



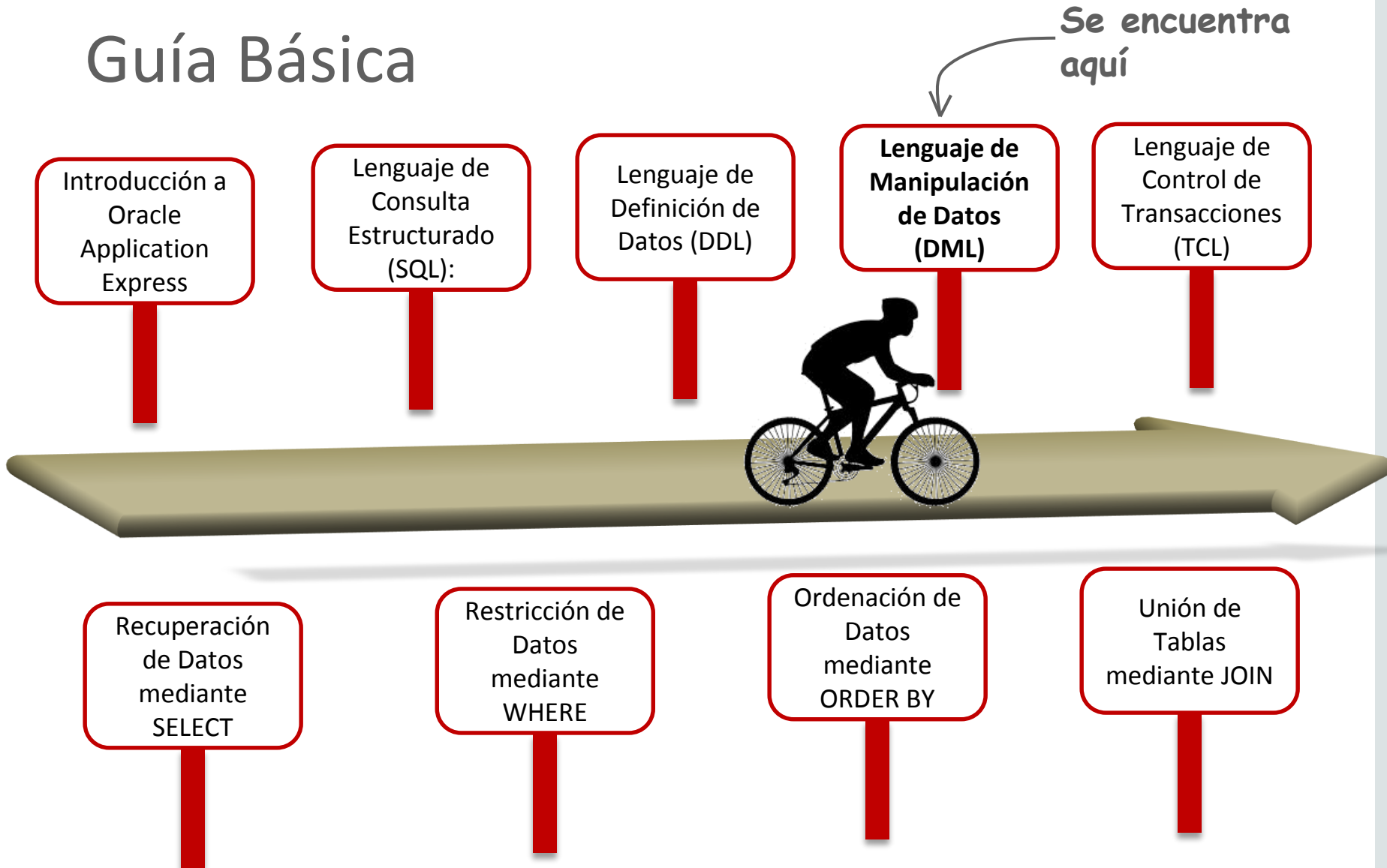
Fundamentos de bases de datos

6-4

Lenguaje de Manipulación de Datos (DML)



Guía Básica



Objetivos

En esta lección se abordan los siguientes objetivos:

- Describir la finalidad del lenguaje de manipulación de datos (DML)
- Explicar las operaciones DML que son necesarias para gestionar los datos de tabla de una base de datos:
 - Insertar
 - Actualización
 - Supresión



Lenguaje de Manipulación de Datos

- Las sentencias DML se ejecutan al:
 - Agregar nuevas filas a una tabla
 - Modificar filas existentes en una tabla
 - Eliminar filas existentes de una tabla
- Una transacción consta de una recopilación de sentencias DML que forman una unidad lógica de trabajo.



