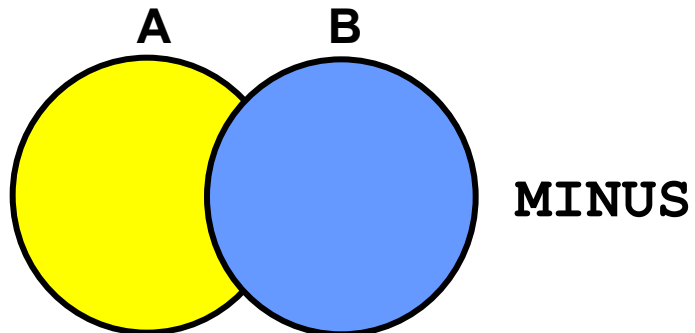
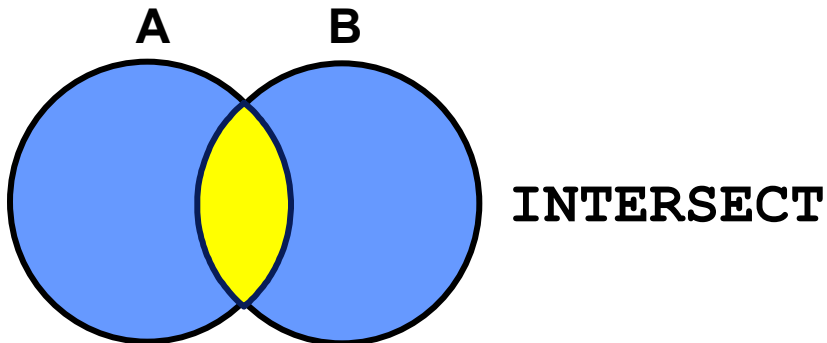
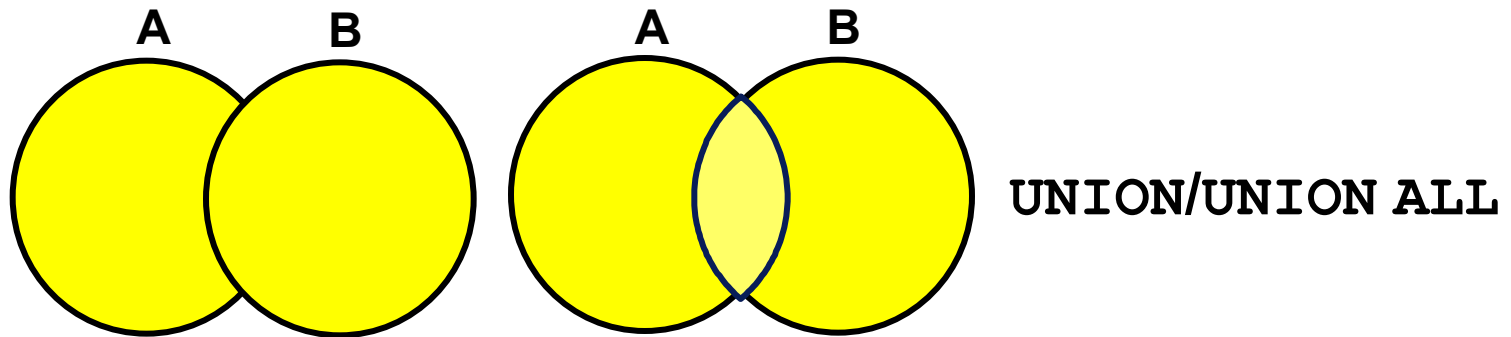


Utiliser des opérateurs ensemblistes

Opérateurs ensemblistes (SET)



Règles relatives aux opérateurs ensemblistes

- Les listes `SELECT` doivent contenir le même nombre d'expressions.
- Le type de données de chaque colonne de la deuxième interrogation doit être identique au type de données de la colonne correspondante dans la première interrogation.
- Il est possible d'utiliser des parenthèses pour modifier la séquence d'exécution.
- La clause `ORDER BY` ne peut apparaître qu'à la toute fin de l'instruction.

