



# **Gestión de Objetos de Esquema**

# Objetivos

Al finalizar esta lección, debería estar capacitado para lo siguiente:

- Agregar restricciones
- Crear índices
- Crear índices mediante la sentencia `CREATE TABLE`
- Crear índices basados en funciones
- Borrar columnas y definir columnas como `UNUSED`
- Realizar operaciones `FLASHBACK`
- Crear y utilizar tablas externas

# Agenda

- Uso de la sentencia `ALTER TABLE` para agregar, modificar y borrar una columna
- Gestión de restricciones:
  - Adición y borrado de una restricción
  - Restricciones diferidas
  - Activación y desactivación de una restricción
- Creación de índices:
  - Uso de la sentencia `CREATE TABLE`
  - Creación de índices basados en funciones
  - Eliminación de índices
- Realización de operaciones de flashback
- Creación y uso de tablas temporales
- Creación y uso de tablas externas

# Sentencia ALTER TABLE

Utilizar la sentencia ALTER TABLE para:

- Agregar una nueva columna
- Modificar una columna existente
- Definir un valor por defecto para la nueva columna
- Borrar una columna

# Sentencia ALTER TABLE

Utilizar la sentencia ALTER TABLE para agregar, modificar y borrar columnas:

```
ALTER TABLE table  
ADD          (column datatype [DEFAULT expr]  
              [, column datatype]...);
```

```
ALTER TABLE table  
MODIFY       (column datatype [DEFAULT expr]  
              [, column datatype]...);
```

```
ALTER TABLE table  
DROP (column [, column] ...);
```

# Adición de Columnas

- Puede utilizar la cláusula `ADD` para agregar columnas:

```
ALTER TABLE dept80  
ADD      (job_id VARCHAR2(9)) ;
```

```
ALTER TABLE dept80 succeeded.
```

- La nueva columna se convierte en la última:

	EMPLOYEE_ID	LAST_NAME	ANNSAL	HIRE_DATE	JOB_ID
1	145	Russell	14000	01-OCT-96	(null)
2	146	Partners	13500	05-JAN-97	(null)
3	147	Errazuriz	12000	10-MAR-97	(null)
4	148	Cambrault	11000	15-OCT-99	(null)
5	149	Zlotkey	10500	29-JAN-00	(null)

# Modificación de Columnas

- Puede cambiar el tipo de dato, tamaño y valor por defecto de una columna.

```
ALTER TABLE dept80  
MODIFY      (last_name VARCHAR2(30)) ;
```

```
ALTER TABLE dept80 succeeded.
```

- El cambio del valor por defecto sólo afecta a las inserciones posteriores en la tabla.

# Borrado de Columnas

Utilizar la cláusula `DROP COLUMN` para borrar columnas que ya no son necesarias de la tabla:

```
ALTER TABLE dept80  
DROP COLUMN job_id;
```

```
ALTER TABLE dept80 succeeded.
```

	EMPLOYEE_ID	LAST_NAME	ANNSAL	HIRE_DATE
1	145	Russell	14000	01-OCT-96
2	146	Partners	13500	05-JAN-97
3	147	Errazuriz	12000	10-MAR-97
4	148	Cambrault	11000	15-OCT-99
5	149	Zlotkey	10500	29-JAN-00



# Opción SET UNUSED

- Puede utilizar la opción SET UNUSED para marcar una o más columnas como no utilizadas.
- Puede utilizar la opción DROP UNUSED COLUMNS para eliminar las columnas marcadas como no utilizadas.

```
ALTER TABLE <table_name>  
SET UNUSED(<column_name> [ , <column_name>]) ;
```

○

```
ALTER TABLE <table_name>  
SET UNUSED COLUMN <column_name> [ , <column_name>];
```

```
ALTER TABLE <table_name>  
DROP UNUSED COLUMNS;
```

# Agenda

- Uso de la sentencia `ALTER TABLE` para agregar, modificar y borrar una columna
- Gestión de restricciones:
  - Adición y borrado de una restricción
  - Restricciones diferidas
  - Activación y desactivación de una restricción
- Creación de índices:
  - Uso de la sentencia `CREATE TABLE`
  - Creación de índices basados en funciones
  - Eliminación de índices
- Realización de operaciones de flashback
- Creación y uso de tablas temporales
- Creación y uso de tablas externas

