

Práctica 1. Bases de datos

Relaciones entre tablas

- **Filas:** instancias o transacciones de la base de datos
 - Cada fila de datos en una tabla está identificada unívocamente por una clave primaria (Primary Key: PK)
 - Se pueden relacionar múltiples tablas usando claves foráneas (Foreign Key: FK)
- **Columnas:** tipo de dato o atributo que identifica el objeto a contener

| Nombre Tabla: EMP | | | | Nombre Tabla : DEPT | | |
|-------------------|-------|-----------|--------|---------------------|------------|----------|
| EMPNO | ENAME | JOB | DEPTNO | DEPTNO | DNAME | LOC |
| 7839 | KING | PRESIDENT | 10 | 10 | ACCOUNTING | NEW YORK |
| 7698 | BLAKE | MANAGER | 30 | 20 | RESEARCH | DALLAS |
| 7782 | CLARK | MANAGER | 10 | 30 | SALES | CHICAGO |
| 7566 | JONES | MANAGER | 20 | 40 | OPERATIONS | BOSTON |

Comandos SQL

SQL: Structured Query Language o Lenguaje de Consultas Estructurado. Es el lenguaje que permite la comunicación con el Sistema Gestor de Bases de Datos

| | |
|---|-----------------------------|
| SELECT | Recuperación de Datos |
| INSERT UPDATE DELETE | Manipulación de Datos (DML) |
| CREATE ALTER DROP RENAME TRUNCATE | Definición de Datos (DDL) |
| COMMIT ROLLBACK SAVEPOINT | Control de Transacciones |
| GRANT REVOKE | Control de Datos (DCL) |

Recuperación de datos

- Obtener todas las columnas de una tabla:

```
SELECT * FROM {table_name};
```

- Los nombres de las columnas se usan como cabeceras de presentación. Si éste nombre es demasiado corto, largo, o críptico, se puede poner un alias en la consulta:

```
SELECT {column_name} "alias" FROM {table_name};  
SELECT {column_name} as "nuevo alias" FROM {table_name};
```

- También se puede recuperar todas las tablas de una base de datos:

```
SELECT table_name FROM user_tables;
```

Recuperación de datos

- La recuperación de datos se realiza mediante la sentencia **SELECT**, la cual consta de cuatro partes:
- Partes obligatorias:
 - La cláusula **SELECT** seguida de la descripción de lo que se desea ver, los nombres de las columnas a seleccionar.
 - La cláusula **FROM** seguida de la especificación de las tablas de las que se han de obtener los datos.
- Partes opcionales:
 - La cláusula **WHERE** seguida por un criterio de selección, una condición.
 - La cláusula **ORDER BY** seguida por el criterio de ordenación.

```
SELECT { * | {columna,}+ }  
FROM {tabla,}+  
[WHERE condición]  
[ORDER BY {expresiónColumna [ASC | DESC],}+];
```

Recuperación de datos

- Obtener todas las columnas de una tabla:

```
SELECT * FROM {table_name};
```

- Los nombres de las columnas se usan como cabeceras de presentación. Si éste nombre es demasiado corto, largo, o críptico, se puede poner un alias en la consulta:

```
SELECT {column_name} "alias" FROM {table_name};  
SELECT {column_name} as "nuevo alias" FROM {table_name};
```

- También se puede recuperar todas las tablas de una base de datos:

```
SELECT table_name FROM user_tables;
```

Recuperación de datos

- Operadores de comparación:

| Operador | Operación | Ejemplo |
|--------------|---|--|
| = | Igualdad | select * from emp where cod_dep = 100; |
| !=, <>, ^= | Desigualdad | select * from emp where cod_dep != 100; |
| < | Menor que | select * from emp where cod_dep < 200; |
| > | Mayor que | select * from emp where cod_dep > 200; |
| <= | Menor o igual que | select * from emp where cod_dep <= 200; |
| >= | Mayor o igual que | select * from emp where cod_dep >= 200; |
| in | Igual a cualquiera de los miembros entre paréntesis | select * from emp where cod_dep in (100, 300); |
| not in | Distinto a cualquiera de los miembros entre paréntesis | select * from emp where cod_dep not in (200); |
| between | Contenido en el rango | select * from emp where cod_emp between 100 and 199; |
| not between | Fuera del rango | select * from emp where cod_emp not between 100 and 199; |
| like '_abc%' | Contiene la cadena 'abc' a partir del segundo carácter y luego cualquier cadena de caracteres | select * from emp where nombre like 'Ma%'; |

