

## Diseño de base de datos

### Unidad 3: Lenguajes de definición y manipulación de datos en SQL

# INFORME DE TALLER: SINTAXIS DDL Y DML

#### Datos de identificación del estudiante

|                  |                         |
|------------------|-------------------------|
| Nombre           | Constanza Fernández     |
| Asignatura       | Diseño de base de datos |
| Sede             | Santiago Centro         |
| Docente          | Carlos Guajardo         |
| Fecha de entrega | 11-11-2018              |

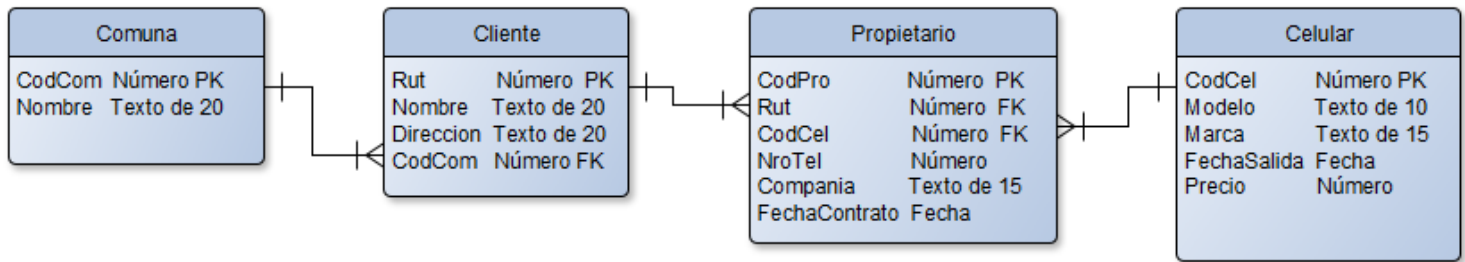
#### Descripción del taller

**Instrucción:** en este espacio deberán describir qué realizarán en este taller y establecer cuál es la finalidad de generar sintaxis DDL y DML.

En este taller se realizara la implementación de bases de datos físicas mediante instrucciones DDL y DML utilizando SQL de Oracle. La finalidad de generar esta sintaxis es comprender el modelo relacional que se muestra en el informe y a partir de esto crear las diferentes tablas que se solicitan, además de entender el tipo de lenguaje que se utiliza en este programa para poder, en el caso que se solicite ingresar, modificar, borrar o actualizar los datos según corresponda.

#### Desarrollo de sintaxis DDL y DML

##### Modelo relacional



1. Incluyan instrucciones para agregar la restricción PK al campo **CodCel** de la tabla **Celular**.

```
ALTER TABLE COMUNA
ADD CONSTRAINT PK_CELULAR_CODCEL
PRIMARY KEY (CODCEL);
```

```
SQL> alter table celular
2 add constraint pk_celular_codcel
3 primary key (codcel);
```

Table altered.

```
SQL> desc celular
```

| Name        | Null?    | Type         |
|-------------|----------|--------------|
| CODCEL      | NOT NULL | NUMBER       |
| MODELO      |          | VARCHAR2(10) |
| MARCA       |          | VARCHAR2(15) |
| FECHASALIDA |          | DATE         |
| PRECIO      |          | NUMBER       |

2. Incluyan instrucciones para crear la tabla **Propietario** sin restricciones de integridad.

```
CREATE TABLE PROPIETARIO(CODPRO NUMBER,RUT NUMBER,CODCEL NUMBER, NROTEL NUMBER, COMPANIA
VARCHAR2(15), FECHACONTRATO DATE);
```

```
SQL> create table propietario (codpro number, rut number, codcel number, nrotel number, compania varchar2(15),
fechaContrato date);
```

Table created.

```
SQL> desc propietario
```

| Name          | Null? | Type         |
|---------------|-------|--------------|
| CODPRO        |       | NUMBER       |
| RUT           |       | NUMBER       |
| CODCEL        |       | NUMBER       |
| NROTEL        |       | NUMBER       |
| COMPANIA      |       | VARCHAR2(15) |
| FECHACONTRATO |       | DATE         |

