



 **ACADEMY**

# Programación de bases de datos con SQL

13-1

Creación de tablas



**ORACLE** ACADEMY

Copyright © 2017, Oracle y/o sus filiales. Todos los derechos reservados.

# Objetivos

En esta lección, aprenderá a:

- Enumerar y clasificar los principales objetos de base de datos
- Revisar la estructura de una tabla
- Describir cómo la base de datos Oracle utiliza los objetos de esquema
- Crear una tabla mediante el tipo de dato adecuado para cada columna
- Explicar el uso de tablas externas
- Consultar el diccionario de datos para obtener los nombres y otros atributos de objetos de la base de datos

# Objetivo

- Hasta ahora, ha seleccionado, actualizado, insertado y suprimido información en las tablas existentes de una base de datos.
- Como administrador de base de datos (DBA), se espera que sepa también cómo crear tablas.
- En esta lección, conocerá los objetos de base de datos que más se utilizan, cómo mirar la estructura de la tabla y cómo crear tablas nuevas.

# Objetivo

- Las tablas serán pequeñas en comparación con las tablas que contienen millones de filas y cientos de columnas, pero la creación de una tabla pequeña implica las mismas sentencias SQL y sintaxis que la creación de una muy grande.
- También aprenderá sobre las tablas externas, tablas que son similares en cuanto a estructura a las tablas de base de datos Oracle normales, pero las filas de datos reales se almacenan de forma externa un archivo plano y solo se acceden a ellas cuando es necesario.

# Objetos de Esquema de la Base de Datos

- Una base de datos Oracle puede contener muchos tipos distintos de objetos.
- En esta sección se presentan los objetos más utilizados y también se describe cómo Oracle Server utiliza la información almacenada en el diccionario de datos cuando realiza el trabajo como resultado de las sentencias SQL que emita.

# Objetos de Esquema de la Base de Datos

- Los principales tipos de objetos de base de datos son:
  - Tabla
  - Índice de
  - ¿La restricción
  - Ver
  - Secuencia
  - Sinónimo
- Algunos de estos tipos de objetos pueden existir de forma independiente y otros no pueden.

# Objetos de Esquema de la Base de Datos

- Algunos de los tipos de objetos ocupan espacio, lo que se denomina almacenamiento, en la base de datos y otros no.
- Los objetos de base de datos que ocupan mucho espacio de almacenamiento se denominan segmentos.
- Las tablas y los índices son ejemplos de segmentos, ya que los valores almacenados en las columnas de cada fila ocupan mucho espacio en el disco físico.
- Las vistas, restricciones, secuencias y sinónimos también son objetos, pero el único espacio que necesitan en la base de datos está en la definición del objeto: ninguno de ellos tiene filas de datos asociadas.



# Objetos de Esquema de la Base de Datos

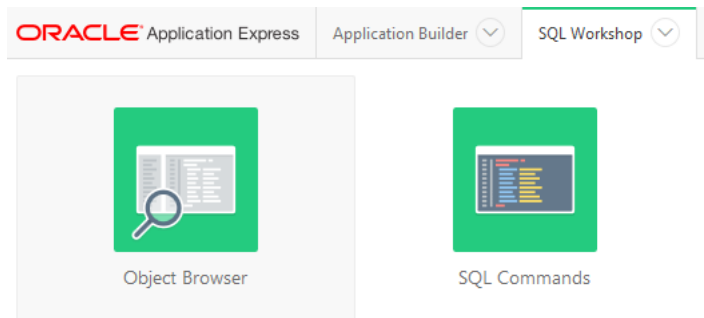
- La base de datos almacena las definiciones de todos los objetos de base de datos en el diccionario de datos y estas definiciones están accesibles para todos los usuarios de la base de datos así como para la base de datos en sí.
- ¿Se ha preguntado alguna vez cómo sabe Oracle qué columnas devolver de una consulta?
- Por ejemplo, si especifica `SELECT * FROM jobs` en lugar de `SELECT job_id, job_title FROM jobs`, ¿cómo sabe Oracle qué columnas devolver?
- La base de datos busca la definición de la tabla utilizada en la consulta, traduce el "\*" en la lista completa de columnas, y devuelve el resultado.

