



TECNICAS DE ALMACENAMIENTO DE DATOS



Sentencias DDL, DML y DCL

DDL (Data Definition Language)

Lenguaje de definición de datos

DML (Data Manipulation Language)

Lenguaje de manipulación de datos

DCL (Data Control Language)

Lenguaje de control de datos

DDL (Data Definition Language)

Lenguaje de definición de datos

CREATE
ALTER
DROP

TABLE
INDEX
VIEW
SYNONYM
FUNCTION
PROCEDURE
PACKAGE
TRIGGER

DML (Data Manipulation Language)

Lenguaje de manipulación de datos

INSERT	INSERT INTO TABLA (CAMPO_1, CAMPO_2, CAMPO_3) VALUES (VALOR1, VALOR2, VALOR3) ;
UPDATE	UPDATE TABLA SET CAMPO_N = VALOR WHERE CAMPO_1 = VALOR1 ;
DELETE	DELETE FROM TABLA WHERE CAMPO = VALOR ;
SELECT	SELECT * FROM TABLA WHERE CAMPO = VALOR ;

DCL (Data Control Language)

Lenguaje de control de datos

- SYSTEM PRIVILEGES

```
SELECT DISTINCT PRIVILEGE  
FROM DBA_SYS_PRIVS;
```

- OBJECT PRIVILEGES

```
SELECT DISTINCT PRIVILEGE  
FROM DBA_TAB_PRIVS;
```

DCL (Data Control Language)

Lenguaje de control de datos

```
--SINTAXIS SYSTEM PRIVILEGES  
GRANT  privileges TO  user;  
REVOKE privileges FROM user;
```



```
CREATE TABLE  
ALTER TABLE  
DROP TABLE  
CREATE SESSION  
ALTER SESSION  
...
```

DCL (Data Control Language)

Lenguaje de control de datos

GRANT

```
--EJEMPLO GRANT (SYSTEM PRIVILEGE)  
GRANT CREATE TABLE  
TO SCOTT;
```

REVOKE

```
--EJEMPLO REVOKE (SYSTEM PRIVILEGE)  
REVOKE CREATE TABLE  
FROM SCOTT;
```

DCL (Data Control Language)

Lenguaje de control de datos

--SINTAXIS OBJECT PRIVILEGES

GRANT **privileges** ON object TO user;

REVOKE **privileges** ON object FROM user;



SELECT

INSERT

UPDATE

DELETE

REFERENCES

ALL

DCL (Data Control Language)

Lenguaje de control de datos

GRANT

--EJEMPLO GRANT (OBJECT PRIVILEGE)
GRANT INSERT, SELECT
ON EMPLOYEES
TO SCOTT
WITH GRANT OPTION;

REVOKE

--EJEMPLO REVOKE (OBJECT PRIVILEGE)
REVOKE DELETE
ON EMPLOYEES
FROM SCOTT;

DCL (Data Control Language)

Lenguaje de control de datos

ROLES

- Un rol es un conjunto de privilegios que pueden ser asignados a un usuario o a otro rol

```
SELECT *  
FROM ROLE_SYS_PRIVS  
ORDER BY ROLE;
```

DCL (Data Control Language)

Lenguaje de control de datos

VER ROLES ASIGNADOS

```
SELECT *  
  FROM DBA_ROLE_PRIVS  
 WHERE GRANTEE = 'HR';
```

GRANTEE	GRANTED_ROLE
HR	RESOURCE
HR	CONNECT

```
SELECT *  
  FROM DBA_SYS_PRIVS  
 WHERE GRANTEE = 'HR';
```

GRANTEE	PRIVILEGE
HR	CREATE VIEW
HR	UNLIMITED TABLESPACE
HR	CREATE DATABASE LINK
HR	DROP PUBLIC SYNONYM
HR	CREATE SEQUENCE
HR	CREATE SESSION
HR	ALTER SESSION
HR	DROP ANY SYNONYM

```
SELECT *  
  FROM ROLE_SYS_PRIVS  
 WHERE ROLE IN ('CONNECT', 'RESOURCE');
```

DCL – EJERCICIO

- En el esquema HR crear el sinónimo público **ALUMNO** que apunte a la tabla ALUMNO

--CREACION DE TABLA

```
CREATE TABLE ALUMNO (  
    LEGAJO                INTEGER,  
    FECHA_DE_INGRESO DATE,  
    NOMBRE                VARCHAR2 (30) ,  
    APELLIDO              VARCHAR2 (30)  
);
```