Serveur Oracle et opérateurs ensemblistes

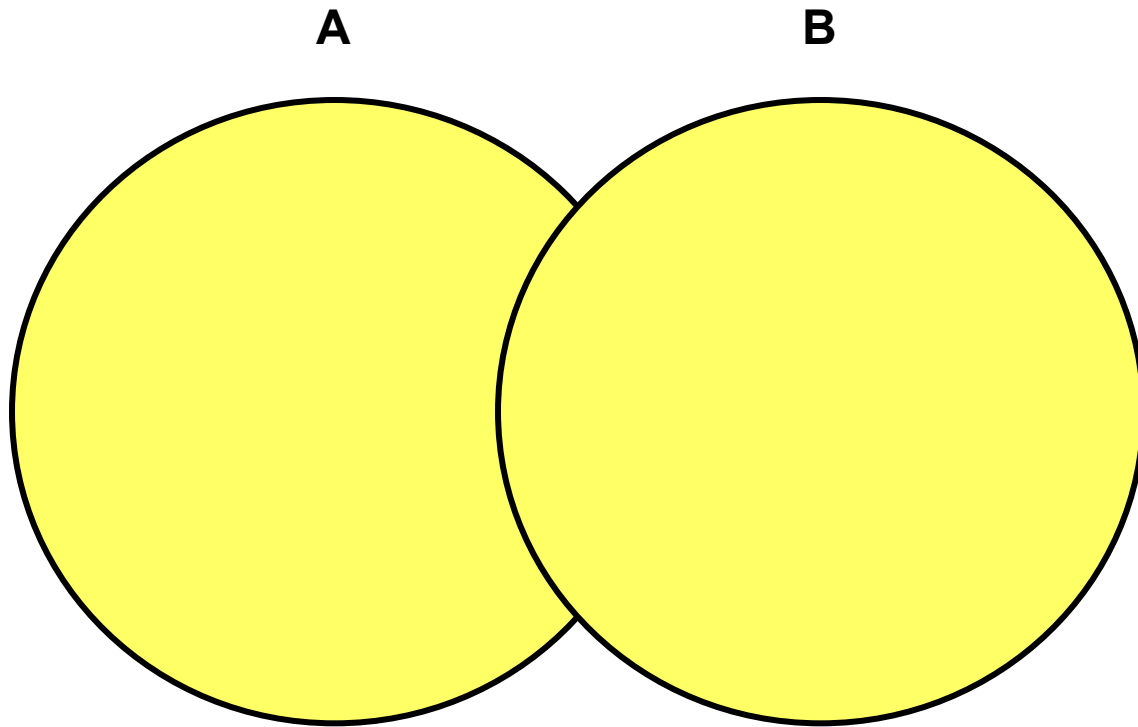
- Les lignes en double sont automatiquement éliminées sauf avec `UNION ALL`.
- Les noms de colonne de la première interrogation apparaissent dans le résultat.
- Par défaut, le résultat est trié par ordre croissant, sauf avec `UNION ALL`.

Tables utilisées dans le chapitre

Les tables suivantes sont utilisées dans ce chapitre :

- `EMPLOYEES` : Fournit des détails sur tous les employés en poste.
- `JOB_HISTORY` : Enregistre la date de début et la date de fin du poste précédent, ainsi que l'ID de poste et le département lorsqu'un employé change de poste.

Opérateur UNION



L'opérateur UNION renvoie les lignes issues des deux interrogations après élimination des doublons.

