

```

177  /*
178  Instrucciones para las consultas SQL:
179  Relación muchos a muchos entre usuarios y grupo:
180  Consulta: - Debes seleccionar los nombres de los usuarios y los nombres de los grupos.
181  - Utiliza las tablas "usuarios", "grupo" y "usuario_grupo".
182  - Realiza una unión entre las tablas "usuarios" y "usuario_grupo" utilizando la condición
183  "usuarios.id = usuario_grupo.us_id".
184  - Realiza una unión entre las tablas "grupo" y "usuario_grupo"
185  utilizando la condición "grupo.id = usuario_grupo.gr_id".
186  */
187
188  select u.nombre,g.nombre from usuarios u, grupo g ,usuario_grupo ug
189  where ug.gr_id=g.id and ug.us_id=u.id;
190  /*
191  Subconsulta: - Debes seleccionar el nombre

```

Data Output Messages Notifications

	nombre character varying (25)	nombre character varying (25)
1	Marilyn	Maternal intensivo 1
2	Romel	Maternal 3
3	Mario	Maternal intensivo 1
4	Johann	Maternal intensivo 2
5	Monserate	Maternal 1

```

/*
Subconsulta:
- Debes seleccionar el nombre de los usuarios.
- Utiliza la tabla "usuarios".
- Utiliza una subconsulta para obtener los "us_id" de la tabla "usuario_grupo" donde "gr_id" sea igual
- Utiliza la condición "id IN (SELECT ***** FROM ***** WHERE ***** = 1)".
*/

select u.nombre from usuarios u where u.id IN (select us_id from usuario_grupo where gr_id=1);
/*
Función de agregación:
- Debes seleccionar el nombre del grupo y contar la cantidad de "us_id" en la tabla "usuario_grupo".
- Utiliza las tablas "grupo" y "usuario_grupo".

```

Data Output Messages Notifications

nombre character varying (25)
Monserate
Berni

```

/*
Función de agregación:
- Debes seleccionar el nombre del grupo y contar la cantidad de "us_id" en la tabla "usuario_grupo".
- Utiliza las tablas "grupo" y "usuario_grupo".
- Realiza una unión entre las tablas "grupo" y "usuario_grupo" utilizando la condición "grupo.id = u
- Utiliza la función de agregación "COUNT(usuario_grupo.us_id)" para contar la cantidad de usuarios
- Utiliza la cláusula "GROUP BY grupo.nombre" para agrupar los resultados por el nombre del grupo.
*/

select g.nombre as grupo, count(ug.us_id) as cantidad_usuarios from grupo g , usuario_grupo ug where

```

Output Messages Notifications

grupo	cantidad_usuarios
character varying (25)	bigint
Maternal 1	2
Maternal intensivo 3	1
Maternal 2	1
Maternal intensivo 1	4
Maternal intensivo 2	1

```

212 /*
213 Consulta 2 :
214 - Debes seleccionar los nombres de los usuarios y los nombres de los grupos.
215 - Utiliza las tablas "usuarios", "grupo" y "usuario_grupo".
216 - Realiza una unión entre las tablas "usuarios" y "usuario_grupo" utilizando la condición "usua
217 - Realiza una unión entre las tablas "grupo" y "usuario_grupo" utilizando la condición "grupo.i
218 - Utiliza la condición "grupo.nombre LIKE '%intensivo%'" para filtrar los grupos que contengan
219 */
220 select u.nombre as nombre_usuario,g.nombre as nombre_grupo from usuarios u, grupo g ,usuario_gr
221 where ug.gr_id=g.id and ug.us_id=u.id and g.nombre LIKE '%intensivo%';
222
223 /*
224 Subconsulta 2 :
225 - Debes seleccionar el nombre de los usuarios.
226 - Utiliza la tabla "usuarios".
227 - Utiliza una subconsulta para obtener los "us_id" de la tabla "usuario_grupo" donde "gr_id" se