# Adición de Sintaxis de Restricción

Utilizar la sentencia `ALTER TABLE` para:

- Agregar o borrar una restricción, pero sin modificar su estructura
- Activar o desactivar restricciones
- Agregar una restricción `NOT NULL` mediante la cláusula `MODIFY`

```
ALTER TABLE  <table_name>
ADD [CONSTRAINT <constraint_name>]
type (<column_name>) ;
```

# Adición de Restricciones

Agregar una restricción FOREIGN KEY a la tabla EMP2 que indique que un gestor ya debe existir como empleado válido en la tabla EMP2.

```
ALTER TABLE emp2  
MODIFY employee_id PRIMARY KEY;
```

```
ALTER TABLE emp2 succeeded.
```

```
ALTER TABLE emp2  
ADD CONSTRAINT emp_mgr_fk  
FOREIGN KEY(manager_id)  
REFERENCES emp2(employee_id);
```

```
ALTER TABLE succeeded.
```

# Cláusula ON DELETE

- Utilizar la cláusula `ON DELETE CASCADE` para suprimir filas secundarias al suprimir una clave principal:

```
ALTER TABLE emp2 ADD CONSTRAINT emp_dt_fk  
FOREIGN KEY (Department_id)  
REFERENCES departments(department_id) ON DELETE CASCADE;
```

```
ALTER TABLE Emp2 succeeded.
```

- Utilizar la cláusula `ON DELETE SET NULL` para definir el valor de filas secundarias en nulo al suprimir una clave principal:

```
ALTER TABLE emp2 ADD CONSTRAINT emp_dt_fk  
FOREIGN KEY (Department_id)  
REFERENCES departments(department_id) ON DELETE SET NULL;
```

```
ALTER TABLE Emp2 succeeded.
```

# Restricciones Diferidas

Las restricciones pueden tener los siguientes atributos:

- DEFERRABLE **O** NOT DEFERRABLE
- INITIALLY DEFERRED **O** INITIALLY IMMEDIATE

```
ALTER TABLE dept2  
ADD CONSTRAINT dept2_id_pk  
PRIMARY KEY (department_id)  
DEFERRABLE INITIALLY DEFERRED
```

Restricción diferida en la creación

```
SET CONSTRAINTS dept2_id_pk IMMEDIATE
```

Cambio de un atributo de restricción específico

```
ALTER SESSION  
SET CONSTRAINTS= IMMEDIATE
```

Cambio de todas las restricciones para una sesión

# Diferencia entre INITIALLY DEFERRED y INITIALLY IMMEDIATE

INITIALLY DEFERRED	Espera a comprobar la restricción hasta que termina la transacción
INITIALLY IMMEDIATE	Comprueba la restricción al final de la ejecución de la sentencia

```
CREATE TABLE emp_new_sal (salary NUMBER
    CONSTRAINT sal_ck
    CHECK (salary > 100)
    DEFERRABLE INITIALLY IMMEDIATE,
    bonus NUMBER
    CONSTRAINT bonus_ck
    CHECK (bonus > 0 )
    DEFERRABLE INITIALLY DEFERRED );
```

```
create table succeeded.
```

# Borrado de Restricciones

- Eliminar la restricción de gestor de la tabla EMP2:

```
ALTER TABLE emp2  
DROP CONSTRAINT emp_mgr_fk;
```

```
ALTER TABLE Emp2 succeeded.
```

- Eliminar la restricción PRIMARY KEY en la tabla DEPT2 y borrar la restricción FOREIGN KEY asociada en la columna EMP2.DEPARTMENT\_ID:

```
ALTER TABLE dept2  
DROP PRIMARY KEY CASCADE;
```

```
ALTER TABLE dept2 succeeded.
```



# Desactivación de Restricciones

- Ejecutar la cláusula `DISABLE` de la sentencia `ALTER TABLE` para desactivar una restricción de integridad.
- Aplicar la opción `CASCADE` para desactivar las restricciones de integridad dependientes.

```
ALTER TABLE emp2  
DISABLE CONSTRAINT emp_dt_fk;
```

```
ALTER TABLE Emp2 succeeded.
```

# Activación de Restricciones

- Activar una restricción de integridad desactivada actualmente en la definición de tabla mediante la cláusula

```
ALTER TABLE      emp2  
ENABLE CONSTRAINT emp_dt_fk;
```

```
ALTER TABLE Emp2 succeeded.
```

- Se crea un índice `UNIQUE` automáticamente si activa una clave `UNIQUE` o una restricción `PRIMARY KEY`.

