

**Comenzado el** viernes, 22 de noviembre de 2024, 19:03**Estado** Finalizado**Finalizado en** viernes, 22 de noviembre de 2024, 20:15**Tiempo empleado** 1 hora 12 minutos**Calificación** 9,50 de 10,00 (95%)**Pregunta 1**

Correcta

Se puntúa 0,50 sobre 0,50

**¿Cuántas filas devuelve?**

```
select count(*)
from (
  select 1 as numero
  union
  select avg(num) as num
  from ( select 0 as num union select 2 ) x
) x
```

Seleccione una:

- ☐ a. 0
- ☐ b. 2
- ☒ c. 1 ✓
- ☐ d. 3

**Respuesta correcta**

La respuesta correcta es: 1

**Pregunta 2**

Correcta

Se puntúa 0,50 sobre 0,50

**¿en qué orden se utilizan las sentencias siguientes para recorrer un cursor? (sin importar si están presentes todas instrucciones necesarias)**

Seleccione una:

- ☒ a. open --> fetch --> close ✓
- ☐ b. fetch --> declare --> open
- ☐ c. fetch --> open --> close
- ☐ d. declare --> fetch --> open

**Respuesta correcta**

La respuesta correcta es: open --&gt; fetch --&gt; close

**Pregunta 3**

Correcta

Se puntúa 0,50 sobre 0,50

¿Qué invocación y resultado son correctos y verdaderos?

```
drop function if exists fun_04;  
create function fun_04(x varchar(20))  
returns varchar(20)  
deterministic return x;
```

Seleccione una:

- ☐ a. call fun\_04("hola") --> devuelve una tabla de una fila y una columna, con el valor "hola"
- ☐ b. call fun\_04("123") --> devuelve 123.00
- ☒ c. select fun\_04("123") --> devuelve 123 ✓
- ☐ d. call fun\_04("hola") --> devuelve el string "hola"

**Respuesta correcta**

La respuesta correcta es: select fun\_04("123") --&gt; devuelve 123

**Pregunta 4**

Correcta

Se puntúa 0,50 sobre 0,50

Dada la tabla Pedidos= { id\_sucursal int, id\_pedido int, id\_cliente int, fecha date }. Sin PK ni FKs. ¿Por qué da error?

```
select id_pedido, count(*)  
from Pedidos  
where count(*)>1  
group by status, id_pedido  
having max(fecha)> "2023-06-01"  
order by count(*)
```

Seleccione una:

- ☐ a. porque el having tiene un agregado
- ☒ b. porque el where es incorrecto ✓
- ☐ c. porque los campos del group by difieren de los campos del select
- ☐ d. porque el order by tiene un agregado

**Respuesta correcta**

La respuesta correcta es: porque el where es incorrecto

**Pregunta 5**

Correcta

Se puntúa 0,50 sobre 0,50

¿qué error da la siguiente consulta?

```
select x.id, p.apellido, p.nombre, d.tipo, p.numero  
      , l.localidad, l.calle, numero  
from estudiante x  
join persona p on p.id_estudiante= x.id  
inner join documento t on t.id= p.id_documento  
left outer join localidad l on l.id= p.id_localidad  
where x.carrera in ( select id from carrera x );
```

Seleccione una:

- ☐ a. no da error
- ☐ b. Nombre nombre de columna es ambiguo (en el where)
- ☐ c. falta inner/left en join de persona
- ☒ d. número es un nombre de columna ambiguo (en el select) ✓

**Respuesta correcta**

La respuesta correcta es: número es un nombre de columna ambiguo (en el select)

**Pregunta 6**

Finalizado

Se puntúa 6,50 sobre 7,00

**Dado el siguiente DER**

EMPLEADO : Tabla								
codigo_c	nombre	edad	oficio	dir	fecha_alt	salario	comision	depto_no
281-160483-0005F	Rocha Vargas Hector	27	Vendedor	Leon	12/05/1983	12000	0	40
281-040483-0056P	López Hernandez Julio	27	Analista	Chinandega	14/07/1982	13000	1500	20
081-130678-0004S	Esquivel José	31	Director	Juigalpa	05/06/1981	16700	1200	30
281-160473-0009Q	Delgado Carmen	37	Vendedor	Leon	02/03/1983	13400	0	40
281-160493-0005F	Castillo Montes Luis	17	Vendedor	Masaya	12/08/1982	16309	1000	40
281-240784-0004Y	Esquivel Leonel Alfonso	26	Presidente	Nagarote	12/09/1981	15000	0	30
281-161277-0008R	Perez Luis	32	Empleado	Managua	02/03/1980	16890	0	10

