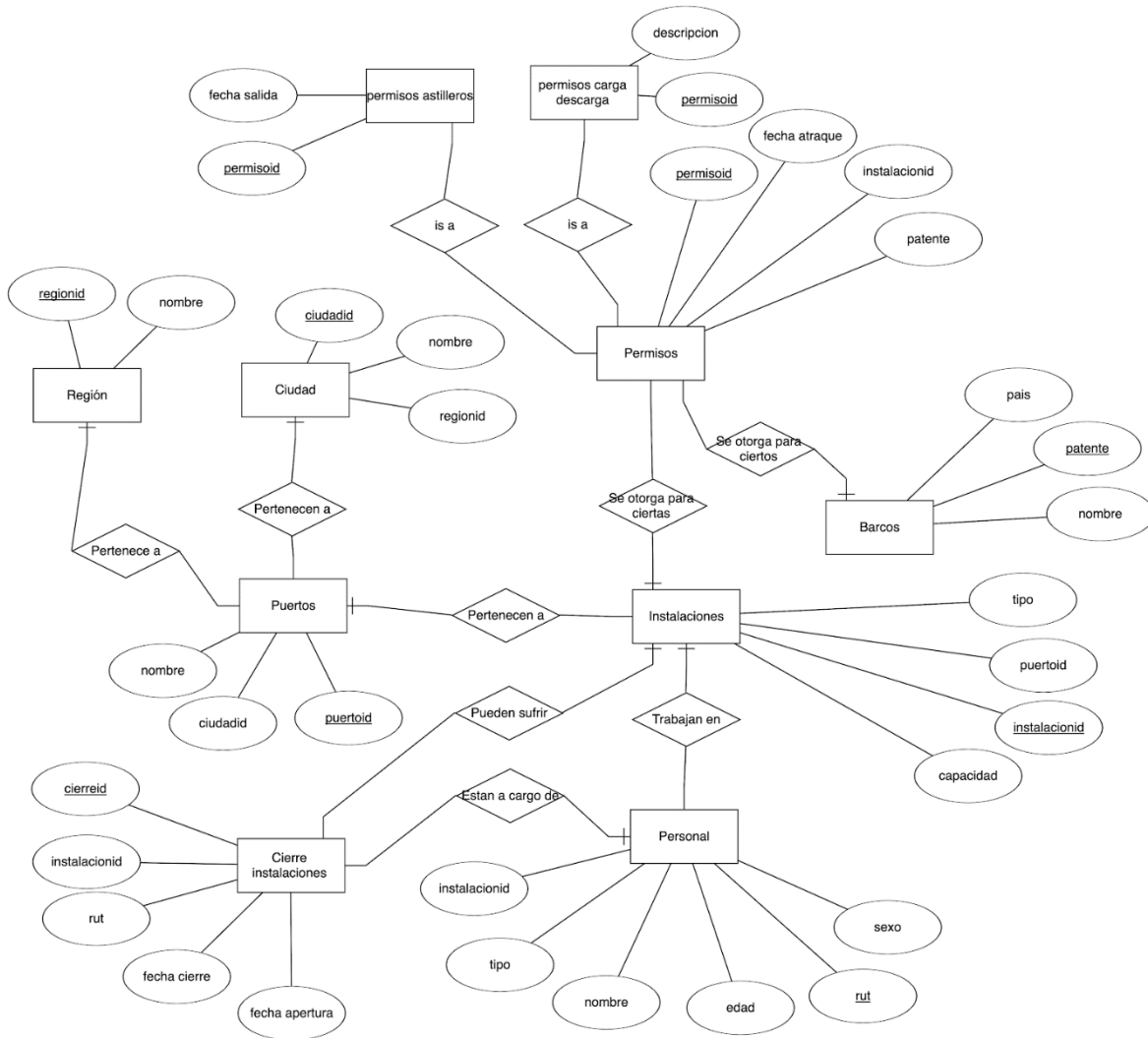


Entrega 2 grupo 77

A continuación se presenta el diagrama E/R, el cual contiene todas las entidades y relaciones utilizadas en la entrega 2, nótese que las rayas pequeñas sobre una línea son el equivalente a una flecha y que tanto permisos carga descarga como permisos astilleros tienen como llave una foreign key (la de permisos), también se escribió de forma “larga”, a modo de ejemplo aparece regionid y no reid, el cual fue usado para el esquema relacional, esto ocurre con varios atributos cuyos nombres cambian ligeramente entre el diagrama y el esquema.

Por otra parte, ninguna de las relaciones del diagrama fueron utilizadas en el esquema relacional ni en sql debido a que como vimos en clase esto se podía evitar poniendo la llave correspondiente en la entidad que solo se asocia con otra. A modo de ejemplo cada instalación pertenece a un puerto, pero en vez de crear la tabla pertenecen a, se optó por incluir la llave de puerto (puid) en instalaciones. Esto se usó con todas las relaciones, evitando tener que crear numerosas tablas innecesarias.