# Restricciones en Cascada

- La cláusula `CASCADE CONSTRAINTS` se utiliza junto con la cláusula `DROP COLUMN`.
- La cláusula `CASCADE CONSTRAINTS` borra todas las restricciones de integridad referencial que hacen referencia a las claves `PRIMARY` y `UNIQUE` definidas en las columnas borradas.
- La cláusula `CASCADE CONSTRAINTS` borra también todas las restricciones de varias columnas definidas en las columnas borradas.

# Restricciones en Cascada

Ejemplo:

```
ALTER TABLE emp2  
DROP COLUMN employee_id CASCADE CONSTRAINTS;
```

```
ALTER TABLE Emp2 succeeded.
```

```
ALTER TABLE test1  
DROP (col1_pk, col2_fk, col1) CASCADE CONSTRAINTS;
```

```
ALTER TABLE test1 succeeded.
```

# Cambio de Nombre de Columnas de Tabla y Restricciones

Utilizar la cláusula `RENAME COLUMN` de la sentencia `ALTER TABLE` para cambiar el nombre de las columnas de la tabla.

**a**

```
ALTER TABLE marketing RENAME COLUMN team_id  
TO id;
```

```
ALTER TABLE marketing succeeded.
```

Utilizar la cláusula `RENAME CONSTRAINT` de la sentencia `ALTER TABLE` para cambiar el nombre de cualquier restricción existente para la tabla.

**b**

```
ALTER TABLE marketing RENAME CONSTRAINT mktg_pk  
TO new_mktg_pk;
```

```
ALTER TABLE marketing succeeded.
```

# Agenda

- Uso de la sentencia `ALTER TABLE` para agregar, modificar y borrar una columna
- Gestión de restricciones:
  - Adición y borrado de una restricción
  - Restricciones diferidas
  - Activación y desactivación de una restricción
- Creación de índices:
  - Uso de la sentencia `CREATE TABLE`
  - Creación de índices basados en funciones
  - Eliminación de índices
- Realización de operaciones de flashback
- Creación y uso de tablas temporales
- Creación y uso de tablas externas

# Visión General de Índices

Los índices se crean:

- Automáticamente
  - Creación de `PRIMARY KEY`
  - Creación de `UNIQUE KEY`
- Manualmente
  - Sentencia `CREATE INDEX`
  - Sentencia `CREATE TABLE`

# CREATE INDEX con la Sentencia CREATE TABLE

```
CREATE TABLE NEW_EMP  
(employee_id NUMBER(6)  
    PRIMARY KEY USING INDEX  
    (CREATE INDEX emp_id_idx ON  
    NEW_EMP(employee_id)),  
first_name VARCHAR2(20),  
last_name VARCHAR2(25));
```

CREATE TABLE succeeded.

```
SELECT INDEX_NAME, TABLE_NAME  
FROM USER_INDEXES  
WHERE TABLE_NAME = 'NEW_EMP';
```

INDEX_NAME	TABLE_NAME
1 EMP_ID_IDX	NEW_EMP



# Índices Basados en Funciones

- Un índice basado en funciones se basa en expresiones.
- La expresión de índice se genera a partir de las columnas de las tablas, restricciones, funciones SQL y funciones definidas por el usuario.

```
CREATE INDEX upper_dept_name_idx  
ON dept2 (UPPER(department_name)) ;
```

```
CREATE INDEX succeeded.
```

```
SELECT *  
FROM   dept2  
WHERE  UPPER(department_name) = 'SALES' ;
```

# Eliminación de Índices

- Eliminar un índice del diccionario de datos mediante el comando `DROP INDEX`:

```
DROP INDEX index;
```

- Eliminar el índice `UPPER_DEPT_NAME_IDX` del diccionario de datos:

```
DROP INDEX upper_dept_name_idx;
```

```
DROP INDEX upper_dept_name_idx succeeded.
```

- Para borrar un índice, debe ser el propietario del mismo o tener el privilegio `DROP ANY INDEX`.

