

# FUNDAMENTOS DE BASES DE DATOS

---

ENERGIA ELECTRICA

ZEYNEP DANIS

Ceuta 2019

---

*Análisis de los requisitos definidos en el enunciado de la práctica.  
Descripción en lenguaje natural de los requisitos codificando cada  
requisito de datos o restricción semántica.*

## DATOS

- PRODUCTOR( ID, Produccion Media, Produccion Maxima, Fecha de Actividad)
- HIDROELECTRICA(Codigo, Ocupacion, Capacidad Maximo, Numero de Turbinas)
- SOLAR(Codigo, Superficie de Paneles, Media Anual Horas de Sol, Tipo)
- NUCLEAR(Codigo, Numero de Reactores, El Volumen de Plutonio, Resudios)
- TERMICA(Codigo, Numero de Hornos, Volumen Carbon, Volumen Emisiones)
- TRANSPORTISTA(ID, Nombre, Matricula)
- SUMINISTRADOR( ID, Pais, Nombre)
- ESTACION PRIMARIA( Codigo, Nombre, Numero de Transformadores)
- REDES DE DISTRIBUCION( Codigo, Numero de Red, Cabecera)
- COMPANIAS ELECTRICAS(ID, Nombre)
- LINEA( Codigo, Numero Secuencial, Longitud)
- SUBESTACION(Codigo)
- ZONAS DE SERVICIO( Codigo, Consumo Medio, Numero de Consumidores)
- PROVINCIA(Codigo, Nombre)

