

## Fake Data for Alumni Association SQL Exercises

### Entity table

entity_id	first_name	last_name	birth_dt	primary_county	primary_state	spouse_entity_id
1001	James	Kang	2/28/1995 0:00	San Francisco	California	1003
1002	Ken	Miller	NULL	NULL	NULL	NULL
1003	Annette	Kang	6/28/1962 0:00	Portland	Oregon	1001

### entity\_record\_type table

entity_id	record_type_code	record_type
1001	AG	Alum: Grad
1002	AU	Alum: Undergrad
1003	GA	Attendee: Grad
1004	UA	Attendee: Undergrad

### membership table

memb_num	entity_id	memb_type_code	memb_type	memb_status
M01	1007	RA	CAA Regular Annual	Paid
M02	1010	RG	CAA Golden Bear Life Paid In Full	Cancelled
M03	1014	BL	CAA Golden Bear Life Upgrade	Complimentary
M04	1017	YA	CAA Young Alumni Annual	Lapsed

### Affiliation table

entity_id	affil_name	affil_level	affil_status
1005	CAA Saudi Arabia Alumni Club	Officer	Current
1007	Berk Haas Entrepreneurship Pgm	Volunteer	Current
1009	CAA Cal Alumni Club of Long Beach	President	Current
1012	CAA Japanese American Women Alumnae	Vice President	Former
1014	University Chorus	Member	Former

### Committee table

entity_id	committee_code	committee_desc	committee_status
1001	BA1	CAA Board of Directors	Current
1009	BL1	L&S Executive Board	Current
1012	BM1	BAMPFA Board of Trustees	Former

## SQL Exercises

1. Write a SQL query to randomly pull 5,000 young alumni in the Bay Area. The output file should include entity id, full name, age, county, state

- Alumni (record\_type\_code: AG, AU, GA, UA)
- Young Alumni (Up to 30 yrs)
- The nine Bay Area counties (Contra Costa, San Mateo, Alameda, Marin, Santa Clara, Solano, Sonoma, San Francisco, Napa)

### My Solution:

```
SELECT entity_record_type.entity_id, first_name, last_name,
DATE_PART('year', CURRENT_DATE)-DATE_PART('year',to_date(birth_dt, 'MM DD
YYYY')) AS age,primary_county AS county, primary_state AS state
FROM Entity
JOIN entity_record_type ON
entity.entity_id=entity_record_type.entity_id
WHERE primary_county IN ('Contra Costa', 'San Mateo', 'Alameda', 'Marin',
'Santa Clara', 'Solano','Sonoma', 'San Francisco', 'Napa') AND
DATE_PART('year', CURRENT_DATE)-DATE_PART('year',to_date(birth_dt, 'MM DD
YYYY'))<31 AND
entity_record_type.record_type_code IN ('AG', 'AU', 'GA', 'UA')
ORDER BY RANDOM()
LIMIT 5000
```

2. Write a SQL query to pull all alumni who fit these criteria

1. CAA Golden Bear Life Members (segment 1)
2. Current Alumni Chapter Presidents and Officers (segment 2)
3. **Exclude** current CAA Board of Directors

The output file should include entity id, first name, last name, segment number, segment description

- CAA Golden Bear Life Members (member type: RG, BL & member status: Paid, Complimentary)
- Current Alumni Chapter Members (affiliation name starts with 'CAA' & affiliation level: president, vice president, officer & affiliation status: current)
- Current CAA Board of Directors (committee code: BA1 & committee status: current)

### My Solution:

```
SELECT Entity.entity_id, first_name, last_name,
CASE
WHEN memb_type_code IN ('RG', 'BL') AND memb_status IN ('Paid',
'Complimentary')
AND (committee_code!='BA1' OR committee_status!='Current' OR committee_code
IS NULL OR committee_status IS NULL)
THEN 1
WHEN SUBSTRING(affil_name,1,3)='CAA' AND
affil_level IN ('President','Vice President','Officer') AND
affil_status='Current'
AND (committee_code!='BA1' OR committee_status!='Current' OR committee_code
IS NULL
```

```

        OR committee_status IS NULL)
THEN 2
END AS segment_number,
CASE
WHEN memb_type_code IN ('RG', 'BL') AND memb_status IN ('Paid',
'Complimentary')
AND (committee_code!='BA1' OR committee_status!='Current' OR committee_code
IS NULL
        OR committee_status IS NULL)
THEN 'CAA Golden Bear Life Members'
WHEN SUBSTRING(affil_name,1,3)='CAA' AND
affil_level IN ('President','Vice President','Officer') AND
affil_status='Current'
AND (committee_code!='BA1' OR committee_status!='Current' OR committee_code
IS NULL
        OR committee_status IS NULL)
THEN 'Current Alumni Chapter Presidents and Officers'
END AS segment_description

FROM Entity

LEFT JOIN membership ON
Entity.entity_id=membership.entity_id
LEFT JOIN Affiliation ON
Entity.entity_id=Affiliation.entity_id
LEFT JOIN Committee ON
Entity.entity_id=Committee.entity_id
ORDER BY segment_number

```

3. Write a SQL query to pull all alumni who live in California. The output file should include entity id, full name, spouse entity id, spouse full name, state

- Alumni (record\_type\_code: AG, AU, GA, UA)

**My Solution:**

```

SELECT entity_record_type.entity_id, ent1.first_name, ent1.last_name,
ent1.spouse_entity_id, ent2.first_name AS spouse_first_name, ent2.last_name
AS spouse_last_name,
ent1.primary_state AS state
FROM Entity ent1
JOIN entity_record_type ON
ent1.entity_id=entity_record_type.entity_id
JOIN Entity ent2 ON
ent1.spouse_entity_id=ent2.entity_id
WHERE ent1.primary_state='California' AND
        entity_record_type.record_type_code IN ('AG', 'AU', 'GA', 'UA')

```

4. For each CAA membership type, list each membership status and number of counts. Only show the unique counts of at least 10,000. Sort the counts from the highest value to the lowest value.

**My Solution:**

```
SELECT memb_type, memb_status, COUNT(memb_status)
5. FROM membership
6. GROUP BY memb_type, memb_status
7. HAVING COUNT(memb_status) >= 10000
8. ORDER BY COUNT(memb_status) DESC
```