# DROP TABLE ... PURGE

```
DROP TABLE dept80 PURGE;
```

```
DROP TABLE dept80 succeeded.
```

# Agenda

- Uso de la sentencia `ALTER TABLE` para agregar, modificar y borrar una columna
- Gestión de restricciones:
  - Adición y borrado de una restricción
  - Restricciones diferidas
  - Activación y desactivación de una restricción
- Creación de índices:
  - Uso de la sentencia `CREATE TABLE`
  - Creación de índices basados en funciones
  - Eliminación de índices
- Realización de operaciones de flashback
- Creación y uso de tablas temporales
- Creación y uso de tablas externas

# Sentencia FLASHBACK TABLE

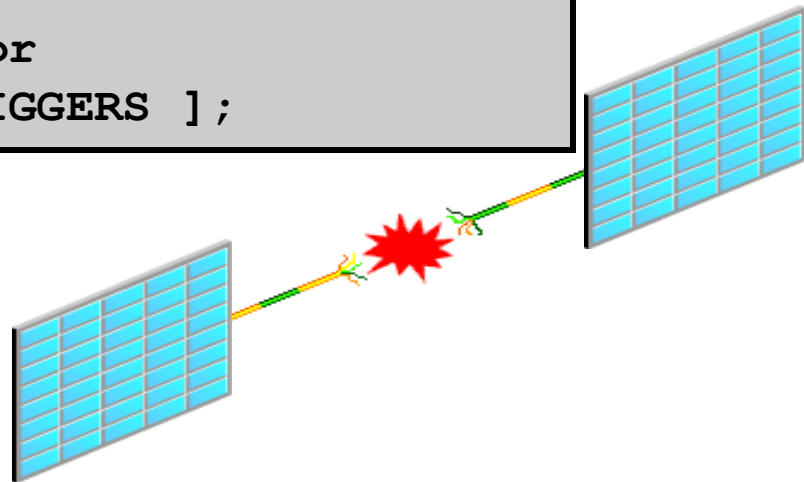
- Permite recuperar tablas a un punto especificado en el tiempo con una única sentencia.
- Restaura los datos de la tabla junto con los índices y las restricciones.
- Permite revertir la tabla y su contenido a un determinado punto en el tiempo o a un número de cambio del sistema (SCN)



# Sentencia FLASHBACK TABLE

- Herramienta de reparación de modificaciones accidentales de tabla
  - Restaura una tabla a un punto anterior en el tiempo.
  - Ventajas: facilidad de uso, disponibilidad y rápida ejecución.
  - Se ejecuta in-situ.
- Sintaxis:

```
FLASHBACK TABLE[schema.]table[,  
[ schema.]table ]...  
TO { TIMESTAMP | SCN } expr  
[ { ENABLE | DISABLE } TRIGGERS ];
```



# Uso de la Sentencia FLASHBACK TABLE

```
DROP TABLE emp2;
```

```
DROP TABLE emp2 succeeded.
```

```
SELECT original_name, operation, droptime FROM  
recyclebin;
```

