

# Proyecto Bases de Datos Global

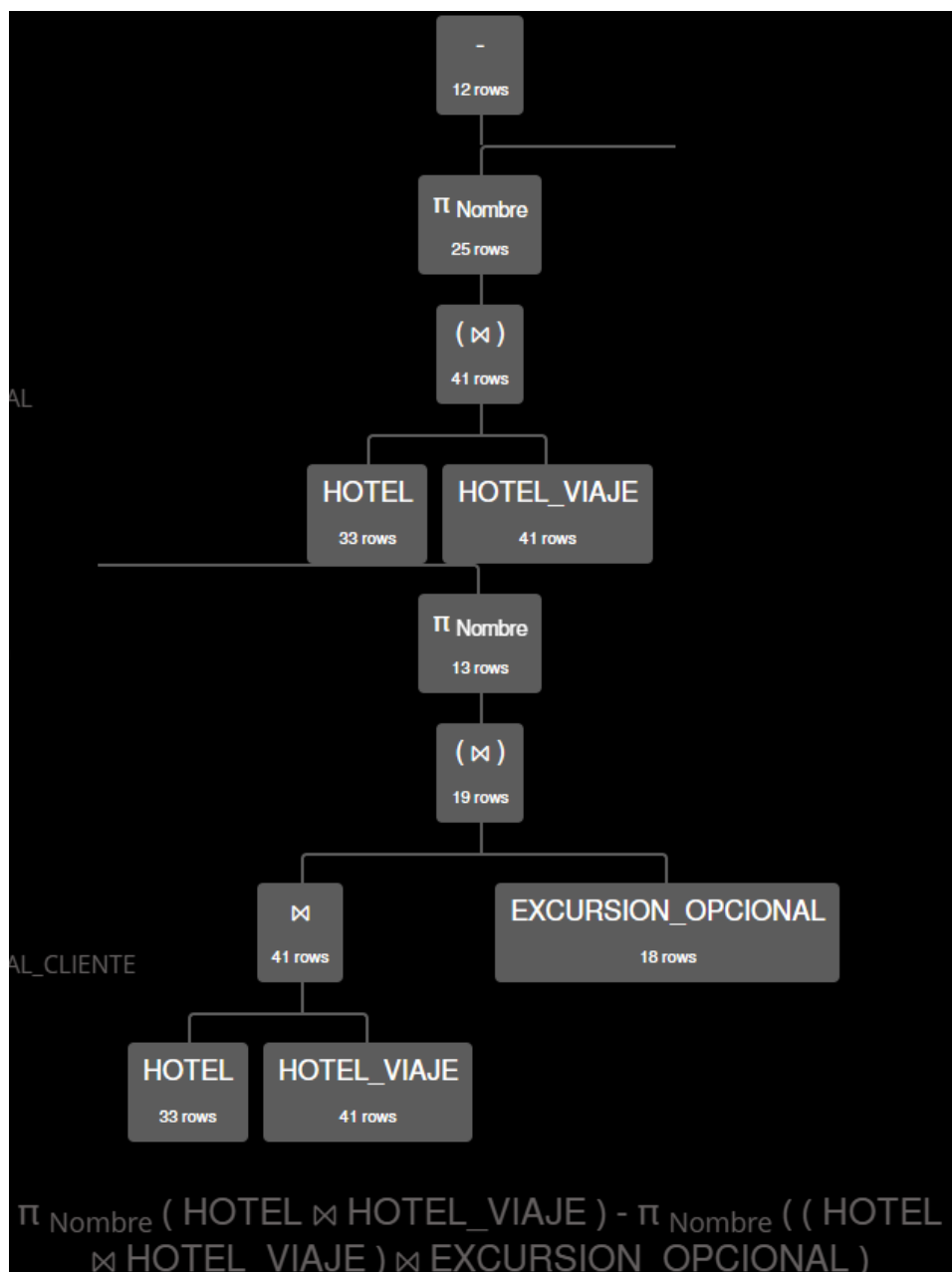
## Algebra relacional

1. Obtener cuál ha sido el número de clientes que se han apuntado en la excursión opcional que más clientes ha tenido. ( No he conseguido hacer esta consulta correctamente )

```
G =  $\pi$  DESTINOEXCURSION, COUNT(DNI) ( $\sigma$  DESTINO = DESTINO AND FECHASALIDA = FECHASALIDA AND NUMERO = NUMERO (EXCURSION_OPCIONAL_CLIENTE  $\bowtie$  EXCURSION_OPCIONAL))  
RESULTADO =  $\pi$  MAX(count) (G)
```