# Objetos de Esquema de la Base de Datos

- La base de datos utiliza el diccionario de datos para todas las sentencias que emita, incluso aunque enumere los nombres de columna en lugar de utilizar "\*".
- Comprueba que las tablas a las que hace referencia en la sentencia existen en la base de datos, comprueba que los nombres de columna son correctos, comprueba si tiene los privilegios correctos para realizar la acción que está solicitando y, finalmente, utiliza el diccionario de datos para decidir el plan de ejecución, cómo realizará en realidad la solicitud.

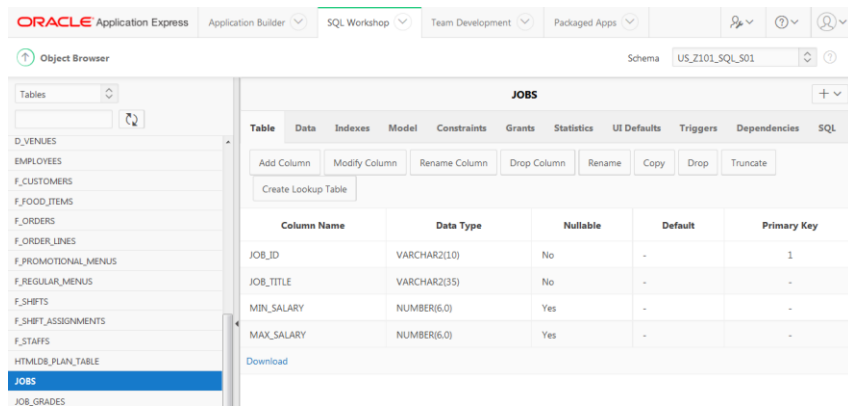
# Objetos de Esquema de la Base de Datos

- Todos los usuarios de la base de datos pueden consultar el diccionario de datos.
- En Application Express, se puede acceder a él mediante sentencias SQL en la interfaz SQL Workshop> SQL Commands y también desde la interfaz SQL Workshop> Object Browser.



# Objetos de Esquema de la Base de Datos

- En la ventana SQL Commands, debe saber los nombres de la tabla que consulta y en la interfaz Object Browser, solo tiene que hacer clic en los objetos mostrados para ver sus detalles.



# Objetos de Esquema de la Base de Datos

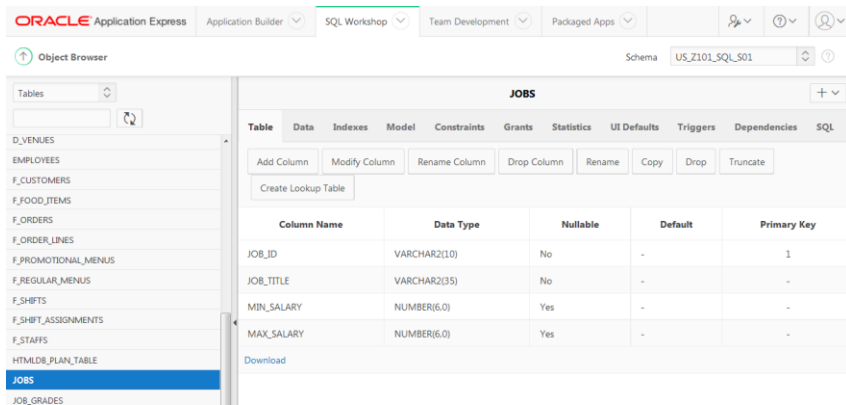
- Por lo tanto, si desea ver los detalles de la tabla JOBS, simplemente haga clic en su título en la tabla que aparece:

The screenshot shows the Oracle Application Express interface. At the top, there's a navigation bar with 'ORACLE Application Express', 'Application Builder', 'SQL Workshop', 'Team Development', and 'Packaged Apps'. Below this, the 'Object Browser' is visible, showing a list of tables in the 'US\_Z101\_SQL\_S01' schema. The 'JOBS' table is selected and highlighted in blue. To the right of the table list, the 'JOBS' table details are displayed. The 'Table' tab is active, showing a table with 5 columns: 'Column Name', 'Data Type', 'Nullable', 'Default', and 'Primary Key'. The table contains 4 rows of data: 'JOB\_ID' (VARCHAR2(10), No, -, 1), 'JOB\_TITLE' (VARCHAR2(35), No, -, -), 'MIN\_SALARY' (NUMBER(6,0), Yes, -, -), and 'MAX\_SALARY' (NUMBER(6,0), Yes, -, -). Below the table, there is a 'Download' link.

Column Name	Data Type	Nullable	Default	Primary Key
JOB_ID	VARCHAR2(10)	No	-	1
JOB_TITLE	VARCHAR2(35)	No	-	-
MIN_SALARY	NUMBER(6,0)	Yes	-	-
MAX_SALARY	NUMBER(6,0)	Yes	-	-

# Objetos de Esquema de la Base de Datos

- En este ejemplo, mediante Object Browser, puede ver los detalles de la tabla JOBS así como las opciones para ver los datos, los índices, las restricciones, los permisos y otros detalles de la tabla.



# Objetos de Esquema de la Base de Datos

- Mediante la ventana SQL Commands, debe solicitar una DESCripción de la tabla.
- Ninguna de las opciones adicionales que ofrece Object Browser está disponible en esta interfaz.

```
DESCRIBE jobs;
```

Results	Explain	Describe	Saved SQL	History					
Object Type		TABLE <span>?</span>	Object						
			JOBS <span>?</span>						
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
JOBS	JOB_ID	VARCHAR2	10	-	-	1	-	-	-
	JOB_TITLE	VARCHAR2	35	-	-	-	-	-	-
	MIN_SALARY	NUMBER	-	6	0	-	✓	-	-
	MAX_SALARY	NUMBER	-	6	0	-	✓	-	-

Puede utilizar DESCRIBE o su forma abreviada DESC para ver descripciones de tablas.

# Creación de Tabla

- Todos los datos de una base de datos relacional se almacenan en tablas.
- Al crear una nueva tabla, utilice las siguientes reglas para los nombres de tabla y columnas:
  - Deben empezar por una letra
  - Deben tener entre 1 y 30 caracteres
  - Solo deben contener A - Z, a - z, 0 - 9, \_ (carácter de subrayado), \$ y #
  - No deben ser un duplicado de otro nombre de objeto propiedad del mismo usuario
  - No deben ser una palabra reservada de Oracle Server



# Reglas de Nomenclatura

- Es mejor utilizar nombres descriptivos para las tablas y otros objetos de la base de datos.
- Si una tabla va a almacenar información sobre los alumnos, asígnele el nombre ALUMNOS, no PERSONAS o NIÑOS.
- Los nombres de tablas no son sensibles a mayúsculas/minúsculas.
- Por ejemplo, ALUMNOS se trata igual que ALuMnoS o alumnos.



# Reglas de Nomenclatura

- Los nombres de tabla deben ser en plural, por ejemplo, ALUMNOS, no alumno.
- La creación de tablas forma parte del lenguaje de definición de datos (DDL) de SQL.
- En las sentencias DDL que se utilizan para configurar, cambiar y eliminar estructuras de datos de tablas se incluyen ALTER, DROP, RENAME y TRUNCATE.



# CREATE TABLE

- Para crear una nueva tabla, debe tener el privilegio CREATE TABLE y un área de almacenamiento para él.
- El administrador de la base datos utiliza sentencias de lenguaje de control de datos (DCL) para otorgar este privilegio a los usuarios y asignar un área de almacenamiento.
- Las tablas que pertenecen a otros usuarios no se incluyen en su esquema.
- Si desea utilizar una tabla que no está en su esquema, utilice el nombre del propietario de la tabla como prefijo para el nombre de tabla:

```
SELECT *  
FROM mary.students;
```

# CREATE TABLE

- Si desea utilizar una tabla que no está en su esquema, utilice el nombre del propietario de la tabla como prefijo para el nombre de tabla:

```
SELECT *  
FROM mary.students;
```

- Se le debe otorgar acceso a una tabla para poder realizar selecciones en ella.

