

{desafío}
latam_

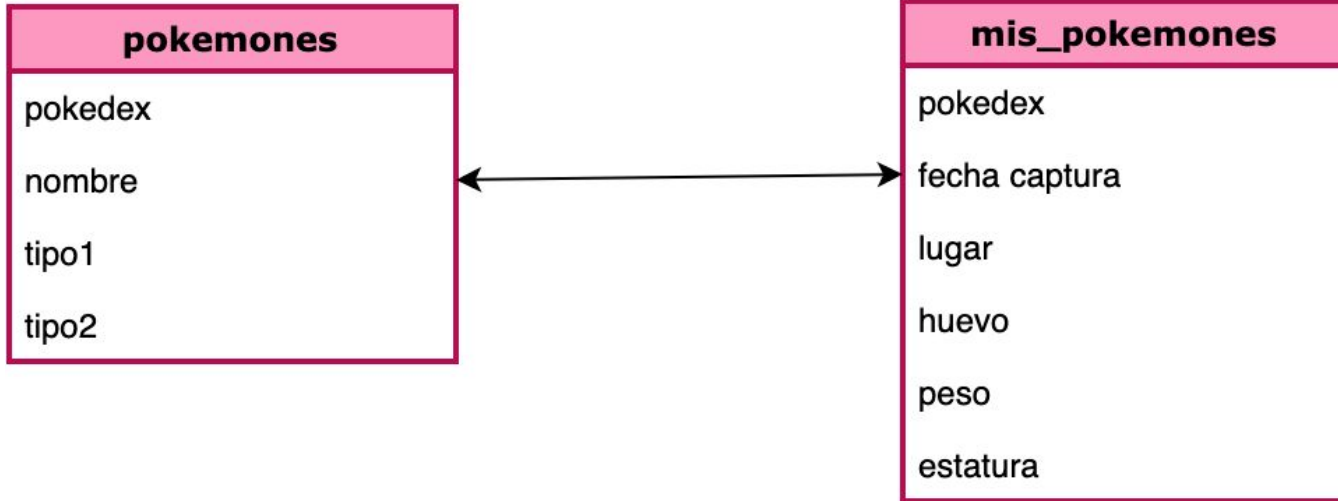
Gestión de bases de datos _

Parte II



Realizando consultas

Ejercicio Pokemones



Mostrar registros de una tabla

```
SELECT *  
FROM MIS_POKEMONES;
```

Salida de Script x Resultado de la Consulta x

SQL | Se han recuperado 50 filas en 0,011 segundos

	POKEDEX	FECHA_CAPTURA	LUGAR	HUEVO	PESO	ESTATURA
1	121	09/07/19	Recoleta	false	879	11
2	121	29/03/19	Providencia	false	798	10
3	121	09/07/19	Providencia	false	801	10
4	121	31/03/19	Vitacura	false	825	12
5	123	31/03/19	Providencia	false	558	15
6	123	16/04/19	Providencia	false	520	14
7	123	06/07/19	Vitacura	false	533	12
8	123	06/07/19	Quinta Normal	false	519	15
9	125	04/01/19	Vitacura	false	451	15

Usando SELECT

Consultar por una columna

```
SELECT columna FROM nombre_tabla;
```

Consultar por más de una columna

```
SELECT columna1, columna2 FROM nombre_tabla;
```

Consultar por todas las columnas de una tabla

```
SELECT * FROM nombre_tabla;
```