Utiliser l'opérateur UNION

Affichez les détails relatifs au poste actuel et au poste précédent de tous les employés. N'affichez qu'une seule fois chaque employé.

```
SELECT employee_id,  
job_id  
FROM employees  
UNION  
SELECT employee_id,  
job_id
```

	EMPLOYEE_ID	JOB_ID
1	100	AD_PRES
2	101	AC_ACCOUNT

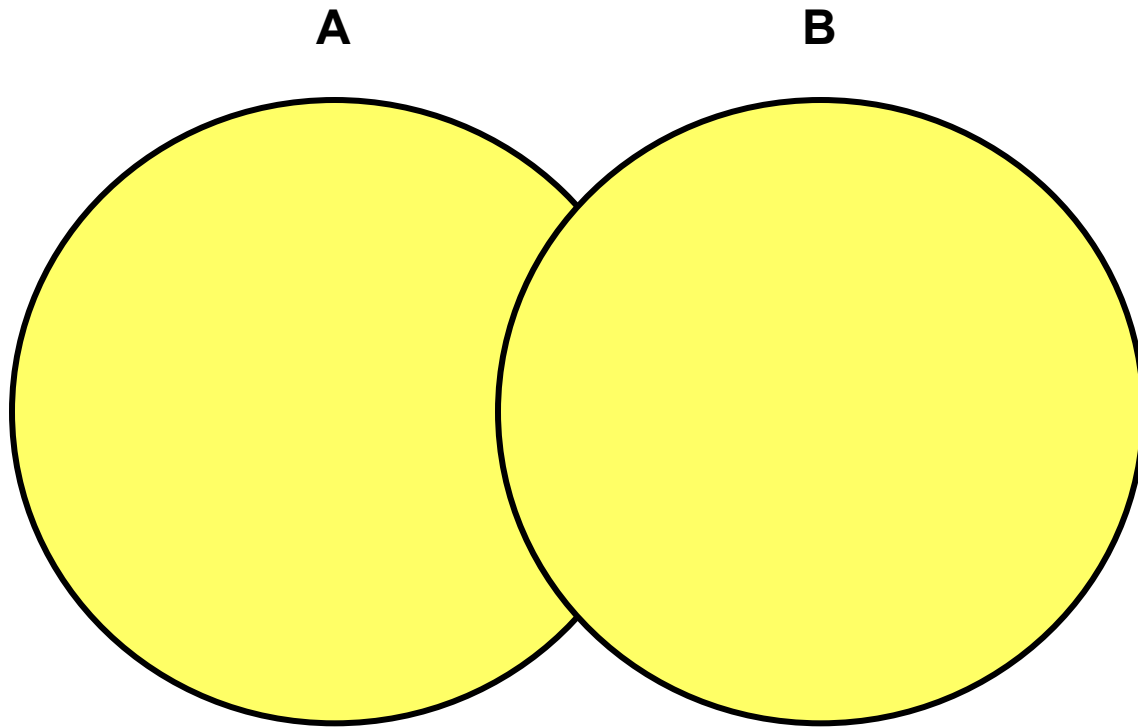
...

22	200	AC_ACCOUNT
23	200	AD_ASST

...

27	205	AC_MGR
28	206	AC_ACCOUNT

Opérateur UNION ALL



L'opérateur UNION ALL renvoie les lignes issues des deux interrogations, y compris tous les doublons.

