Introducción a SQL

Mg. Hilmar Hinojosa Lazo

Base de Datos

2024-2

Comandos para gestión de tablas

Comando para crear una tabla

CREATE TABLE nombre de la tabla
(columnas);

Ejemplo:

```
CREATE TABLE DISTRITO
( discod  number primary key,
  disnom varchar2(20) not null );
```

Comandos para gestión de tablas

Ejemplo:

```
CREATE TABLE PERSONA
( percod  number primary key,
  perapp varchar2(20) not null,
  perapm varchar2(20) not null,
  pernom varchar2(20) not null,
  persex varchar2(1) not null,
  perfna date ) ;
```

Comandos para gestión de tablas

Columna con restricción UNIQUE

```
CREATE TABLE PERSONA
( percod  number primary key,
  perapp varchar2(20) not null,
  perapm varchar2(20) not null,
  pernom varchar2(20) not null,
  persex varchar2(1) not null,
  perfna date,
  percor varchar2(50),
  constraint u_percor UNIQUE( percor) ) ;
```

Comandos para gestión de tablas

Columna con restricción CHECK

```
CREATE TABLE PERSONA
( percod  number primary key,
  perapp varchar2(20) not null,
  perapm varchar2(20) not null,
  pernom varchar2(20) not null,
  persex varchar2(1) not null,
  perfna date,
  perpeso number,
  constraint ch_perpeso CHECK (perpeso > 0) ) ;
```

Comandos para gestión de tablas

Columna con restricción FOREIGN KEY

```
CREATE TABLE PERSONA
( percod  number primary key,
  perapp varchar2(20) not null,
  perapm varchar2(20) not null,
  pernom varchar2(20) not null,
  persex varchar2(1) not null,
  perfna date,
  discod number,
  constraint fk_discod foreign key (discod) references  DISTRITO (discod) );
```

Comandos para gestión de tablas

Columna con restricción FOREIGN KEY con ON DELETE CASCADE

```
CREATE TABLE PERSONA
( percod  number primary key,
  perapp varchar2(20) not null,
  perapm varchar2(20) not null,
  pernom varchar2(20) not null,
  persex varchar2(1) not null,
  perfna date,
  discod number,
  constraint fk_discod foreign key (discod) references  DISTRITO (discod)
  on delete cascade );
```

Comandos para gestión de tablas

Columna con restricción FOREIGN KEY con ON DELETE SET NULL

```
CREATE TABLE PERSONA
( percod  number primary key,
  perapp varchar2(20) not null,
  perapm varchar2(20) not null,
  pernom varchar2(20) not null,
  persex varchar2(1) not null,
  perfna date,
  discod number,
  constraint fk_discod  foreign key (discod) references  DISTRITO
 (discod) on delete set null );
```

Comandos para gestión de tablas

```
CREATE TABLE PROFESION
( procod  number primary key,
  pronom varchar2(20) not null );

CREATE TABLE TELEFONO
( percod  number,
  pertel varchar2(10),
  primary key ( percod, pertel ),
  constraint fk_percod foreign key (percod) references PERSONA ( percod) );
```

Comandos para gestión de tablas

```
CREATE TABLE PERSONA_PROFESION
( percod number,
  procod  number,
  añotit number,
  primary key ( percod, procod ),
  constraint fk_percod foreign key (percod) references PERSONA ( percod),
  constraint fk_procod foreign key (procod) references PROFESION( procod) );
```

Comandos para gestión de tablas

Comando para ver la estructura de una tabla (las columnas que tiene)

DESCRIBE nombre de tabla;

Ejemplo:

DESCRIBE PERSONA;

Comandos para gestión de tablas

Comando para agregar una nueva columna a una tabla

ALTER TABLE nombre de tabla
ADD nombre de columna tipo restricción;

Ejemplo:

ALTER TABLE PERSONA
ADD pertalla number;

ALTER TABLE PERSONA
ADD año number;

