Edward Ramirez

Sql Assignment 8

1.
```sql
select distinct first_name,last_name, to_Char(registration_date, 'Mon DD
YYYY') as REGDATE
from student s
inner join enrollment e
on s.student_id = e.student_id
where enroll_date =
(select(min (enroll_Date))
from enrollment)
order by last_name,first_name;
```
2.
```sql
select description, s.course_no, cost, capacity
from course c
inner join section s
on c.course_no = s.course_no
where cost =
(select min(cost)
from course)
and capacity <=
(select avg(capacity)
from section)
order by description, course_no;
```
3.
```sql
select c.course_no, description, sum(capacity) as TOTAL_CAPACITY
from course c
inner join section s
on c.course_no = s.course_no
group by c.course_no, description
having sum(capacity) <
(select avg(capacity)
from section)
order by description, "TOTAL_CAPACITY";
```
4.
```sql
select s.student_id, first_name,last_name, count(*) as MOST_CLASSES
from student s
inner join enrollment e
on s.student_id = e.student_id
group by s.student_id, first_name,last_name
having count(*) =
(select max(MOST_CLASSES)
from
(select s.student_id, first_name,last_name, count(*) as MOST_CLASSES
from student s
inner join enrollment e
on s.student_id = e.student_id
group by s.student_id, first_name,last_name))
order by first_name,last_name;
```
5.
```sql
select distinct first_name,last_name
from student s
```

```
inner join enrollment e
on s.student_id = e.student_id
inner join zipcode z
on s.zip = z.zip
where z.zip in
(select z.zip
from student s
inner join enrollment e
on s.student_id = e.student_id
inner join zipcode z
on s.zip = z.zip
group by z.zip
having count(*) =
    (select max(CountNum)
    from
        (select z.zip,count(*) as CountNum
        from student s
        inner join enrollment e
        on s.student_id = e.student_id
        inner join zipcode z
        on s.zip = z.zip
        group by z.zip)))
order by last_name,first_name;
6.
select c.course_no, description
from course c
inner join section s
on c.course_no = s.course_no
inner join instructor i
on s.instructor_id = i.instructor_id
where first_name = 'Anita' and last_name= 'Morris'
order by description;
7.
SELECT first_name, last_name
FROM student s
WHERE student_id NOT IN
    (SELECT s.student_id
        FROM student s
        INNER JOIN enrollment e
        ON s.student_id = e.student_id)
ORDER BY last_name, first_name;

8.
SELECT s.first_name, s.last_name, c.description, e.section_id
FROM student s
INNER JOIN enrollment e
ON s.student_id = e.student_id
INNER JOIN grade g
ON e.student_id = g.student_id
AND e.section_id = g.section_id
INNER JOIN section sec
ON e.section_id = sec.section_id
INNER JOIN course c
ON c.course_no = sec.course_no
```

```sql
WHERE grade_type_code = 'PJ'
AND numeric_grade =
    (SELECT MIN(numeric_grade)
        FROM grade
        WHERE grade_type_code = 'PJ')
ORDER BY last_name, first_name, description;
9.
SELECT COUNT(8) AS Enrolled, sec.section_id, capacity
FROM enrollment e
INNER JOIN section sec
ON e.section_id = sec.section_id
GROUP BY sec.section_id, capacity
HAVING COUNT(*)>= capacity
ORDER BY capacity DESC, section_id;
10.
SELECT course_no, description, cost
FROM course
WHERE COST =
    (SELECT MIN(cost) AS COST
        FROM course)
ORDER BY course_no, description;
```