A continuación se presenta el esquema relacional con las tablas respectivas de la entrega 2 correspondiente a los grupos impares. Cada tabla está escrita en: primero sql, luego en álgebra relacional y finalmente una breve descripción donde se aclaran los atributos que pueden generar confusión. Luego para las consultas se adjuntan imágenes de cada una escrita en Relax.

Esquema:

- Puertos(puid INT PRIMARY KEY, punombre VARCHAR(100), puciudad VARCHAR(100), puregion VARCHAR(100))
Puertos(puid: int, punombre: string, puciudad: string, puregion: string)
Puertos tiene: id (key), nombre, ciudad, región.
- Ciudad(ciuid INT PRIMARY KEY, ciunombre VARCHAR(100))
Ciudad tiene: id (key), nombre.
- Region(reid INT PRIMARY KEY, renombre VARCHAR(100))
Region tiene: id (key), nombre.
- Instalaciones (inid INT PRIMARY KEY, puid INT, incapacidad INT, intipo VARCHAR(100))
Instalaciones(inid: int, puid: int, incapacidad: int, intipo: string)
Instalaciones tiene: id (key), puerto id, capacidad, tipo.
- CierreInstalaciones (cid INT PRIMARY KEY, inid INT, psid INT, ccierre DATE, capertura DATE)
CierreInstalaciones(cid: int, inid: int, psid: int, ccierre: date, capertura: date)
CierreInstalaciones tiene: id (key), id instalacion, id persona, fecha cierre, fecha apertura.
- Personal (psrut VARCHAR(20) PRIMARY KEY, psnombre VARCHAR(100), psedad INT, pssexo VARCHAR(100), inid INT, psesjefe VARCHAR(2))
Personal(psrut: str, psnombre: string, psedad: int, pssexo: string, inid: int, psesjefe: string)
Personal tiene: rut (key), nombre, edad, sexo, id instalacion, es jefe o no
- Barcos (bpatente VARCHAR(100) PRIMARY KEY, bnombre VARCHAR(100), bpais VARCHAR(100))
Barcos(bpatente: string, bnombre: string, bpais: string)
Barcos tiene: patente (key), nombre, país.
- Permisos (pmid INT PRIMARY KEY, inid INT, bpatente VARCHAR(20), pmaatraque DATE)

- Permisos(pmid: int, inid: int, bpatente:str, pmatraque:date)
Permisos tiene: id (key), id instalacion, patente barco, fecha atraque.
- PermisosAstilleros (pmid INT PRIMARY KEY, pmasalida DATE)
- PermisosAstilleros(pmid: int, pmasalida: date)
PermisosAstilleros tiene: id (key), fecha salida.
- PermisosCargaDescarga (pmcdid INT PRIMARY KEY, pmcddescripcion VARCHAR(1000))
- PermisosCargaDescarga(pmcdid: int, descripcion: string)
PermisosCargaDescarga tiene: id (key), descripción.

Justificación de porque el esquema se halla en BCNF:

Este esquema se halla en BCNF porque todas sus relaciones se encuentran en BCNF: Específicamente, para toda dependencia funcional no trivial $X \rightarrow Y$, X es llave.

En nuestro caso:

Para puertos, puertoid (puid) determina el nombre, el id de la ciudad (ciuid) y el id de la región (reid), si bien se puede pensar que una ciudad determina la región, en el caso de que la ciudad se encuentre entre 2 regiones no habría forma de determinar esto, por lo tanto normalizar esto no sería correcto.

Para ciudad, ciuid determina el nombre de la ciudad (ciunombre).

Para región, reid determina el nombre de la región (renombre).

Para instalaciones, instalacionid (inid) determina el id del puerto (puid), la capacidad de la instalacion (incapacidad) y el tipo (intipo).

Para cierre instalaciones, cierre id (cid) determina, el id de la instalacion (inid), el rut del encargado (psrut), la fecha de cierre (ccierre) y la fecha de apertura (apertura).

Para personal, el rut (psrut) determina el nombre (psnombre), la edad (psedad), el sexo (pssexo) y si es jefe o no (psesjefe).

Para barcos, la patente (bpatente) determina el nombre (bnombre) y el pais (bpais).

Para permisos, el id del permiso (pmid) determina la patente del barco (bpatente) y la fecha de atraque (pmatraque).

Para permisos astilleros, el id del permiso (pmid) determina la fecha de salida (pmasalida).

Para permisos carga descarga, el id del permiso (pmid) determina la descripción (pmcddescripcion).