Operaciones de unión entre Tablas

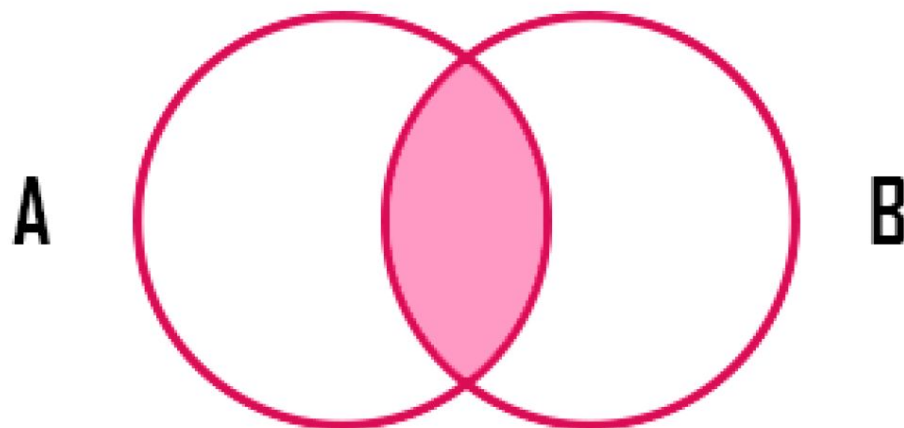
Caso hipotético

```
-- Seleccionamos las columnas desde la tabla1
SELECT columnas FROM tabla1
-- Posterior a la selección de la columna, indicamos que vamos a generar la unión
-- con la columna de la tabla2
JOIN tabla2 ON tabla1.columna=tabla2.columna
[WHERE condicion];
```

Tipos de unión

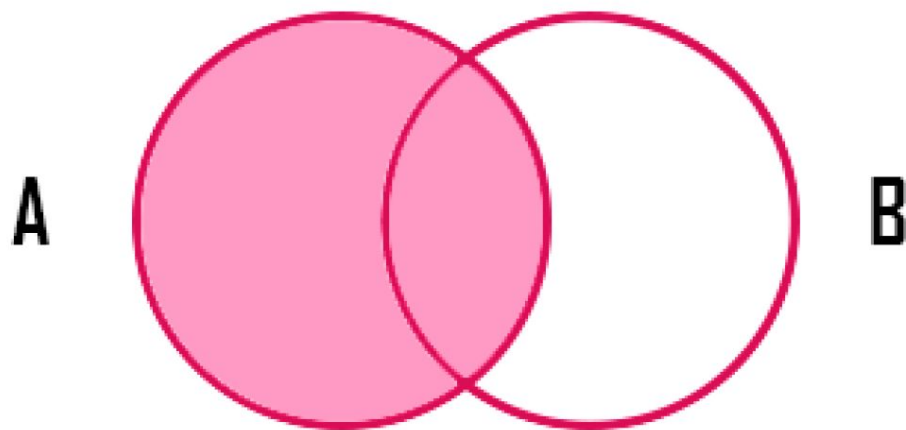
- INNER JOIN
- LEFT JOIN
- FULL OUTER JOIN

INNER JOIN



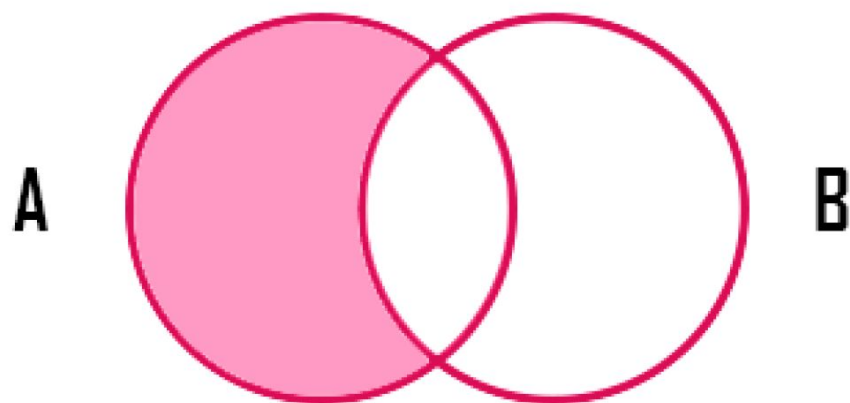
```
SELECT columnas  
FROM A  
INNER JOIN B  
ON A.columna=B.columna
```

