

Assignment 17. Tatyana Li

```
postgres=# SELECT * FROM artist;
 artid |
-----+-----
      1 | Muse
      2 | Mr. Scruff
      3 | DeadMau5
      4 | Mark Ronson
      5 | Mark Ronson & The Business Intl
      6 | Animal Collective
      7 | Kings of Leon
      8 | Maroon 5
(8 rows)
```

```
postgres=# SELECT * FROM cd;
 cdid | artid | cdtitle | cdprice | cdgenre | cdnumtracks |
-----+-----+-----+-----+-----+-----+
      1 |      1 | Black Holes and Revelations | 9.99 | Rock |
      2 |      1 | The Resistance | 11.99 | Rock |
      3 |      2 | Ninja Tuna | 9.99 | Electronica |
      4 |      3 | For Lack of a Better Name | 9.99 | Electro House |
      5 |      4 | Version | 12.99 | Pop |
      6 |      5 | Record Collection | 11.99 | Alternative Rock |
      7 |      6 | Merriweather Post Pavilion | 12.99 | Electronica |
      8 |      7 | Only By The Night | 9.99 | Rock |
      9 |      7 | Come Around Sundown | 12.99 | Rock |
     10 |      8 | Hands All Over | 11.99 | Pop |
(10 rows)
```

```
postgres=# SELECT cdtitle, cdprice
```

1. Find a list of CD titles and prices where the title contains exactly one word

```
postgres=# SELECT cdtitle, cdprice
postgres=# FROM cd
postgres=# WHERE LENGTH(cdtitle)-LENGTH(REPLACE(cdtitle, ' ', '')) = 0;
 cdtitle | cdprice |
-----+-----+
  Version |    12.99 |
(1 row)

postgres=#
```

2. Find all CD titles and the corresponding artist names where title contains the string "ro"

```
postgres=# SELECT cdtitle, artname
postgres=# FROM cd
postgres=# JOIN artist ON cd.artid = artist.artid
postgres=# WHERE cdtitle LIKE '%ro%';
 cdtitle | artname |
-----+-----+
Come Around Sundown | Kings of Leon |
(1 row)
```

3. Find all the information from CD where title contains the letters “he” containing the single character at the beginning

```
postgres=# SELECT * FROM cd
postgres=# WHERE cdtitle LIKE '_he%';
 cdid | artid |  cdtitle  | cdprice | cdgenre | cdnumtracks
-----+-----+-----+-----+-----+-----
    2 |     1 | The Resistance |    11.99 | Rock    |
(1 row)
```

4. Use a subquery to find a list of CDs that have the same genre as ‘Version’

```
postgres=# SELECT *
postgres=# FROM cd
postgres=# WHERE cdgenre = (SELECT cdgenre FROM cd WHERE cdtitle = 'Version')
postgres=# AND cdtitle != 'Version';
 cdid | artid |  cdtitle  | cdprice | cdgenre | cdnumtracks
-----+-----+-----+-----+-----+-----
   10 |     8 | Hands All Over |    11.99 | Pop     |
(1 row)
```

5. Use IN to find a list of the titles of albums that are the same price as any ‘Electronica’ album

```
postgres=# SELECT cdtitle
postgres=# FROM cd
postgres=# WHERE cdprice IN (SELECT cdprice FROM cd WHERE cdgenre = 'Electronica')
postgres=# AND cdGenre != 'Electronica';
 cdtitle
-----
Black Holes and Revelations
For Lack of a Better Name
Version
Only By The Night
Come Around Sundown
(5 rows)
```

6. Use ANY to find the titles of CDs that cost less than at least one other CD

```
postgres=# SELECT cdtitle
postgres=# FROM cd
postgres=# WHERE cdprice < ANY (SELECT cdprice FROM cd WHERE cdprice > 0);
 cdtitle
-----
Black Holes and Revelations
The Resistance
Ninja Tuna
For Lack of a Better Name
Record Collection
Only By The Night
Hands All Over
(7 rows)
```

7. Use ALL to find a list of CD titles with the least price

```
postgres=# SELECT cdtitle
postgres=# FROM cd
postgres=# WHERE cdprice <= ALL(SELECT cdprice FROM cd);
           cdtitle
-----
Black Holes and Revelations
Ninja Tuna
For Lack of a Better Name
Only By The Night
(4 rows)
```

8. Use EXISTS to find a list of Artists who produced a “Pop” CD

```
postgres=# SELECT a.artname
postgres=# FROM artist AS a
postgres=# WHERE EXISTS (
postgres(#      SELECT 1
postgres(#      FROM cd AS c
postgres(#      WHERE c.artid = a.artid
postgres(#      AND c.cdgenre = 'Pop'
postgres(# );
           artname
-----
Mark Ronson
Maroon 5
(2 rows)
```

