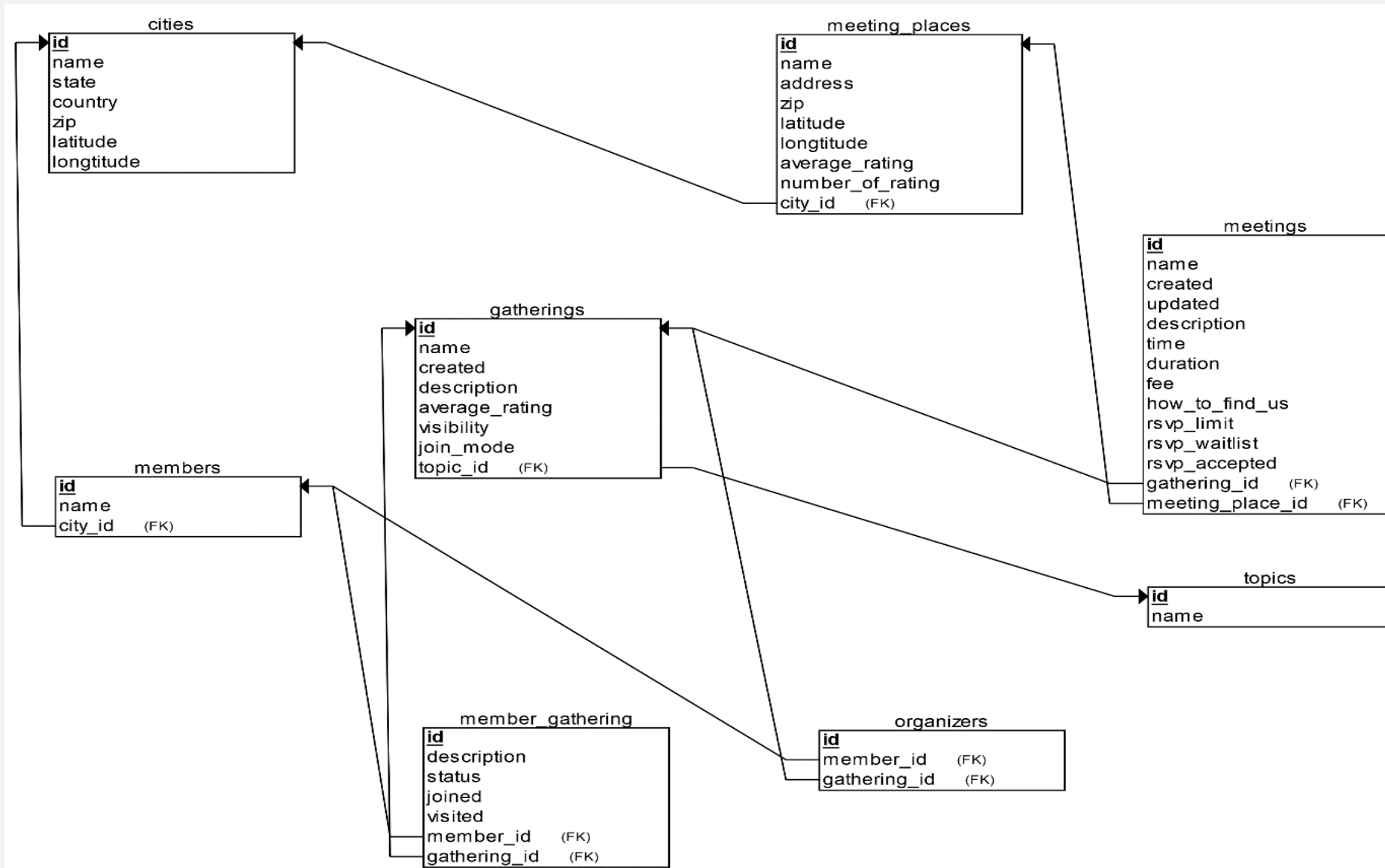


데이터 관리와 분석 PROJECT2

박소정, 이준현, 이창민, 한승우

DATABASE SCHEMA



RI

(1) DATABASE와 TABLE 구축

(2) CSV파일을 읽어서 데이터 삽입하기

(3) 외래키 설정하기

```
cnx = mysql.connector.connect(host=host, user=user, password=password)
cursor = cnx.cursor()
cursor.execute('SET GLOBAL innodb_buffer_pool_size=2*1024*1024*1024;')
```

```
cursor.execute("""DROP DATABASE IF EXISTS DMA_team12;""")
cursor.execute("""CREATE DATABASE DMA_team12;""")
cursor.execute("""USE DMA_team12;""")
```

```
#cities TABLE ? ?
cursor.execute("""
    CREATE TABLE IF NOT EXISTS cities(
        id VARCHAR(22) NOT NULL,
        name VARCHAR(255) NOT NULL,
        state VARCHAR(30) NOT NULL,
        country VARCHAR(30) NOT NULL,
        zip INT NULL,
        latitude FLOAT NULL,
        longitude FLOAT NULL,
        PRIMARY KEY (id));
    """)
```