Adición de una Nueva Fila a una Tabla

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
80	Sales	145	2500

Nueva fila

70	Public Relations	204	2700
----	------------------	-----	------

Insertar la nueva fila en la
tabla 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

Sintaxis de las Sentencias INSERT

- Agregar filas a una tabla mediante la sentencia INSERT:

```
INSERT INTO  table [(column [, column...])]  
VALUES      (value [, value...]);
```

- Con esta sintaxis, sólo se inserta una fila cada vez.

Inserción de Filas

- Insertar una fila que contenga los valores de cada columna.

```
INSERT INTO departments  
VALUES (80, 'Marketing', 400, 1900);
```

- Mostrar valores en el orden por defecto de las columnas de la tabla.
- Opcionalmente, mostrar la lista de columnas en la cláusula INSERT.

```
INSERT INTO departments(department_id,  
                        department_name, manager_id, location_id)  
VALUES (70, 'Public Relations', 100, 1700);
```


Inserción de Filas

- Encerrar valores de caracteres y de fecha entre comillas simples.

```
INSERT INTO departments(department_id,  
                        department_name, manager_id, location_id)  
VALUES (70, 'Public Relations', 100, 1700);
```

Supuesto de Caso: Inserción de Filas

Entiendo que la sentencia **INSERT** se utiliza para agregar filas a una tabla. ¿Es posible insertar una fila si algunas columnas no tienen valores?



Alumno



Profesor

Por supuesto. Puede utilizar la sentencia **INSERT** para especificar las columnas para las que no se introducirán valores.