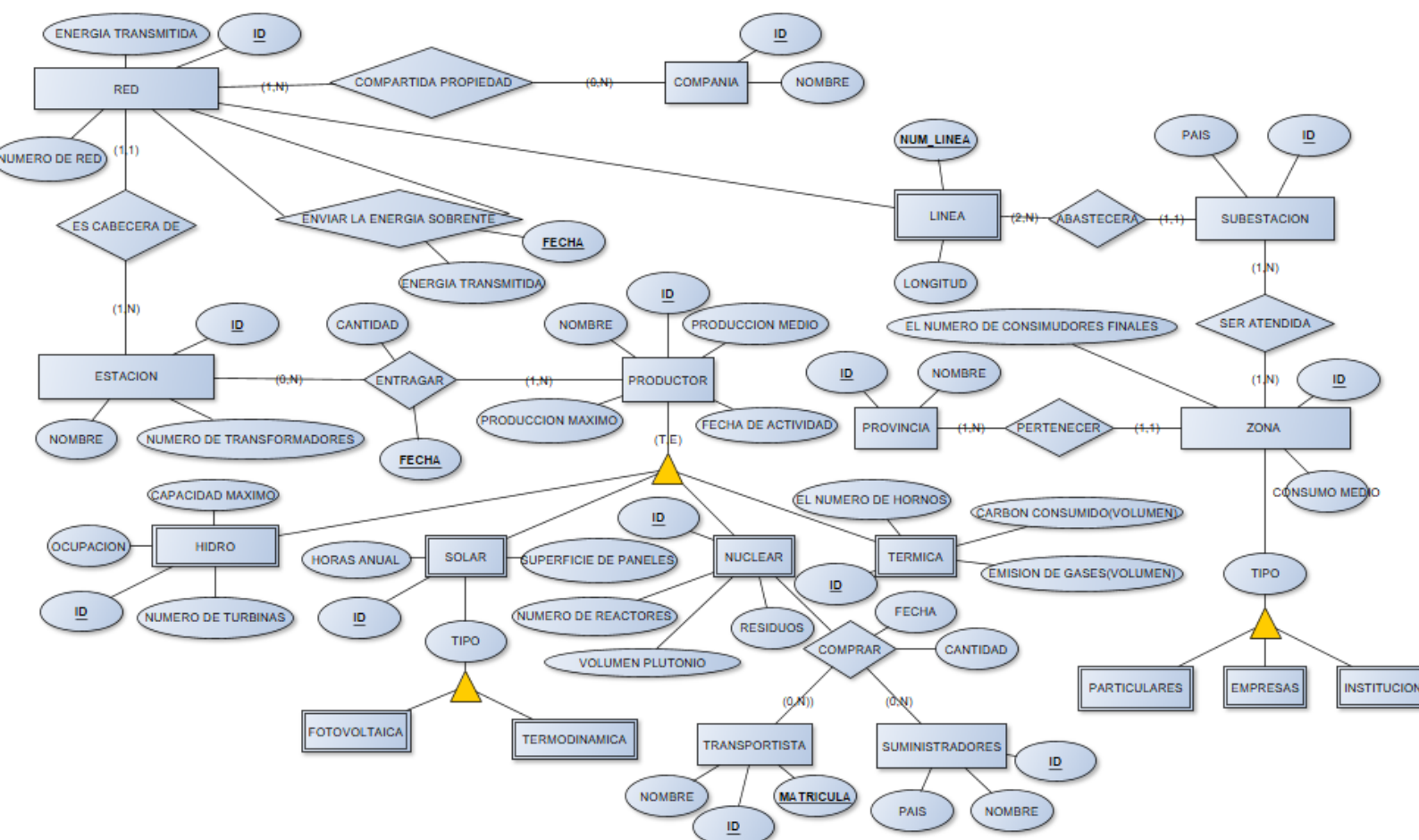
## RESTRICCIONES

- Un Productor produce energia hidroelectrica, solar, nuclear o termica.
- Un Central Solar puede ser de tipo fotovoltaica o termodinamica.
- Control sobre la cantidad de plutonio que compra cada central nuclear a cada uno de sus suministradores.
- Cada compra es entregada por un determinado transportista.
- Un mismo suministrador puede vender a varias centrales nucleares.
- Cada compra puede realizarla un transportista diferente.
- Los productores entregan cada dia la energia producida a una o varias estaciones primarias.
- Los productores entregan su produccion total.
- Cada estacion primaria es cabecera de una o mas redes de distribucion.
- Una red de distribucion solo puede tener una estacion primaria como cabecera.

- La propiedad sobrante de una red puede ser compartida por varias compañías eléctricas.
- La energía sobrante de una red se puede enviar a otra red.
- Se registra el volumen total de energía intercambiada entre dos redes.
- Cada red esta compuesta por líneas.
- Cada línea se identifica por un numero secuencial dentro del numero de red y tiene una determinada longitud.
- La menor de las líneas posibles abastecera al menos a dos subestaciones.
- Las provincias se encuentran divididas en zonas de servicio.
- No puede haber zonas de servicio que pertenezcan a mas de una provincia.
- Cada zona de servicio puede ser atendida por mas de una subestacion.
- Cada zona de servicio contempla tres categorias: particulares, empresas e instituciones.
- De cada categoria se registra el consumo medio y el numero de consumidores finales.




*Obtención de un diseño conceptual mediante diagrama Entidad Relación.*

## MODELO E/R (DISEÑO RELACIONAL)



## *Traslación del modelo E/R (diseño conceptual) al modelo relacional (diseño lógico).*

### MODELO RELACIONAL (DISEÑO LÓGICO)

-  ⇒ Clave Primaria
-  ⇒ Clave Foranea
-  ⇒ Clave Primaria de tabla de relaciones

- PRODUCTOR(**ID**, Nombre, Produccion Media, Produccion Maxima, Fecha Actividad)
- ESTACION (**ID**, Nombre, Numero de Transformadores)
- HIDRO (**ID**, Ocupacion, Capacidad Maximo, Numero Turbinas)
- SOLAR(**ID**, Superficie Paneles, Media Anual Horas Sol, Tipo)
- NUCLEAR(**ID**, Numero de Reactores, El Volumen Plutonio, Resudios)
- TERMICA(**ID**, Numero de Hornos, Volumen Carbon, Volumen Emisiones)
- TRANSPORTISTA(**ID**, **Matricula**, Nombre)
- SUMINISTRADOR(**ID**, Nombre, Pais)
- COMPRA( **ID[PRODUCTOR]**, **ID[TRANSPORTISTA]**, **ID[SUMINISTRADOR]**, **Fecha**, Cantidad)
- ENTRAGA ( **ID[PRODUCTOR]**, **ID[ESTACION]**, **Fecha**, Cantidad )
- REDES (**ID**, Numero de Red, energia\_transmitida, **ID[ESTACION]**)
- ENVIAR SOBRENTE(**ID1[REDES]**, **ID2[REDES]**, **Fecha**, El Volumen Total de Energia)
- COMPANIAS (**ID**, Nombre)
- COMPARTIDA PROPIEDAD (**ID[REDES]**, **ID[COMPANIAS]**)
- LINEA( **ID\_RED[REDES]**, **Num\_Linea**, Longitud)
- SUBESTACION(**ID**, **ID\_RED[LINIA]**, **NUM\_LINEA[LINIA]**)
- PROVINCIA(**ID**, Nombre)
- ZONAS (**ID**, Consumo Medio, Tipo, **ID\_PROVINCIA[PROVINCIA]**)
- ATENDER(**ID [SUBESTACION]**, **ID[ZONAS]**)