```

Data Output Messages Notifications

	nombre_usuario	nombre_grupo
	character varying (25)	character varying (25)
1	Marilyn	Maternal intensivo 1
2	Mario	Maternal intensivo 1
3	Johann	Maternal intensivo 2
4	Martín	Maternal intensivo 1

ata Output

Messages











Notifications

nombre

character varying (25)

Lucio

Data Output		Messages	Notifications
<div>       </div>			
grupo	máximo_valor	minimo_valor	
character varying (25)	integer	integer	
Maternal 1	10	5	
Maternal intensivo 3	9	9	
Maternal 2	6	6	
Maternal intensivo 1	8	1	
Maternal intensivo 2	4	4	

Data Output		Messages	Notifications	
				
				
				
	nombre character varying (25) 	fecha_creacion date 		
1	Romel	2020-03-08		
2	Monserate	2020-03-08		
3	Lucio	2020-03-08		
4	Berni	2020-03-08		

```

259  /*
260  Subconsultas 3 :
261  - Debes seleccionar el nombre de los usuarios.
262  - Utiliza la tabla "usuarios".
263  - Utiliza una subconsulta para obtener los "us_id" de la tabla "usuario_grupo" donde "gr_id" sea igual
264  - Utiliza la condición "id IN (SELECT ***** FROM ***** WHERE ***** = 3)".
265  */
266  select u.nombre from usuarios u where u.id IN (select us_id from usuario_grupo where gr_id=3);
267  /*
268  Función de agregación 3:

```

Data Output	Messages	Notifications
<div> <div>nombre</div> <div>character varying (25)</div> </div>		
1	Romel	

```

/*
Función de agregación 3:
- Debes seleccionar la descripción del grupo y contar la cantidad de "us_id" en la tabla "usuario_grupo"
- Utiliza las tablas "grupo" y "usuario_grupo".
- Realiza una unión entre las tablas "grupo" y "usuario_grupo" utilizando la condición "grupo.id = usuario_grupo.gr_id"
- Utiliza la función de agregación "COUNT(usuario_grupo.us_id)" para contar la cantidad de usuarios en cada grupo
- Utiliza la condición "grupo.descripcion LIKE '%matutino%'" para filtrar los grupos que contengan la palabra "matutino"
- Utiliza la cláusula "GROUP BY grupo.descripcion" para agrupar los resultados por la descripción del grupo
*/

select g.descripcion as descripcion_grupo, count(ug.us_id) as cantidad_usuarios
from grupo g , usuario_grupo ug
where ug.gr_id=g.id and g.descripcion LIKE '%matutino%' group by (g.descripcion);

```

Data Output	Messages	Notifications
<div>descripcion_grupo</div> <div>character varying (75)</div>	<div>cantidad_usuarios</div> <div>bigint</div>	
Grupo de maternal matutino	2	
Grupo de maternal matutino días sabados	4	

```

/*
Relación muchos a muchos entre habitaciones y huéspedes:
Consulta:
- Debes seleccionar el número de habitación, los nombres y apellidos de los huéspedes.
- Utiliza las tablas "habitaciones", "huespedes" y "reservas".
- Realiza una unión entre las tablas "habitaciones" y "reservas" utilizando la condición "habitaciones.id = reservas.habitacion_id"
- Realiza una unión entre las tablas "huespedes" y "reservas" utilizando la condición "huespedes.id = reservas.huespedes_id"
*/
select ha.habitacion_numero, hu.nombres, hu.apellidos from habitaciones ha, huespedes hu, reservas r
where ha.id = r.habitacion_id and hu.id = r.huespedes_id;

/*
Subconsulta:

```

Data Output	Messages	Notifications
<div>habitacion_numero</div> <div>integer</div>	<div>nombres</div> <div>character varying (45)</div>	<div>apellidos</div> <div>character varying (45)</div>
2	Bryan	Moncada
2	Diana	Monroy
1	Ricardo	Montero
1	Sofía	Martinez

```

4 Subconsulta:
5 - Debes seleccionar los nombres y apellidos de los huéspedes.
6 - Utiliza la tabla "huespedes".
7 - Utiliza una subconsulta para obtener los "huesped_id" de la tabla "reservas" donde "habitacion" sea
8 - Utiliza la condición "id IN (SELECT huesped_id FROM reservas WHERE habitacion = 2)".
9 */
10 select h.nombres,h.apellidos from huespedes h where h.id IN (SELECT huesped_id FROM reservas WHERE ha
11 /*
12 Función de agregación:
13 - Debes seleccionar el número de habitación y contar la cantidad de "huesped_id" en la tabla "reserva
14 - Utiliza las tablas "habitaciones" y "reservas".
15 - Realiza una unión entre las tablas "habitaciones" y "reservas" utilizando la condición "habitacione

```

ta Output Messages Notifications

nombres	apellidos
character varying (45)	character varying (45)
Diana	Monroy
Bryan	Moncada

```

301 /*
302 Función de agregación:
303 - Debes seleccionar el número de habitación y contar la cantidad de "huesped_id" en la tabla "reservas
304 - Utiliza las tablas "habitaciones" y "reservas".
305 - Realiza una unión entre las tablas "habitaciones" y "reservas" utilizando la condición "habitaciones
306 - Utiliza la función de agregación "COUNT(reservas.huesped_id)" para contar la cantidad de huéspedes e
307 - Utiliza la cláusula "GROUP BY habitaciones.habitacion_numero" para agrupar los resultados por el nú
308 */
309
310
311 select h.habitacion_numero,count(r.huesped_id) as cantidad_huespedes from habitaciones h, reservas r v

```

Data Output Messages Notifications

	habitacion_numero	cantidad_huespedes
	[PK] integer	bigint
1	11	1
2	9	1
3	10	1
4	2	2
5	12	1

```

13 /*
14 Consulta 2:
15 - Debes seleccionar el número de habitación, el piso, los nombres y apellidos de los huéspedes.
16 - Utiliza las tablas "habitaciones", "huespedes" y "reservas".
17 - Utiliza la condición "habitaciones.piso = 4" para filtrar las habitaciones que estén en el piso
18 - Realiza una unión entre las tablas "habitaciones" y "reservas" utilizando la condición "habitac
19 - Realiza una unión entre las tablas "huespedes" y "reservas" utilizando la condición "huespedes.
20 */
21 select ha.habitacion_numero,ha.piso, hu.nombres, hu.apellidos
22 from habitaciones ha, huespedes hu, reservas r
23 where ha.habitacion_numero = r.habitacion and hu.id = r.huesped_id
24 and ha.piso = 4;
25

```

ata Output Messages Notifications

habitacion_numero	piso	nombres	apellidos
integer	integer	character varying (45)	character varying (45)
2	4	Bryan	Moncada
2	4	Diana	Monroy
1	4	Ricardo	Montero
1	4	Sofia	Martinez

```

26  /*
27  Subconsulta 2:
28  - Debes seleccionar los nombres y apellidos de los huéspedes.
29  - Utiliza la tabla "huespedes".
30  - Utiliza una subconsulta para obtener los "huesped_id" de la tabla "reservas" donde "habitacion" sea
31  - Utiliza la condición "id IN (SELECT huesped_id FROM reservas WHERE habitacion = 3)".
32  */
33
34  select h.nombres,h.apellidos from huespedes h where h.id IN (SELECT huesped_id FROM reservas WHERE hab
35  /*
36  Función de agregación 2:

```

Data Output Messages Notifications

nombres	apellidos
character varying (45)	character varying (45)

```

35  /*
36  Función de agregación 2:
37  - Debes seleccionar el número de habitación y obtener el promedio de "huesped_id" en la tabla "res
38  - Utiliza las tablas "habitaciones" y "reservas".
39  - Realiza una unión entre las tablas "habitaciones" y "reservas" utilizando la condición "habitaci
40  - Utiliza la función de agregación "AVG(reservas.huesped_id)" para obtener el promedio de huéspedes
41  - Utiliza la cláusula "GROUP BY habitaciones.habitacion_numero" para agrupar los resultados por el
42  */
43  select h.habitacion_numero,round(avg(r.huesped_id),2) as promedio_huespedes from habitaciones h, r

```

Data Output Messages Notifications

habitacion_numero [PK] integer	promedio_huespedes numeric
11	6.00
9	8.00
10	7.00
2	9.50
12	5.00

Consulta 3:

- Debes seleccionar el número de habitación, los nombres y apellidos de los huéspedes.
- Utiliza las tablas "habitaciones", "huespedes" y "reservas".
- Realiza una unión entre las tablas "habitaciones" y "reservas" utilizando la condición "habitaciones.habitacion_numero = reservas.habitacion_numero"
- Realiza una unión entre las tablas "huespedes" y "reservas" utilizando la condición "huespedes.id = reservas.huesped_id"

```

/*
select ha.habitacion_numero, hu.nombres, hu.apellidos
from habitaciones ha, huespedes hu, reservas r
where ha.habitacion_numero = r.habitacion and hu.id = r.huesped_id;

```

Data Output Messages Notifications

habitacion_numero integer	nombres character varying (45)	apellidos character varying (45)
2	Bryan	Moncada
2	Diana	Monroy
1	Ricardo	Montero
1	Sofía	Martinez

```

Subconsulta 3:
- Debes seleccionar los nombres y apellidos de los huéspedes.
- Utiliza la tabla "huespedes".
- Utiliza una subconsulta para obtener los "huesped_id" de la tabla "reservas" donde "habitacion" sea igual
- Utiliza la condición "id IN (SELECT huesped_id FROM reservas WHERE habitacion = 4)".
*/
select nombres,apellidos from huespedes where id IN (SELECT huesped_id FROM reservas WHERE habitacion = 4);
/*

```

Output Messages Notifications

nombres character varying (45)	apellidos character varying (45)
-----------------------------------	-------------------------------------

```

366 Función de agregación 3:
367 - Debes seleccionar el número de habitación y obtener la suma de "precio_por_noche" en la tabla "habitaciones"
368 - Utiliza las tablas "habitaciones" y "reservas".
369 - Realiza una unión entre las tablas "habitaciones" y "reservas" utilizando la condición "habitaciones.habi
370 - Utiliza la función de agregación "SUM(habitaciones.precio_por_noche)" para obtener el total recaudado por
371 - Utiliza la cláusula "GROUP BY habitaciones.habitacion_numero" para agrupar los resultados por el número de
372 */
373
374 select h.habitacion_numero, sum(h.precio_por_noche) from habitaciones h,reservas r where h.habitacion_numero

```

Data Output Messages Notifications

	habitacion_numero [PK] integer	sum numeric
1	11	15.00
2	9	20.00
3	10	15.00

Relación muchos a muchos entre municipio y proyectos:

Consulta:

```

- Debes seleccionar el nombre del municipio y el nombre del proyecto.
- Utiliza las tablas "municipio", "proyecto" y "proyecto_municipio".
- Realiza una unión entre las tablas "municipio" y "proyecto_municipio" utilizando la condición "municipio.id
- Realiza una unión entre las tablas "proyecto" y "proyecto_municipio" utilizando la condición "proyecto.id
*/
select m.nombre as nombre_municipio, pr.proyecto as nombre_proyecto from municipio m, proyecto pr,proyecto_m
/*

```

Subconsulta:

```

- Debes seleccionar el nombre del proyecto.
- Utiliza la tabla "proyecto".
- Utiliza una subconsulta para obtener los "proyecto_id" de la tabla "proyecto_municipio" donde "municipio_id
- Utiliza la condición "id IN (SELECT ***** FROM ***** WHERE ***** = 1)".