R I

(2) CSV파일 읽어서 데이터 삽입하기

```
sql_cities = """INSERT INTO cities VALUES (%s, %s, %s, %s, %s, %s, %s)"""
```

```
f_cities = open(directory+'cities.csv', 'r', encoding='utf-8')  
next(f_cities)
```

```
while True:  
    line = f_cities.readline()  
    line_list = line.replace('\n', '')  
    line_list = line_list.split(',')  
    for i, j in enumerate(line_list):  
        if line_list[i] == '':  
            line_list[i] = None  
        try:  
            line_list[i] = int(j)  
        except:  
            pass  
        if line:  
            cursor.execute(sql_cities, line_list)  
    else:  
        break
```

Exception) Meeting_places - int(float(j))

RI
(3) 외래키 설정하기

```
cursor.execute("ALTER TABLE gatherings ADD CONSTRAINT FOREIGN KEY (topic_id)  
REFERENCES topics(id);")
```


R2~R5

```
def requirement2(host, user, password):  
    cnx = mysql.connector.connect(host=host, user=user, password=password)  
    cursor = cnx.cursor()  
    cursor.execute('SET GLOBAL innodb_buffer_pool_size=2*1024*1024*1024;')  
    cursor.execute('USE DMA_team%02d;' % team)
```

R2

```
1 • SELECT workson7.Year as Year, IFNULL(NewUser, 0) as NewUser, IFNULL(NewGroup, 0) as NewGroup, IFNULL(NewMeeting,0) as NewMeeting
2 FROM (SELECT year(joined) as Year, COUNT(*) as NewUser
3 FROM member_gathering
4 GROUP BY year(joined)) as workson1
5 RIGHT JOIN
6 (SELECT Year, NewGroup, NewMeeting
7 FROM ((SELECT workson2.Year, NewGroup, NewMeeting
8 FROM (SELECT year(created) as Year, COUNT(*) as NewGroup
9 FROM gatherings
10 GROUP BY year(created)) as workson2
11 LEFT JOIN (SELECT year(time) as Year, COUNT(*) as NewMeeting
12 FROM meetings
13 GROUP BY year(time)) as workson3
14 ON workson2.Year = workson3.Year)
15 UNION
16 (SELECT workson5.Year, NewGroup, NewMeeting
17 FROM (SELECT year(created) as Year, COUNT(*) as NewGroup
18 FROM gatherings
19 GROUP BY year(created)) as workson4
20 RIGHT JOIN (SELECT year(time) as Year, COUNT(*) as NewMeeting
21 FROM meetings
22 GROUP BY year(time)) as workson5
23 ON workson4.Year = workson5.Year)
24 ) as workson6
25 ) as workson7
26 ON workson1.Year = workson7.Year
27 ORDER BY Year
```

```
SELECT workson7.Year as Year,  
IFNULL(NewUser, 0) as NewUser,  
IFNULL(NewGroup, 0) as NewGroup,  
IFNULL(NewMeeting, 0) as NewMeeting  
FROM workson1  
RIGHT JOIN workson7  
ON workson1.Year = workson7.Year  
ORDER BY Year;
```



NewUser



NewGroup,
NewMeeting

```
SELECT Year, NewGroup, NewMeeting  
FROM
```

```
(
```

```
(SELECT workson2.Year, NewGroup, NewMeeting  
FROM (SELECT year(created) as Year, COUNT(*) as NewGroup  
FROM gatherings  
GROUP BY (created)) as workson2
```

```
LEFT JOIN (SELECT year(time) as Year, COUNT(*) as NewMeeting  
FROM meetings  
GROUP BY year(time)) as workson3  
ON workson2.Year = workson3.Year)
```

```
UNION
```

```
(SELECT workson2.Year, NewGroup, NewMeeting  
FROM (SELECT year(created) as Year, COUNT(*) as NewGroup  
FROM gatherings  
GROUP BY (created)) as workson4
```

```
RIGHT JOIN (SELECT year(time) as Year, COUNT(*) as NewMeeting  
FROM meetings  
GROUP BY year(time)) as workson5  
ON workson4.Year = workson5.Year)) as workson6  
) as workson7
```