Comandos para gestión de tablas

Comando para modificar una columna de una tabla

ALTER TABLE nombre de tabla
MODIFY nombre de columna tipo;

Ejemplo:

ALTER TABLE PERSONA
MODIFY pernom varchar2(30);

Comandos para gestión de tablas

Comando para cambiar el nombre de una columna de una tabla

ALTER TABLE nombre de tabla
RENAME nombre actual de columna TO nuevo nombre;

Ejemplo:

ALTER TABLE PERSONA
RENAME percor TO permail;

Comandos para gestión de tablas

Comando para eliminar una columna de una tabla

ALTER TABLE nombre de tabla
DROP COLUMN nombre de columna;

Ejemplo:

ALTER TABLE PERSONA
DROP COLUMN perpeso;

Comandos para gestión de tablas

Comando para agregar una restricción en una tabla

ALTER TABLE nombre de tabla
ADD CONSTRAINT restricción;

Ejemplo:

ALTER TABLE PERSONA
ADD CONSTRAINT ch_pertalla CHECK (pertalla > 1.65);

Comandos para gestión de tablas

Comando para eliminar una restricción en una tabla

ALTER TABLE nombre de tabla
DROP CONSTRAINT nombre de restricción;

Ejemplo:

ALTER TABLE PERSONA
DROP CONSTRAINT ch_pertalla;

Comandos para gestión de tablas

Comando para eliminar una tabla

DROP TABLE nombre de tabla CASCADE CONSTRAINTS;

Ejemplo:

DROP TABLE DISTRITO CASCADE CONSTRAINTS;

Comandos para gestión de tablas

Comando para renombrar una tabla

RENAME nombre de tabla TO nuevo nombre;

Ejemplo:

RENAME PERSONA TO PEOPLE;

Sentencia INSERT

- Se utiliza para insertar nuevas filas en una tabla
- Sintaxis:

```
INSERT  INTO  tabla
VALUES  ( valor1, valor2, ..., valorN )
```

```
INSERT INTO  tabla
( col1, col2, ..., colN )
VALUES  ( valor1, valor2, ..., valorN )
```

Sentencia INSERT

Inserción de datos para todas las columnas

```
INSERT INTO  persona
( percod, perapp, perapm, pernom, persex, perfna, perpeso,
pertalla, percor, discod, año )
VALUES  ( 52, 'Valdez', 'Durand', 'Dora' , 'F',
                to_date('02-07-1991', 'DD-MM-YYYY'), 59,
                1.62, 'dvaldez@gmail.com', 5, 2 )
```

Sentencia INSERT

Inserción de datos para algunas columnas

```
INSERT INTO  persona
( percod, perapp, perapm, pernom, persex, perpeso )
VALUES  ( 53, 'Miranda', ‘Flores', ‘Roberto' , ‘M',  73 );
```

Sentencia UPDATE

- Se utiliza para modificar valores en una tabla
- Sintaxis:

  UPDATE tabla
  SET col1 = valor1, col2 = valor2, … colN = valorN
  WHERE condición
- La cláusula SET establece los nuevos valores para las columnas indicadas
- La cláusula WHERE sirve para seleccionar las filas que se quiere modificar
- Si se omite WHERE se modificará los valores en todas las filas de la tabla

Sentencia UPDATE

Asignar la fecha de nacimiento 17 de mayo de 1999 y el sueldo 3400 soles a la persona cuyo código es 35

```
UPDATE  persona
SET  perfna =  TO_DATE('17-05-1999', 'DD-MM-YYYY'),
       persue = 3400
WHERE  percod = 35
```

Sentencia UPDATE

Modificar el nombre de Victor Segovia Tudela debiendo ser ahora Victoria

```
UPDATE  persona
SET  pernom = 'Victoria' ,  persex = 'F'
WHERE  pernom = 'Victor' AND perapp = 'Segovia'
            AND perapm = 'Tudela'
```