## *Paso del modelo relacional a SQL de ORACLE*

### **CREACION DE LAS TABLAS SQL**

```
CREATE TABLE PRODUCTORR(  
    ID NUMBER(*,0) CONSTRAINT PRODUCTOR_ID_PK PRIMARY KEY,  
    NOMBRE NVARCHAR2(20) NOT NULL,  
    PRODUCCION_MEDIA NUMBER(*,4),  
    PRODUCCION_MAXIMA NUMBER(*,4),  
    FECHA_ACTIVIDAD DATE  
);
```

```
CREATE TABLE ESTACIONN (  
    ID NUMBER(*,0) CONSTRAINT ESTACION_ID_PK PRIMARY KEY,  
    NOMBRE NVARCHAR2(20) NOT NULL,  
    NUMERO_TRANS NUMBER(*,0)  
);
```

```
CREATE TABLE HIDROO(  
    ID NUMBER(*,0) CONSTRAINT HIDRO_ID_PK PRIMARY KEY,  
    CAPACIDAD_MAXIMO NUMBER(*,4),  
    OCUPACION NUMBER(*,4),  
    NUMERO_TURBINAS NUMBER(*,0),  
    CONSTRAINT HIDRO_ID_FK FOREIGN KEY(ID)  
        REFERENCES PRODUCTORR(ID)  
        ON DELETE CASCADE  
);
```

```

CREATE TABLE SOLARR(
    ID NUMBER(*,0) CONSTRAINT SOLAR_ID_PK PRIMARY KEY,
    HORAS_ANUAL NUMBER(*,4),
    TIPO NVARCHAR2(15) CONSTRAINT SOLAR_TIPO CHECK(TIPO
IN('TERMODINAMICA','FOTOVOLTAICA')),
    SUP_PANELES NUMBER(*,4),
    CONSTRAINT SOLAR_ID_FK FOREIGN KEY(ID)
REFERENCES PRODUCTORR(ID)
    ON DELETE CASCADE
);

```

```

CREATE TABLE NUCLEARR(
    ID NUMBER(*,0) CONSTRAINT NUCLEAR_ID_PK PRIMARY KEY,
    NUMERO_REACTORES NUMBER(*,4),
    VOLUMEN_PLUTONIO NUMBER(*,4),
    RESIDUOS NUMBER(*,4),
    CONSTRAINT NUCLEAR_ID_FK FOREIGN KEY(ID)
REFERENCES PRODUCTORR(ID)
    ON DELETE CASCADE
);

```

```

CREATE TABLE TERMICAA(
    ID NUMBER(*,0) CONSTRAINT TERMICA_ID_PK PRIMARY KEY,
    NUMERO_HORNOS NUMBER(*,0),
    CARBON_VOLUMEN NUMBER(*,4),
    GAS_VOLUMEN NUMBER(*,4),
    CONSTRAINT TERMICA_ID_FK FOREIGN KEY(ID)

```

REFERENCES PRODUCTORR(ID)

ON DELETE CASCADE

);

CREATE TABLE TRANSPORTISTAA(

ID NUMBER(\*,0) NOT NULL,

NOMBRE NVARCHAR2(20),

MATRICULA NUMBER(\*,0) NOT NULL,

CONSTRAINT TRANSPORTISTA\_PK PRIMARY KEY(ID,MATRICULA)

);

CREATE TABLE SUMINISTRADORR(

ID NUMBER(\*,0) CONSTRAINT SUMINISTRADORES\_PK PRIMARY KEY,

NOMBRE NVARCHAR2(20),

PAIS NVARCHAR2(20)

);