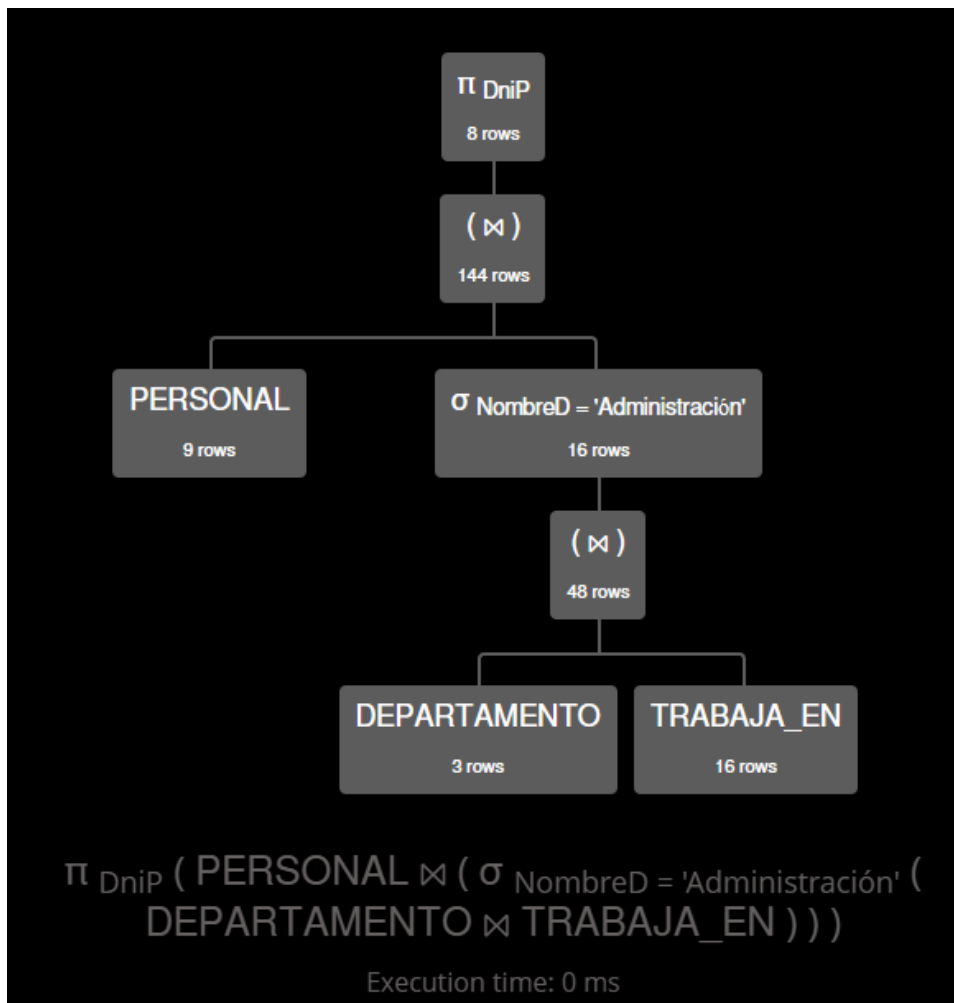
2. Obtener el nombre de los hoteles que se ofertan en los viajes que no tienen excursiones opcionales.

```
 $\pi$  Nombre (HOTEL  $\bowtie$  HOTEL_VIAJE) -  $\pi$  Nombre (HOTEL  $\bowtie$  HOTEL_VIAJE  $\bowtie$  EXCURSION_OPCIONAL)
```



3. Obtener el DNI de los empleados que trabajan en todos los proyectos del departamento 'Administracion'

```
 $\pi_{DniP} (PERSONAL \bowtie (\sigma_{NombreD = 'Administración'} (DEPARTAMENTO \bowtie TRABAJA\_EN)))$ 
```



## SQL

1. Obtener el teléfono de los clientes que únicamente han realizado viajes con destino a 'Toledo'.

```
SELECT c.NTelefono
FROM CLIENTE c
WHERE c.DNI IN (
    SELECT hvc.DNI
    FROM HOTEL_VIAJE_CLIENTE hvc
    WHERE hvc.Destino = 'Toledo'
)
AND c.DNI NOT IN (
    SELECT hvc.DNI
    FROM HOTEL_VIAJE_CLIENTE hvc
    WHERE hvc.Destino != 'Toledo'
)
```