Consultas sql:

A continuación, se presentan todas las consultas en sql tal como se encuentran en la aplicación, es decir para las consultas del enunciado donde salía algún lugar, puerto, barco o fecha específica se reemplazo con variables y herramientas para poder adaptarlas y hacerlas case sensitive y con matching parcial. Esto significa que las consultas como están solo funcionan en el contexto de nuestra aplicación.

Cuando este es el caso también se adjunta la consulta que satisface el enunciado según nuestra base de datos (directamente en psql).

- 1) `SELECT punombre, ciunombre FROM puertos, ciudad WHERE puertos.ciuid = ciudad.ciuid;`
- 2) `SELECT psnombre from personal, instalaciones, puertos where personal.inid = instalaciones.inid and personal.psesjefe = 'si' and puertos.puid = instalaciones.puid and LOWER (puertos.punombre) LIKE LOWER ('%$nombre%');`

`select psnombre from personal, instalaciones where personal.inid = instalaciones.inid and personal.psesjefe = 'si' and puertos.puid = instalaciones.puid and puertos.punombre = 'Mejillones'`
- 3) `SELECT punombre from puertos, instalaciones where puertos.puid = instalaciones.puid and instalaciones.intipo = 'astillero';`
- 4) `SELECT barcos.bnombre, puertos.punombre, ciudad.ciunombre, permisos.pmatraque FROM PERMISOS, BARCOS, INSTALACIONES, CIUDAD, PUERTOS WHERE permisos.inid = instalaciones.inid AND instalaciones.puid = puertos.puid AND barcos.bpatente = permisos.bpatente AND ciudad.ciuid = puertos.ciuid AND LOWER(barcos.bnombre) LIKE`

```
LOWER('%$barco%') AND LOWER(ciudad.ciunombre) LIKE  
LOWER('%$puerto%');
```

```
SELECT barcos.bnombre, puertos.punombre, ciudad.ciunombre,  
permisos.pmatraque FROM PERMISOS, BARCOS,  
INSTALACIONES, CIUDAD, PUERTOS WHERE permisos.inid =  
instalaciones.inid AND instalaciones.puid = puertos.puid AND  
barcos.bpatente = permisos.bpatente AND ciudad.ciuid =  
puertos.ciuid AND LOWER(barcos.bnombre) LIKE  
LOWER('Calypso') AND LOWER(ciudad.ciunombre) LIKE  
LOWER('Arica');
```

5) SELECT puertos.punombre, AVG(personal.psedad) FROM
PERSONAL, INSTALACIONES, PUERTOS WHERE personal.inid =
instalaciones.inid AND puertos.puid = instalaciones.puid GROUP BY
puertos.punombre;

6) SELECT puertos.punombre, COUNT(barcos.bpatente) as llegadas
FROM PERMISOS, BARCOS, INSTALACIONES, PUERTOS
WHERE permisos.inid = instalaciones.inid AND instalaciones.puid =
puertos.puid AND barcos.bpatente = permisos.bpatente AND
permisos.pmatraque >= '%\$new_date_01%' AND
permisos.pmatraque <= '%\$new_date_31%' GROUP BY
(puertos.puid) HAVING COUNT(barcos.bpatente) = (SELECT
MAX(llegadas) FROM (SELECT puertos.punombre,
COUNT(barcos.bpatente) as llegadas FROM PERMISOS, BARCOS,
INSTALACIONES, PUERTOS WHERE permisos.inid =
instalaciones.inid AND instalaciones.puid = puertos.puid AND
barcos.bpatente = permisos.bpatente AND permisos.pmatraque >=
'%\$new_date_01%' AND permisos.pmatraque <=
'%\$new_date_31%' GROUP BY (puertos.puid)) as FOO);

```
SELECT puertos.punombre, COUNT(barcos.bpatente) as llegadas  
FROM PERMISOS, BARCOS, INSTALACIONES, PUERTOS  
WHERE permisos.inid = instalaciones.inid AND instalaciones.puid =  
puertos.puid AND barcos.bpatente = permisos.bpatente AND  
permisos.pmatraque >= '2020-08-01' AND permisos.pmatraque <=  
'2020-08-31' GROUP BY (puertos.puid) HAVING  
COUNT(barcos.bpatente) = (SELECT MAX(llegadas) FROM  
(SELECT puertos.punombre, COUNT(barcos.bpatente) as llegadas  
FROM PERMISOS, BARCOS, INSTALACIONES, PUERTOS  
WHERE permisos.inid = instalaciones.inid AND instalaciones.puid =  
puertos.puid AND barcos.bpatente = permisos.bpatente AND
```

```
permisos.pmatraque >= '2020-08-01' AND permisos.pmatraque <=
'2020-08-31' GROUP BY (puertos.puid)) as FOO);
```