

Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC2413 — Bases de Datos — 2' 2024

Ayudantía 3 – Guía de ejercicios SQL

Consulta 1

Álgebra Relacional:
Encuentre los nombres de todo el personal con edad menor a 25, o que son de nacionalidad Chilena.

$$\Pi_{\text{nombre}}(\sigma_{\text{edad}<25 \vee \text{nacionalidad}=\text{"}Chile\text{"}}(\text{Personal}))$$

SQL:

```
SELECT nombre
FROM Personal
WHERE edad < 25 OR nacionalidad = 'Chile';
```

Consulta 2

Álgebra Relacional:
Encuentre los ids de todo el personal que trabaja en un buque de la naviera "Lagos del Sur".

$$\Pi_{\text{personalId}}(\text{TrabajaEn} \bowtie_{\text{BuqueId}} \text{Navegaciones} \bowtie_{\text{navieraId}} (\sigma_{\text{nombre}=\text{"}LagosdelSur\text{"}}(\text{Navieras})))$$

SQL:

```
SELECT personalId
FROM TrabajaEn
JOIN Navegaciones ON TrabajaEn.BuqueId = Navegaciones.BuqueId
JOIN Navieras ON Navegaciones.NavieraId = Navieras.NavieraId
WHERE Navieras.nombre = 'Lagos-del-Sur';
```

De forma alternativa:

```
SELECT personalId
FROM TrabajaEn, Navieras, Navegaciones
WHERE Navieras.nombre = 'Lagos-del-Sur'
  AND TrabajaEn.BuqueId = Navegaciones.BuqueId
  AND Navegaciones.NavieraId = Navieras.NavieraId;
```

Consulta 3

Álgebra Relacional:
Para validar si los datos son consistentes, asegure que cada capitán de un bote también está registrado en la tabla Personal. Escribe la consulta que encuentre todos los capitanes que no aparecen en la tabla de Personal.

$$\Pi_{\text{capitanId}}(\text{Buques}) - \Pi_{\text{Buques.capitanId}}(\text{Buques} \bowtie_{\text{Buques.capitanId=Personal.personalId}} \text{Personal})$$

SQL:

```
SELECT capitanId
FROM Buques
WHERE capitanId NOT IN (
    SELECT personalId
    FROM Personal
);
```

Esto es una consulta anidada (lo verán mas adelante)

Consulta 4

Álgebra Relacional:
Encuentre los nombres de los buques operando para más de una naviera.

$$\rho(N1, \Pi_{\text{buqueId,NavieraId}}(\text{Navegaciones}))$$

$$\rho(N2, \Pi_{\text{buqueId,NavieraId}}(\text{Navegaciones}))$$

$$\rho(N3, \Pi_{N1.\text{buqueId}}(\sigma_{N1.\text{buqueId}=N2.\text{buqueId} \wedge N1.\text{NavieraId} \neq N2.\text{NavieraId}}(N1 \times N2)))$$

$$\Pi_{\text{nombre}}(\text{Buques} \bowtie_{\text{Buques.buqueId}=N3.\text{buqueId}} N3)$$

SQL:

```
SELECT DISTINCT B.nombre
FROM Buques B
JOIN Navegaciones N1 ON B.buqueId = N1.buqueId
JOIN Navegaciones N2 ON B.buqueId = N2.buqueId
WHERE N1.NavieraId != N2.NavieraId;
```

DISTINCT nos permite evitar la repetición, quedándonos solo con una de las tuplas que se repitan

Consulta 5

Álgebra Relacional:
Encuentre el nombre del buque con la capacidad más grande.

$$\rho(B1, \Pi_{\text{nombre,capacidad}}(\text{Buques}))$$

$$\rho(B2, \Pi_{\text{nombre,capacidad}}(\text{Buques}))$$

$$\rho(B4, \Pi_{B1.\text{nombre},B1.\text{capacidad}}(\sigma_{B1.\text{capacidad}<B2.\text{capacidad}}(B1 \times B2)))$$

$$\Pi_{\text{nombre}}(B1 - B4)$$

SQL:

```
SELECT nombre
FROM Buques
ORDER BY capacidad DESC
LIMIT 1;
```

El ordenamiento también lo verán mas adelante, pero puede reducir mucho una consulta, en este caso ordenamos por orden decreciente según la capacidad y nos quedamos con el primer valor (el mayor).

Consulta 6

Álgebra Relacional:
Encuentre los nombres de todo el personal que trabaja para dos o más buques distintos.

$$\rho(T1, \text{TrabajaEn})$$

$$\rho(T2, \text{TrabajaEn})$$

$$\rho(T3, \Pi_{T1.\text{personalId}}(\sigma_{T1.\text{personalId}=N2.\text{personalId} \wedge T1.\text{buqueId} \neq T2.\text{buqueId}}(T1 \times T2)))$$

$$\Pi_{\text{nombre}}(\text{Personal} \bowtie_{\text{Personal.personalId}=T3.\text{personalId}} T3)$$

SQL:

```
SELECT DISTINCT P.nombre
FROM Personal P
JOIN TrabajaEn T1 ON P.personalId = T1.personalId
JOIN TrabajaEn T2 ON P.personalId = T2.personalId
WHERE T1.buqueId != T2.buqueId;
```