```

Función de agregación:

```

- Debes seleccionar el nombre del municipio y contar la cantidad de "proyecto_id" en la tabla "proyecto_mun

```

Output Messages Notifications

nombre_municipio character varying (45)	nombre_proyecto character varying (50)
MUNIUCIPIO DE GUAYAQUIL	Proyectos de Gestión Ambiental
MUNIUCIPIO DE GUAYAQUIL	Proyectos en Cambio Climático
GAD MUNICIPAL DE CUENCA	Proyectos de Gestión Ambiental
GAD MUNICIPAL QUITO	Proyectos en Fomento y Desarrollo Productivo
MUNICIPALIDAD DE MACAHALA	Proyectos de Gestión Ambiental

Subconsulta:

- Debes seleccionar el nombre del proyecto.
- Utiliza la tabla "proyecto".
- Utiliza una subconsulta para obtener los "proyecto_id" de la tabla "proyecto_municipio" donde "municipio_id" = 1.
- Utiliza la condición "id IN (SELECT ***** FROM ***** WHERE ***** = 1)".

```
*/
select pr.proyecto as nombre_proyecto from proyecto pr where pr.id IN (SELECT proyecto_id FROM proyecto_municipio where municipio_id = 1)
/*
```

Output Messages Notifications

nombre_proyecto	
character varying (50)	
Proyectos en Fomento y Desarrollo Productivo	

/*

Función de agregación:

- Debes seleccionar el nombre del municipio y contar la cantidad de "proyecto_id" en la tabla "proyecto_municipio".
- Utiliza las tablas "municipio" y "proyecto_municipio".
- Realiza una unión entre las tablas "municipio" y "proyecto_municipio" utilizando la condición "municipio_id = proyecto_municipio.municipio_id".
- Utiliza la función de agregación "COUNT(proyecto_municipio.proyecto_id)" para contar la cantidad de proyecto_id.
- Utiliza la cláusula "GROUP BY municipio.nombre" para agrupar los resultados por el nombre del municipio.

```
select m.nombre,COUNT(pm.proyecto_id) as cantidad_proyecto from municipio m,proyecto_municipio pm where m.municipio_id = pm.municipio_id
/*
```

Output Messages Notifications

nombre	cantidad_proyecto	
character varying (45)	bigint	
MUNICIPIO DE LOJA	1	
MUNICIPALIDAD DE MACAHALA	1	
MUNICIPIO DE OTAVALO	1	
GAD MUNICIPAL QUITO	1	
MUNICIPIO DE LATACUNGA	1	

/*

Consulta 2:

- Debes seleccionar el nombre del municipio y el nombre del proyecto.
- Utiliza las tablas "municipio", "proyecto" y "proyecto_municipio".
- Realiza una unión entre las tablas "municipio" y "proyecto_municipio" utilizando la condición "municipio_id = proyecto_municipio.municipio_id".
- Realiza una unión entre las tablas "proyecto" y "proyecto_municipio" utilizando la condición "proyecto_id = proyecto_municipio.proyecto_id".
- Utiliza la condición "municipio.nombre LIKE '%GAD%'" para filtrar los municipios que contengan la palabra "GAD".

```
*/
select m.nombre as nombre_municipio, p.proyecto as nombre_proyecto
from municipio m, proyecto p,proyecto_municipio pm
where m.id=pm.municipio_id and p.id=pm.proyecto_id and m.nombre LIKE '%GAD%';
/*
```

Función de agregación 2:

- Debes seleccionar el nombre del municipio y obtener el mínimo valor de "proyecto_id" en la tabla "proyecto_municipio".
- Utiliza las tablas "municipio" y "proyecto_municipio".

