



ACADEMY

Fundamentos de bases de datos

6-2

Lenguaje de consulta estructurado (SQL)



ORACLE ACADEMY

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

Guía básica

Se encuentra aquí

Introducción a
Oracle
Application
Express (APEX)

Lenguaje de
consulta
estructurado
(SQL)

Lenguaje de
definición de
datos (DDL)

Lenguaje de
manipulación
de datos
(DML)

Lenguaje de
control de
transacciones
(TCL)

Recuperación
de datos
mediante
SELECT

Restricción de
datos
mediante
WHERE

Ordenación de
datos
mediante
ORDER BY

Unión de
tablas
mediante
JOIN

ORACLE ACADEMY

DFo 6-2
Lenguaje de consulta estructurado (SQL)

Copyright © 2017, Oracle y/o sus filiales. Todos los derechos reservados. 3

Objetivos

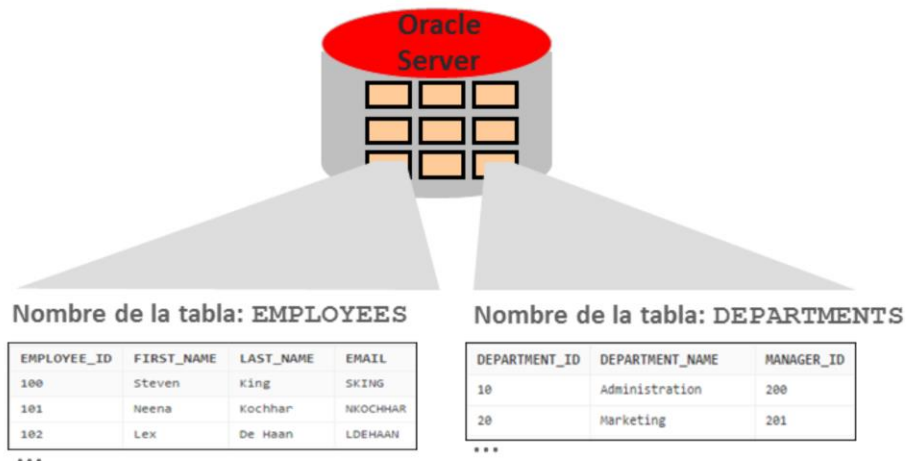
En esta lección se abordan los siguientes objetivos:

- Describir cómo se organizan los datos en una base de datos relacional
- Explicar las distintas terminologías de bases de datos relacionales
- Definir el lenguaje de consulta estructurado y sus funciones
- Describir cómo se produce el procesamiento SQL
- Identificar las herramientas que se utilizan para acceder a la base de datos relacional



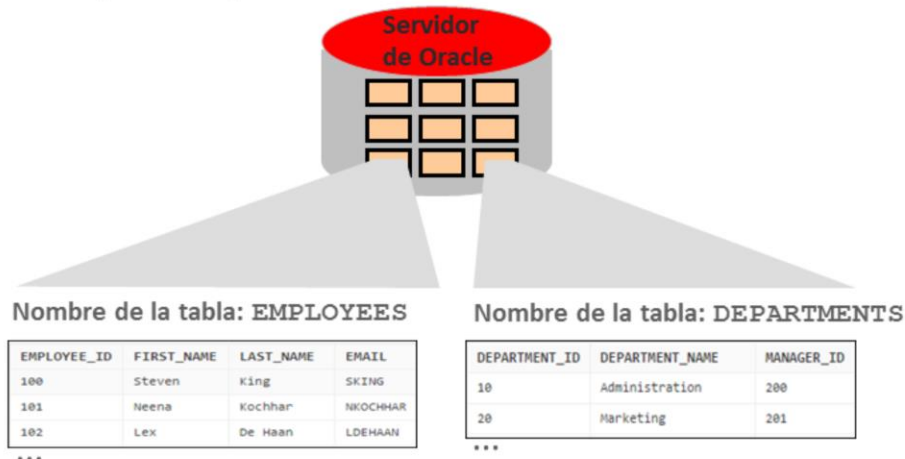
¿Cómo se organizan los datos en bases de datos relacionales?