--CREACION DE SINONIMO PUBLICO

```
CREATE PUBLIC SYNONYM ALUMNO FOR HR.ALUMNO;
```

DCL – EJERCICIO

- En el esquema HR crear el sinónimo público **ALUMNO** que apunte a la tabla ALUMNO

--CREACION DE TABLA

```
CREATE TABLE ALUMNO (  
    LEGAJO                INTEGER,  
    FECHA_DE_INGRESO DATE,  
    NOMBRE                VARCHAR2 (30) ,  
    APELLIDO              VARCHAR2 (30)  
);
```

--CREACION DE SINONIMO PUBLICO

```
CREATE PUBLIC SYNONYM ALUMNO FOR HR.ALUMNO;
```

--ERROR

Error SQL: ORA-01031: insufficient privileges

DCL – EJERCICIO

- En el esquema HR crear el sinónimo público **ALUMNO** que apunte a la tabla ALUMNO

--EN USER SYS

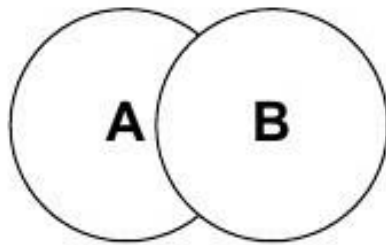
```
GRANT CREATE PUBLIC SYNONYM TO HR;  
REVOKE CREATE PUBLIC SYNONYM FROM HR;
```

--EN USER HR

```
CREATE PUBLIC SYNONYM ALUMNO FOR HR.ALUMNO;  
DROP PUBLIC SYNONYM ALUMNO;
```

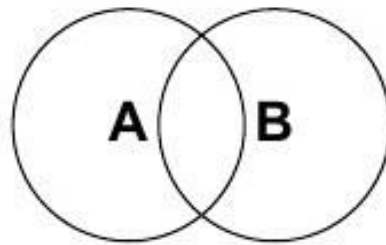
Operaciones de conjunto

Ambos set de datos deben tener el mismo número de campos y el mismo tipo de dato en cada campo.



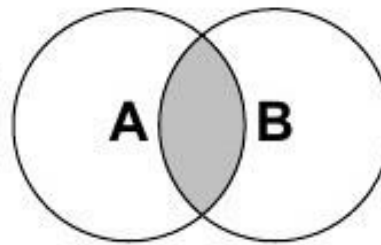
UNION

Una distintas consultas, no muestra los registros duplicados.



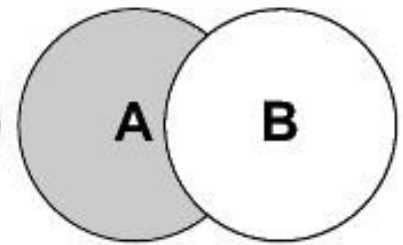
UNION ALL

Una distintas consultas, si muestra los registros duplicados.



INTERSECT

Una distintas consultas, retorna únicamente los registros que son duplicados.



EXCEPT

Una distintas consultas, y solo muestra los registros de A y no los de B.



Operaciones de conjunto

--VISTA 1: EMPLEADOS DEPTO FINANZAS

CREATE OR REPLACE VIEW V_EMPLE1

AS

SELECT EMPLOYEE_ID, FIRST_NAME, LAST_NAME, JOB_ID,
SALARY, DEPARTMENT_ID

FROM EMPLOYEES E

WHERE E.DEPARTMENT_ID = 100;

--6 REGISTROS



EMPLOYEE_ID	FIRST_NAME	LAST_NAME	JOB_ID	SALARY	DEPARTMENT_ID
108	Nancy	Greenberg	FI_MGR	12008	100
109	Daniel	Faviet	FI_ACCOUNT	9000	100
110	John	Chen	FI_ACCOUNT	8200	100
111	Ismael	Sciarra	FI_ACCOUNT	7700	100
112	Jose Manuel	Urman	FI_ACCOUNT	7800	100
113	Luis	Popp	FI_ACCOUNT	6900	100

--VIEW 2: EMPLEADOS CON SALARIO MAYOR A 11000

CREATE OR REPLACE VIEW V_EMPLE2

AS

SELECT EMPLOYEE_ID, FIRST_NAME, LAST_NAME, JOB_ID,
SALARY, DEPARTMENT_ID

FROM EMPLOYEES E

WHERE E.SALARY > 11000;