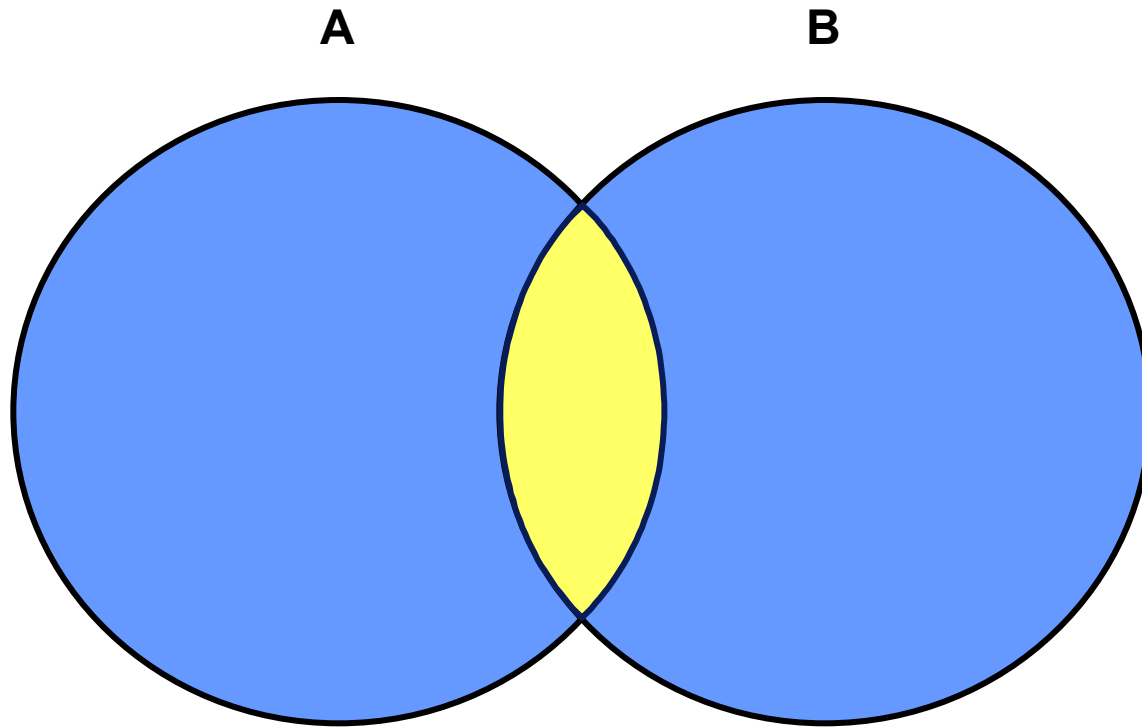
Utiliser l'opérateur UNION ALL

Affichez le département actuel et le département précédent de chaque employé.

```
SELECT employee_id, job_id, department_id
FROM   employees
UNION ALL
SELECT employee_id, job_id, department_id
FROM   job_history
ORDER BY employee_id;
```

	EMPLOYEE_ID	JOB_ID	DEPARTMENT_ID
1	100	AD_PRES	90
...			
17	149	SA_MAN	80
18	174	SA_REP	80
19	176	SA_REP	80
20	176	SA_MAN	80
21	176	SA_REP	80
22	178	SA_REP	(null)
23	200	AD_ASST	10
...			
30	206	AC_ACCOUNT	110

Opérateur INTERSECT



L'opérateur **INTERSECT** renvoie les lignes communes aux deux interrogations.