ORIGINAL_NAME	OPERATION	DROPTIME
EMP2	DROP	2009-05-20:18:00:39

...

```
FLASHBACK TABLE emp2 TO BEFORE DROP;
```

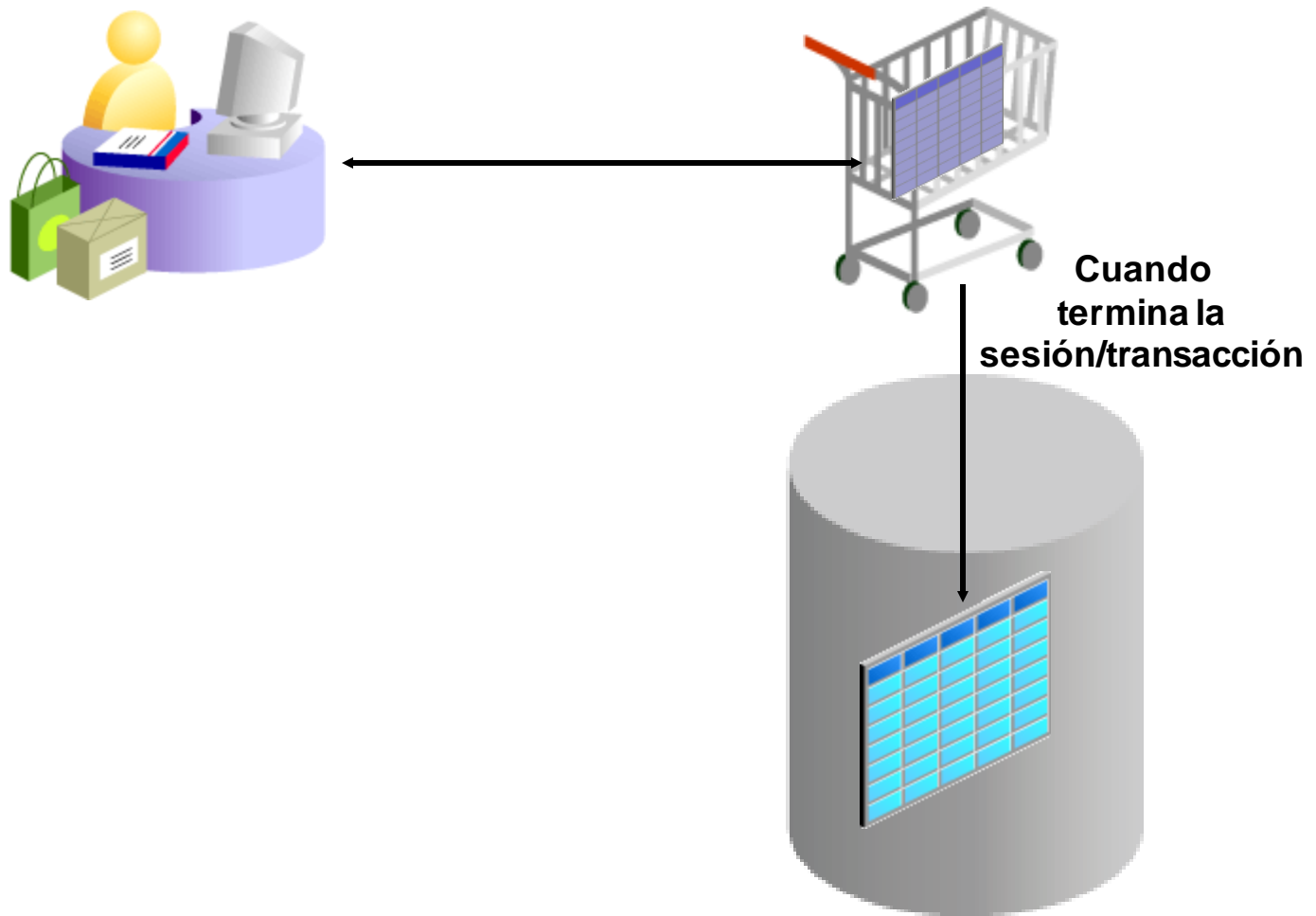
```
FLASHBACK TABLE succeeded.
```

# Agenda

- Uso de la sentencia `ALTER TABLE` para agregar, modificar y borrar una columna
- Gestión de restricciones:
  - Adición y borrado de una restricción
  - Restricciones diferidas
  - Activación y desactivación de una restricción
- Creación de índices:
  - Uso de la sentencia `CREATE TABLE`
  - Creación de índices basados en funciones
  - Eliminación de índices
- Realización de operaciones de flashback
- **Creación y uso de tablas temporales**
- Creación y uso de tablas externas