3. Incluyan instrucciones para modificar el campo **NroTel** ahora se llama **Numero** en la tabla **Propietario**.

```
ALTER TABLE PROPIETARIO
RENAME COLUMN NROTEL TO NUMERO;
```

```
SQL> alter table propietario
2 rename column nrotel to numero;
```

Table altered.

```
SQL> desc propietario
```

| Name          | Null? | Type         |
|---------------|-------|--------------|
| -----         | ----- | -----        |
| CODPRO        |       | NUMBER       |
| RUT           |       | NUMBER       |
| CODCEL        |       | NUMBER       |
| NUMERO        |       | NUMBER       |
| COMPANIA      |       | VARCHAR2(15) |
| FECHACONTRATO |       | DATE         |

4. Incluyan instrucciones para eliminar el campo **FechaSalida** de la tabla **Celular**.

```
ALTER TABLE CELULAR
DROP COLUMN FECHASALIDA;
```

```
SQL> alter table celular
2 drop column fechaSalida;
```

Table altered.

```
SQL> desc celular
```

| Name   | Null?    | Type         |
|--------|----------|--------------|
| -----  | -----    | -----        |
| CODCEL | NOT NULL | NUMBER       |
| MODELO |          | VARCHAR2(10) |
| MARCA  |          | VARCHAR2(15) |
| PRECIO |          | NUMBER       |

5. Incluyan instrucciones para cambiar el nombre de la tabla **Celular** a **Movil**.

```
ALTER TABLE CELULAR
RENAME TO MOVIL;
```

```
SQL> alter table celular rename to movil;
```

Table altered.

```
SQL> desc movil
```

| Name   | Null?    | Type         |
|--------|----------|--------------|
| -----  | -----    | -----        |
| CODCEL | NOT NULL | NUMBER       |
| MODELO |          | VARCHAR2(10) |
| MARCA  |          | VARCHAR2(15) |
| PRECIO |          | NUMBER       |

6. Incluyan instrucciones para cambiar el campo **compania**, ahora debe ser un texto de 20 y no puede quedar como nulo en la tabla **propietario**.

```
ALTER TABLE PROPIETARIO
MODIFY (COMPANIA VARCHAR2(20) NOT NULL);
```

```
SQL> alter table propietario
2 modify (compania varchar2(20) not null);
```

Table altered.

```
SQL> desc propietario
```

| Name          | Null?    | Type         |
|---------------|----------|--------------|
| -----         | -----    | -----        |
| CODPRO        |          | NUMBER       |
| RUT           |          | NUMBER       |
| CODCEL        |          | NUMBER       |
| NUMERO        |          | NUMBER       |
| COMPANIA      | NOT NULL | VARCHAR2(20) |
| FECHACONTRATO |          | DATE         |

```
SQL>
```

7. Incluyan instrucciones para crear la FK del campo **codcom** en la tabla **Cliente**.

```
ALTER TABLE CLIENTE
ADD CONSTRAINT FK_CLIENTE_CODCOM
FOREIGN KEY (CODCOM) REFERENCES COMUNA (CODCOM);
```

```
SQL> alter table cliente
2 add constraint fk_cliente_codcom
3 foreign key(codcom) references comuna (codcom);
```

Table altered.

```
SQL> desc cliente
```

| Name      | Null? | Type         |
|-----------|-------|--------------|
| -----     | ----- | -----        |
| RUT       |       | NUMBER       |
| NOMBRE    |       | VARCHAR2(20) |
| DIRECCION |       | VARCHAR2(20) |
| CODCOM    |       | NUMBER       |

8. Incluyan instrucciones para eliminar a todos los propietarios de la compañía **Movistar**.

```
DELETE FROM PROPIETARIO
```

