

IIC2413 – Bases de Datos

Álgebra Relacional

Esquema

Trabajaremos con el actual esquema del sistema de *EnoTours*, una agencia de enoturismo que organiza visitas a varias viñas ubicadas a lo largo de Chile. Para esto, queremos modelar el escenario donde un tour puede visitar a varias viñas, y donde una viña puede participar en varios tours.

Para soportar este caso de uso, contamos con las siguientes tres tablas:

- Tours(tid, nombre, duración, precio)
- Viñas(vid, nombre, región)
- SeVisita(tid, vid, degustación)

Lo siguiente es una instancia para el esquema:

Tours	tid	nombre	duración	precio	Viñas	vid	nombre	región
	1	Quinta	4	15.000		1	Casas del Bosque	V
	2	Quinta lujo	5	25.000		2	Viña VIK	X
	3	Chile premium	48	109.999		3	Koyle	X
	4	Vinos del Mar	12	58.000		4	Matetic	V

SeVisita	tid	vid	degustación
	2	1	básica
	1	4	premium
	4	3	premium
	4	3	básica
	3	5	básica
	3	2	básica

Consultas

Escriba una consulta en álgebra relacional que entregue lo pedido en cada uno de estos apartados. Para ahorrar espacio, puedes definir consultas intermedias en algunos apartados y volver a usarlas más abajo.

Importante: recuerda que tus consultas deben funcionar para cualquier instancia, la instancia de arriba es solo para que se use como referencia.

- Encuentre los nombres de las viñas de la Quinta región.

Sol: De la tabla Viñas, seleccionar aquellas tales que la región es “V”. Proyectar el nombre.

$$\pi_{nombre}(\sigma_{region = "V"}(Viñas))$$

- Encuentre los nombres de los tours que visitan una viña de la Quinta Región, o una viña de la Décima Región.

Sol: Tiene que haber un join de Tour, Viña y SeVisita. Seleccionar las filas tales que la región es V o X. Alternativamente puede hacer una consulta para la región V, una para X y luego unirlos. Proyectar los nombres de los tours.

$$\pi_{Tours.nombre}(\sigma_{Vinas.region="V" \vee Vinas.region="X"}(Tours \bowtie SeVisita \bowtie Vinas))$$

- Encuentre los tours que visitan al menos 2 viñas.

Sol: Esto se puede hacer con un self join de la tabla SeVisita consigo misma a través del atributo tid. Luego, se hace selección en las filas tales que SeVisita.vid != SeVisita2.vid. Acá no era necesario proyectar el nombre, podían dejar solo el tid.

$$\begin{aligned} &\rho(SV1, \pi_{tid, vid}(SeVisita)) \\ &\rho(SV2, \pi_{tid, vid}(SeVisita)) \\ &\pi_{SV1.tid}(\sigma_{SV1.vid \neq SV2.vid}(SV1 \bowtie_{SV1.tid=SV2.tid} SV2)) \end{aligned}$$

- Encuentre los nombres de los tours que visitan una viña de la Quinta Región, y una viña de la Décima Región.

Sol: Hacer 2 consultas: una para los tour que visitan la quinta y una para los tours que visitan la décima región. Luego, intersectarlas.

$$\begin{aligned} &\rho(ToursV, \pi_{Tours.nombre}(\sigma_{Vinas.region="V"}(Tours \bowtie SeVisita \bowtie Vinas))) \\ &\rho(ToursX, \pi_{Tours.nombre}(\sigma_{Vinas.region="X"}(Tours \bowtie SeVisita \bowtie Vinas))) \\ &ToursV \cap ToursX \end{aligned}$$

- Encuentre los nombres de los tours que no visitan ninguna viña de la Quinta Región.

Sol: Hacer el join para obtener los tours que SI visitan la quinta región. Luego hacer la diferencia: todos los tour - los que visitan la quinta.

$$\begin{aligned} &\rho(ToursV, \pi_{Tour.nombre}(\sigma_{Vinas.region="V"}(Tours \bowtie SeVisita \bowtie Vinas))) \\ &Tour - ToursV \end{aligned}$$

- Encuentre los nombres de los tours que no visitan ninguna viña de la Quinta Región, y visitan una viña de la Décima Región.

Sol: Hacer el join para obtener los tours que SI visitan la quinta región y los que SI visitan la décima. Luego hacer la diferencia entre los que visitan la décima - los que visitan la quinta.

$$\begin{aligned} & \rho(ToursV, \pi_{Tours.nombre}(\sigma_{Vinas.region="V"}(Tours \bowtie SeVisita \bowtie Vinas))) \\ & \rho(ToursX, \pi_{Tours.nombre}(\sigma_{Vinas.region="X"}(Tours \bowtie SeVisita \bowtie Vinas))) \\ & ToursX - ToursV \end{aligned}$$

- Encuentre el nombre del tour más caro.

Sol: Primero identificamos los tours con mayor precio. Para eso hacemos producto cruz de Tours consigo mismo, digamos T1 X T2. Ahora, seleccionamos las filas tales que T1.precio es menor que T2.precio. Con esto nos quedamos con los tours que NO SON LOS MÁS CAROS. La proyección correspondiente de esto, la restamos de la tabla original Tours y así obtenemos el tour más caro. Ojo acá que si se seleccionan las filas tales que T1.precio es mayor que T2.precio, solo se están quedando con los tours que NO SON EL MAS BARATO, así que eso no cumple lo que se pide.

$$\begin{aligned} & \rho(T1, Tours) \\ & \rho(T2, Tours) \\ & \rho(ToursNoMasCaros, \pi_{T1.tid, T1.nombre, T1.duracion, T1.precio}(\sigma_{T1.precio < T2.precio}(T1 \times T2))) \\ & \pi_{Tours.nombre}(Tours - ToursNoMasCaros) \end{aligned}$$