

Examen Base de Datos I

dcr7juilland@gmail.com Cambiar de cuenta



Borrador guardado

*Obligatorio

Preguntas

12/5/22, 10:18 PM

Examen Base de Datos I Basándonos en el siguiente DER. Esta query retorna un listado de los usuarios cuyo ID esté entre 1 y 100 y que su nombre de usuario comience con A: SELECT * FROM usuarios WHERE idUsuario BETWEEN 1 AND 100 AND nombreUsuario LIKE = "A%"; Usuarios Bibliotecas PK idBiblioteca int NOT NULL idUsuario int NOT NULL Contrasenas nombreUsuario char(50) NOT NULL FK1 idUsuario int NOT NULL idContrasena int NOT NULL credito float cantidadJuegos int NOT NULL idUsuario int NOT NULL fechalnscripcion date NOT NULL contrasena varchar(50) NOT NULL pais varchar(50) NOT NULL JuegoXBiblioteca fecha date NOT NULL ciudad varchar(50) NOT NULL idJuegoXBiblioteca NOT NULL dirreccion varchar(50) NOT NULL idBiblioteca int NOT NULL IdJuego int NOT NULL Desarrolladoras VideoJuegos fechaAdg date NOT NULL idDesarrolladora int NOT NULL idVideoJuego int NOT NULL nombre varchar(50) NOT NULL

nombre varchar(50) NOT NULL

idDesarrolladora int NOT NULL

idEditorial int NOT NULL

precio float NOT NULL

Editorales

idEditorial int NOT NULL

cantidadJuegos int pais varchar(50) NOT NULL

nombre varchar(50) NOT NULL

VERDADERO

FALSO

cantidadJuegos int

pais varchar(50) NOT NULL

La siguiente query *

SELECT country.Name, COALESCE(country.Population, city.Population)
FROM country
LEFT JOIN city ON country.Code = city.CountryCode
WHERE country.Population IS NOT NULL

Siempre muestra la población de los países

Siempre muestra la población de las ciudades

A veces muestra la población de los países y a veces la de las ciudades, dependiendo de los datos de la base de datos.

La ejecución de consulta del siguiente gráfico, ¿Qué datos devuelve? *

SELECT cl.id, cl.color FROM coche ch
INNER JOIN marca ma ON ma.id = ch.id_marca
RIGHT JOIN color cl ON cl.id = ch.id_color
WHERE ch.id_color IS NULL;

	coche					
id	d_marca	modelo	anio	d_color		precio
1	3	UNO	2000	1	\$	250.000,00
2	4	FIESTA MA	2015	3	\$	514.000,00
3	3	SIENA	2008	2	\$	288.000,00
4	3	GRAN SIEN	2020	1	\$	1.720.400,00
5	4	FALCON	1980	4	\$	170.000,00
6	3	UNO	2009	2	\$	385.000,00
7	8	COROLLA	2018	1	\$	2.250.500,00
8	1	CORSA	2012	1	\$	413.500,00
9	7	SANDERO	2016	4	\$	937.500,00
10	4	FIESTA MA	2012	3	\$	394.800,00

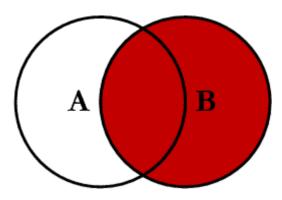
color				
id color				
1	ROJO			
2	AZUL			
3	VERDE			
4	CELESTE			
5	BLANCO			
6	GRIS			
7	NEGRO			

marca			
id	marca		
1	CHEVROL		
2	DODGE		
3	FIAT		
4	FORD		
6	NISSAN		
7	RENAULT		
8	TOYOTA		