--10 REGISTROS



EMPLOYEE_ID	FIRST_NAME	LAST_NAME	JOB_ID	SALARY	DEPARTMENT_ID
100	Steven	King	AD_PRES	24000	90
101	Neena	Kochhar	AD_VP	17000	90
102	Lex	De Haan	AD_VP	17000	90
108	Nancy	Greenberg	FI_MGR	12008	100
145	John	Russell	SA_MAN	14000	80
146	Karen	Partners	SA_MAN	13500	80
147	Alberto	Errazuriz	SA_MAN	12000	80
168	Lisa	Ozer	SA_REP	11500	80
201	Michael	Hartstein	MK_MAN	13000	20
205	Shelley	Higgins	AC_MGR	12008	110

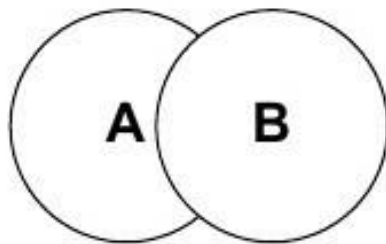
UNION

--UNION

```
SELECT *  
  FROM V_EMPLE1  
UNION  
SELECT *  
  FROM V_EMPLE2 ;
```

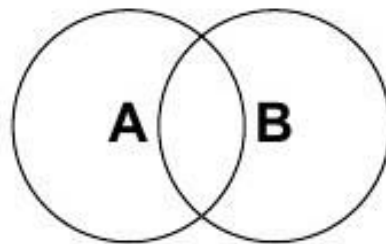
	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	JOB_ID	SALARY	DEPARTMENT_ID
1	100	Steven	King	AD_PRES	24000	90
2	101	Neena	Kochhar	AD_VP	17000	90
3	102	Lex	De Haan	AD_VP	17000	90
4	108	Nancy	Greenberg	FI_MGR	12008	100
5	109	Daniel	Faviet	FI_ACCOUNT	9000	100
6	110	John	Chen	FI_ACCOUNT	8200	100
7	111	Ismael	Sciarra	FI_ACCOUNT	7700	100
8	112	Jose Manuel	Urman	FI_ACCOUNT	7800	100
9	113	Luis	Popp	FI_ACCOUNT	6900	100
10	145	John	Russell	SA_MAN	14000	80
11	146	Karen	Partners	SA_MAN	13500	80
12	147	Alberto	Errazuriz	SA_MAN	12000	80
13	168	Lisa	Ozer	SA_REP	11500	80
14	201	Michael	Hartstein	MK_MAN	13000	20
15	205	Shelley	Higgins	AC_MGR	12008	110

Operaciones de conjunto



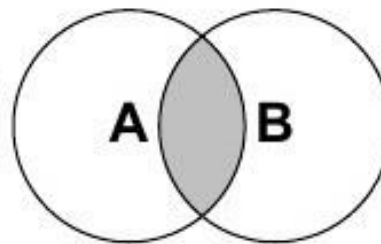
UNION

Una distintas consultas, no muestra los registros duplicados.



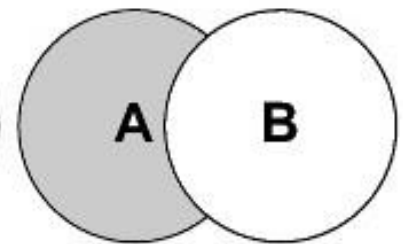
UNION ALL

Una distintas consultas, si muestra los registros duplicados.



INTERSECT

Una distintas consultas, retorna únicamente los registros que son duplicados.



EXCEPT

Una distintas consultas, y solo muestra los registros de A y no los de B.

