

9

Usando comandos DDL Creación y Gestión de Tablas

ORACLE®
DATABASE **11^g**

ORACLE®

Objetivos

Al finalizar esta lección, debería estar capacitado para:

- Describir los principales objetivos de base de datos
- Consultar la estructura de las tablas
- Describir los tipos de dato que se pueden utilizar al especificar la definición de columna
- Creación de tablas
- Entender cómo se crean las restricciones cuando se crean las tablas
- Modificar definiciones de tabla
- Describir cómo trabajan los objetos del esquema

Objetos de Base de Datos

Objeto	Descripción
Tabla	Unidad básica de almacenamiento, está formada por filas y columnas.
Vista	Representa lógicamente subconjuntos de datos de una o varias tablas.
Secuencia	Generador de valor numérico.
Índice	Mejora el rendimiento de algunas consultas
Sinónimo	Proporciona nombres alternativos a objetos.

Reglas de Nomenclatura

Los nombre y tablas de columnas:

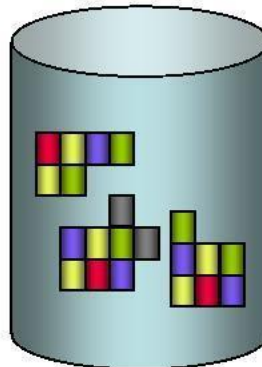
- Deben empezar por una letra.
- Deben tener entre 1 y 30 caracteres.
- Solo deben contener A-Z, a-z, 0-9, _, \$ y #.
- No deben duplicar el nombre de otro objeto propiedad del mismo usuario.
- No deben ser palabras reservadas de Oracle Server.

La Sentencia CREATE TABLE

- Debe tener:
 - El privilegio CREATE TABLE
 - Un área de almacenamiento

```
CREATE TABLE [ schema.] table  
                (column datatype [DEFAULT expr] [, ..]);
```

- Especifique:
 - Nombre de tabla
 - Nombre de columna, tipo de dato de columna y tamaño de columna



Creación de Tablas

- Cree la tabla

```
CREATE TABLE dept
```

```
    (deptno  NUMBER(2),  
     dname   VARCHAR2(14),  
     loc     VARCHAR(13),  
     create_date DATE DEFAULT SYSDATE);
```

Table created.

- Confirme la creación de la tabla.

```
DESCRIBE dept;
```

La Opción DEFAULT

- Especifique un valor por defecto para una columna durante la inserción

```
... Hire_date DATE DEFAULT SYSDATE, ...
```

- Los valores literales, las expresiones o las funciones SQL son valores válidos.
- El nombre de otra columna o una pseudocolumna son valores no válidos.
- El tipo de datos por defecto debe coincidir con el tipo de dato de columna.

```
CREATE TABLE hire_date  
    (id      NUMBER(8),  
    hire_date DATE DEFAULT SYSDATE);
```

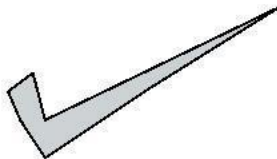
Table created

Tipos de Dato

Tipo de Dato	Descripción
VARCHAR2 (size)	Dato de caracteres de longitud variable
CHAR(size)	Dato de caracteres de longitud fija
NUMBER(p,s)	Dato numérico de longitud variable
DATE	Valores de Fecha y hora
LONG	Dato de caracteres de longitud variable de hasta 2 gigabytes
CLOB	Dato de caracteres de hasta 4 gigabytes
RAW y LONG RAW	Dato raw binario
BLOB	Dato binario de hasta 4 gigabytes
BFILE	Dato binario almacenado en un archivo externo; hasta 4 gigabytes
ROWID	Sistema numérico de base 64 que representa la dirección única de una fila en su tabla.

Incluyendo restricciones

- Las restricciones fuerzan las reglas a nivel de tabla.
- Las restricciones evitan la supresión de una tabla si hay dependencias.
- Son validos los siguientes tipos de restricción:
 - NOT NULL
 - UNIQUE
 - PRIMARY KEY
 - FOREIGN KEY
 - CHECK



Instrucciones sobre Restricciones

- Asigne un nombre a una restricción; si no lo hace, Oracle Server genera un nombre con el formato SYS_Cn.
- Cree una restricción:
 - Al mismo tiempo que se crea la tabla, o bien
 - Una vez creada la tabla.
- Defina una restricción a nivel de columna o de tabla.
- Visualice una restricción en el diccionario de datos.