9. Find the names of Artists who have albums of “Rock” category. Provide at least two different queries producing the same solution

```
postgres=# SELECT DISTINCT artname
postgres=# FROM cd
postgres=# JOIN artist ON cd.artid = artist.artid
postgres=# WHERE cdgenre = 'Rock';
           artname
-----
Kings of Leon
Muse
(2 rows)

postgres=# SELECT artname
postgres=# FROM artist AS a
postgres=# WHERE EXISTS (
postgres(#      SELECT 1
postgres(#      FROM cd AS c
postgres(#      WHERE c.artid = a.artid
postgres(#      AND c.cdgenre = 'Rock'
postgres(# );
           artname
-----
Kings of Leon
Muse
(2 rows)
```

10. Find names of CDs produced by those Artists who have at least two words as their name

```
postgres=# SELECT cdtitle
postgres=# FROM cd
postgres=# JOIN artist AS a
postgres=# ON a.artid = cd.artid
postgres=# WHERE LENGTH(a.artname) - LENGTH(REPLACE(a.artname, ' ', '')) + 1 >= 2;
           cdtitle
-----
Ninja Tuna
Version
Record Collection
Merriweather Post Pavilion
Only By The Night
Come Around Sundown
Hands All Over
(7 rows)
```

11. Find all information about CDs which are cheaper than others in the 'Rock' and 'Pop' categories only

```
postgres=# SELECT *
postgres=# FROM cd
postgres=# WHERE cdprice <= ALL(
postgres=#     SELECT cdprice
postgres=#     FROM cd
postgres=#     WHERE cdgenre IN ('Rock', 'Pop')
postgres=# )
postgres=# AND cdgenre IN ('Rock', 'Pop');
 cdid | artid |          cdtitle          | cdprice | cdgenre | cdnumtracks
-----+-----+-----+-----+-----+-----
    1 |    1 | Black Holes and Revelations |    9.99 | Rock    |
    8 |    7 | Only By The Night          |    9.99 | Rock    |
(2 rows)
```

12. Find the Artist names and their ID (in this order) for albums which cost £12.99. Provide three different queries producing the same solution

```
postgres=# SELECT a.artname, a.artid
postgres=# FROM artist as a
postgres=# JOIN cd ON a.artid = cd.artid
postgres=# WHERE cd.cdprice = '12.99';
```

artname	artid
Mark Ronson	4
Animal Collective	6
Kings of Leon	7

(3 rows)

```
postgres=# SELECT artname, artid
postgres=# FROM artist
postgres=# WHERE artid IN (
postgres=# SELECT artid
postgres=# FROM cd
postgres=# WHERE cdprice = '12.99'
postgres=# );
```

artname	artid
Mark Ronson	4
Kings of Leon	7
Animal Collective	6

(3 rows)

```
postgres=# SELECT a.artname, a.artid
postgres=# FROM artist AS a
postgres=# WHERE EXISTS (
postgres=# SELECT 1
postgres=# FROM cd AS c
postgres=# WHERE c.artid = a.artid
postgres=# AND c.cdprice = '12.99'
postgres=# );
```

artname	artid
Mark Ronson	4
Kings of Leon	7
Animal Collective	6

(3 rows)

13. List the cd titles in reverse alphabetical order

```
postgres=# SELECT cdtitle
postgres=# FROM cd
postgres=# ORDER BY cdtitle DESC;
cdtitle
```

cdtitle
Version
The Resistance
Record Collection
Only By The Night
Ninja Tuna
Merriweather Post Pavilion
Hands All Over
For Lack of a Better Name
Come Around Sundown
Black Holes and Revelations

(10 rows)

14. List the titles, genres and prices of CDs in order of price from lowest to highest

```
postgres=# SELECT cdtitle, cdgenre, cdprice
postgres=# FROM cd
postgres=# ORDER BY cdprice;
```

cdtitle	cdgenre	cdprice
Black Holes and Revelations	Rock	9.99
Ninja Tuna	Electronica	9.99
Only By The Night	Rock	9.99
For Lack of a Better Name	Electro House	9.99
Hands All Over	Pop	11.99
The Resistance	Rock	11.99
Record Collection	Alternative Rock	11.99
Merriweather Post Pavilion	Electronica	12.99
Come Around Sundown	Rock	12.99
Version	Pop	12.99

(10 rows)

15. List the titles, genres and prices of CDs in order of price from highest to lowest

```
postgres=# SELECT cdtitle, cdgenre, cdprice
postgres=# FROM cd
postgres=# ORDER BY cdprice DESC;
```

cdtitle	cdgenre	cdprice
Merriweather Post Pavilion	Electronica	12.99
Version	Pop	12.99
Come Around Sundown	Rock	12.99
Hands All Over	Pop	11.99
The Resistance	Rock	11.99
Record Collection	Alternative Rock	11.99
For Lack of a Better Name	Electro House	9.99
Only By The Night	Rock	9.99
Ninja Tuna	Electronica	9.99
Black Holes and Revelations	Rock	9.99

(10 rows)