	Year	NewUser	NewGroup	NewMeeting
▶	2002	69	52	0
	2003	2150	80	0
	2004	1785	25	0
	2005	2529	36	0
	2006	10819	207	0
	2007	26265	280	0
	2008	41665	310	0
	2009	71118	415	0
	2010	91002	431	0
	2011	139059	573	0
	2012	254796	823	0
	2013	393928	989	0
	2014	556617	1292	0
	2015	1196683	1631	0
	2016	1514295	2191	0
	2017	1591087	3212	1446
	2018	0	0	4102

📄 Action Output ▼


#	Time	Action	Message
1	18:14:02	SELECT workson7.Year as Year, IFNULL(NewUser, 0) as NewUser, IFNULL(NewGroup...	17 row(s) returned

Duration / Fetch
3.938 sec / 0.000 sec

R3

```
1 • SELECT MP.id as id, city_id, name, address
2 FROM meeting_places as MP, (SELECT view1.id as id
3 FROM (SELECT id
4 FROM meeting_places
5 WHERE number_of_rating>=5 AND average_rating>=4.5) as view1
6 LEFT JOIN (SELECT meeting_place_id as id
7 FROM meetings
8 GROUP BY meeting_place_id
9 HAVING COUNT(*)>3) as view2
10 ON view1.id=view2.id
11 WHERE view2.id is NULL) as view3
12 WHERE MP.id=view3.id
13 ORDER BY id;
```

```
SELECT MP.id as id, city_id, name, address  
FROM meeting_places as MP, view3  
WHERE MP.id=view3.id  
ORDER BY id
```



리뷰 수 5개 이상, 평균 평점 4.5 이상,
만남 3번 이하

```
SELECT view1.id as id
FROM (SELECT id
      FROM meeting_places
      WHERE number_of_rating >= 5 AND average_rating >= 4.5) as view1
LEFT JOIN (SELECT meeting_place_id as id
           FROM meetings
           GROUP BY meeting_place_id
           HAVING COUNT(*) > 3) as view2
ON view1.id = view2.id
WHERE view2.id is NULL
```

만남 3번 초과

리뷰 수 5개 이상, 평균
평점 4.5 이상,

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	id	city_id	name	address
▶	1000009	94101	Pier 38 (ashburymusichall.com office)	Embarcadero @ Townsend
	1000071	60601	Black Walnut Gallery	220 N. Aberdeen
	1000124	60601	Leone Beach Park c/o Loyola Park	1222 W Touhy Ave
	1000131	94101	SPUR	645 Mission St.
	10001772	10001	The Monterey Lounge	175 East 96th Street
	1000231	94101	Private location	Civic Center
	10003022	94101	Rackspace	Suite 100 620 Folsom Street

Result 9 ×

Read Only

Context Help

Snippets

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 1	17:06:57	SELECT MP.id as id, city_id, name, address FROM meeting_places as MP, (SELECT vi...	12336 row(s) returned	0.016 sec / 0.609 sec

R4

```
1 ● SELECT id, city_id, name
2   FROM members as m
3   WHERE m.id IN (SELECT member_id
4                   FROM member_gathering
5                   WHERE TIMESTAMPDIFF(YEAR, joined, visited) >= 1
6                   GROUP BY member_id
7                   HAVING COUNT(gathering_id) >= 5)
8   ORDER BY id;
```

```
SELECT id, city_id, name  
FROM members as m  
WHERE m.id IN view1  
ORDER BY id
```



1년 이상 활동한 그룹이 5개
이상인 사람

	id	city_id	name
▶	10000441	10001	danny
	10000561	94101	Kasia Rachuta
	100016592	10001	George Slavin
	100016832	94101	Claudia
	100017382	94101	Chelsea
	100017922	10001	Justin
	10001946	10001	Andrew
	100021792	94101	Jen
	10002568	10001	RUTH ODAMTTEN
	10003397	94101	Alex
	10003574	60601	Stella
	10003993	60601	OtherJoe
	10004127	10001	mike
	10004261	60601	Jessica Kirsch
	10004653	10001	glenn