Definición de Restricciones

- Sintaxis:

```
CREATE TABLE [ schema.] table  
          (column datatype [DEFAULT expr] [, ..]  
          [column_constraint],  
          ...  
          [table_constraint] [, . . .];
```

- Restricción a nivel de columna:

```
Column    [CONSTRAINT constraint_name] constraint_type,
```

- Restricción a nivel de tabla

```
Column  
    [CONSTRAINT constraint_name] constraint_type  
    (column, . . .)]
```

Definición de Restricciones

- Restricción a nivel de columna:

```
CREATE TABLE employees (  
    employees_id    NUMBER(6)  
    CONSTRAINT emp_emp_id_pk PRIMARY KEY,  
    first_name      VARCHAR(20),  
    . . .);
```

1

- Restricción a nivel de tabla

```
CREATE TABLE employees (  
    employees_id    NUMBER(6),  
    first_name      VARCHAR(20),  
    . . .  
    job_id          VARCHAR2(10) NOT NULL,  
    CONSTRAINT emp_emp_id_pk  
        PRIMARY KEY (EMPLOYEE_ID) );
```

2

La Restricción NOT NULL

Asegura que no se permiten valores nulos para la columna:

EMPLOYEE_ID	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	DEPARTMENT_ID
100	King	SKING	515.123.4567	17/06/87	AD_PRES	24000	90
101	Kochhar	NKOCHHAR	515.123.4568	21/09/89	AD_VP	17000	90
102	De Haan	LDEHAAN	515.123.4569	13/01/93	AD_VP	17000	90
103	Hunold	AHUNOLD	590.423.4567	03/01/90	IT_PROG	9000	60
104	Ernst	BERNST	590.423.4568	21/05/91	IT_PROG	6000	60
105	Austin	DAUSTIN	590.423.4569	25/06/97	IT_PROG	4800	60
106	Pataballa	VPATABAL	590.423.4560	05/02/98	IT_PROG	4800	60
107	Lorentz	DLORENTZ	590.423.5567	07/02/99	IT_PROG	4200	60
108	Greenberg	NGREENBE	515.124.4569	17/08/94	FI_MGR	12000	100
109	Faviet	DFAVIET	515.124.4169	16/08/94	FI_ACCOUNT	9000	100



**Restricción
NOT NULL**
(Ninguna fila
puede
contener un
valor nulo para
esta columna)



**Restricción
NOT NULL**

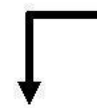


**Ausencia de
Restricción NOT
NULL** (Cualquier fila
puede contener un
valor nulo para esta
columna)

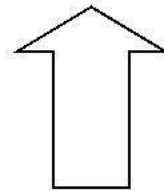
La Restricción UNIQUE

EMPLOYEES

EMPLOYEE_ID	LAST_NAME	EMAIL
100	King	SKING
101	Kochhar	NKOCHHAR
102	De Haan	LDEHAAN
103	Hunold	AHUNOLD
104	Ernst	BERNST
105	Austin	DAUSTIN
106	Pataballa	VPATABAL



Restricción UNIQUE



INSERT INTO

206	Gietz	WGIEZT
-----	-------	--------



Permitido

101	Kochhar	NKOCHHAR
-----	---------	----------



**No permitido:
ya existe**

ORACLE

La Restricción UNIQUE

Definida a nivel de tabla o columna:

```
CREATE TABLE employees (  
    employee_id    NUMBER (6),  
    last_name      VARCHAR2(25) NOT NULL,  
    email          VARCHAR2(25),  
    salary         NUMBER(8,2)  
    commission_pct NUMBER(2,2),  
    hire_date      DATE NOT NULL,  
    ...  
    CONSTRAINT emp_email_uk UNIQUE(email) );
```

La Restricción PRIMARY KEY

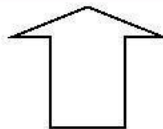
DEPARTMENTS

PRIMARY KEY



DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500
60	IT	103	1400
70	Public Relations	204	2700
80	Sales	145	2500

INSERT INTO

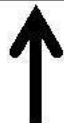


No permitido
(Valor nulo)



	Public Accounting		1400
50	Finance	124	1500

No Permitido
(50 ya existe)



ORACLE

La Restricción FOREIGN KEY

DEPARTMENTS

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500
60	IT	103	1400
70	Public Relations	204	2700
80	Sales	145	2500

PRIMARY
KEY →

EMPLOYEES

EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID
100	King	90
101	Kochhar	90
102	De Haan	90
103	Hunold	60
104	Ernst	60
105	Austin	60
106	Pataballa	60
107	Lorentz	60

