ORACLE

15

Managing Schema Objects

Lesson Agenda

- Managing constraints
- Creating and using temporary tables
- Creating and using external tables

Adding a Constraint Syntax

Use the ALTER TABLE statement to:

- Add or drop a constraint, but not to modify its structure
- Enable or disable constraints
- Add a NOT NULL constraint by using the MODIFY clause

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

3

Adding a Constraint

Add a FOREIGN KEY constraint to the EMP2 table indicating that a manager must already exist as a valid employee in the EMP2 table.

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

```
Table EMP2 altered.
```

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

```
Table EMP2 altered.
```

4

Dropping a Constraint

- The drop_constraint_clause enables you to drop an integrity constraint from a database.
- Remove the manager constraint from the EMP2 table:

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

```
Table EMP2 altered.
```

- Remove the PRIMARY KEY constraint on the EMP2 table and drop the associated FOREIGN KEY constraint on the EMP2.MANAGER_ID column:

```
ALTER TABLE emp2
DROP PRIMARY KEY CASCADE;
```

```
Table EMP2 altered.
```


ON DELETE Clause

- Use the ON DELETE CASCADE clause to delete child rows when a parent key is deleted:

```
ALTER TABLE dept2 ADD CONSTRAINT dept_lc_fk
FOREIGN KEY (location_id)
REFERENCES locations(location_id) ON DELETE CASCADE;
```

```
Table DEPT2 altered.
```

- Use the ON DELETE SET NULL clause to set the child rows value to null when a parent key is deleted:

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

```
Table EMP2 altered.
```

Renaming Table Columns and Constraints

- Use the RENAME table clause of the ALTER TABLE statement to rename tables.

```
ALTER TABLE marketing RENAME to new_marketing;
```

1

- Use the RENAME COLUMN clause of the ALTER TABLE statement to rename table columns.

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

2

- Use the RENAME CONSTRAINT clause of the ALTER TABLE statement to rename any existing constraint for a table.

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

3

Disabling Constraints

- Execute the DISABLE clause of the ALTER TABLE statement to deactivate an integrity constraint.
- Apply the CASCADE option to disable the primary key. It will also disable all dependent FOREIGN KEY constraints automatically.

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

```
Table EMP2 altered.
```

```
ALTER TABLE    dept2
DISABLE primary key CASCADE;
```

```
Table DEPT2 altered.
```

Enabling Constraints

- Activate an integrity constraint that is currently disabled in the table definition by using the ENABLE clause.

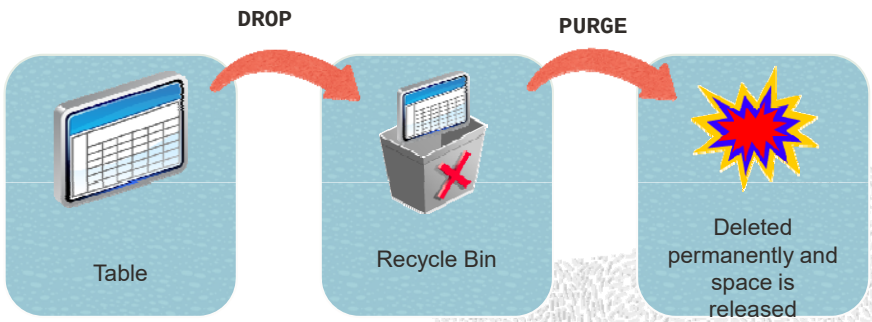
```
ALTER TABLE       emp2
ENABLE CONSTRAINT emp_dt_fk;
```

```
Table EMP2 altered.
```


DROP TABLE … PURGE

```
DROP TABLE emp_new_sal PURGE;
```

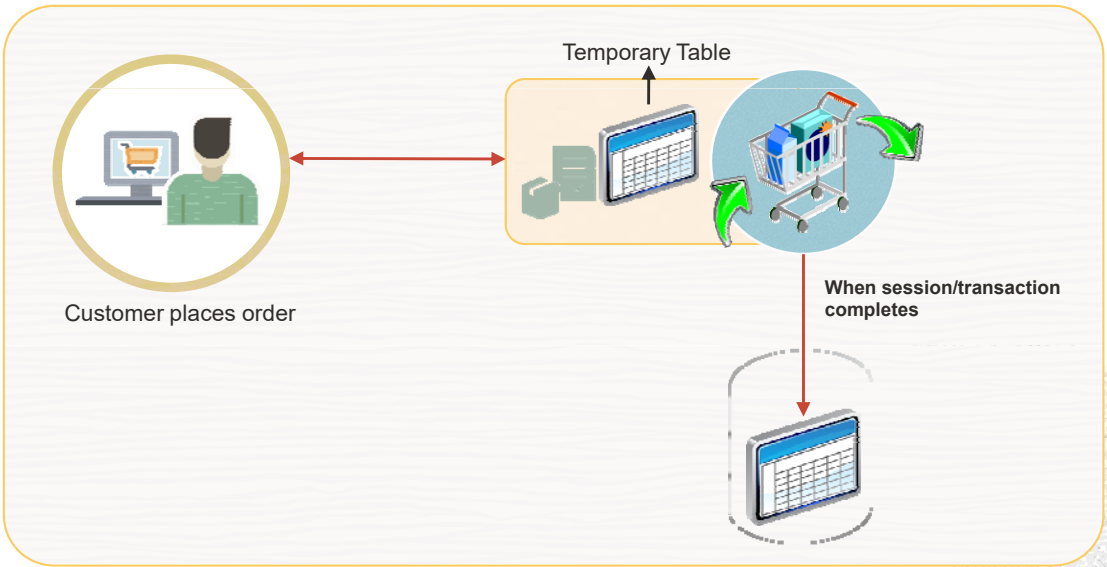
```
Table EMP_NEW_SAL dropped.
```



Lesson Agenda

- Managing constraints
- Creating and using temporary tables
- Creating and using external tables