CREATE TABLE COMPRAA(

FECHA DATE NOT NULL,

CANTIDAD NUMBER(\*,4) NOT NULL,

TRANS\_ID NUMBER(\*,0) NOT NULL,

MATRICULA NUMBER(\*,0) NOT NULL,

COMPRA\_ID NUMBER(\*,0) NOT NULL,

CONSTRAINT COMPRA\_ID\_REF FOREIGN KEY(COMPRA\_ID) REFERENCES SUMINISTRADORR(ID),

ID\_NUCLEAR NUMBER(\*,0) NOT NULL,

CONSTRAINT NUCLEAR\_FK FOREIGN KEY(ID\_NUCLEAR) REFERENCES NUCLEARR(ID),

```
CONSTRAINT CLAVE_FORANEA_COMPRA FOREIGN KEY(MATRICULA, TRANS_ID) REFERENCES  
TRANSPORTISTAA(MATRICULA,ID),
```

```
CONSTRAINT CLAVE_PRIMARIA_COMPRA PRIMARY KEY(MATRICULA, TRANS_ID, COMPRA_ID,  
ID_NUCLEAR, FECHA)
```

```
);
```

```
CREATE TABLE ENTRAGAA(
```

```
ID_PRODUCTOR NUMBER(*,0) NOT NULL,
```

```
CONSTRAINT ID_PRODUCTOR_REF FOREIGN KEY(ID_PRODUCTOR) REFERENCES PRODUCTORR(ID),
```

```
ID_ESTACION NUMBER(*,0) NOT NULL,
```

```
CONSTRAINT ID_ESTACION_REF FOREIGN KEY(ID_ESTACION) REFERENCES ESTACIONN(ID),
```

```
CANTIDAD NUMBER(*,4),
```

```
FECHA DATE,
```

```
CONSTRAINT CLAVE_PRIMARIA_ENTRAGA PRIMARY KEY(FECHA, ID_PRODUCTOR, ID_ESTACION)
```

```
);
```

```
CREATE TABLE REDD(
```

```
    ID NUMBER(*,0) CONSTRAINT RED_ID_PK PRIMARY KEY,
```

```
    ENERGIA_TRANSMITIDA NUMBER(*,4),
```

```
    NUMERO NUMBER(*,0),
```

```
    ID_ESTACION NUMBER(*,0) NOT NULL,
```

```
    CONSTRAINT ID_RED_EST FOREIGN KEY(ID_ESTACION) REFERENCES ESTACIONN(ID)
```

```
);
```

```
CREATE TABLE ENVIAR_SOBRENTTEE(
```

```
    ID_RED_TOMAR NUMBER(*,0) NOT NULL,
```



```
CONSTRAINT RED_TOMAR FOREIGN KEY(ID_RED_TOMAR) REFERENCES REDD(ID),  
ID_RED_DAR NUMBER(*,0) NOT NULL,  
CONSTRAINT RED_DAR FOREIGN KEY(ID_RED_DAR) REFERENCES REDD(ID),  
FECHA DATE NOT NULL,  
ENERGIA_TRANSMITIDA NUMBER(*,4),  
CONSTRAINT CLAVE_PRIMARIA_SOBRENTE PRIMARY KEY(ID_RED_TOMAR, ID_RED_DAR, FECHA)  
);
```

```
CREATE TABLE COMPANIAA(  
    ID NUMBER(*,0) CONSTRAINT COMPANIA_ID_PK PRIMARY KEY,  
    NOMBRE NVARCHAR2(30) NOT NULL  
);
```

```
CREATE TABLE COMPARTIDAA(  
    ID_RED NUMBER(*,0) NOT NULL,  
    CONSTRAINT COMPARTIDA_RED FOREIGN KEY(ID_RED) REFERENCES REDD(ID),  
    ID_COMPANIA NUMBER(*,0) NOT NULL,  
    CONSTRAINT COMPARTIDA_COMPANIA FOREIGN KEY(ID_COMPANIA) REFERENCES COMPANIAA(ID),  
    CONSTRAINT CLAVE_PRIMARIA_COMPARTIDA PRIMARY KEY(ID_RED, ID_COMPANIA)  
);
```

```
CREATE TABLE LINEAA(  
    ID_RED NUMBER(*,0) NOT NULL,  
    CONSTRAINT LINEA_RED FOREIGN KEY(ID_RED) REFERENCES REDD(ID),  
    NUM_LINEA NUMBER(*,0) NOT NULL,  
    LONGITUD NUMBER(*,4),
```

```

        CONSTRAINT CLAVE_PRIMARIA_LINEA PRIMARY KEY(ID_RED, NUM_LINEA)

);

CREATE TABLE SUBESTACIONN(

    ID NUMBER(*,0) CONSTRAINT SUB_ID_PK PRIMARY KEY,

    ID_RED NUMBER(*,0) NOT NULL,

    NUM_LINEA NUMBER(*,0) NOT NULL,

    PAIS NVARCHAR2(20),

    CONSTRAINT CLAVE_FORANEA_SUBESTACION FOREIGN KEY(ID_RED, NUM_LINEA) REFERENCES
LINEAA(ID_RED, NUM_LINEA)

);

CREATE TABLE PROVINCIAA(

    ID NUMBER(*,0) CONSTRAINT PROVINCIA_ID_PK PRIMARY KEY,

    NOMBRE NVARCHAR2(20)

);

CREATE TABLE ZONAA(

    ID NUMBER(*,0) CONSTRAINT ZONA_ID_PK PRIMARY KEY,

    CONSUMO_MEDIO NUMBER(*,4),

    TIPO NVARCHAR2(20) CONSTRAINT TIPO_ZONA CHECK(TIPO IN('PARTICULAR', 'EMPRESA',
'INSTITUCION')),

    ID_PROVINCIA NUMBER(*,0) NOT NULL,

    CONSTRAINT ZONA_PROVINCIA FOREIGN KEY(ID_PROVINCIA)
REFERENCES PROVINCIAA(ID)

);