Sentencia UPDATE

Incrementar en 100 soles el sueldo de todas las mujeres

```
UPDATE  persona
SET  persue = persue + 100
WHERE  persex = 'F'
```

Sentencia DELETE

- Se utiliza para eliminar filas de una tabla
- Sintaxis:

  DELETE FROM tabla

  WHERE condición

- La cláusula WHERE sirve para seleccionar las filas que se quiere eliminar
- Si se omite WHERE se eliminarán todas las filas de la tabla

Sentencia DELETE

Eliminar todas las filas de la tabla PERSONA

```
DELETE  FROM  persona
```

Eliminar los datos de la persona con código 27

```
DELETE FROM  persona
WHERE  percod = 27
```

Eliminar los datos de las personas que ganan más de 4000 soles

```
DELETE FROM  persona
WHERE  persue > 4000
```

Sentencia SELECT

- Es una de las sentencias SQL más importantes, ya que permite realizar consultas sobre los datos almacenados en la base de datos.
- Sintaxis:

SELECT * ó Lista de columnas

FROM tablas

WHERE condición

ORDER BY columnas

Sentencia SELECT

- Se usa * cuando se quiere visualizar todas las columnas.
- La cláusula FROM permite indicar las tablas que serán consultadas.
- La cláusula WHERE permite especificar la condición que deben cumplir las filas que serán consideradas en la consulta.
- La cláusula ORDER BY permite ordenar el resultado de la consulta en base a una o varias columnas.

Sentencia SELECT

Mostrar todos los datos de todas las personas

```
SELECT * FROM persona
```

Mostrar todos los datos de las mujeres

```
SELECT *
FROM persona
WHERE  persex = 'F'
```

Sentencia SELECT

Mostrar todos los datos de las personas que ganan más de 3000 soles ordenados por apellido paterno

```
SELECT  *
FROM  persona
WHERE  persue > 3000
ORDER BY  perapp
```

Sentencia SELECT

Mostrar apellidos y nombres de las personas. Cada columna debe tener un título

```
SELECT  perapp AS Apellido_Paterno,
        perapm AS Apellido_Materno,
        pernom AS Nombre
FROM  persona
```

Sentencia SELECT

Mostrar todos los sueldos. Si hay sueldos repetidos solo deben mostrarse una sola vez. Los sueldos deben mostrarse de mayor a menor.

```
SELECT  DISTINCT persue

FROM  persona

ORDER BY persue DESC
```

Operadores lógicos AND y OR

- Las condiciones de la cláusula WHERE se pueden combinar usando los operadores AND y OR.

- Los operadores AND y OR se utilizan para filtrar registros en función de más de una condición.

Operador AND

- El operador AND muestra una fila si todas las condiciones separadas por AND son verdaderas.

Mostrar todos los datos de las mujeres que ganan menos de 2500 soles

```
SELECT *
FROM persona
WHERE persex = 'F' AND persue < 2500
```

Operador OR

- El operador OR muestra una fila si alguna de las condiciones separadas por OR es verdadera.

Mostrar apellidos y nombre de todos los que se llaman Jorge o Juan

```
SELECT  perapp, perapm, pernom
FROM  persona
WHERE  pernom = 'Juan'  OR  pernom = 'Jorge'
```

Operador Between

- El operador BETWEEN selecciona valores dentro de un rango dado.
- Los valores pueden ser números, texto o fechas.

Mostrar apellidos, nombre, fecha de nacimiento y sueldo de las personas que ganan desde 2000 hasta 4000 soles

```
SELECT  perapp, perapm, pernom, perfna, persue
FROM  persona
WHERE  persue  BETWEEN 2000 AND 4000
```

Operador IN

- El operador IN permite especificar un conjunto de valores en una cláusula WHERE evitando el uso de múltiples condiciones OR.