WHERE COMPANIA = 'MOVISTAR';

```
SQL> select * from propietario;
```

| CODPRO | RUT | CODCEL | NUMERO | COMPANIA | FECHACON |
|--------|-----|--------|--------|----------|----------|
|        |     |        |        | movistar |          |
|        |     |        |        | entel    |          |

```
SQL> delete from propietario
2 where compania = 'movistar';
```

1 row deleted.

```
SQL> select * from propietario;
```

| CODPRO | RUT | CODCEL | NUMERO | COMPANIA | FECHACON |
|--------|-----|--------|--------|----------|----------|
|        |     |        |        | entel    |          |

9. Incluyan instrucciones para insertar al **Ciente** Rut 20230245-5 de nombre Juan que vive en la calle Omega 443 de la comuna Ancud (Ancud es código 11).

```
INSERT INTO CLIENTE(RUT, NOMBRE, DIRECCION, COMUNA)
```

```
VALUES(20230245, 'JUAN', CALLE OMEGA 443, 11);
```

```
SQL> insert into cliente (rut, nombre, direccion, codcom) values (20230245, 'Juan', 'calle omega 443', 11);
```

1 row created.

```
SQL> desc cliente
```

| Name      | Null? | Type         |
|-----------|-------|--------------|
| RUT       |       | NUMBER       |
| NOMBRE    |       | VARCHAR2(20) |
| DIRECCION |       | VARCHAR2(20) |
| CODCOM    |       | NUMBER       |

```
SQL> select * from cliente
2 ;
```

| RUT      | NOMBRE | DIRECCION       | CODCOM |
|----------|--------|-----------------|--------|
| 20230245 | Juan   | calle omega 443 | 11     |

10. Incluyan instrucciones para disminuir un 15% los precios de los teléfonos marca 'Samsung'.

UPDATE MOVIL

```
SET PRECIO = PRECIO*0.85
WHERE MARCA = 'SAMSUNG';
```

```
SQL> update movil
  2  set precio = precio*0.85
  3  where marca = 'samsung';

0 rows updated.
```

| CODCEL | MODELO | MARCA   | PRECIO |
|--------|--------|---------|--------|
| 123456 | s9     | samsung | 5000   |

```
SQL> update movil
  2  set precio = precio*0.85
  3  where marca = 'samsung';

1 row updated.
```

```
SQL> select * from movil;
```

| CODCEL | MODELO | MARCA   | PRECIO |
|--------|--------|---------|--------|
| 123456 | s9     | samsung | 4250   |

## Reflexiones finales

**Instrucción:** establezcan qué aprendieron en este taller y qué les pareció más fácil y/o difícil. Finalmente, indiquen por qué es importante generar esta sintaxis.

Durante la realización de este taller aprendí a utilizar el lenguaje correcto para la creación de las diferentes tablas que se solicitaban según el modelo relacional, ingresar los datos a cada tabla, crear las PK o FK correspondientes, cambiar el nombre de las tablas o columnas, eliminar campos o columnas de un determinado valor o tipo, insertar valores a las filas y actualizar los datos de estas. Lo que me pareció difícil de este taller fue modificar el nombre de una tabla ya que al hacerlo no estaba utilizando la sintaxis correcta, pero al verificar la información con el material dispuesto en la plataforma pude corregir ese error, y lo más fácil fue crear las tablas guiándome por el modelo relacional. Entiendo que es importante generar esta sintaxis ya que de esta forma se encuentran unificados los criterios y los datos se mantienen almacenados de forma ordenada lo que permite acceder a ellos de forma rápida y precisa.

## Bibliografía

Material Unidad N°3: Lenguaje de definición de datos (S.F). *INACAP Chile*. Obtenido de <https://lms.inacap.cl/course/view.php?id=55449>

Material Unidad N°3: Uso de DDL en Oracle (S.F). *INACAP Chile*. Obtenido de <https://lms.inacap.cl/course/view.php?id=55449>