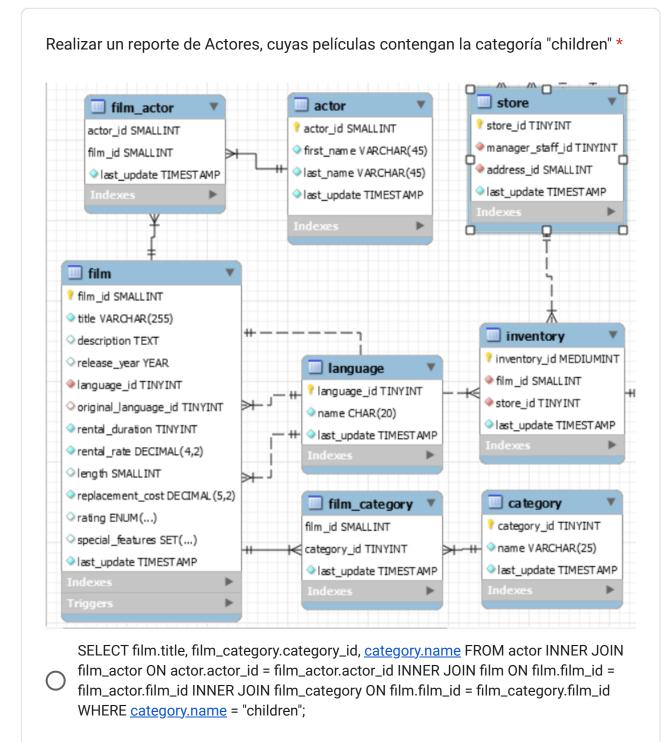
Devuelve el id y color de coches que no existen en la base de datos.

- Devuelve el id y color de coches que existen en la base de datos.
- La ejecución de la consulta no retorna datos.

La siguiente imagen representa a *



- O INNER JOIN
- LEFT JOIN
- RIGHT JOIN
- GROUP BY



SELECT actor.first_name, film.title, film_category.category_id, category.name FROM actor LEFT JOIN film_actor ON film_actor.actor_id = film_actor.actor_id LEFT JOIN film ON film.film_id = film.film_id LEFT JOIN film_category ON film_category.film_id = film_category.film_id LEFT JOIN category ON category.category_id = category.category_id WHERE category.name = "children";

SELECT actor.first_name, film.title, film_category.category_id, <u>category.name</u> FROM actor INNER JOIN film_actor ON actor.actor_id = film_actor.actor_id INNER JOIN film ON film.film_id = film_actor.film_id INNER JOIN film_category ON film.film_id = film_category.film_id INNER JOIN category ON film_category.category_id = category.category_id WHERE <u>category.name</u> = "children";

Observe la consulta del siguiente gráfico y seleccione la respuesta correcta *

SELECT * FROM cliente JOIN telefono;

cliente				
id apellido nombre		nombre	correo	
1	AGUILERA	ESTEBAN	aguilera80@gmail.com	
2	AGUIRRE	LUCIANA	aguirre.sj@gmail.com	
3	FLORES	ROBERTO	flores2000@gmail.com	
4	MANRIQUE	MARCELA	manrique.mar@gmail.com	
5	PEREZ	ALEJANDRO	sur25perez@gmail.com	

	telefono				
id	id_cliente	prefijo	numero		
1	3	0264	4245588		
2	1	0261	4218877		
3	2	0351	43355141		
4	2	011	43377125		
5	1	0264	4214277		
6	5	0261	4285500		
7	1	011	43281214		

4		1. 1.			1
()	I a consulta no se	elecutara	nordije tjene i	in error de sintaxis
1		La consulta no se	cjecutara	porque tierie t	ın error de sintaxis.

La consulta no se ejecutar	á porque tiene un	error en la sem	nántica del lenguaie
La consulta no se ejecular	a porque tiene un	i citor città scit	iaritica aci icrigaaje

(La sintaxis de la	consulta es o	correcta v se ei	iecutará satis	factoriamente.
м	 La ciritanio ac id	i oonoanta co t	concota , cc c	jeodiaia oatio	i a o to i a i i ci i to.

	Ninguna	respuesta	es correcta.

¿Cuál de las siguientes sentencias SQL nos muestra los datos de las facturas y de los clientes cuando la ciudad de envío es Berlin? facturas clientes 💡 FacturaID INT dientesdemo ClienteID CHAR(5) ClienteID CHAR(5) ClienteID CHAR(5) Compania VARCHAR(40) EmpleadoID INT TipoClienteID CHAR(10) Contacto VARCHAR(30) → FechaFactura DATETIME → Titulo VARCHAR(30) FechaRegistro DATETIME Direction VARCHAR(60) FechaEnvio DATETIME Ciudad VARCHAR(15) EnvioVia INT Regiones VARCHAR(15) Transporte DOUBLE CodigoPostal VARCHAR(10) clientesdemografia Nom breEnvio VARCHAR (40) Pais VARCHAR(15) TipoClienteID CHAR(10) DireccionEnvio VARCHAR(60) Telefono VARCHAR(24) □ TipoClien teDesc LONGTEXT ○ CiudadEnvio VARCHAR(15) Fax VARCHAR (24) RegionEnvio VARCHAR(15) CodigoPostalEnvio VARCHAR(10) PaisEnvio VARCHAR(15) SELECT * FROM emarket.facturas f INNER JOIN clientes c ON f.ClienteID = c.ClienteID WHERE CiudadEnvio LIKE "Berlin"; SELECT * FROM emarket.facturas f INNER JOIN clientes c ON f.ClienteID = c.ClienteID LIKE "Berlin"; SELECT * FROM emarket.facturas f INNER JOIN clientes c ON f.Cliente = "Berlin"; Ninguna de las anteriores