# Sintaxis de CREATE TABLE

- Para crear una nueva tabla, utilice los siguientes detalles de sintaxis:
  - table es el nombre de la tabla.
  - column es el nombre de la columna.
  - datatype es el tipo de dato y la longitud de la columna.
  - La expresión DEFAULT especifica un valor por defecto si se omite un valor en la sentencia INSERT.

```
CREATE TABLE table  
(column data type [DEFAULT expression],  
 (column data type [DEFAULT expression],  
 (.....[ ] );
```

Las definiciones de columnas se separan con comas.

Los tipos de dato se tratan con más detalle en la siguiente lección.

# Ejemplo de CREATE TABLE

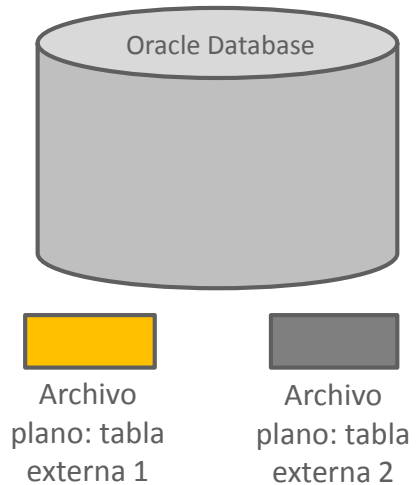
- Los ejemplos siguientes muestran la sentencia CREATE TABLE:

```
CREATE TABLE my_cd_collection
(cd_number NUMBER(3),
 title VARCHAR2(20),
 artist VARCHAR2(20),
 purchase_date DATE DEFAULT SYSDATE);
```

```
CREATE TABLE my_friends
(first_name VARCHAR2(20),
 last_name VARCHAR2(30),
 email VARCHAR2(30),
 phone_num VARCHAR2(12),
 birth_date DATE);
```

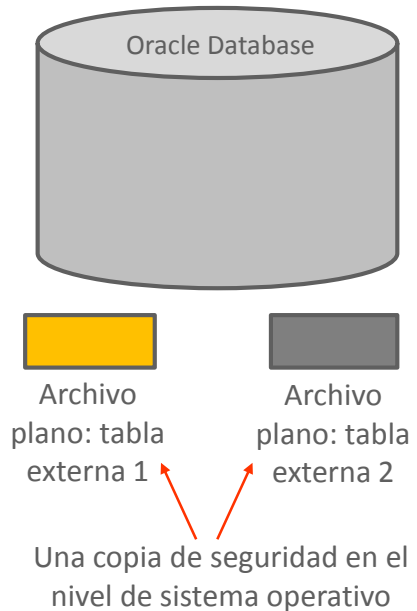
# Tablas Externas

- Oracle también soporta otro tipo de tabla: tabla externa.
- En una tabla externa, las filas de datos no están en los archivos de la base de datos, sino que en su lugar se encuentran en un archivo plano, almacenado de forma externa a la base de datos.
- Normalmente, una tabla externa se utiliza para almacenar datos migrados de versiones anteriores de las bases de datos utilizadas por una compañía.