# Tablas Temporales



# Creación de una Tabla Temporal

```
CREATE GLOBAL TEMPORARY TABLE cart  
ON COMMIT DELETE ROWS;
```

1

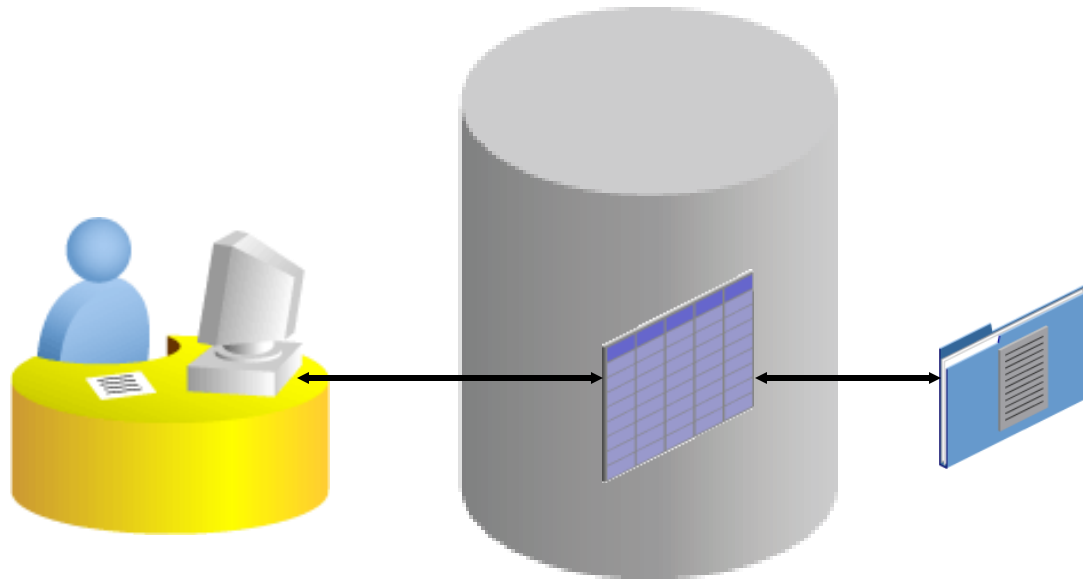
```
CREATE GLOBAL TEMPORARY TABLE today_sales  
ON COMMIT PRESERVE ROWS AS  
    SELECT * FROM orders  
        WHERE order_date = SYSDATE;
```

2

# Agenda

- Uso de la sentencia `ALTER TABLE` para agregar, modificar y borrar una columna
- Gestión de restricciones:
  - Adición y borrado de una restricción
  - Restricciones diferidas
  - Activación y desactivación de una restricción
- Creación de índices:
  - Uso de la sentencia `CREATE TABLE`
  - Creación de índices basados en funciones
  - Eliminación de índices
- Realización de operaciones de flashback
- Creación y uso de tablas temporales
- Creación y uso de tablas externas

# Tablas Externas



# Creación de un Directorio para la Tabla Externa

Crear un objeto `DIRECTORY` que corresponda al directorio en el sistema de archivos en el que reside el origen de datos externo.

```
CREATE OR REPLACE DIRECTORY emp_dir  
AS '/.../emp_dir';  
  
GRANT READ ON DIRECTORY emp_dir TO ora_21;
```

# Creación de Tablas Externas

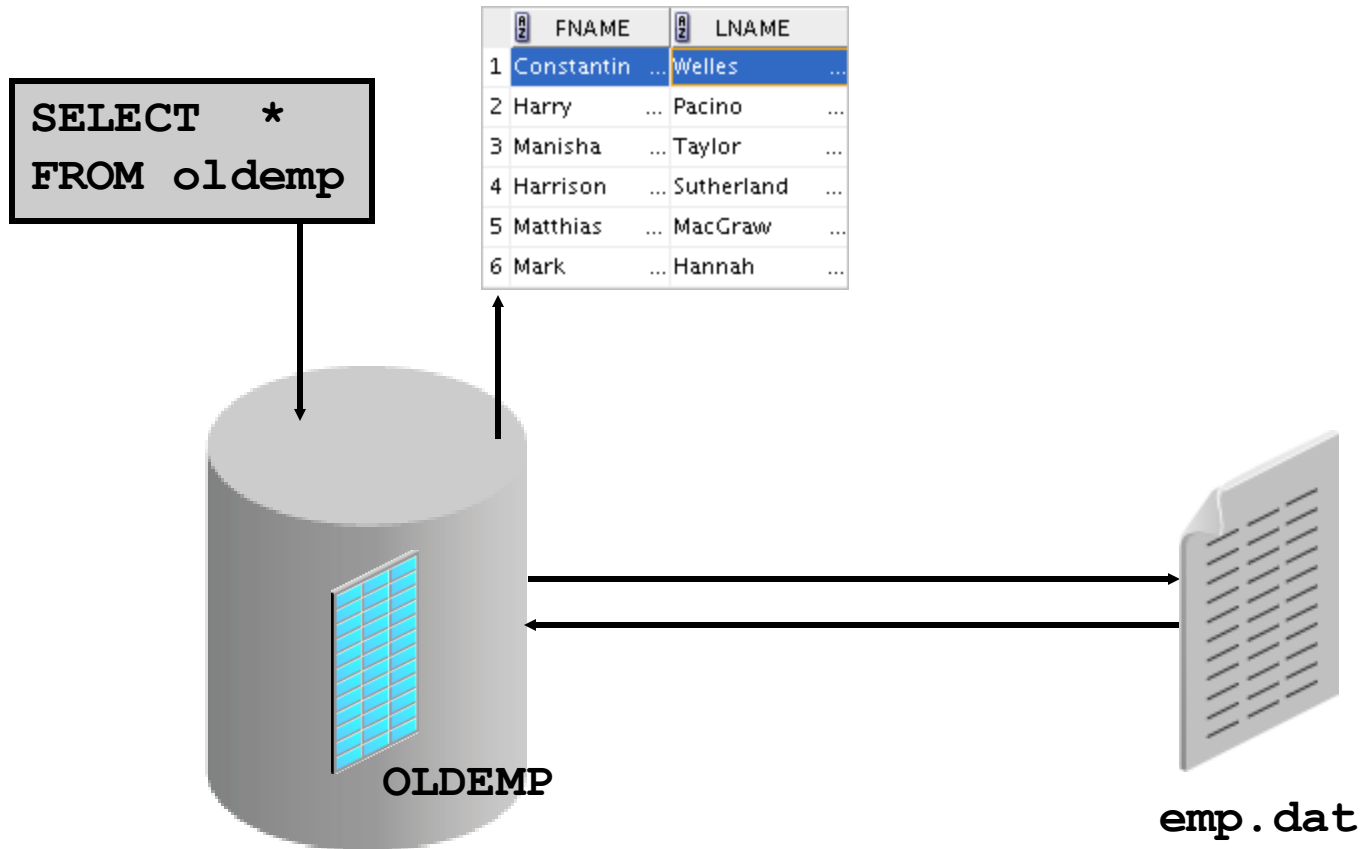
```
CREATE TABLE <table_name>
  ( <col_name> <datatype>, ... )
ORGANIZATION EXTERNAL
  (TYPE <access_driver_type>
    DEFAULT DIRECTORY <directory_name>
    ACCESS PARAMETERS
      (... ) )
  LOCATION ('<location_specifier>')
REJECT LIMIT [0 | <number> | UNLIMITED];
```