LEFT JOIN



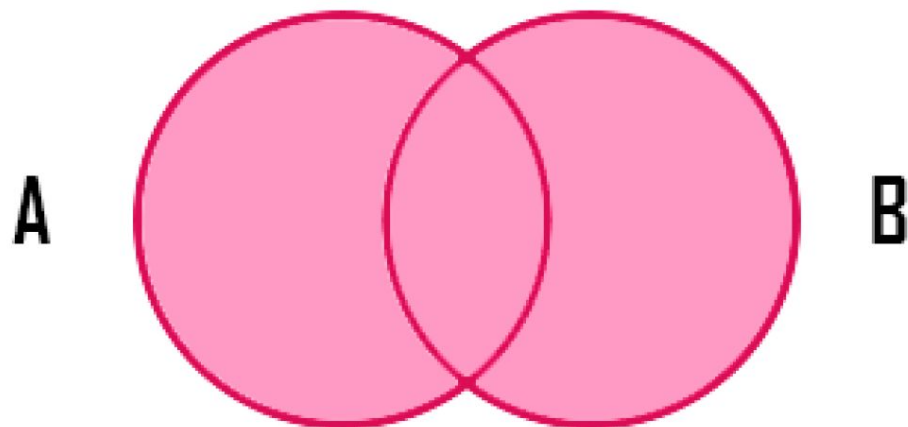
```
SELECT columnas  
FROM A  
LEFT JOIN B  
ON A.columna=B.columna
```

LEFT JOIN



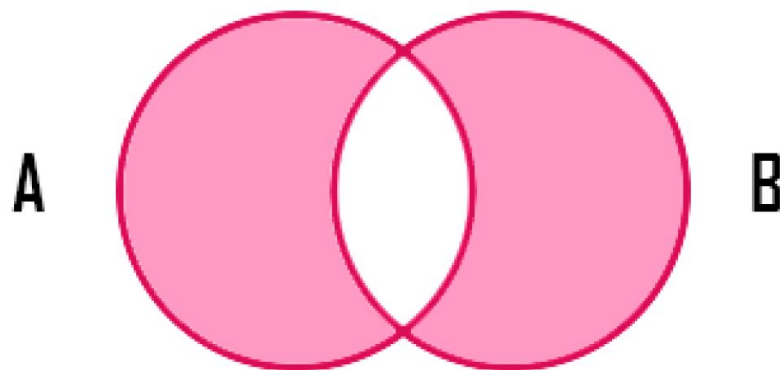
```
SELECT columnas  
FROM A  
LEFT JOIN B  
ON A.columna=B.columna  
WHERE B.columna IS NULL
```

FULL JOIN



```
SELECT columnas  
FROM A  
FULL OUTER JOIN B  
ON A.columna=B.columna
```

FULL JOIN



```
SELECT columnas  
FROM A  
FULL OUTER JOIN B  
ON A.columna=B.columna  
WHERE A.columna IS NULL  
OR B.columna IS NULL
```

Subquery - Operador WHERE

```
SELECT columna1,columna2,...  
FROM nombre_tabla  
WHERE columna1 IN  
    (SELECT columna1  
     FROM nombre_tabla2  
     WHERE condicion);
```

Subquery - Operador FROM

```
SELECT x.columna1, x.columna2, ...  
FROM (  
    SELECT columna1, columna2, ...  
    FROM nombre_tabla2  
    WHERE condicion  
    ) as x  
INNER JOIN nombre_tabla1 as y on x.columna1 = y.columna1;
```