# Tablas Externas

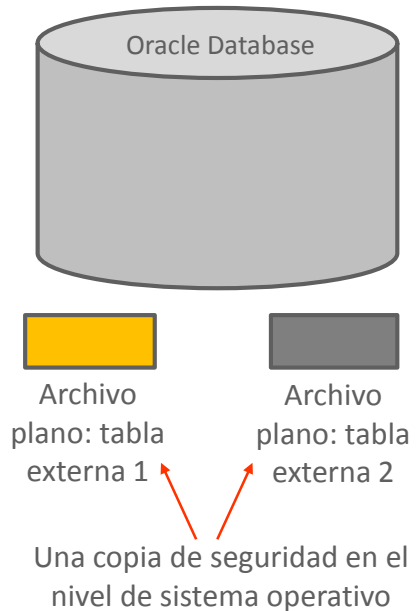
- Cuando una compañía implanta una nueva aplicación y base de datos, suele tener que importar la mayoría de los datos de los sistemas antiguos al nuevo sistema para el acceso de lectura y escritura normal, pero puede haber algunos datos que no se utilicen con frecuencia y, por lo tanto, solo se tenga que acceder a ellos para lectura.
- Este tipo de dato podría alojarse en una tabla externa.





# Tablas Externas

- Una de las principales ventajas de Oracle es que solo se tiene que hacer una copia de seguridad una vez de los datos alojados en tablas externas y, a continuación, nunca más a menos que el contenido del archivo cambie.
- La sintaxis para crear una tabla externa es muy similar a la de crear una tabla estándar, excepto por el hecho de que tiene sintaxis adicional al final.



Las compañías a menudo compran datos de referencia a fuentes externas, por ejemplo, información de marketing sobre clientes potenciales o archivos de direcciones con códigos postales. En lugar de cargar todos estos datos en la base de datos, las compañías suelen dejar los datos en archivos planos externos a la base de datos y acceder a ellos como una tabla externa.

# Tablas Externas

- La nueva sintaxis (que aparece en rojo) de las siguientes diapositivas no se utiliza en sentencias SQL estándar para la creación de tablas.
- ORGANIZATION EXTERNAL: indica a Oracle que cree una tabla externa
- TYPE ORACLE\_LOADER: de tipo Oracle Loader (producto de Oracle)
- DEFAULT DIRECTORY def\_dir1: nombre del directorio para el archivo

# Tablas Externas

- ACCESS PARAMETERS: cómo leer el archivo
- RECORDS DELIMITED BY NEWLINE: cómo identificar el principio de una nueva fila
- FIELDS: especificaciones de nombre de campo y tipo de dato
- LOCATION: nombre del archivo real que contiene los datos
- Un ejemplo de la nueva sintaxis se muestra en rojo en la siguiente diapositiva.

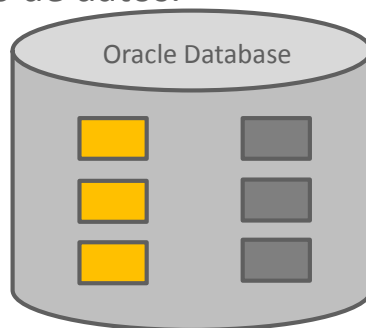
# Ejemplo de Tablas Externas

```
CREATE TABLE emp_load
  (employee_number CHAR(5),
  employee_dob CHAR(20),
  employee_last_name CHAR(20),
  employee_first_name CHAR(15),
  employee_middle_name CHAR(15),
  employee_hire_date DATE)
ORGANIZATION EXTERNAL
  (TYPE ORACLE_LOADER
  DEFAULT DIRECTORY def_dir1
  ACCESS PARAMETERS
  (RECORDS DELIMITED BY NEWLINE
  FIELDS (employee_number CHAR(2),
  employee_dob CHAR(20),
  employee_last_name CHAR(18),
  employee_first_name CHAR(11),
  employee_middle_name CHAR(11),
  employee_hire_date CHAR(10) date_format DATE mask "mm/dd/yyyy"))
  LOCATION ('info.dat'));
```

Esta consulta no se ejecutará en APEX porque no hay conexión a una tabla externa .