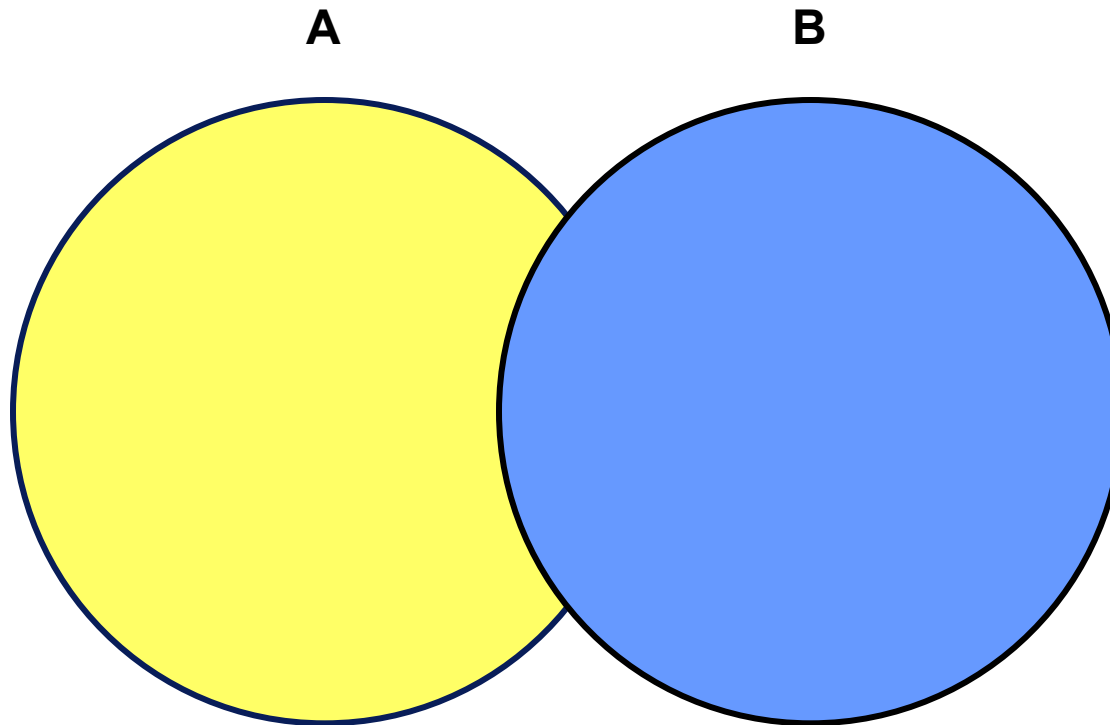
Utiliser l'opérateur INTERSECT

Affichez les ID d'employé et de poste des employés dont l'intitulé de poste actuel est identique au précédent. (Les employés ont changé de poste mais sont maintenant revenus à leur poste précédent.)

```
SELECT employee_id, job_id
FROM employees
INTERSECT
SELECT employee_id, job_id
FROM job_history;
```

	EMPLOYEE_ID	JOB_ID
1	176	SA_REP
2	200	AD_ASST

Opérateur MINUS



L'opérateur MINUS renvoie toutes les lignes distinctes sélectionnées par la première interrogation, mais non présentes dans l'ensemble de résultats de la deuxième interrogation.

Utiliser l'opérateur MINUS

Affichez l'ID des employés qui n'ont jamais changé de poste.

```
SELECT employee_id  
FROM employees  
MINUS  
SELECT employee_id  
FROM job_history;
```

	EMPLOYEE_ID
1	100
2	103
3	104

...

13	202
14	205
15	206

Assurer la correspondance des instructions SELECT

- A l'aide de l'opérateur `UNION`, affichez l'ID de lieu, le nom de département et l'état dans lequel le lieu est situé.
- Si des colonnes n'existent pas dans l'une des tables, vous devez assurer la correspondance du type de données (à l'aide de la fonction `TO_CHAR` ou de n'importe quelle autre fonction de conversion).

```
SELECT location_id, department_name "Department",  
       TO_CHAR(NULL) "Warehouse location"  
FROM departments  
UNION  
SELECT location_id, TO_CHAR(NULL) "Department",  
       state_province  
FROM locations;
```

Assurer la correspondance des instructions

SELECT : Exemple

A l'aide de l'opérateur UNION, affichez l'ID d'employé, l'ID de poste et le salaire de tous les employés.

```
SELECT employee_id, job_id, salary
FROM   employees
UNION
SELECT employee_id, job_id, 0
FROM   job_history;
```

	EMPLOYEE_ID	JOB_ID	SALARY
1	100	AD_PRES	24000
2	101	AC_ACCOUNT	0
3	101	AC_MGR	0
4	101	AD_VP	17000
5	102	AD_VP	17000
...			
29	205	AC_MGR	12000
30	206	AC_ACCOUNT	8300

Utiliser la clause ORDER BY dans des opérations ensemblistes

- La clause ORDER BY ne peut apparaître qu'une fois à la fin de l'interrogation composée.
- Les sous-interrogations ne peuvent pas comporter de clauses ORDER BY individuelles.
- La clause ORDER BY reconnaît uniquement les colonnes de la première interrogation SELECT.
- Par défaut, la première colonne de la première interrogation SELECT est utilisée pour le tri du résultat par ordre croissant.