← FOREIGN KEY

INSERT INTO

200	Ford	9
201	Ford	60

No permitido:
(9 no existe)

Permitido

ORACLE

La Restricción FOREIGN KEY

Defina a nivel de tabla o de columna:

```
CREATE TABLE employees (  
    employee_id      NUMBER(6),  
    last_name        VARCHAR2(25) NOT NULL,  
    email            VARCHAR2(25),  
    salary            NUMBER(8,2),  
    commission_pct   NUMBER(2,2),  
    hire_date        DATE NOT NULL,  
    ...  
    department_id    NUMBER(4),  
    CONSTRAINT emp_dept_fk FOREIGN KEY (department_id)  
        REFERENCES departments(department_id),  
    CONSTRAINT emp_email_uk UNIQUE(email) );
```

Palabras Clave de la Restricción FOREIGN KEY

- FOREIGN KEY: Define la columna de la tabla secundaria a nivel de restricción de tabla.
- REFERENCES: Identifica la tabla y la columna en la tabla principal.
- ON DELETE CASCADE: Suprime las filas dependientes de la tabla secundaria cuando se suprime una fila en la tabla principal.
- ON DELETE SET NULL: Convierte los valores de clave ajena dependientes en valores nulos.

Restricción CHECK

- Define una condición que debe satisfacer cada fila.
- No se permiten las siguientes expresiones:
 - Referencias a las pseudocolumnas CURRVAL, NEXTVAL, LEVEL, ROWNUM
 - Llamadas a las funciones SYSDATE, UID, USER y USERENV
 - Consultas que hagan referencia a otros valores de otras filas

```
... ,    salary    NUMBER(8, 2)
        CONSTRAINT emp_salary_min
        CHECK (salary > 0), ...
```

Crear tabla : Ejemplo

```
CREATE TABLE employees
( employee_id  NUMBER(6)
  CONSTRAINT emp_employee_id PRIMARY KEY,
  first_name  VARCHAR2(20),
  last_name   VARCHAR2(25)
  CONSTRAINT emp_last_name_nn  NOT NULL,
  email       VARCHAR2(25)
  CONSTRAINT emp_email_nn     NOT NULL
  CONSTRAINT emp_email_uk     UNIQUE,
  phone_number VARCHAR2(20),
  hire_date    DATE
  CONSTRAINT emp_hire_date_nn NOT NULL,
  job_id       VARCHAR2(10)
  CONSTRAINT emp_job_nn      NOT NULL,
  salary       NUMBER(8,2)
  CONSTRAINT emp_salary_ck   CHECK (salary >0),
  commission_pct  NUMBER(2,2),
  manager_id     NUMBER(6),
  department_id  NUMBER(4)
  CONSTRAINT emp_dept_fk REFERENCES
    departments (department_id) );
```

Violando constraints

```
UPDATE employees
SET    department_id = 55
WHERE  department_id = 110;
```

```
UPDATE    employees
          *
ERROR at line 1:
ORA-02291: Integrity constraint (HR.EMP_DEPT_FK);
Violate - parent key not found
```

El departamento 55 no existe.

Violando constraints

No puede suprimir una fila que contenga una clave primaria utilizada como clave ajena en otra tabla.

```
DELETE FROM departments
WHERE department_id = 60;
```

```
DELETE FROM departments
      *
ERROR at line 1:
ORA-02292: integrity constraint (HR.EMP_DEPT_FK)
Violated - child record found
```

Creación de una Tabla Utilizando una Sintaxis de Subconsulta

- Cree una tabla e inserte filas combinando la sentencia CREATE TABLE y la opción AS subquery.

```
CREATE TABLE table  
           [ (column, column . . .) ]  
AS subquery
```

- Haga coincidir el número de columnas especificadas con el número de columnas de subconsulta.
- Defina columnas con nombres de columna y valores por defecto.