```

```

CREATE TABLE ATENDERR(
    ID_SUB NUMBER(*,0) NOT NULL,
    CONSTRAINT ID_SUB_ATENDER FOREIGN KEY(ID_SUB) REFERENCES SUBESTACIONN(ID),
    ID_ZONA NUMBER(*,0) NOT NULL,
    CONSTRAINT ID_ZONA_ATENDER FOREIGN KEY(ID_ZONA) REFERENCES ZONAA(ID),
    CONSTRAINT CLAVE_PRIMARIA_ATENDER PRIMARY KEY(ID_SUB, ID_ZONA)
);

```

## CONSULTAS SQL

1 . Selecciona la zona con mayor consumo medio.

```

SELECT ID AS ID_ZONA, CONSUMO_MEDIO, TIPO, ID_PROVINCIA
FROM ZONAA
WHERE CONSUMO_MEDIO=(SELECT MAX(CONSUMO_MEDIO) FROM ZONAA);

```

2. Selecciona al productor que ha entregado mayor cantidad de energia en el mes de mayo del 2015.

```

SELECT E.CANTIDAD, E.FECHA, E.ID_PRODUCTOR, P.NOMBRE
FROM ENTRAGAA E, PRODUCTORR P
WHERE CANTIDAD = (SELECT MAX(CANTIDAD) FROM ENTRAGAA
                  WHERE
                  FECHA BETWEEN DATE'2015-05-01' AND DATE'2015-05-31'
                  AND E.ID_PRODUCTOR = P.ID);

```

3. Determina las companias electricas que abastecen a la Provincia de Ceuta.

```
SELECT DISTINCT C.ID AS "ID DE COMPANIA", C.NOMBRE AS "NOMBRE DE COMPANIA"

FROM PROVINCIAA P, ZONAA Z, ATENDERR A, SUBESTACIONN S, COMPARTIDAA CT, COMPANIAA C

WHERE P.NOMBRE = 'CEUTA' AND Z.ID_PROVINCIA = P.ID

AND A.ID_ZONA = Z.ID AND S.ID = A.ID_SUB

AND CT.ID_RED = S.ID_RED AND CT.ID_COMPANIA = C.ID;
```

4. Realiza un listado con todos las redes, su estacion cabecera y la cantidad total de energia recibida por esta estacion.

```
SELECT R.ID AS ID_RED, E.ID_ESTACION AS "ESTACION CABECERA", SUM(NVL(E.CANTIDAD,0)) AS

"TOTAL ENERGIA"

FROM ENTRAGAA E, REDD R

WHERE E.ID_ESTACION = R.ID_ESTACION

GROUP BY R.ID, E.ID_ESTACION;
```

5. Determina los productores solares que entregan energia a todas las estaciones primarias existentes.

```
SELECT P.NOMBRE, S.ID FROM SOLARR S, PRODUCTORR P
WHERE NOT EXISTS (SELECT *FROM ESTACIONN E
    WHERE NOT EXISTS(SELECT *FROM ENTRAGAA A
        WHERE A.ID_PRODUTOR=S.ID AND
        E.ID = A.ID_ESTACION
    )
) AND S.ID = P.ID;
```