La siguiente consulta: SELECT nombre_pelicula, genero_pelicula ORDER BY * nombre_pelicula WHERE nombre_pelicula LIKE 'Harry Potter'
O Devuelve todas las películas de Harry Potter ordenadas alfabéticamente por nombre
Oa error porque al LIKE le falta el signo '%'
O Devuelve la película llamada 'Harry Potter'.
Da error de sintaxis.

Basándonos en el siguiente DER. Si queremos obtener un listado con los países y * la cantidad de usuarios que tienen la query sería: Bibliotecas Usuarios PK idUsuario int NOT NULL PK idBiblioteca int NOT NULL Contrasenas nombreUsuario char(50) NOT NULL FK1 idUsuario int NOT NULL idContrasena int NOT NULL credito float cantidadJuegos int NOT NULL idUsuario int NOT NULL fechalnscripcion date NOT NULL contrasena varchar(50) NOT NULL pais varchar(50) NOT NULL JuegoXBiblioteca fecha date NOT NULL ciudad varchar(50) NOT NULL idJuegoXBiblioteca NOT NULL dirreccion varchar(50) NOT NULL idBiblioteca int NOT NULL IdJuego int NOT NULL Desarrolladoras VideoJuegos fechaAdg date NOT NULL idDesarrolladora int NOT NULL idVideoJuego int NOT NULL nombre varchar(50) NOT NULL nombre varchar(50) NOT NULL Editorales cantidadJuegos int idEditorial int NOT NULL idEditorial int NOT NULL pais varchar(50) NOT NULL idDesarrolladora int NOT NULL nombre varchar(50) NOT NULL precio float NOT NULL cantidadJuegos int pais varchar(50) NOT NULL SELECT pais, COUNT(idUsuario) FROM usuarios GROUP BY idUsuario; SELECT pais, SUM(pais) FROM usuarios GROUP BY pais; SELECT pais, COUNT(idUsuario) FROM usuarios GROUP BY pais; SELECT pais, SUM(idUsuario) FROM usuarios GROUP BY pais;

¿Cuál de las siguientes sentencias de SQL lista en orden alfabético (sin repetir) las marcas de los aviones que tienen vuelo en la fecha 3/10/21? dientes reservas 💡 ClienteId INT 💡 ReservaId INT Nom bre VARCHAR(45) FechaHoraRegistro DATE Apelido VARCHAR (45) Estado VARCHAR(45) FechaNacimiento DATETIME ♦ VueloId INT Sexo CHAR(1) ClienteId INT aviones AvionId INT uelos 🛄 Marca VARCHAR (45) 💡 VueloId INT partidasdestinos V aeropuertos Estado VARCHAR(45) AeropuertoId INT Disponibilidad INT 💡 PartidaDestinoId INT Num eroVuelo VARCHAR (45) Nom bre VARCHAR (45) Nombre VARCHAR(45) ♦ TipoVueloId INT AeropuertoId INT Codigo CHAR(3) AvionId INT PartidaId INT tiposvuelos horarios DestinoId INT TipoVueloId INT HorarioId INT Fecha DATE Nombre VARCHAR (45) Hora TIME HoraSalidaId INT Horal legadaId INT SELECT * FROM aviones INNER JOIN vuelos ON aviones. AvionId = vuelos. VueloId WHERE Fecha = '2021-10-03' ORDER BY Marca SELECT DISTINCT Marca FROM aviones INNER JOIN vuelos ON aviones. AvionId = vuelos. Vuelold ORDER BY Marca SELECT DISTINCT Marca FROM aviones INNER JOIN vuelos ON aviones. AvionId = vuelos. Vuelold WHERE Fecha = '2021-10-03' Ninguna de las anteriores