16. List the titles, genres and prices of CDs in alphabetical order by genre, then by price from the highest price to the lowest one

```
postgres=# SELECT cdtitle, cdgenre, cdprice
postgres=# FROM cd
postgres=# ORDER BY cdgenre, cdprice DESC;
```

cdtitle	cdgenre	cdprice
Record Collection	Alternative Rock	11.99
For Lack of a Better Name	Electro House	9.99
Merriweather Post Pavilion	Electronica	12.99
Ninja Tuna	Electronica	9.99
Version	Pop	12.99
Hands All Over	Pop	11.99
Come Around Sundown	Rock	12.99
The Resistance	Rock	11.99
Only By The Night	Rock	9.99
Black Holes and Revelations	Rock	9.99

(10 rows)

17. Find a list of artist names, the number of CDs they have produced, and the average price for their CDs. Only return results for artists with more than one CD

```
postgres=# SELECT
postgres=#     a.artname,
postgres=#     COUNT(c.cdidd) AS num_cds,
postgres=#     AVG(c.cdprice) AS avg_price
postgres=# FROM artist AS a
postgres=# JOIN cd AS c ON a.artid = c.artid
postgres=# GROUP BY a.artid
postgres=# HAVING COUNT(c.cdidd) > 1;
 artname      | num_cds |      avg_price
-----+-----+-----
Kings of Leon |       2 | 11.489999771118164
Muse          |       2 | 10.989999771118164
(2 rows)
```

18. Find a list of artist names, the number of CDs by that artist and the average price for their CDs but not including 'Electronica' albums (you might like to use a WHERE in this one too)

```
postgres=# SELECT
postgres=# a.artname,
postgres=# COUNT(c.cdidd) AS num_cds,
postgres=# AVG(c.cdprice) AS avg_price
postgres=# FROM artist AS a
postgres=# JOIN cd AS c ON c.artid = a.artid
postgres=# WHERE c.cdgenre != 'Electronica'
postgres=# GROUP BY a.artname;
 artname      | num_cds |      avg_price
-----+-----+-----
Kings of Leon |       2 | 11.489999771118164
Mark Ronson & The Business Intl |       1 | 11.989999771118164
Mark Ronson   |       1 | 12.989999771118164
Muse          |       2 | 10.989999771118164
DeadMau5     |       1 |  9.989999771118164
Maroon 5      |       1 | 11.989999771118164
(6 rows)
```

19. Find the difference between the average price of the rock albums and the average price of all albums excluding Muse's CDs (ABS() will produce the absolute value of a calculation)

```
postgres=# SELECT ABS(
postgres=#     (SELECT AVG(cdprice)
postgres=# FROM cd
postgres=# WHERE cdgenre = 'Rock') -
postgres=#     (SELECT AVG(cdprice)
postgres=# FROM cd
postgres=# WHERE artid NOT IN (SELECT artid FROM artist WHERE artname = 'Muse'))
postgres=# ) AS price_difference;
 price_difference
-----
0.375
(1 row)
```

20. Find the artist name(s) with the most expensive CDs by average

```
postgres=# SELECT a.artname,
postgres=#         AVG(c.cdprice) AS avg_price
postgres=# FROM cd AS c
postgres=# JOIN artist AS a ON c.artid = a.artid
postgres=# GROUP BY a.artname
postgres=# HAVING AVG(c.cdprice) >= (
postgres=#     SELECT MAX(avg_price)
postgres=#     FROM (
postgres=#         SELECT AVG(cdprice) AS avg_price
postgres=#         FROM cd
postgres=#         GROUP BY artid
postgres=#     )
postgres=# );
 artname      |      avg_price
-----+-----
Animal Collective | 12.989999771118164
Mark Ronson    | 12.989999771118164
(2 rows)
```

21. Find the most expensive genre of music by calculating the minimum of the averages of all genres (NOTE: Should the problem state “Find the least expensive genre...” ?)

```
postgres=# SELECT cdgenre, AVG(cdprice) AS avg_price
postgres=# FROM cd
postgres=# GROUP BY cdgenre
postgres=# ORDER BY avg_price ASC
postgres=# LIMIT 1;
 cdgenre      |      avg_price
-----+-----
Electro House | 9.989999771118164
(1 row)
```

22. Find a list of artist names, the number of CDs by that artist and the average price for their CDs but not including ‘Pop’ albums

```
postgres=# SELECT a.artname, COUNT(c.cdprice) AS num_cds, AVG(c.cdprice) AS avg_price
postgres=# FROM cd AS c
postgres=# JOIN artist AS a ON c.artid = a.artid
postgres=# WHERE c.cdgenre != 'Pop'
postgres=# GROUP BY a.artname;
 artname      | num_cds |      avg_price
-----+-----+-----
Kings of Leon |      2 | 11.489999771118164
Mr. Scruff    |      1 | 9.989999771118164
Mark Ronson & The Business Intl |      1 | 11.989999771118164
Animal Collective |      1 | 12.989999771118164
Muse          |      2 | 10.989999771118164
DeadMau5      |      1 | 9.989999771118164
(6 rows)
```