Mostrar apellidos, nombre y sueldo de las mujeres cuyo apellido paterno sea Torres, Cuba, Diaz o Vargas

```
SELECT  perapp, perapm, pernom, persue
FROM  persona
WHERE  persex = 'F'  AND
            perapp IN ( 'Torres', 'Cuba',  'Diaz', 'Vargas' )
```

Operador ||

- El operador || permite concatenar o juntar varios textos.

Mostrar apellidos y nombre de las personas, todo junto en una sola columna. Ordenar por apellido paterno.

```
SELECT  perapp || ' ' || perapm || ' ' || ' ' pernom
FROM  persona
ORDER BY perapp
```

Operador LIKE

- El operador LIKE se usa en una cláusula WHERE para buscar un patrón específico en una columna.
- Hay dos comodines que se usan con frecuencia junto con el operador LIKE:
  - % representa cero, uno o varios caracteres
  - _ representa un solo carácter

Operador LIKE

Mostrar apellidos y nombre de las personas cuyo apellido paterno empieza con la letra F

```
SELECT  perapp, perapm, pernom
FROM  persona
WHERE  perapp  LIKE  'F%'
```

Mostrar apellidos y nombre de las personas cuyo nombre tiene una a en la penúltima posición

```
SELECT  perapp, perapm, pernom
FROM  persona
WHERE  pernom  LIKE  '%a_'
```

Funciones para el manejo de fechas

- **extract(year from fecha):** retorna el año de una fecha
- **extract(month from fecha):** retorna el mes de una fecha
- **extract(day from fecha):** retorna el día de una fecha

Funciones para el manejo de fechas

Mostrar apellidos, nombre y fecha de nacimiento de las personas que nacieron el mismo día de navidad después del año 1996

```
SELECT  perapp, perapm, pernom, perfna
FROM  persona
WHERE  extract( day from perfna ) = 25  AND
            extract( month from perfna ) = 12 AND
            extract( year from perfna ) > 1996
```

Funciones para el manejo de fechas

Mostrar apellidos, nombre y fecha de nacimiento de las personas que nacieron en febrero, abril, junio, agosto o noviembre, ordenado por fecha de nacimiento

```
SELECT  perapp, perapm, pernom, perfna
FROM  persona
WHERE  extract( month from perfna) in ( 2, 4, 6, 8, 11 )
ORDER BY perfna
```

Consultas multitabla

Mostrar apellidos y nombres de las personas, el distrito donde viven y desde qué año viven en él

```
SELECT  perapp, perapm, pernom, disnom, año
FROM  persona P, distrito D
WHERE  P.discod = D.discod
ORDER  BY perapp
```

Consultas multitabla

Mostrar apellidos y nombres de los varones que viven en Surco

```
SELECT  perapp, perapm, pernom, disnom

FROM  persona P, distrito D

WHERE  P.discod = D.discod  AND  persex = 'M'

            AND disnom = 'Surco'

ORDER BY  perapp
```

Consultas multitabla

Mostrar apellidos, nombre y teléfonos de las personas cuyos teléfonos terminan en 5

```
SELECT  perapp, perapm, pernom, pertel

FROM  persona P, telefono T

WHERE  P.percod = T.percod  AND  pertel like '%5'

ORDER BY  perapp
```

Consultas multitabla

Mostrar apellidos, nombre y profesión de todas las personas

```
SELECT  perapp, perapm, pernom, pronom
FROM  persona P, personaprofesion PP, profesion R
WHERE  P.percod = PP.percod
            AND PP.procod = R.procod
ORDER BY  perapp
```

Consultas multitabla

Mostrar apellidos y nombre de todos los ingenieros que viven en La Molina

```
SELECT  perapp, perapm, pernom, pronom, disnom

FROM  persona P,  personaprofesion PP,  profesion R, distrito D

WHERE  P.percod = PP.percod AND  PP.procod = R.procod AND

            P.discod = D.discod AND disnom = 'La Molina' AND

            pronom like 'Ingeniero%'

ORDER  BY perapp
```