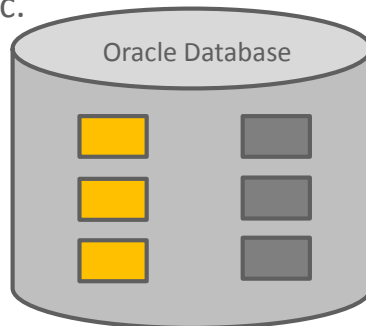
# Data Dictionary

- Existen dos tipos de tablas en una base de datos Oracle: tablas de usuario y tablas de diccionario de datos.
- Se pueden emitir sentencias SQL para acceder a ambos tipos de tablas: puede seleccionar, insertar, actualizar y suprimir datos de las tablas de usuario y puede seleccionar datos de la tabla de diccionario de datos.



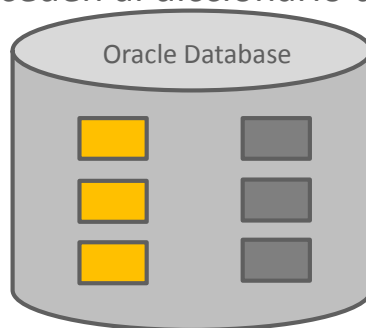
# Data Dictionary

- Las tablas de usuario creadas por el usuario que contiene sus datos:
  - empleados, departamentos, cargos, etc.
- Tablas de diccionario de datos:
  - DICTIONARY, USER\_OBJECTS, USER\_TABLES, USER\_SEGMENTS, USER\_INDEXES, etc.



# Data Dictionary

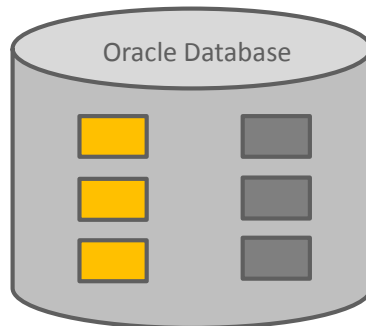
- Las tablas de diccionario de datos son propiedad de un usuario de Oracle especial llamado SYS y solo se deben utilizar sentencias SELECT al trabajar con cualquiera de estas tablas.
- Para proteger a estas tablas del acceso accidental de usuario, todas tienen vistas creadas a través de las que los usuarios de la base de datos acceden al diccionario de datos.



Las vistas se describirán más detalladamente más adelante en el curso.

# Data Dictionary

- Si cualquier usuario de Oracle intenta realizar inserciones, actualizaciones o supresiones en una de las tablas de diccionario de datos, no se permite la operación ya que puede poner en peligro la integridad de toda la base de datos.





# Data Dictionary

- Al utilizar las vistas del diccionario de datos en la interfaz SQL Commands, debe saber los nombres de las vistas del diccionario con las que está trabajando.
- En Oracle, esto es bastante sencillo: ponga como prefijo al tipo de objeto que está buscando USER\_XXX o ALL\_XXX, donde XXX es el tipo de objeto.

```
SELECT table_name, status  
FROM USER_TABLES;
```

```
SELECT table_name, status  
FROM ALL_TABLES;
```

# Data Dictionary

- Por lo tanto, si desea investigar los índices, simplemente seleccione USER\_INDEXES; si desea información acerca de las secuencias, la tabla es USER\_SEQUENCES, etc.

```
SELECT *  
FROM user_indexes;
```

```
SELECT *  
FROM user_objects  
WHERE object_type = 'SEQUENCE';
```

Si no está seguro de la vista del diccionario de datos que necesita, puede utilizar `SELECT * FROM DICTIONARY` y se devolverá el nombre de las vistas incluidas en el diccionario con una descripción de su contenido.

# Terminología

Entre los términos clave utilizados en esta lección se incluyen:

- CREATE TABLE
- Diccionario de datos
- Tabla
- Tabla externa
- Esquema
- DEFAULT

# Resumen

En esta lección, debe haber aprendido lo siguiente:

- Clasificación de los Principales Objetos de Base de Datos
- Revisar la estructura de una tabla
- Describir cómo se utilizan los objetos de esquema
- Crear una tabla mediante el tipo de dato adecuado para cada columna
- Explicar el uso de tablas externas
- Utilizar el diccionario de datos para obtener los nombres y otros atributos de objetos de la base de datos

