars FROM Business WHERE
Is_open = 0) as t2;
```

3. ## Are there more tips for business that are open than business that are closed

```
SELECT open_tips, close_tips,
CASE
  WHEN open_tips > close_tips THEN 'More tips with open
business'
  WHEN close_tips > open_tips THEN 'More tips with closed
business'
  ELSE 'Equal'
END AS Result
FROM (SELECT SUM(Tip.amount) as open_tips FROM Tip
      WHERE Tip.Business_id IN (SELECT ID as open_id FROM
Business WHERE Is_open = 1)) as t1,
     (SELECT SUM(Tip.amount) as close_tips FROM Tip
      WHERE Tip.Business_id IN (SELECT ID as open_id FROM
Business WHERE Is_open = 0)) as t2;
```