Creación de una Tabla Utilizando una Subconsulta

```
CREATE TABLE dept80
```

```
AS
```

```
SELECT  employee_id, last_name,  
        salary*12 ANNSAL,  
        hire_date  
FROM    employees  
WHERE   department_id = 80;
```

Table Created.

```
DESCRIBE dept80;
```

Name	Null?	Type
EMPLOYEE_ID		NUMBER(6)
LAST_NAME	NOT NULL	VARCHAR2(25)
ANNSAL		NUMBER
HIRE_DATE	NOT NULL	DATE

La sentencia ALTER TABLE

Utilice la sentencia ALTER TABLE para:

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

La sentencia ALTER TABLE

Utilice la sentencia ALTER TABLE para agregar, modificar o 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) ;
```

Adición de una Columna

DEPT80

EMPLOYEE_ID	LAST_NAME	ANNSAL	HIRE_DAT
145	Russell	168000	01/10/96
146	Partners	162000	05/01/97
147	Errazuriz	144000	10/03/97
148	Cambrault	132000	15/10/99

Columna nueva

JOB_ID

“Agregue una columna nueva a la tabla DEPT80 ”

DEPT80

EMPLOYEE_ID	LAST_NAME	ANNSAL	HIRE_DAT	JOB_ID
145	Russell	168000	01/10/96	
146	Partners	162000	05/01/97	
147	Errazuriz	144000	10/03/97	
148	Cambrault	132000	15/10/99	
149	Zlotkey	126000	29/01/00	

Adición de una Columna

- Puede utilizar la cláusula ADD para agregar columnas.

```
ALTER TABLE dept80
ADD          ( job_id  VARCHAR2(9) );
Table altered.
```

- La nueva columna pasa a ser la última.

EMPLOYEE_ID	LAST_NAME	ANNSAL	HIRE_DAT	JOB_ID
145	Russell	168000	01/10/96	
146	Partners	162000	05/01/97	
147	Errazuriz	144000	10/03/97	
148	Cambrault	132000	15/10/99	
149	Zlotkey	126000	29/01/00	

Modificación de una Columna

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

```
ALTER TABLE dept80  
MODIFY      ( last_name VARCHAR2(30) );  
Table altered.
```

- Un cambio en el valor por defecto sólo afecta a las inserciones posteriores en la tabla.

Eliminación de una Columna

Utilice la cláusula DROP COLUMN para borrar las columnas que ya no necesite de la tabla.

```
ALTER TABLE dept80  
DROP COLUMN job_id;  
Table altered.
```

Borrando una tabla

Utilice la sentencia DROP TABLE para:

- Todos los datos y la estructura son borrados
- Se comprometen algunas transacciones pendientes
- Todos los índices son borrados
- Todas las restricciones son borradas
- No se puede ejecutar ROLLBACK a una tabla que fue borrada.

```
DROP TABLE dept80;
```

```
Table dropped.
```


Recuperando una Tabla

- En Oracle 10g, podemos recuperar BD, y Tablas
- Cuando se borra una tabla, Oracle no reclama inmediatamente el espacio asociado a la misma, sino que coloca la tabla y cualquier objeto asociado a ella en la Papelera de Reciclaje.
- En caso de que la tabla se haya borrado por error, se podrá recuperar por medio del comando Flashback
- Si el espacio del área de recuperación es menor al tamaño de la tabla, no hace uso del área de recuperación y la tabla no será posible recuperarla.

FLASHBACK TABLE dept80 TO BEFORE DROP;

Tabla recuperada

Cambio de Nombre de un Objeto

- Para cambiar el nombre de una tabla, una vista, una secuencia o un sinónimo, ejecute la sentencia RENAME.

```
RENAME dept80 TO detail_dept;  
Table renamed.
```

- Debe ser el propietario del objeto.

Truncamiento de una Tabla

- La sentencia TRUNCATE TABLE:
 - Elimina todas las filas de una tabla.
 - Libera el espacio de almacenamiento utilizado por dicha tabla.

```
TRUNCATE TABLE detail_dept;  
Table truncated.
```

- No puede realizar rollback de la eliminación de filas si utiliza TRUNCATE.
- También puede eliminar filas utilizando la sentencia DELETE.

Resumen

En esta lección, debería haber aprendido a utilizar sentencias DDL para crear, modificar, borrar y cambiar el nombre a tablas.

Sentencia	Descripción
CREATE TABLE	Crea una tabla.
ALTER TABLE	Modifica la estructura de la tabla.
DROP TABLE	Elimina las filas y la estructura de la tabla.
RENAME	Cambia el nombre de una tabla, vista, secuencia o sinónimo.
TRUNCATE	Elimina todas las filas de una tabla y libera el espacio de almacenamiento.

Visión General de la Práctica 9

Esta práctica cubre los siguientes temas:

- Creación de tablas nuevas
- Creación de una tabla nueva utilizando la sintaxis CREATE TABLE AS
- Modificación de definiciones de columna
- Verificación de que existen las tablas
- Eliminación de tablas
- Modificación de tablas