Inserción de Filas con Valores Nulos

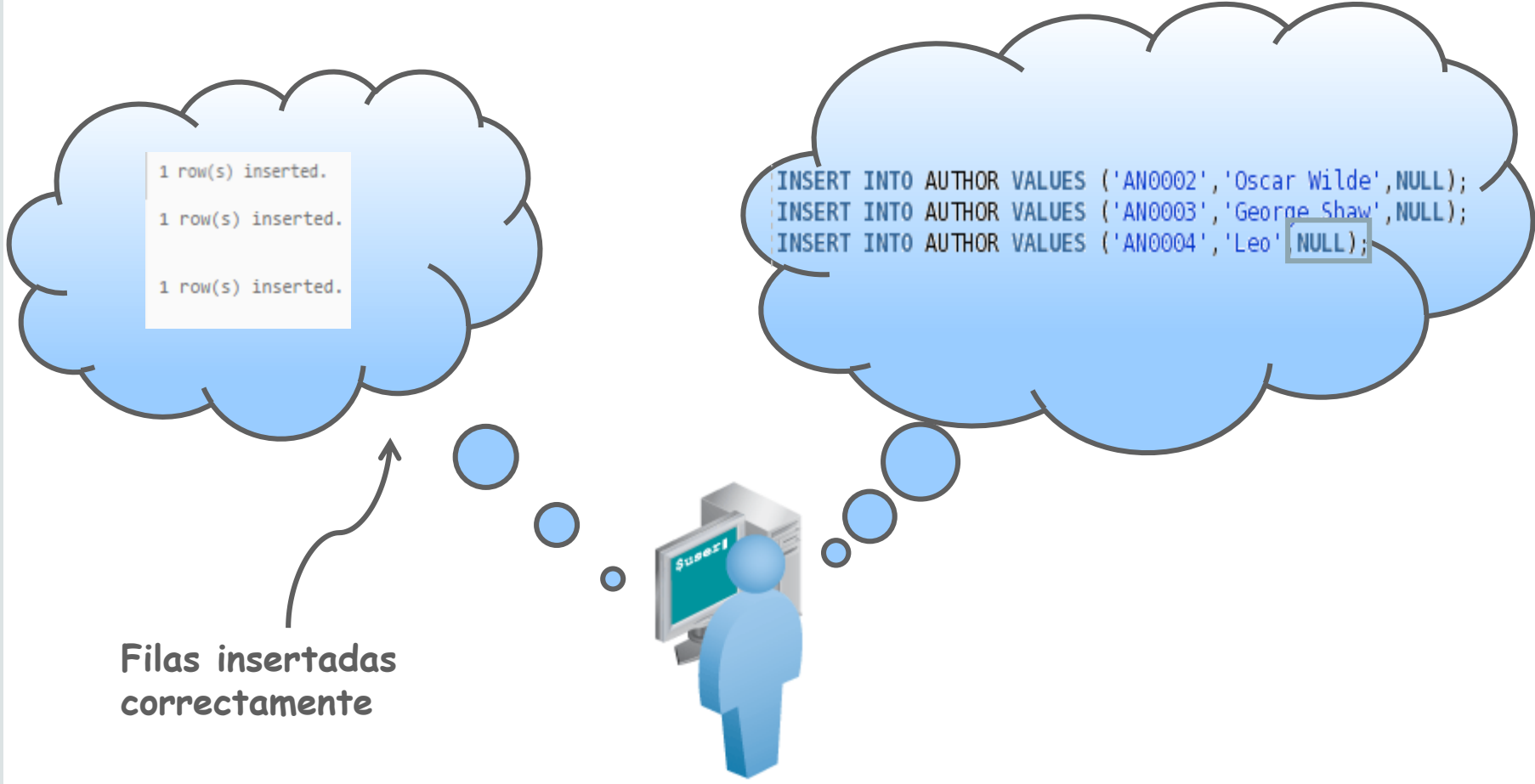
- Método implícito: omitir la columna de la lista de columnas.

```
INSERT INTO departments (department_id,  
                        department_name)  
VALUES      (30, 'Purchasing');
```

- Método explícito: especificar la palabra clave NULL en la cláusula VALUES.

```
INSERT INTO departments  
VALUES      (100, 'Finance', NULL, NULL);
```

Supuesto de Caso: Inserción de Filas



```
1 row(s) inserted.  
1 row(s) inserted.  
1 row(s) inserted.
```

Filas insertadas
correctamente

```
INSERT INTO AUTHOR VALUES ('AN0002', 'Oscar Wilde', NULL);  
INSERT INTO AUTHOR VALUES ('AN0003', 'George Shaw', NULL);  
INSERT INTO AUTHOR VALUES ('AN0004', 'Leo', NULL);
```

Inserción de Valores Especiales

La función `SYSDATE` registra la fecha y la hora actuales.

```
INSERT INTO employees (employee_id,  
                        first_name, last_name,  
                        email, phone_number,  
                        hire_date, job_id, salary,  
                        commission_pct, manager_id,  
                        department_id)  
VALUES  
      (113,  
       'Louis', 'Popp',  
       'LPOPP', '515.124.4567',  
       SYSDATE, 'AC_ACCOUNT', 6900,  
       NULL, 205, 110);
```

Inserción de Valores de Fecha y Hora Específicos

- Agregar un empleado.

```
INSERT INTO employees
VALUES      (114, 'Den', 'Raphealy',
             'DRAPHEAL', '515.127.4561',
             TO_DATE('Dec 7, 2002', 'MON DD, YYYY'),
             'SA_REP', 11000, 0.2, 100, 60, NULL);
```

1 rows inserted

114	Den	Raphealy	DRAPHEAL	515.127.4561	07-Dec-2002
-----	-----	----------	----------	--------------	-------------


- Verificar la adición.