Una agenda telefónica en Excel, ¿Es un SGBD? *	
No, Excel no es un SGBD	
Sí, Excel es considerado como una base de datos.	
No, una base de datos debe contener relaciones	
O Si, es una base de datos SQL	

¿Cuál de las siguientes sentencias de SQL muestra el nombre y apellido del cliente y la cantidad de cuentas que tiene asociado a él, ordenado de menor a mayor? clients deposits 💡 id INT (11) ? id INT(11) name VARCHAR (50) date TIMESTAMP √ last_name VARCHAR(50) amount FLOAT ◇identification_number VARCHAR(50) id account INT (11) age INT 🔾 id_city INT(11) extractions > accounts __ cities ? id INT(11) id INT(11) ? id INT(11) date DATETIME account_number VARCHAR(50) name VARCHAR (50) amount FLOAT id_client INT(11) id_account INT (11) SELECT c.name, c.last_name, COUNT(a.id) as cant FROM clients c INNER JOIN accounts a ON c.id = a.id_client GROUP BY c.name, c.last_name ORDER BY cant SELECT c.last_name, COUNT(a.id) as cant FROM clients c INNER JOIN accounts a ON c.id = a.id_client GROUP BY c.name, c.last_name ORDER BY cant

SELECT <u>c.name</u>, COUNT(<u>a.id</u>) as cant FROM clients c INNER JOIN accounts a ON <u>c.id</u> = a.id_client GROUP BY <u>c.name</u>, c.last_name ORDER BY cant La ejecución de la consulta del siguiente gráfico, ¿Devuelve uno o más registros con algún dato nulo?

SELECT * FROM cliente

RIGHT JOIN telefono ON telefono.id_cliente = cliente.id;

	cliente					
id apellido nombre		nombre	correo			
1	AGUILERA	ESTEBAN	aguilera80@gmail.com			
2	AGUIRRE	LUCIANA	aguirre.sj@gmail.com			
3	FLORES	ROBERTO	flores2000@gmail.com			
4	MANRIQUE	MARCELA	manrique.mar@gmail.com			
5	PEREZ	ALEJANDRO	sur25perez@gmail.com			

	t	elefono	
id	id_cliente	prefijo	numero
1	3	0264	4245588
2	1	0261	4218877
3	2	0351	43355141
4	2	011	43377125
5	1	0264	4214277
6	5	0261	4285500
7	1	011	43281214

- Si, devuelve un registro con datos parcialmente nulos.
- Si, devuelve cinco registros con datos parcialmente nulos.
- Si, devuelve siete registros con datos parcialmente nulos.

No devuelve registros con datos parcialmente nulos.

Observe la consulta del siguiente gráfico y seleccione la respuesta correcta *

SELECT * FROM cliente JOIN telefono;

		client	te
id	apellido	nombre	correo
1	AGUILERA	ESTEBAN	aguilera80@gmail.com
2	AGUIRRE	LUCIANA	aguirre.sj@gmail.com
3	FLORES	ROBERTO	flores2000@gmail.com
4	MANRIQUE	MARCELA	manrique.mar@gmail.com
5	PEREZ	ALEJANDRO	sur25perez@gmail.com

	t	elefono	
id	id_cliente	prefijo	numero
1	3	0264	4245588
2	1	0261	4218877
3	2	0351	43355141
4	2	011	43377125
5	1	0264	4214277
6	5	0261	4285500
7	1	011	43281214

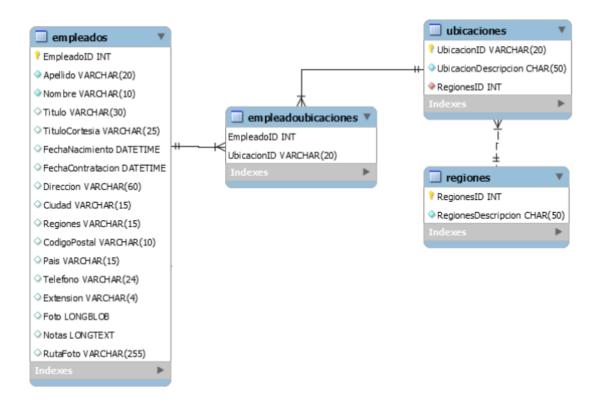
\bigcirc	La consulta generará un error de si	intaxis porque le falta	el RIGHT en la cláusula del
\cup	JOIN.		