- Los datos se almacenan en una matriz bidimensional conocida como tabla.



¿Cómo se organizan los datos en bases de datos relacionales?

- Se utiliza el software de DBMS para gestionar la lectura y manipulación de datos.



El software de Oracle se ejecuta en muchas arquitecturas de hardware y sistemas operativos. La computadora en la que reside el software de Oracle Database se conoce como el servidor de Oracle Database. Además, el servidor de Oracle Database puede hacer referencia al software de Oracle Database y sus datos.

Terminología de bases de datos relacionales

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	SALARY	COMMISSION_PCT	DEPARTMENT_ID
100	Steven	King	24000	-	90
101	Neena	Kochhar	17000	-	90
102	Lex	De Haan	17000	-	90
103	Alexander	Hunold	9000	-	60
104	Bruce	Ernst	6000	-	60
105	David	Austin	4800	-	60
106	Valli	Pataballa	4800	-	60
107	Diana	Lorentz	4200	-	60
108	Nancy	Greenberg	12008	-	100
109	Daniel	Faviet	9000	-	100
110	John	Chen	8200	-	100
111	Ismael	Sciarra	7700	-	100
112	Jose Manuel	Urman	7800	-	100

Nota: Consulte las notas para obtener explicaciones sobre las notaciones numéricas

ORACLE

ACADEMY

DFo 6-2
Lenguaje de consulta estructurado (SQL)

Copyright © 2017, Oracle y/o sus filiales. Todos los derechos reservados. 7

Una base de datos relacional puede contener una o más tablas. Una tabla es la estructura básica de almacenamiento de RDBMS. Una tabla contiene todos los datos necesarios sobre algo del mundo real, como empleados, facturas o clientes. La diapositiva muestra el contenido de la relación o tabla EMPLOYEES. Los números indican lo siguiente:

1. Una única fila (o tupla), que representa todos los datos necesarios para un empleado concreto. Cada fila de una tabla se debe identificar por una clave primaria, que no permite duplicar filas. El orden de las filas no es importante. Se puede especificar cuando se recuperen los datos.
2. Una columna o un atributo que contiene el valor de la clave primaria. El número de employee_id identifica a un único empleado en la tabla EMPLOYEES. En este ejemplo, la columna del número de employee_id está designada como clave primaria. Una clave primaria debe contener un valor y el valor debe ser único.
3. Una columna que no es un valor clave. Una columna representa un tipo de dato en una tabla; en este ejemplo, los datos son los salarios de todos los empleados. El orden de las columnas no es importante al almacenar datos; se puede especificar el orden de la columna al recuperar los datos.
4. Una columna o atributo que contiene el valor de clave ajena, que también es una clave ajena. Una clave ajena es una columna que define cómo se relacionan las tablas entre sí. Una clave ajena hace referencia a una clave primaria o única en la misma tabla o en otra tabla. En el ejemplo, DEPARTMENT_ID identifica de forma única un departamento en la tabla DEPARTMENTS.
5. Un campo puede estar en la intersección de una fila y una columna. Solo puede contener un valor.
6. Un campo puede no contener ningún valor. Esto se denomina valor nulo. En la tabla EMPLOYEES, solo aquellos empleados que tienen el rol de representante de ventas tienen un valor en el campo COMMISSION_PCT (comisiones).

Relación de varias tablas

- Cada fila de datos de una tabla se puede identificar de forma única mediante una clave primaria.
- Puede relacionar datos de forma lógica desde varias tablas mediante claves ajenas.

Nombre de la tabla: **EMPLOYEES**

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	DEPARTMENT_ID
100	Steven	King	90
101	Neena	Kochhar	90
102	Lex	De Haan	90
103	Alexander	Hunold	60
104	Bruce	Ernst	60
105	David	Austin	60
106	Valli	Pataballa	60
107	Diana	Lorentz	60

...

Clave primaria

Nombre 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

...