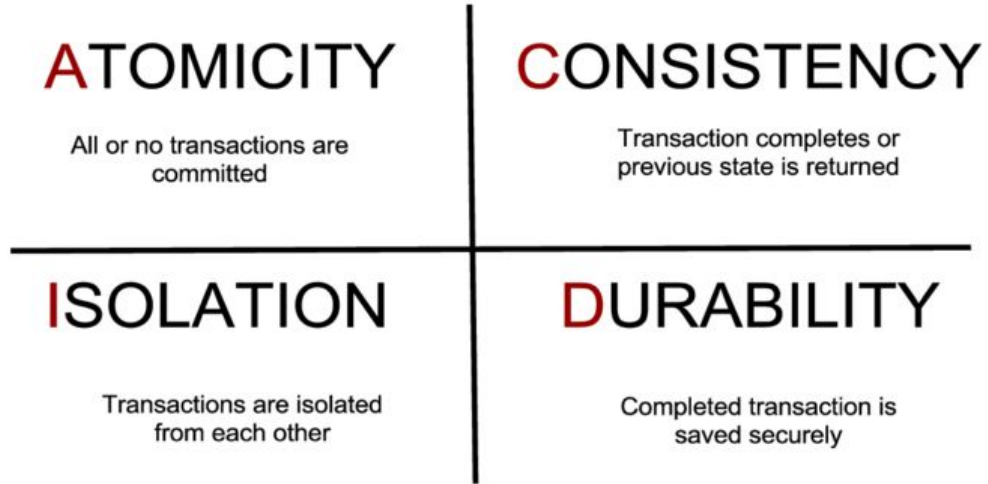
Algunas reglas que deben seguir las subqueries

- Las consultas internas deben estar encapsuladas entre paréntesis.
- Una subquery puede tener sólo una columna especificada en SELECT , con la excepción de múltiples columnas definidas en la consulta principal.
- El comando ORDER BY no se puede utilizar en una consulta interna. La excepción es que esta instrucción si puede ser incluída en la consulta principal.
- Para obtener un resultado similar a ORDER BY dentro de una consulta interna, se puede implementar el comando GROUP BY .
- Aquellas consultas internas que retornen más de una fila sólo pueden ser utilizadas con operadores de múltiples valores como IN .

Transacciones

Transacciones

- **A**tomicidad
- **C**onsistencia
- **I**solamiento
- **D**urabilidad



Comandos para las transacciones

- BEGIN: El sistema permite que se ejecuten todas las sentencias SQL que necesitamos.
- COMMIT: Guarda los cambios de la transacción.
- ROLLBACK: Retrocede los cambios realizados.
- SAVEPOINT: Guarda el punto de partida al cual volver a la hora de aplicar ROLLBACK
- SET TRANSACTION:: Le asigna nombre a la transacción.

Respaldo una base de datos usando SQL Developer

Respaldo de una tabla

Asistente de Exportación: Paso 1 de 3

Origen/Destino

Conexión: HR

☒ Exportar DDL

☒ Impresión con Sangría ☒ Mostrar Esquema ☐ Permisos ☒ Agregar Palabra Clave BYTE