6. Realiza un listado con las compras acumuladas realizadas por centrales nucleares en el mes de mayo de 2015 ordenados de mayor a menor total acumulado. Los datos a mostrar seran: Nombre de Productor, Cantidad acumulada en el mes.

```
SELECT P.NOMBRE AS "NOMBRE DE PRODUCTOR", SUM(C.CANTIDAD) AS "CANTIDAD ACUMULADA"
FROM PRODUCTORR P, NUCLEARR N, COMPRAA C
WHERE P.ID=N.ID AND C.ID_NUCLEAR= N.ID AND C.FECHA BETWEEN DATE'2015-05-01' AND
DATE'2015-05-31'
GROUP BY P.NOMBRE
ORDER BY "CANTIDAD ACUMULADA" DESC;
```

.7. Realiza un listado de los intercambios de energia entre redes mostrando el total enviado, recibido y la diferencia entre ambos cada red.

```
SELECT ID_RED_TOMAR, ID_RED_DAR, ENERGIA_TRANSMITIDA
FROM ENVIAR_SOBRENTTEE ;
```

8. Realiza un listado de provincias con sus consumos medios ordenado por este valor, Los campos a mostrar seran: Nombre de Provincia, Consumo Medio.

```
SELECT P.NOMBRE, SUM(Z.CONSUMO_MEDIO) AS CONSUMO_MEDIO
FROM PROVINCIAA P
LEFT JOIN ZONAA Z
ON P.ID = Z.ID_PROVINCIA
GROUP BY P.NOMBRE
ORDER BY CONSUMO_MEDIO ASC;
```

9. Selecciona las subestaciones no abastecidas por ninguna línea.

10. Realiza un listado con los suministradores que hayan efectuado más de 2 ventas y la cantidad de material suministrada supere los 1000 kg. Los campos a mostrar seran: Nombre, N de ventas, Cantidad Total.

```
SELECT S.NOMBRE, COUNT(*) AS VENTAS, SUM(C.CANTIDAD) AS CANTIDAD_TOTAL
FROM SUMINISTRADORR S, COMPRAA C
WHERE C.COMPRA_ID = S.ID
GROUP BY S.NOMBRE
HAVING COUNT(C.COMPRA_ID) >= 2 AND SUM(C.CANTIDAD) >= 1000;
```

11. Crear una vista que muestre los consumos acumulados por provincia. Los campos a mostrar seran: Codigo de provincia, consumo medio de la provincia.

```
CREATE VIEW CONSUMO_PROVINCIA
(NOMBRE_PROVINCIA, CONSUMO_MEDIO)
AS
(SELECT P.NOMBRE AS NOMBRE_PROVINCIA, SUM(Z.CONSUMO_MEDIO) AS CONSUMO_MEDIO
FROM PROVINCIAA P
LEFT JOIN ZONAA Z
ON P.ID = Z.ID_PROVINCIA
GROUP BY P.NOMBRE
);
```

12. Crear una vista que muestre la cantidad entregada de los productores por meses ordenado por Productor y Mes. Los campos a mostrar seran: Nombre de Productor, Mes, Cantidad.

```
CREATE VIEW CANTIDAD_ENTREGADA
(NOMBRE_PRODUCTOR, MES, CANTIDAD_TOTAL)
AS
(SELECT P.NOMBRE AS NOMBRE_PRODUCTOR, EXTRACT(MONTH FROM E.FECHA) AS MES,
SUM(E.CANTIDAD) AS CANTIDAD_TOTAL
FROM PRODUCTORR P
LEFT JOIN ENTRAGAA E
ON E.ID_PRODUCTOR = P.ID
GROUP BY P.NOMBRE, EXTRACT(MONTH FROM E.FECHA) );
```

13. Crear un indice sobre la tabla zona para mejorar la eficiencia de consulta de esta tabla al acceder a traves del codigo de provincia.

```
CREATE INDEX INDEX_ZONA_ID_PROVINCIA ON ZONAA(ID_PROVINCIA);
```

14. Crear un indice que mejore la eficiencia de las consultas de compras por codigo de productor nuclear.

```
CREATE INDEX INDEX_COMPRA_ID_NUCLEAR ON COMPRAA(ID_NUCLEAR);
```