(La consulta se	eiecutará	satisfactorian	nente v deve	olverá 35	reaistros.
- 1	La consulta sc	cjccatara	Satisfactoriari	incline y acv	orvera oo	regiotios.

C La consulta generará un error de sintaxis porque le falta el LEFT en la cláusula del JOIN.

Ninguna respuesta es correcta.
¿Cuál es la definición que está más relacionada a "Entidad"? *
Es un objeto, real o abstracto, acerca del cual se recoge información de interés para la base de datos.
Describe solo las características principales de un Objeto, ejemplo: Nombre,Apellido, Edad
Conjunto de uno o más atributos cuyos valores identifican unívocamente cada ocurrencia de la entidad.
on la representación de la realidad no comprometida con ningún entorno informático: modelo entidad-relación.

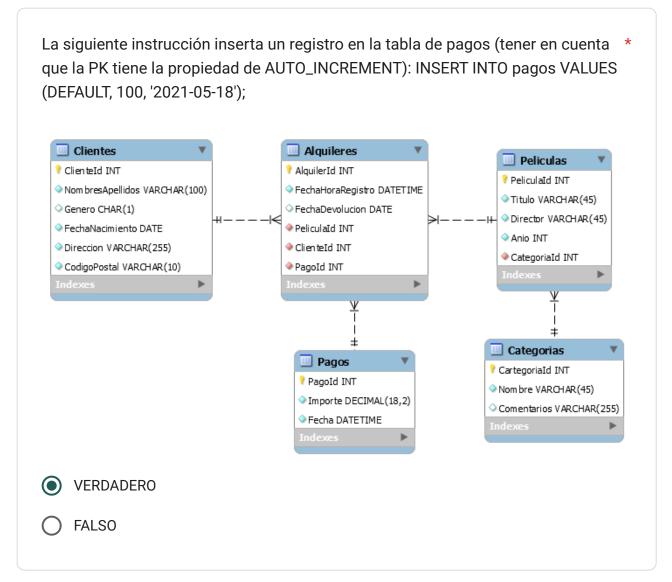
Conseguir un listado de todos los empleados cuyo nombre empiece con la letra F * y la descripción de la región sea "Southern"



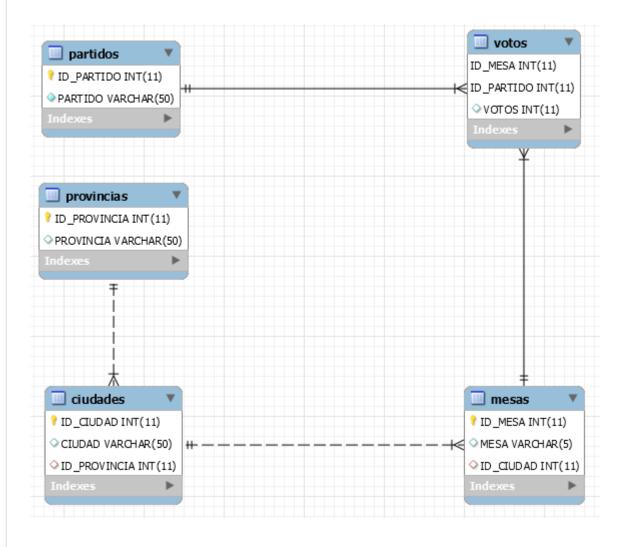
- SELECT e.Apellido, e.Nombre, r.RegionesDescripcion FROM empleadoubicaciones eu INNER JOIN empleados e ON eu.EmpleadoID = e.EmpleadoID INNER JOIN ubicaciones u ON eu.UbicacionID = u.UbicacionID INNER JOIN regiones r ON u.RegionesID = r.RegionesID WHERE r.RegionesDescripcion LIKE "Southern" AND e.Nombre LIKE 'F_';
- SELECT eu.Apellido, eu.Nombre, r.RegionesDescripcion FROM empleadoubicaciones eu INNER JOIN ubicaciones u ON eu.UbicacionID = u.UbicacionID INNER JOIN regiones r ON u.RegionesID = r.RegionesID WHERE r.RegionesDescripcion LIKE "Southern" AND e.Nombre LIKE '%F%';