Result
Grid

Form
Editor

Field
Types

Query
Stats

members 6 x

Apply

Revert

Context Help

Snippets

Output



Action Output

#	Time	Action	Message	Duration / Fetch
✓ 1	19:59:41	SELECT id, city_id, name FROM members as m WHERE m.id IN (SE...	59320 row(s) returned	29.109 sec / 2.719 sec

R5

```
1 • ALTER TABLE gatherings ADD remarks VARCHAR(20);
```

```
1 • UPDATE gatherings SET remarks = CASE
2   WHEN year(created)>=2014 AND id NOT IN (SELECT DISTINCT gathering_id
3     FROM member_gathering)
4     THEN "need promotion"
5   WHEN year(created)<2014 AND id NOT IN (SELECT DISTINCT gathering_id
6     FROM member_gathering)
7     THEN "to be deleted" END;
```

```
1  SELECT id, name, created, remarks
2  FROM gatherings
3  WHERE id NOT IN (SELECT DISTINCT gathering_id
4    FROM member_gathering)
5  ORDER BY id
```

```
UPDATE gatherings SET remarks=CASE
WHEN year(created)>=2014 AND id NOT IN view1
THEN "need promotion"
WHEN year(created)<2014 AND id NOT IN view1
THEN "to be deleted" END
```



그룹원이 한 명이라도
있는 그룹

	id	name	created	remarks
▶	10104	NYC Pit Bull Group	2003-10-22 21:39:00	to be deleted
	10359	NYC International Arabic Language & Culture Club	2003-05-22 14:19:00	to be deleted
✱	NULL	NULL	NULL	NULL

Result
Grid



aatherinos 2 x

Apply

Revert

Context Help

Snippets

Output

Action Output

	#	Time	Action	Message	Duration / Fetch
✓	1	23:36:50	ALTER TABLE gatherings ADD remarks VARCHAR(20)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.047 sec
✓	2	23:37:12	UPDATE gatherings SET remarks = CASE WHEN year(created)>=201...	2 row(s) affected Rows matched: 12547 Changed: 2 Warnings: 0	0.187 sec
✓	3	23:37:30	SELECT id, name, created, remarks FROM gatherings ...	2 row(s) returned	0.422 sec / 0.000 sec

R6 정의 및 QUERY 설명

R6 문제 정의

“사이트 A는 현재 가입자들이 어떤 주제에 관심이 많은지, 어떤 주제에서 양질의 교류가 이루어지고 있는지, 관심사에 대한 거시적인 정보를 파악하기 위해 가입자들에게 좋은 평가를 받은 그룹이 많이 있는 주제를 알아보고자 한다. 이를 위해 주제 별로 평균 평점 4.8이상을 받은 그룹의 개수를 구해 주제 이름과 평균 평점이 4.8이상인 그룹의 개수를 출력하라. 출력된 결과는 그룹의 개수를 기준으로 내림차순으로 정렬되어야 하며 column 순서는 주제 이름, 그룹의 개수여야 한다.”

R6 QUERY 설명

```
1 • SELECT T.name, COUNT(g.id) as total
2 FROM gatherings as G, topics as T
3 WHERE G.average_rating >= 4.8 AND G.topic_id = T.id
4 GROUP BY T.id
5 ORDER BY total DESC
```

	name	total
▶	Tech	1117
	Career & Business	716
	Health & Wellbeing	299
	Socializing	231
	New Age & Spirituality	224
	Language & Ethnic Identity	186
	Outdoors & Adventure	165
	Food & Drink	155
	Sports & Recreation	132
	Music	129
	Arts & Culture	115
	Education & Learning	115
	Fitness	108
	Community & Environment	93
	Games	82
	LGBT	79
	Movies & Film	71
	Dancing	70
	Photography	54
	Book Clubs	47

Result Grid

Form Editor

Field Types

Query Stats

Execution Plan

Result 6 x

Read Only

Context Help

Snippets

Output



Action Output

#	Time	Action	Message	Duration / Fetch
✓ 1	00:04:36	SELECT T.name, COUNT(g.id) as total FROM gatherings as G, topics ...	33 row(s) returned	0.016 sec / 0.000 sec

R2~R6 결과 출력 코드

R2~R6 결과 출력 코드

```
fopen = open('project2_team%02d_req6.txt' % team, 'w', encoding='utf8')

rows = cursor.fetchall()
for line in rows:
    for i in range(len(line)):
        fopen.write(str(line[i]))
        if i < len(line) - 1: fopen.write(';')
    if line != rows[len(rows) - 1]: fopen.write('\n')

fopen.close()
cursor.close()
```

늘어 주셔서 감사합니다