Clave ajena Clave primaria

Debido a que los datos sobre las diferentes entidades se almacenan en diferentes tablas, puede que tenga que combinar dos o más tablas para responder a una pregunta determinada. Por ejemplo, puede que desee conocer la ubicación del departamento en el que trabaja un empleado. En este caso, necesita información de la tabla EMPLOYEES (que contiene datos sobre los empleados) y la tabla DEPARTMENTS (que contiene información sobre los departamentos). Con RDBMS, puede relacionar los datos de una tabla con los datos de otra utilizando las claves ajenas. Una clave ajena es una columna (o un juego de columnas) que hace referencia a una clave primaria en la misma tabla o en otra tabla.

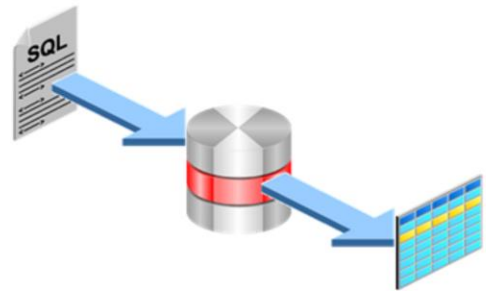
Tiene la posibilidad de relacionar datos de una tabla con datos de otra tabla para organizar la información en unidades separadas y gestionables. Los datos de empleado se pueden mantener, de forma lógica, separados de los datos de departamento almacenándolos en una tabla independiente.

¿Qué es SQL?

- En una base de datos relacional, no es necesario especificar la ruta de acceso a las tablas y tampoco es necesario saber cómo se organizan los datos de forma física.
- Para acceder a la base de datos, ejecute una sentencia SQL, que es el lenguaje estándar de ANSI (American National Standards Institute) para el funcionamiento las de bases de datos relacionales.
- SQL también es compatible con el estándar ISO (SQL 1999).

¿Qué es SQL?

- El lenguaje de consulta estructurado (SQL) es el lenguaje declarativo basado en juegos utilizado para acceder a los datos de una base de datos Oracle.
- SQL proporciona una interfaz para una base de datos relacional y proporciona las sentencias que ayudan a trabajar con la base de datos.
- SQL es:
 - Eficiente, fácil de aprender y de utilizar
 - Funcionalidad completa (con SQL, puede definir, recuperar y manipular datos en las tablas).



Funciones de SQL

- Creación, sustitución, modificación y borrado de objetos de base de datos
- Inserción, actualización y supresión de filas en una tabla
- Consulta de datos almacenados en la base de datos
- Control de acceso a la base de datos y los objetos de base de datos
- Garantía de integridad y consistencia de la base de datos



SQL ofrece ventajas para todos los tipos de usuarios:

- Programadores de aplicaciones
- Administradores de base de datos
- Gestores
- Usuarios finales

SQL procesa los juegos de datos como grupos en lugar de como unidades individuales. Permite a los usuarios:

- Acceder y describir los datos almacenados en la base de datos
- Definir los datos almacenados en la base de datos y manipular esos datos

Tipos de comandos SQL

- **DDL** (Lenguaje de definición de datos):
Define estructuras de bases de datos
- **DML** (Lenguaje de manipulación de datos):
Manipula los datos (INSERT, UPDATE, DELETE)
- **DQL** (Lenguaje de consulta de datos):
Selecciona datos (SELECT)
- **DCL** (Lenguaje de control de datos):
Controla el acceso de usuario
- **TCL** (Lenguaje de control transaccional):
Gestiona las transacciones de bases de datos