SELECT e.Apellido, e.Nombre, r.RegionesDescripcion FROM empleadoubicaciones eu INNER JOIN empleados e ON eu.EmpleadoID = e.EmpleadoID INNER JOIN ubicaciones u ON eu.UbicacionID = u.UbicacionID INNER JOIN regiones r ON u.RegionesID = r.RegionesID WHERE r.RegionesDescripcion LIKE "Southern" AND e.Nombre LIKE '%F%';

SELECT e.Apellido, e.Nombre, r.RegionesDescripcion FROM empleadoubicaciones eu INNER JOIN empleados e ON eu.EmpleadoID = e.EmpleadoID INNER JOIN ubicaciones u ON eu.UbicacionID = u.UbicacionID INNER JOIN regiones r ON u.RegionesID = r.RegionesID WHERE r.RegionesDescripcion LIKE "Southern" AND e.Nombre LIKE 'F%';



¿Cuál de las siguientes sentencias de SQL cuenta la cantidad promedio de votos * para cada partido ordenado de menor a mayor por cantidad de votos?



SELECT P.PARTIDO, AVG (V.VOTOS) as 'cantidad de votos del partido' FROM

PARTIDOS as P INNER JOIN VOTOS as V ON P.ID_PARTIDO = V.ID_PARTIDO GROUP
BY P.PARTIDO ORDER BY AVG(V.VOTOS) ASC

SELECT P.PARTIDO, AVG (V.VOTOS) as 'cantidad de votos del partido' FROM PARTIDOS as P INNER JOIN VOTOS as V ON P.ID_PARTIDO = V.ID_PARTIDO GROUP BY V.VOTOS, P.PARTIDO ORDER BY AVG(V.VOTOS) DESC
SELECT P.PARTIDO, AVG (V.VOTOS) as 'cantidad de votos del partido' FROM PARTIDOS as P INNER JOIN VOTOS as V ON P.ID_PARTIDO = V.ID_PARTIDO GROUP BY P.PARTIDO ORDER BY AVG(V.VOTOS) DESC
SELECT P.PARTIDO, COUNT(V.VOTOS) as 'cantidad de votos del partido' FROM PARTIDOS as P INNER JOIN VOTOS as V ON P.ID_PARTIDO = V.ID_PARTIDO GROUP BY P.PARTIDO ORDER BY COUNT(V.VOTOS) ASC
La cláusula DISTINCT permite: *
La cláusula DISTINCT permite: * Devolver valores únicos
·
Devolver valores únicos

Se necesita conocer el total del rating de las películas por cada uno de los géneros al que pertenecen ¿Cuál de las sentencias SQL es la correcta? actor_movie created_at TIMEST AMP updated at TIMESTAME id INT actors actor_episode 77II bi 🕯 movie_id INT ? id INT title VARCHAR(500) created_at TIMESTAMP rating DECIMAL(3,1) first_nam e VARCHAR(100) awards INT actor_id INT last name VARCHAR(100) episode_id INT rating DECIMAL(3,1) length INT favorite_movie_id INT series 💡 id INT ? id INT 🕴 id INT created at TIMESTAME created_at TIMESTAMP created_at TIMESTAMP updated_at TIMESTAMP updated_at TIMESTAM updated_at TIMESTAM title VARCHAR(500) title VARCHAR(500) 💡 id INT created_at TIMESTAMP release date DATETIME release_date DATETIME release_date DATETIM! updated at TIMESTAMP end_date DATETIME rating DECIMAL(3,1) ond_date DATETIME name VARCHAR (100) genre_id INT serie_id INT season_id INT ranking INT active TINYINT (1) SELECT g.name 'Genero', sum(m.rating) "Total de Rating" FROM movies m INNER JOIN genres g ON m.genre_id = g.id GROUP BY g.name; SELECT g.name 'Genero', sum(m.rating) "Total de Rating" FROM movies m INNER JOIN genres g ON m.genre_id = g.id; SELECT g.name 'Genero', m.rating "Total de Rating" FROM movies m INNER JOIN genres g ON m.genre_id = g.id GROUP BY g.name; Ninguna de las anteriores

Atrás

Enviar

Página 2 de 2

Borrar formulario

Nunca envíes contraseñas a través de Formularios de Google.

Este formulario se creó en Digital House. Notificar uso inadecuado

Google Formularios