✓ Showing rows 0 - 0 (1 total, Query took 0.0019 seconds.)

```
SELECT c.NTelefono FROM CLIENTE c WHERE c.DNI IN ( SELECT hvc.DNI FROM HOTEL_VIAJE_CLIENTE hvc WHERE hvc.Destino = 'Toledo' ) AND c.DNI NOT IN ( SELECT hvc.DNI FROM HOTEL_VIAJE_CLIENTE hvc WHERE hvc.Destino != 'Toledo' )
```

☐ Profiling [Edit inline] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

← T → NTelefono

☐ Edit ☐ Copy ☐ Delete 943000006

2. Obtener el identificador y el nombre de los hoteles que han sido seleccionadas como máximo dos veces, además de cuantas veces ha sido seleccionada, siempre que el identificador del hotel empiece con la letra 'h'. El resultado estará ordenado en orden descendente por el número obtenido.

```
SELECT h.idhotel, h.Nombre, COUNT(hvc.IdHotel)
FROM hotel_viaje_cliente hvc RIGHT JOIN hotel h
ON hvc.IdHotel=h.IdHotel
WHERE h.IdHotel LIKE 'h%'
GROUP BY h.IdHotel
HAVING COUNT(h.IdHotel)<=2
```

✓ Showing rows 0 - 15 (16 total, Query took 0.0015 seconds.)

```
SELECT h.idhotel, h.Nombre, COUNT(h.IdHotel) FROM hotel_viaje_cliente hvc RIGHT JOIN hotel h ON hvc.IdHotel=h.IdHotel WHERE h.IdHotel LIKE 'h%' GROUP BY h.IdHotel HAVING COUNT(h.IdHotel)<=2
```

☐ Profiling [Edit inline] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

idhotel	Nombre	COUNT(h.IdHotel)
h20	Mezkitor	1
h21	Camp now hostel	1
h08	Eight Hotel	1
h07	Tinto Dreams Hostel	1
h18	K2	2
h16	Mercury	2
h05	Sunshine	2
h01	Sheraton	1
h04	Hilton	1
h13	Nini	2
h06	River Hostel	1
h17	Kulungu	2
h14	Campamento Torre	1
h11	King	1
h12	Tango	1
h15	Sunshine	1

3. Obtener el identificador, nombre y capacidad de los hoteles cuya capacidad es mayor que la capacidad de los hoteles en la ciudad de 'Lima'. El resultado estará ordenado en orden ascendente según la capacidad.

```
SELECT h.IdHotel, h.Nombre, h.Capacidad
FROM HOTEL h
WHERE h.Capacidad > (
```

```
SELECT MAX(h2.Capacidad)
FROM HOTEL h2
WHERE h2.Localidad = 'Lima')
ORDER BY h.Capacidad ASC;
```

✓ Showing rows 0 - 4 (5 total, Query took 0.0009 seconds.) [Capacidad: 1500... - 2700...]

```
SELECT h.IdHotel, h.Nombre, h.Capacidad FROM HOTEL h WHERE h.Capacidad > ( SELECT MAX(h2.Capacidad) FROM HOTEL h2
WHERE h2.Localidad = 'Lima') ORDER BY h.Capacidad ASC
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 ▼ Filter rows: Search this table Sort by key: None ▼

+ Options

				IdHotel	Nombre	Capacidad
<input type="checkbox"/>				h10	Campsite	1500
<input type="checkbox"/>				h11	King	2000
<input type="checkbox"/>				h19	Hotel Sol	2700
<input type="checkbox"/>				h20	Mezkitor	2700
<input type="checkbox"/>				h21	Camp now hostel	2700

4. Obtener por cada guía cuantos idiomas habla y cuantos destinos distintos ha visitado (en viajes que tienen clientes).

```
SELECT g.dni, COUNT(i.DNI), COUNT(v.Destino)
FROM (guia g inner JOIN idioma i ON g.dni=i.DNI)
WHERE g.dni IN (
    SELECT viaje.DNI
    FROM viaje INNER JOIN hotel_viaje_cliente
    ON viaje.Destino=hotel_viaje_cliente.Destino AND
    viaje.FechaSalida=hotel_viaje_cliente.FechaSalida
)
GROUP BY g.dni
```

✓ Showing rows 0 - 10 (11 total, Query took 0.0021 seconds.)