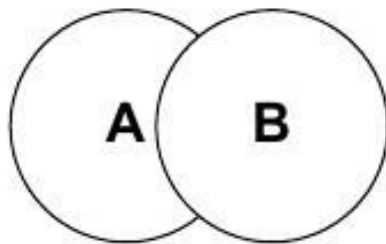


UNION ALL

```
--UNION ALL  
SELECT *  
  FROM V_EMPLE1  
UNION ALL  
SELECT *  
  FROM V_EMPLE2 ;
```

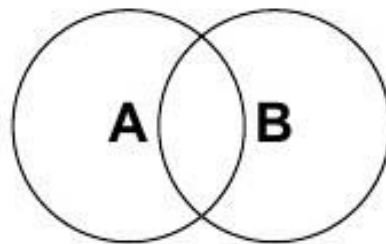
	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	JOB_ID	SALARY	DEPARTMENT_ID
1	108	Nancy	Greenberg	FI_MGR	12008	100
2	109	Daniel	Faviet	FI_ACCOUNT	9000	100
3	110	John	Chen	FI_ACCOUNT	8200	100
4	111	Ismael	Sciarra	FI_ACCOUNT	7700	100
5	112	Jose Manuel	Urman	FI_ACCOUNT	7800	100
6	113	Luis	Popp	FI_ACCOUNT	6900	100
7	100	Steven	King	AD_PRES	24000	90
8	101	Neena	Kochhar	AD_VP	17000	90
9	102	Lex	De Haan	AD_VP	17000	90
10	108	Nancy	Greenberg	FI_MGR	12008	100
11	145	John	Russell	SA_MAN	14000	80
12	146	Karen	Partners	SA_MAN	13500	80
13	147	Alberto	Errazuriz	SA_MAN	12000	80
14	168	Lisa	Ozer	SA_REP	11500	80
15	201	Michael	Hartstein	MK_MAN	13000	20
16	205	Shelley	Higgins	AC_MGR	12008	110

Operaciones de conjunto



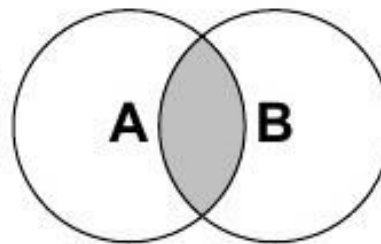
UNION

Una distintas consultas, no muestra los registros duplicados.



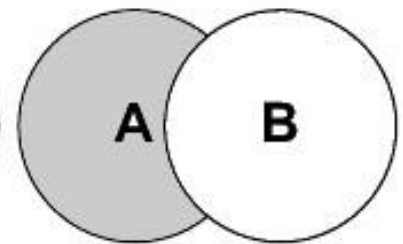
UNION ALL

Una distintas consultas, si muestra los registros duplicados.



INTERSECT

Una distintas consultas, retorna únicamente los registros que son duplicados.



EXCEPT

Una distintas consultas, y solo muestra los registros de A y no los de B.

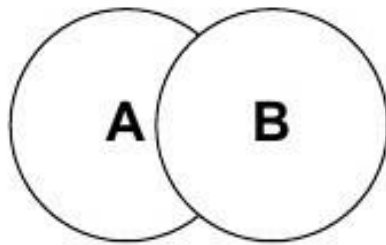


INTERSECT

```
--INTERSECT  
SELECT *  
  FROM V_EMPLE1  
INTERSECT  
SELECT *  
  FROM V_EMPLE2 ;
```

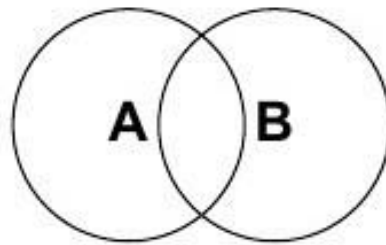
EMPLOYEE_ID	FIRST_NAME	LAST_...	JOB_ID	SALARY	DEPARTMENT_ID
108	Nancy	Greenberg	FI_MGR	12008	100

Operaciones de conjunto



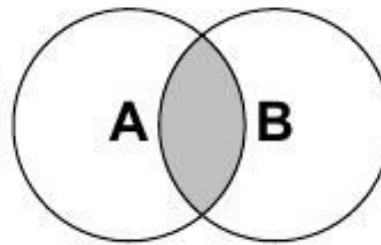
UNION

Una distintas consultas, no muestra los registros duplicados.



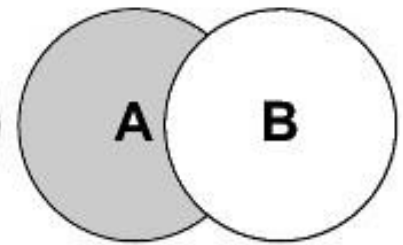
UNION ALL

Una distintas consultas, si muestra los registros duplicados.



INTERSECT

Una distintas consultas, retorna únicamente los registros que son duplicados.



EXCEPT

Una distintas consultas, y solo muestra los registros de A y no los de B.



EXCEPT (MINUS)

```
--MINUS (EXCEPT)
SELECT *
  FROM V_EMPLE1
MINUS
SELECT *
  FROM V_EMPLE2 ;
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	JOB_ID	SALARY	DEPARTMENT_ID
109	Daniel	Faviet	FI_ACCOUNT	9000	100
110	John	Chen	FI_ACCOUNT	8200	100
111	Ismael	Sciarra	FI_ACCOUNT	7700	100
112	Jose Manuel	Urman	FI_ACCOUNT	7800	100
113	Luis	Popp	FI_ACCOUNT	6900	100

EJEMPLO UNION

```
--EMPLEADOS + PASANTES  
SELECT * FROM EMPLOYEES  
UNION  
SELECT * FROM PASANTES;
```