Using Temporary Tables



Temporary Table

A temporary table:

- Holds data that exists only for the duration of a transaction or session
  - Data is private to the session.
- Can be either a Global Temporary Table or a Private Temporary Table

Temporary Table Characteristics

Characteristic	Global	Private
Naming rules	Same as for permanent tables	Must be prefixed with ORA$PTT_
Visibility of table definition	All sessions	Only the session that created the table
Storage of table definition	Disk	Memory only
Types	Transaction-specific (ON COMMIT DELETE ROWS) or session-specific (ON COMMIT PRESERVE ROWS)	Transaction-specific (ON COMMIT DROP DEFINITION) or session-specific (ON COMMIT PRESERVE DEFINITION)

Creating a Global Temporary Table

1

```
CREATE GLOBAL TEMPORARY TABLE cart(n NUMBER,d DATE)
ON COMMIT DELETE ROWS;
```

2

```
CREATE GLOBAL TEMPORARY TABLE emp_details(id NUMBER,name VARCHAR2(30))
ON COMMIT PRESERVE ROWS;
```


Creating a Private Temporary Table

1

```
CREATE PRIVATE TEMPORARY TABLE ORA$PTT_sales_trans
    (time_id      DATE,
     amt_sold  NUMBER(8,2))
   ON COMMIT DROP DEFINITION;
```

2

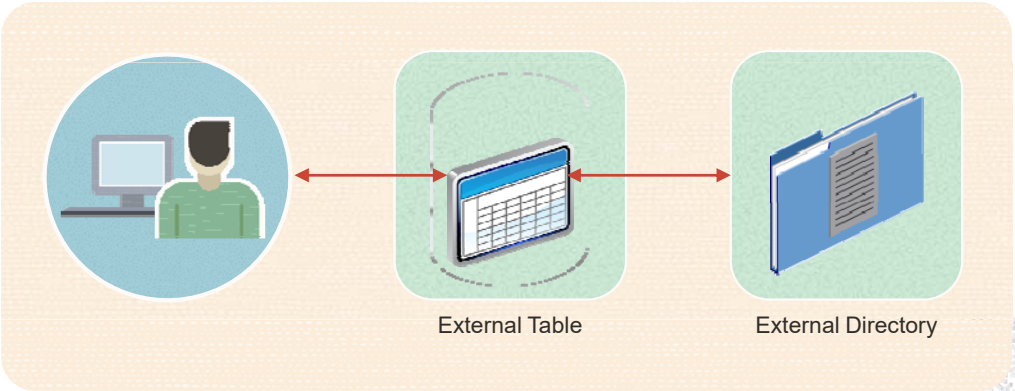
```
CREATE PRIVATE TEMPORARY TABLE ORA$PTT_sales_sess
    (time_id      DATE,
     amt_sold  NUMBER(8,2))
   ON COMMIT PRESERVE DEFINITION;
```

Lesson Agenda

- Managing constraints
- Creating and using temporary tables
- Creating and using external tables

17

External Tables



18

Creating a Directory for the External Table

Create a `DIRECTORY` object that corresponds to the directory on the file system where the external data source resides.

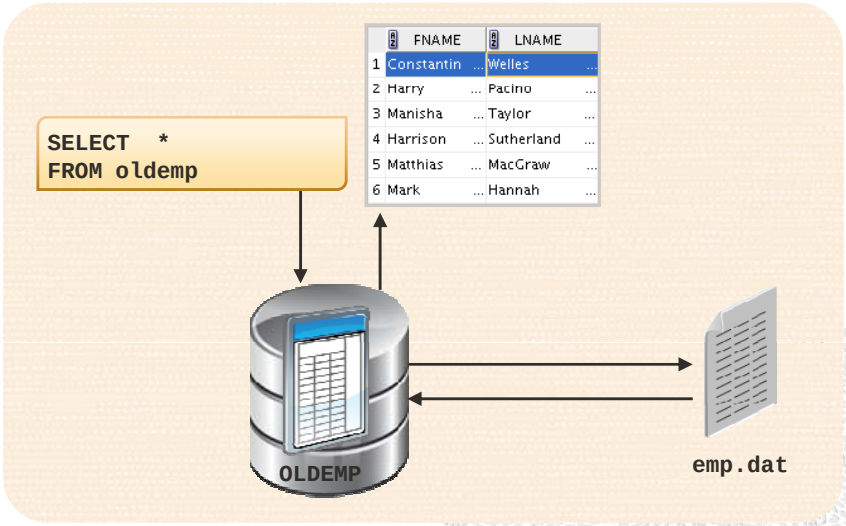
```
CREATE OR REPLACE DIRECTORY emp_dir
AS '/…/emp_dir';

GRANT READ ON DIRECTORY emp_dir TO ora_21;
```


Creating an External Table by Using 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
  FIELDS(fname POSITION ( 1:20) CHAR,
  lname POSITION (22:41) CHAR))
 LOCATION ('emp.dat'));
```

```
Table OLDEMP created.
```

Querying External Tables



Summary

In this lesson, you should have learned how to:

- Manage constraints
- Create and use temporary tables
- Create and use external tables