

1. The HR department needs a list of department IDs for departments that do not contain the job ID ST_CLERK. Use set operators to create this report.

```
1  SELECT department_id
2  FROM departments
3  MINUS
4  SELECT DISTINCT department_id
5  FROM employees
6  WHERE job_id = 'ST_CLERK';
7
```

Results Explain Describe Saved SQL History

DEPARTMENT_ID

10
20
30
50
80

2. The HR department needs a list of countries that have no departments located in them. Display the country ID and the name of the countries. Use set operators to create this report.

```
1  SELECT country_id, country_name
2  FROM countries
3  MINUS
4  SELECT DISTINCT c.country_id, c.country_name
5  FROM countries c
6  JOIN locations l ON c.country_id = l.country_id
7  JOIN departments d ON l.location_id = d.location_id;
8
```

Results Explain Describe Saved SQL History

COUNTRY_ID

COUNTRY_NAME

AU	Australia
CA	Canada
US	United States

3. Produce a list of jobs for departments 10, 50, and 20, in that order. Display job ID and department ID using set operators.

```

1  SELECT job_id, department_id FROM employees WHERE department_id = 10
2  UNION ALL
3  SELECT job_id, department_id FROM employees WHERE department_id = 50
4  UNION ALL
5  SELECT job_id, department_id FROM employees WHERE department_id = 20;
6

```

Results	Explain	Describe	Saved SQL	History	Bottom Splitter
		JOB_ID		DEPARTMENT_ID	
AD_ASST			10		
SA_REP			10		
IT_PROG			10		
SA_MAN			20		
IT_PROG			20		

4. Create a report that lists the employee IDs and job IDs of those employees who currently have a job title that is the same as their job title when they were initially hired by the company (that is, they changed jobs but have now gone back to doing their original job).

```

1  SELECT employee_id, job_id
2  FROM employees
3  INTERSECT
4  SELECT employee_id, job_id
5  FROM job_history;
6

```

Results	Explain	Describe	Saved SQL	History	Bottom Splitter
		EMPLOYEE_ID		JOB_ID	
201				SA_REP	
202				IT_PROG	
204				SA_MAN	

5. The HR department needs a report with the following specifications:

- Last name and department ID of all the employees from the EMPLOYEES table, regardless of whether or not they belong to a department.
- Department ID and department name of all the departments from the DEPARTMENTS table, regardless of whether or not they have employees working in them Write a compound query to accomplish this.

```
1 SELECT last_name AS name, department_id, NULL AS department_name
2 FROM employees
3 UNION
4 SELECT NULL AS name, department_id, department_name
5 FROM departments;
```

Results		
NAME	DEPARTMENT_ID	DEPARTMENT_NAME
Brown	20	-
Hughes	20	-
Hunter	10	-
Junior	10	-
Lee	30	-
Miller	10	-
-	10	IT
-	20	Human Resources