# Creación de una Tabla Externa mediante ORACLE\_LOADER

```
CREATE TABLE oldemp (  
  fname char(25), lname CHAR(25))  
  ORGANIZATION EXTERNAL  
  (TYPE ORACLE_LOADER  
  DEFAULT DIRECTORY emp_dir  
  ACCESS PARAMETERS  
  (RECORDS DELIMITED BY NEWLINE  
  NOBADFILE  
  NOLOGFILE  
  FIELDS TERMINATED BY ', '  
  (fname POSITION ( 1:20) CHAR,  
  lname POSITION (22:41) CHAR))  
  LOCATION ('emp.dat'))  
  PARALLEL 5  
  REJECT LIMIT 200;
```

CREATE TABLE succeeded.

# Consulta de Tablas Externas





# Creación de una Tabla Externa mediante ORACLE\_DATAPUMP: Ejemplo

```
CREATE TABLE emp_ext  
  (employee_id, first_name, last_name)  
  ORGANIZATION EXTERNAL  
  (  
    TYPE ORACLE_DATAPUMP  
    DEFAULT DIRECTORY emp_dir  
    LOCATION  
      ('emp1.exp', 'emp2.exp')  
  )  
  PARALLEL  
AS  
SELECT employee_id, first_name, last_name  
FROM   employees;
```

# Prueba

Una restricción `FOREIGN KEY` fuerza la siguiente acción:

Al suprimir datos de la clave principal, todas las filas de la tabla secundaria que dependen de los valores de clave principal se suprimen también.

1. Verdadero
2. Falso

# Prueba

En todos los casos, al ejecutar el comando `DROP TABLE`, la base de datos cambia el nombre de la tabla y la coloca en la papelera de reciclaje, de donde se puede recuperar posteriormente mediante la sentencia `FLASHBACK TABLE`.

1. Verdadero
2. Falso

# Resumen

En esta lección, debe haber aprendido lo siguiente:

- Agregar Restricciones
- Crear índices
- Crear índices mediante la sentencia `CREATE TABLE`
- Crear índices basados en funciones
- Borrar columnas y definir columnas como `UNUSED`
- Realizar operaciones `FLASHBACK`
- Crear y utilizar tablas externas

# Práctica 2: Visión General

En esta práctica se abordan los siguientes temas:

- Modificación de tablas
- Adición de columnas
- Borrado de columnas
- Creación de índices
- Creación de tablas externas