Set A

1. For each job, count the number of managers. [Note: You are not allowed to use subquery.]

```
SELECT j.job_title, COUNT(*) AS manager_count
FROM hr.employees m
  JOIN hr.employees e ON (e.manager_id = m.employee_id)
  JOIN hr.jobs j ON (m.job_id = j.job_id)
GROUP BY j.job_title
```

2. For each department, find the three most junior employees (It is not necessary for an employee to be serving currently in this case). If there is a draw, print all of them (hence, there could be more than three employees printed for some department). Print department name, full name and hire date of employee. Ensure that the same departments are put in consecutively. [Note: You are not allowed to use subquery.]

```
SELECT
  d.department_name,
  (e.first_name || ' ' || e.last_name) AS employee_name,
  e.hire_date
FROM hr.departments d
  JOIN hr.employees e
  ON (e.department_id = d.department_id)
  LEFT JOIN hr.employees f
  ON (f.department_id = e.department_id
    AND f.hire_date < e.hire_date)
GROUP BY d.department_name, e.first_name, e.last_name, e.hire_date
HAVING COUNT(f.employee_id) < 3
ORDER BY d.department_name
```

3. Find the employees whose managers come from the same region as them. Print the full name and region name of the employee.

```
SELECT
  (e.first_name || ' ' || e.last_name) AS employee_name,
  r.region_name
FROM (
  SELECT e.first_name, e.last_name, e.manager_id, c.region_id
  FROM hr.employees e
    JOIN hr.departments d USING(department_id)
    JOIN hr.locations l USING(location_id)
    JOIN hr.countries c USING(country_id)
) e
JOIN (
  SELECT e.employee_id, c.region_id
  FROM hr.employees e
    JOIN hr.departments d USING(department_id)
    JOIN hr.locations l USING(location_id)
    JOIN hr.countries c USING(country_id)
) m
ON (e.manager_id = m.employee_id AND e.region_id = m.region_id)
JOIN hr.regions r ON (e.region_id = r.region_id)
```

4. Find the employees who are senior to at least half of the employees in his job (All the employees are currently serving in this case).

```
SELECT
  (e.first_name || ' ' || e.last_name) AS employee_name,
  e.job_id,
  e.hire_date
FROM hr.employees e
WHERE (
  SELECT COUNT(*)
  FROM hr.employees e2
  WHERE e2.job_id = e.job_id AND e2.hire_date > e.hire_date
) >= (
  SELECT COUNT(*)/2
  FROM hr.employees e2
```

```
   WHERE e2.job_id = e.job_id
)
```

5. Rank the jobs by number of managers.

```
SELECT 1+COUNT(t2.job_id) AS rank, t1.job_id, t1.manager_count
FROM
(
   SELECT e.job_id, COUNT(e.employee_id) AS manager_count
   FROM hr.employees e
   GROUP BY e.job_id
) t1
LEFT JOIN
(
   SELECT e.job_id, COUNT(e.employee_id) AS manager_count
   FROM hr.employees e
   GROUP BY e.job_id
) t2
ON (t1.manager_count < t2.manager_count)
GROUP BY t1.job_id, t1.manager_count
ORDER BY rank
```

Set B

1. For each department, count the number of managers. [Note: You are not allowed to use subquery.]

```
SELECT d.department_name, COUNT(*) AS manager_count
FROM hr.employees m
  JOIN hr.employees e ON (e.manager_id = m.employee_id)
  JOIN hr.departments d ON (m.department_id = d.department_id)
GROUP BY d.department_name
```

2. For each department, find the three most senior employees (It is not necessary for an employee to be serving currently in this case). If there is a draw, print all of them (hence, there could be more than three employees printed for some department). Print department name, full name and hire date of employee. Ensure that the same departments are put in consecutively. [Note: You are not allowed to use subquery.]

```
SELECT
  d.department_name,
  (e.first_name || ' ' || e.last_name) AS employee_name,
  e.hire_date
FROM hr.departments d
  JOIN hr.employees e
  ON (e.department_id = d.department_id)
  LEFT JOIN hr.employees f
  ON (f.department_id = e.department_id
    AND f.hire_date > e.hire_date)
GROUP BY d.department_name, e.first_name, e.last_name, e.hire_date
HAVING COUNT(f.employee_id) < 3
ORDER BY d.department_name
```

3. Find the employees that are managed by an employee who is not of his own country. Print the full name and country name of both the employee and his manager.

```
SELECT
  (e.first_name || ' ' || e.last_name) AS employee_name,
  e.country_name,
  (m.first_name || ' ' || m.last_name) AS manager_name,
  m.country_name
FROM (
  SELECT e.first_name, e.last_name, e.manager_id, c.country_name
  FROM hr.employees e
     JOIN hr.departments d USING(department_id)
     JOIN hr.locations l USING(location_id)
     JOIN hr.countries c USING(country_id)
) e
JOIN (
  SELECT e.first_name, e.last_name, e.employee_id, c.country_name
  FROM hr.employees e
     JOIN hr.departments d USING(department_id)
     JOIN hr.locations l USING(location_id)
     JOIN hr.countries c USING(country_id)
) m
ON (e.manager_id = m.employee_id AND e.country_name <> m.country_name)
```

4. Find the employees who are junior to at most half of the employees in his job (All the employees are currently serving in this case).

```
SELECT
   (e.first_name || ' ' || e.last_name) AS employee_name,
   e.job_id,
   e.hire_date
FROM hr.employees e
WHERE (
   SELECT COUNT(*)
   FROM hr.employees e2
   WHERE e2.job_id = e.job_id AND e2.hire_date < e.hire_date
) <= (
   SELECT COUNT(*)/2
   FROM hr.employees e2
   WHERE e2.job_id = e.job_id
)
```

5. Rank the departments by number of managers.

```
SELECT 1+COUNT(t2.department_id) AS rank, t1.department_id, t1.manager_count
FROM
(
   SELECT e.department_id, COUNT(e.employee_id) AS manager_count
   FROM hr.employees e
   GROUP BY e.department_id
) t1
LEFT JOIN
(
   SELECT e.department_id, COUNT(e.employee_id) AS manager_count
   FROM hr.employees e
   GROUP BY e.department_id
) t2
ON (t1.manager_count < t2.manager_count)
GROUP BY t1.department_id, t1.manager_count
ORDER BY rank
```