☒ Terminador ☒ Forzar Vistas ☐ Borrados ☐ Borrados en Cascada

☒ Almacenamiento ☐ Dependientes

Versión: COMPATIBLE

☒ Exportar Datos

Formato: insert ☒ Mostrar Esquema

Terminador de Línea: valor por defecto de entorno

☐ Incluir Confirmación Cada 100 Filas

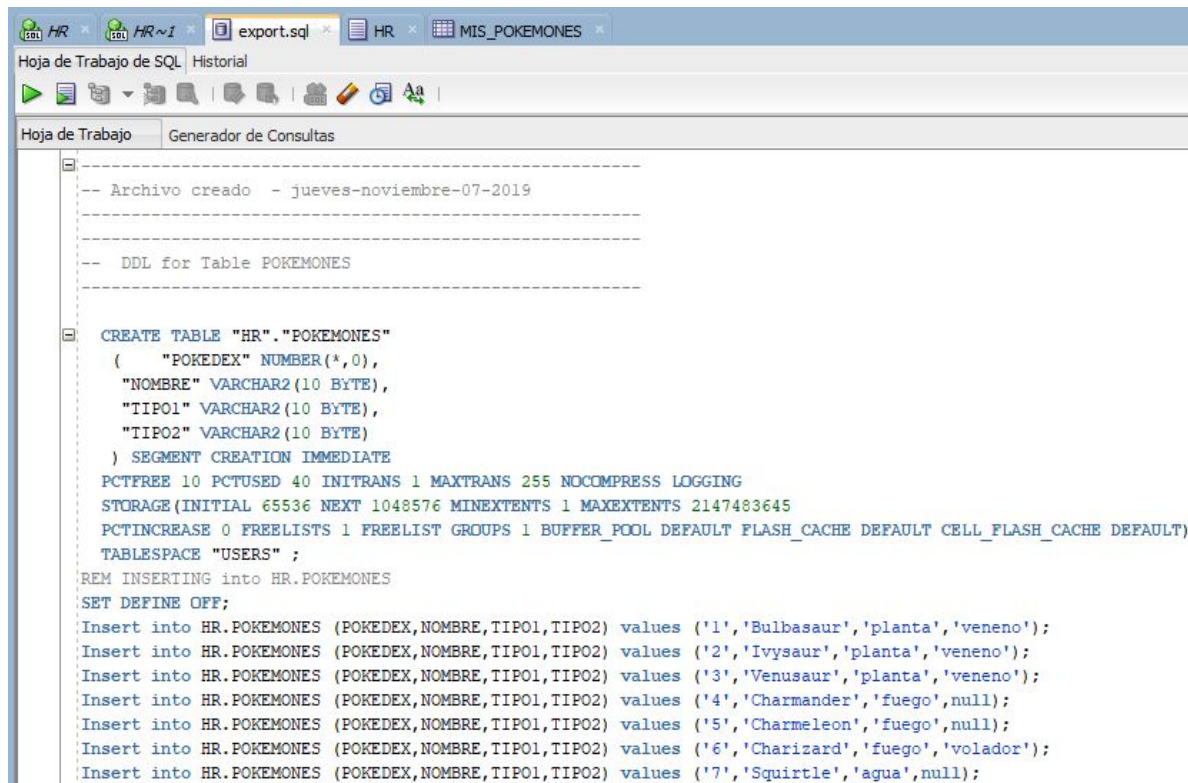
Guardar como: Archivo Único ☐ Comprimido Codificación: Cp1252

Archivo: C:\Users\ygtha\export.sql

☐ Ir al Resumen.

Restaurar una base de datos

Se ejecuta el script
generado



```
-- Archivo creado - jueves-noviembre-07-2019

-- DDL for Table POKEMONES

CREATE TABLE "HR"."POKEMONES"
(
  "POKEDEX" NUMBER(*,0),
  "NOMBRE" VARCHAR2(10 BYTE),
  "TIPO1" VARCHAR2(10 BYTE),
  "TIPO2" VARCHAR2(10 BYTE)
) SEGMENT CREATION IMMEDIATE
PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "USERS" ;

REM INSERTING into HR.POKEMONES
SET DEFINE OFF;
Insert into HR.POKEMONES (POKEDEX,NOMBRE,TIPO1,TIPO2) values ('1','Bulbasaur','planta','veneno');
Insert into HR.POKEMONES (POKEDEX,NOMBRE,TIPO1,TIPO2) values ('2','Ivysaur','planta','veneno');
Insert into HR.POKEMONES (POKEDEX,NOMBRE,TIPO1,TIPO2) values ('3','Venusaur','planta','veneno');
Insert into HR.POKEMONES (POKEDEX,NOMBRE,TIPO1,TIPO2) values ('4','Charmander','fuego',null);
Insert into HR.POKEMONES (POKEDEX,NOMBRE,TIPO1,TIPO2) values ('5','Charmeleon','fuego',null);
Insert into HR.POKEMONES (POKEDEX,NOMBRE,TIPO1,TIPO2) values ('6','Charizard','fuego','volador');
Insert into HR.POKEMONES (POKEDEX,NOMBRE,TIPO1,TIPO2) values ('7','Squirtle','agua',null);
```

{desafío}
latam_

*Academia de
talentos digitales*

www.desafiolatam.com