Procesamiento SQL

- Fases del procesamiento SQL



* Consulte las definiciones de las fases en las notas

- **Análisis SQL:** La primera etapa del procesamiento SQL es el análisis, que implica la separación de partes de una sentencia SQL en una estructura de datos que puedan procesar otras rutinas. La base de datos analiza una sentencia cuando lo indica la aplicación, lo que significa que solo la aplicación, no solo la base de datos en sí, puede reducir el número de análisis. Durante el análisis, la base de datos realiza una comprobación de la sintaxis, una comprobación de la semántica y una comprobación del pool compartido.
- **Optimización SQL:** Es el proceso de selección del medio más eficaz de ejecutar una sentencia SQL. La base de datos optimiza las consultas basándose en las estadísticas recopiladas sobre los datos reales a los que se está accediendo. El optimizador utiliza el número de filas, el tamaño del juego de datos y otros factores para generar posibles planes de ejecución, asignando un costo numérico a cada plan. La base de datos utiliza el plan con el costo más reducido.
- **Generación de origen de fila SQL:** El generador de origen de fila es el software que recibe el plan de ejecución óptimo desde el optimizador y produce un plan iterativo, denominado plan de consulta, que puede utilizar el resto de la base de datos. El plan iterativo es un programa binario que, cuando se ejecuta, genera el juego de resultados. El plan de consulta adopta la forma de una combinación de pasos. Cada paso devuelve un juego de filas. Las filas de este juego se utilizan en el siguiente paso o, en el último paso, se devuelven a la aplicación mediante la emisión de una sentencia SQL.
- **Ejecución de SQL:** El motor SQL ejecuta cada origen de fila del árbol producido por el generador de orígenes de fila.

Acceso a los datos en el servidor de Oracle Database

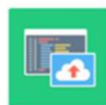
Herramientas especializadas instaladas en las computadoras de los usuarios les permiten acceder a los datos en el servidor de Oracle Database. Estas herramientas se denominan clientes y se utilizan para enviar instrucciones SQL (comandos) al servidor. Tres de estas herramientas son:



Oracle Application Express



Application Builder



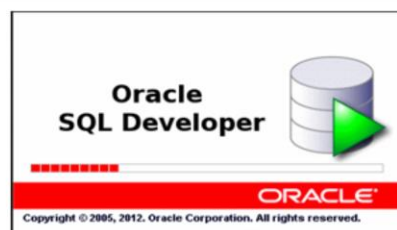
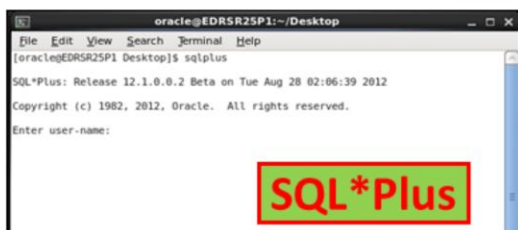
SQL Workshop



Team Development



Packaged Apps



* Nota: En esta clase utilizaremos Oracle Application Express.

Escenario de caso: Necesidad de extraer datos



Profesor

Sean, ya debería comprender la necesidad de utilizar SQL para acceder a los datos de una base de datos relacional. ¿Puede pensar en los casos en los que sería necesario recuperar datos de una tabla de base de datos?

Claro. Déjeme pensar en algunos casos de uso.



Alumno

Casos de uso

Piense y analice el uso de las bases de datos para almacenar datos y producir información en estas y en otras áreas.



Hospitales



Venta al por menor



Aeropuertos



Escuelas



DFo 6-2
Lenguaje de consulta estructurado (SQL)

Copyright © 2017, Oracle y/o sus filiales. Todos los derechos reservados. 16

La necesidad de almacenar y manipular datos se produce en distintos sectores. Entre los ejemplos se incluyen servicios de salud, transporte, venta al por menor y educación.

- Los datos de los pacientes se almacenan en un hospital y se denominan registros médicos.
- Las aerolíneas almacenan datos para que los pasajeros puedan adquirir sus billetes y realizar la facturación en línea.
- Las escuelas mantienen registros de los alumnos y los profesores para realizar transcripciones.

Resumen

En esta lección, debe haber aprendido a hacer lo siguiente:

- Describir cómo se organizan los datos en una base de datos relacional
- Explicar las distintas terminologías de bases de datos relacionales
- Definir el lenguaje de consulta estructurado y sus funciones
- Describir cómo se produce el procesamiento SQL
- Identificar las herramientas que se utilizan para acceder a la base de datos relacional





ACADEMY