Output Messages Notifications

nombre_municipio	nombre_proyecto
character varying (45)	character varying (50)
GAD MUNICIPAL DE CUENCA	Proyectos de Gestión Ambiental
GAD MUNICIPAL QUITO	Proyectos en Fomento y Desarrollo Productivo
GAD MUNICIPALIDAD DE AMBATO	Proyectos de Gestión Ambiental

Función de agregación 2:

- Debes seleccionar el nombre del municipio y obtener el mínimo valor de "proyecto_id" en la tabla "proyecto_municipio".
- Utiliza las tablas "municipio" y "proyecto_municipio".
- Realiza una unión entre las tablas "municipio" y "proyecto_municipio" utilizando la condición "municipio.id=proyecto_municipio.municipio_id".
- Utiliza la función de agregación "MIN(proyecto_municipio.proyecto_id)" para obtener el valor mínimo de proyecto_id.
- Utiliza la cláusula "GROUP BY municipio.nombre" para agrupar los resultados por el nombre del municipio.

*/

```
select m.nombre as nombre_municipio,min(pm.proyecto_id) as minimo_valor from municipio m, proyecto_municipio pm
```

Output Messages Notifications



nombre_municipio	minimo_valor
character varying (45)	integer
MUNICIPIO DE LOJA	2
MUNICIPALIDAD DE MACAHALA	1
MUNICIPIO DE OTAVALO	3
GAD MUNICIPAL QUITO	2
MUNICIPIO DE LATACUNGA	2

1 row(s) of 2 columns Query complete 00:00:00.288

Ln 425, Col 152

/*

Consulta 3 :

Selecciona el nombre del municipio y el nombre de la ciudad.

Se realiza una unión entre las tablas "municipio" y "ciudad" utilizando la condición adecuada.

*/

```
select m.nombre as nombre_municipio, c.nombre as nombre_ciudad from municipio m,ciudad c where m.ciudad_id=c.ciudad_id
```

/*

Subconsulta 3:

Selecciona el proyecto de la tabla "proyecto" donde el id está presente en la subconsulta que obtiene los proyectos de la tabla "proyecto_municipio".

Función de agregación 3:

Output Messages Notifications



nombre_municipio	nombre_ciudad
character varying (45)	character varying (45)
GAD MUNICIPAL QUITO	Quito
GAD MUNICIPAL DE CUENCA	Cuenca
GAD MUNICIPALIDAD DE AMBATO	Ambato
MUNICIPALIDAD DE MACAHALA	Machala
MUNICIPIO DE GUAYAQUIL	Guayaquil

10 row(s) of 2 columns Query complete 00:00:00.351

Ln 433, Col 38

/*

Subconsulta 3:

Selecciona el proyecto de la tabla "proyecto" donde el id está presente en la subconsulta que obtiene los proyectos de la tabla "proyecto_municipio".

*/

```
select *from proyecto where id IN (select proyecto_id from proyecto_municipio where municipio_id=3);
```

/*

Función de agregación 3:

Selecciona el nombre del municipio y se obtiene el valor máximo de proyecto_id en la tabla "proyecto_municipio".

Output Messages Notifications



id	proyecto	monto	fecha_inicio	fecha_entrega
[PK] integer	character varying (50)	money	date	date
1	Proyectos de Gestión Ambiental	\$9,543.33	2022-02-10	2023-02-10

```
/*
Función de agregación 3:
Selecciona el nombre del municipio y se obtiene el valor máximo de proyecto_id en la tabla "proyecto_municipio"
Se realiza una unión entre las tablas "municipio" y "proyecto_municipio" y se agrupa por el nombre del municipio
*/

select m.nombre, max(pm.proyecto_id) from municipio m, proyecto_municipio pm where m.id=pm.municipio_id group by m.nombre
```

Output Messages Notifications

nombre	max
character varying (45)	integer
MUNICIPIO DE LOJA	2
MUNICIPALIDAD DE MACAHALA	1
MUNICIPIO DE OTAVALO	3
GAD MUNICIPAL QUITO	2
MUNICIPIO DE LATACUNGA	2