```
SELECT g.dni, COUNT(i.DNI), COUNT(g.DNI) FROM (guia g inner JOIN idioma i ON g.dni=i.DNI) WHERE g.dni IN ( SELECT
viaje.DNI FROM viaje INNER JOIN hotel_viaje_cliente ON viaje.Destino=hotel_viaje_cliente.Destino AND
viaje.FechaSalida=hotel_viaje_cliente.FechaSalida ) GROUP BY g.dni
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 ▼ Filter rows: Search this table

+ Options

dni	COUNT(i.DNI)	COUNT(g.DNI)
21212121	1	1
66666666	1	1
72515642	3	3
72515650	5	5
72515651	1	1
72515655	2	2
72515657	1	1
72515666	1	1
72515667	4	4
72515671	1	1
72515673	10	10

5. Obtener por cada viaje que tiene excursiones opcionales el identificador del viaje, el DNI del cliente que más excursiones opcionales ha realizado y cuantas excursiones han sido.

```
SELECT eoc.Destino, eoc.FechaSalida, MAX(eoc.DNI), COUNT(eoc.DNI)
FROM EXCURSION_OPCIONAL_CLIENTE eoc
WHERE EXISTS (
    SELECT 1
    FROM EXCURSION_OPCIONAL_CLIENTE eoc2
    WHERE eoc2.Destino = eoc.Destino
    GROUP BY eoc2.Destino AND eoc2.FechaSalida
    HAVING COUNT(DISTINCT eoc2.DNI) > 1)
GROUP BY eoc.Destino, eoc.FechaSalida
```

✓ Mostrando filas 0 - 4 (total de 5, La consulta tardó 0.0020 segundos.)

```
SELECT eoc.Destino, eoc.FechaSalida, MAX(eoc.DNI), COUNT(eoc.DNI) FROM
EXCURSION_OPCIONAL_CLIENTE eoc WHERE EXISTS ( SELECT 1 FROM
EXCURSION_OPCIONAL_CLIENTE eoc2 WHERE eoc2.Destino = eoc.Destino GROUP BY
eoc2.Destino AND eoc2.FechaSalida HAVING COUNT(DISTINCT eoc2.DNI) > 1) GROUP BY
eoc.Destino, eoc.FechaSalida
```

☐ Perfilando [Editar en línea] [ Editar ] [ Explicar SQL ] [ Crear código PHP ] [ Actualizar ]

☐ Mostrar todo

Número de filas:

25 ▼

Filtrar filas:

Buscar en esta tabla

+ Opciones

Destino	FechaSalida	MAX(eoc.DNI)	COUNT(eoc.DNI)
Donostia	2019-01-01	10000004	1
Donostia	2022-01-02	10000013	1
Namibia	2020-11-21	72515657	10
Sahara	2020-09-10	10000008	1
Sahara	2020-11-02	10000011	1

6. Obtener el DNI y el nombre de los clientes que han viajado durante el año 2019, siempre que en dichos viajes no se ofertó excursiones opcionales. En el resultado también aparecerá el identificador del viaje.

```
SELECT c.DNI, c.Nombre, hvc.Destino, hvc.FechaSalida
FROM CLIENTE c NATURAL JOIN hotel_viaje_cliente hvc natural join cliente
LEFT JOIN EXCURSION_OPCIONAL_CLIENTE eoc ON hvc.Destino = eoc.Destino
AND hvc.FechaSalida = eoc.FechaSalida
WHERE hvc.FechaSalida like '2019%'
AND eoc.Destino IS NULL
```

✓ Mostrando filas 0 - 0 (total de 1, La consulta tardó 0.0008 segundos.)

```
SELECT c.DNI, c.Nombre, hvc.Destino, hvc.FechaSalida FROM CLIENTE c NATURAL JOIN
hotel_viaje_cliente hvc natural join cliente LEFT JOIN EXCURSION_OPCIONAL_CLIENTE
eoc ON hvc.Destino = eoc.Destino AND hvc.FechaSalida = eoc.FechaSalida WHERE
hvc.FechaSalida like '2019%' AND eoc.Destino IS NULL
```

☐ Perfilando [Editar en línea] [ Editar ] [ Explicar SQL ] [ Crear código PHP ] [ Actualizar]

☐ Mostrar todo

Número de filas:

25 ▼

Filtrar filas:

Buscar en esta tabla

+ Opciones

DNI	Nombre	Destino	FechaSalida
10000004	Julio Rey	Senegal	2019-03-01