```sql
/* 1. Display last name and job id for all employees who perform the same job as Davies. Exclude Davies
from this query.
*/

SELECT LAST_NAME "Last Name", JOB_ID "Job ID" FROM EMPLOYEES WHERE JOB_ID =
(SELECT JOB_ID FROM EMPLOYEES WHERE LAST_NAME = 'Davies') AND LAST_NAME != 'Davies';
```

Script Output

Task completed in 0.046 seconds

```
Last Name                 Job ID
------------------------- ----------
Rajs                      ST_CLERK
Matos                     ST_CLERK
Vargas                    ST_CLERK
```

```sql
/* 2. Display last name, job id and hire date for all employees hired after Grant.
Sort the output by the most recent hire date.
*/

SELECT LAST_NAME "Last Name", JOB_ID "Job ID", HIRE_DATE "Hire Date" FROM EMPLOYEES
WHERE HIRE_DATE > (SELECT HIRE_DATE FROM EMPLOYEES WHERE LAST_NAME = 'Grant')
ORDER BY "Hire Date" DESC;
```

Script Output x | Query Result x

SQL | All Rows Fetched: 2 in 0.007 seconds

	Last Name	Job ID	Hire Date
1	Zlotkey	SA_MAN	29-JAN-00
2	Mourgos	ST_MAN	16-NOV-99

```sql
/* 3. Display city, province name and postal code for all departments located in countries that start with
letter I (meaning Italy, Israel and India). If the province is blank, show message Unknown and the
heading should be Province.
Sort the output by city ascending.
*/

SELECT CITY "City", STATE_PROVINCE || CASE WHEN STATE_PROVINCE IS NULL THEN 'Unknown' END "Province", POSTAL_CODE "Postal Code" FROM LOCATIONS
WHERE COUNTRY_ID LIKE 'I%'
ORDER BY 1;
```

Script Output

Task completed in 0.048 seconds

City	Province	Postal Code
Bombay	Maharashtra	490231
Roma	Unknown	00989
Venice	Unknown	10934

```
/* 4. Display last name, job id and salary for all employees who earn less than the Average salary in the
Sales department. Do NOT use Join method.
Sort the output by top salaries first and then by job_title.
*/

SELECT LAST_NAME "Last Name", JOB_ID "Job Title", SALARY "Salary" FROM EMPLOYEES
WHERE SALARY < (SELECT AVG(SALARY)FROM EMPLOYEES WHERE DEPARTMENT_ID = (SELECT DEPARTMENT_ID FROM DEPARTMENTS WHERE DEPARTMENT_NAME = 'Sales'))
ORDER BY "Salary" DESC, "Job Title" ;
```

Script Output x | Query Result x

SQL | All Rows Fetched: 13 in 0.006 seconds

	Last Name	Job Title	Salary
1	Hunold	IT_PROG	9000
2	Taylor	SA_REP	8600
3	Gietz	AC_ACCOUNT	8300
4	Grant	SA_REP	7000
5	Ernst	IT_PROG	6000
6	Fay	MK_REP	6000
7	Mourgos	ST_MAN	5800

```sql
/* 5. Display last name, job id and salary for all employees whose salary matches any of
the salaries from the IT Department.
Sort the output by salary ascending first and then by last_name.
*/

SELECT LAST_NAME "Last Name", JOB_ID "Job ID", SALARY "Salary" FROM EMPLOYEES
WHERE SALARY IN (SELECT SALARY FROM EMPLOYEES WHERE DEPARTMENT_ID = (SELECT DEPARTMENT_ID FROM DEPARTMENTS WHERE DEPARTMENT_NAME = 'IT'))
ORDER BY "Salary", "Last Name";
```

Script Output × | Query Result ×

Task completed in 0.046 seconds

Last Name	Job ID	Salary
Lorentz	IT_PROG	4200
Ernst	IT_PROG	6000
Fay	MK_REP	6000
Hunold	IT_PROG	9000

```sql
/* 5. Display last name, job id and salary for all employees whose salary matches any of
the salaries from the IT Department.
Sort the output by salary ascending first and then by last_name.
*/

SELECT LAST_NAME "Last Name", JOB_ID "Job ID", SALARY "Salary" FROM EMPLOYEES
WHERE SALARY IN (SELECT SALARY FROM EMPLOYEES WHERE DEPARTMENT_ID = (SELECT DEPARTMENT_ID FROM DEPARTMENTS WHERE DEPARTMENT_NAME = 'IT'))
ORDER BY "Salary", FIRST_NAME, "Last Name";
```

ript Output x | Query Result x

SQL | All Rows Fetched: 4 in 0.003 seconds

	Last Name	Job ID	Salary
1	Lorentz	IT_PROG	4200
2	Ernst	IT_PROG	6000
3	Fay	MK_REP	6000
4	Hunold	IT_PROG	9000

```
/* 6. Display last name and salary for all employees who earn less than the Lowest salary in ANY
department.
Sort the output by top salaries first and then by last name.
*/

SELECT LAST_NAME "Last Name", SALARY FROM EMPLOYEES
WHERE SALARY < ANY (SELECT MIN(SALARY) FROM EMPLOYEES
GROUP BY DEPARTMENT_ID)
ORDER BY 2 DESC, "Last Name";
```

Script Output x | Query Result x

SQL | All Rows Fetched: 17 in 0.003 seconds

	Last Name	SALARY
1	Hartstein	13000
2	Higgins	12000
3	Abel	11000
4	Zlotkey	10500
5	Hunold	9000
6	Taylor	8600
7	Gietz	8300
8	Grant	7000
9	Ernst	6000
10	Fay	6000
11	Mourgos	5800
12	Whalen	4400
13	Lorentz	4200
14	Rajs	3500
15	Davies	3100
16	Matos	2600
17	Vargas	2500