Cambio de Datos en la Tabla

EMPLOYEES

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
100	Steven	King	SKING	515.123.4567	17-Jun-1987	AD_PRES	24000	-	-	90
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-Sep-1989	AD_VP	17000	-	100	90
102	Lex	De Haan	LDEHAAN	515.123.4569	13-Jan-1993	AD_VP	17000	-	100	90
200	Jennifer	Whalen	JWHALEN	515.123.4444	17-Sep-1987	AD_ASST	4400	-	101	10

Actualizar filas en la tabla **EMPLOYEES**:



EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
100	Steven	King	SKING	515.123.4567	17-Jun-1987	AD_PRES	24000	-	-	60
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-Sep-1989	AD_VP	17000	-	100	60
102	Lex	De Haan	LDEHAAN	515.123.4569	13-Jan-1993	AD_VP	17000	-	100	60
200	Jennifer	Whalen	JWHALEN	515.123.4444	17-Sep-1987	AD_ASST	4400	-	101	10

Sintaxis de Sentencias UPDATE

- Modificar los valores existentes en una tabla con la sentencia UPDATE:

```
UPDATE      table
SET         column = value [, column = value, ...]
[WHERE      condition];
```

- Actualizar más de una fila cada vez (si es necesario).

Actualización de Filas en una Tabla

- Si se especifica la cláusula `WHERE`, se modifican los valores de una fila o varias filas específicas:

```
UPDATE employees
SET    department_id = 50
WHERE  employee_id = 113;
```

- Si se omite la cláusula `WHERE`, se modifican los valores de todas las filas de la tabla:

```
UPDATE    copy_emp
SET       department_id = 110;
```

- Especifique `SET column_name= NULL` para actualizar un valor de columna a `NULL`.

Violación de Restricciones

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

```
ORA-02291: integrity constraint (APEXWS2.EMP_DEPT_FK) violated - parent  
key not found
```

El departamento 55 no existe.

Eliminación de Filas de Tablas

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
90	Executive	100	1700

Suprimir una fila de la
tabla 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
80	Sales	145	2500
90	Executive	100	1700

Sentencia DELETE

Puede eliminar filas existentes de una tabla mediante la sentencia `DELETE`:

```
DELETE [FROM] table  
[WHERE condition];
```

Supresión de Filas de Tablas

- Se suprimen las filas concretas si se incluye la cláusula `WHERE`:

```
DELETE FROM departments  
WHERE department_name = 'Finance';
```

- Se suprimen todas las filas de la tabla si omite la cláusula `WHERE`:

```
DELETE FROM copy_emp;
```

Violación de Restricciones

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

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

```
ORA-02292: integrity constraint (APEXWS2.JHIST_DEPT_FK) violated - child  
record found
```

Supuesto de Caso: Supresión de Filas

¿Es posible suprimir todas las filas de una tabla, pero dejar la estructura intacta?



Alumno



Profesor

Por supuesto. Puede utilizar la sentencia TRUNCATE para ello.

Sentencia TRUNCATE

- Elimina todas las filas de una tabla, dejando la tabla vacía y la estructura de la misma intacta.
- Es una sentencia DDL en lugar de una sentencia DML; no se puede deshacer fácilmente
- Sintaxis:

```
TRUNCATE TABLE nombre_tabla;
```

- Ejemplo:

```
TRUNCATE TABLE copy_emp;
```


La recopilación de sentencias DML que forman una unidad lógica de trabajo se denomina transacción.

- ☐ Verdadero
- ☐ Falso

Section6_Lesson4_Quiz

Quiz - 6 questions

Last Modified: May 25, 2016 at 07:02 AM

PROPERTIES

On passing, 'Finish' button: [Goes to Next Slide](#)

On failing, 'Finish' button: [Goes to Next Slide](#)

Allow user to leave quiz: [At any time](#)

User may view slides after quiz: [At any time](#)

Show in menu as: [Multiple items](#)



Edit in Quizmaker



Edit Properties

Resumen

En esta lección, debe haber aprendido lo siguiente:

- Describir el objetivo de DML
- Mostrar las operaciones DML que son necesarias para gestionar los datos de tabla de una base de datos:
 - Insertar
 - Actualización
 - Supresión





 **ACADEMY**