- Operadores aritméticos:

| Operador | Operación | Ejemplo |
|----------|-----------|---|
| + | Suma | select nombre, salario+comision from emp where oficio='VENDEDOR'; |
| - | Resta | select nombre from emp where sysdate-fecha_alta > 365; |
| * | Producto | select nombre, salario*12 from emp; |
| / | División | select nombre, salario/31 from emp; |

Recuperación de datos

- Operadores de cadenas de caracteres:

| <i>Operador</i> | <i>Operación</i> | <i>Ejemplo</i> |
|-----------------|------------------|--|
| | Concatenación | <code>select nombre oficio from emp;</code> |

Manipulación de datos

- Inserción. Permite añadir filas en las tablas indicadas.

```
INSERT INTO tabla [(columna,)*] VALUES (expresión,+);
```

- Actualización. Modifica la información almacenada en las tablas.

```
UPDATE tabla SET {columna = expresión,+} [WHERE condición];
```

- Borrado. Elimina las filas que cumplan una determinada condición.

```
DELETE FROM tabla [WHERE condición];
```

Definición de datos

- Creación de una tabla:

```
CREATE TABLE [esquema.]tabla ({columna tipoColumna [NOT NULL],}+,  
    {CONSTRAINT nombreRestricción  
        {UNIQUE ([column],}+)|  
        DEFAULT expresion|  
        CHECK (condicion)|  
        PRIMARY KEY ([column],}+)|  
        FOREIGN KEY (column) REFERENCES tabla(columna)},}*)
```

- Tipo de columnas:

| <i>Tipo de Dato</i> | <i>Descripción</i> |
|----------------------------|--|
| VARCHAR2(tamaño) | Almacena datos de tipo carácter alfanumérico de longitud variable, con un tamaño máximo de 2.000. |
| CHAR(tamaño) | Almacena datos de tipo carácter alfanumérico de longitud fija, con un tamaño máximo de 255. |
| LONG | Almacena datos de tipo carácter alfanumérico de longitud variable con un tamaño máximo de hasta 2 Gb. |
| NUMBER(dig,dec) | Almacena datos numéricos de dig dígitos, de los cuales dec son decimales. El tamaño máximo es de 38 dígitos. |
| DATE | Almacena fechas desde el 1-Ene-4712 AC hasta el 31-Dic-4712 DC. |
| RAW(tamaño) | Almacena datos de longitud variable, con un tamaño máximo de 255 bytes. |
| LONG RAW | Almacena datos de longitud variable, con un tamaño máximo de 2 Gb. |

Definición de datos

- Restricciones. Se imponen para asegurarnos que los datos cumplen una serie de condiciones predefinidas para cada tabla. Ayudan a conseguir la integridad de referencia:
 - **NOT NULL.** Establece la obligatoriedad de que esta columna tenga un valor no nulo.
 - **UNIQUE.** Evita valores repetidos en una columna, admitiendo valores nulos.
 - **DEFAULT.** Establece un valor por defecto para esa columna, si no se le asigna ninguno.
 - **CHECK.** Comprueba que se cumpla una condición determinada al rellenar esa columna. Esta condición sólo debe estar construida con columnas de esta misma tabla.
 - **PRIMARY KEY.** Establece el conjunto de columnas que forman la clave primaria de esa tabla.
 - **FOREIGN KEY.** Establece que el contenido de esta columna será uno de los valores contenidos en una columna de otra tabla maestra.

Definición de datos

- Cuando una tabla ya no es útil debe ser borrada:

`DROP TABLE {table_name};`

- Si una columna de esta tabla sirve como clave ajena, se impide la eliminación de la tabla. Se puede solucionar con:

`DROP TABLE {table_name} CASCADE CONSTRAINTS;`

- **Describe:** comando que resulta muy útil cuando queremos conocer la estructura de una tabla, las columnas que la forman y su tipo y restricciones.

`DESCRIBE tabla`