  

DEPARTAMENTO : Tabla		
depto_no	nombre_depto	localizacion
10	Desarrollo Software	El Coyolar
20	Analisis Sistema	Guadalupe
30	Contabilidad	Subtiava
40	Ventas	San Felipe

**drop database if exists parcial;****create database parcial;****use parcial;****drop table if exists empelado;****drop table if exists departamento;****create table departamento (****depto\_nro int not null,****nombre\_depto varchar(100) not null,****localizacion varchar(100) not null,****CONSTRAINT PK\_depto\_nro PRIMARY KEY (depto\_nro)****);****create table empleado (****codigo\_c varchar(100) not null,****nombre varchar(100) not null,****edad tinyint not null,****oficio varchar(100) not null,****dir varchar(100) not null,****fecha\_alt date not null,****salario int not null,****comision int not null,****depto\_nro int not null,****CONSTRAINT PK\_codigo\_empleado PRIMARY KEY (codigo\_c),****CONSTRAINT FK\_depto\_nro FOREIGN KEY (depto\_nro) REFERENCES departamento(depto\_nro)****);****insert into departamento (depto\_nro,nombre\_depto,localizacion)****values (10,'desarrollo software','El Coyolar'),****(20,'analisis sistema','Guadalupe'),****(30,'contabilidad','Subtiava'),****(40,'ventas','san felipe');**

```

insert into empleado (codigo_c,nombre,edad,oficio,dir,fecha_alt,salario,comision,depto_nro)
values ('281-160483-0005f','Racha Vargas Hector',27,'Vendedor','Leon','1983-05-12',12000,0,40);
insert into empleado (codigo_c,nombre,edad,oficio,dir,fecha_alt,salario,comision,depto_nro)
values ('281-040483-0056p','Lopez Hernandez Julio',27,'Analista','Chinandega','1982-07-14',13000,1500,20);
insert into empleado (codigo_c,nombre,edad,oficio,dir,fecha_alt,salario,comision,depto_nro)
values ('281-130678-0004s','Esquivel Jose',31,'Director','Juigalpa','1981-06-05',16700,1200,30);
insert into empleado (codigo_c,nombre,edad,oficio,dir,fecha_alt,salario,comision,depto_nro)
values ('281-160473-0009q','Delgado Carmen',37,'Vendedor','Leon','1983-03-02',13400,0,40);
insert into empleado (codigo_c,nombre,edad,oficio,dir,fecha_alt,salario,comision,depto_nro)
values ('281-160493-0005f','Castillo Monthes Luis',17,'Vendedor','Masaya','1982-08-12',16309,1000,40);
insert into empleado (codigo_c,nombre,edad,oficio,dir,fecha_alt,salario,comision,depto_nro)
values ('281-243784-0004y','Esquivel Leonel Alfansa',26,'Presidente','Nandaime','1981-09-12',15000,0,30);
insert into empleado (codigo_c,nombre,edad,oficio,dir,fecha_alt,salario,comision,depto_nro)
values ('281-161277-0008r','Perez Luis',32,'Empleado','Managua','1980-03-02',16890,0,10);

```

### Se pide lo siguiente

- 1) Seleccionar el nombre, el oficio y la localizacion de los departamentos donde trabajan los vendedores.
- 2) Listar los nombres de los empleados cuyo nombre termine con la letra "o"
- 3) Seleccionar el nombre, salario y localidad donde trabajan los empleados que tengan un salario entre 10000 y 13000
- 4) Visualizar el nombre de los departamentos con más de 5 empleados
- 5) Mostrar los datos de los empleados que trabajan en el departamento de "contabilidad", ordenados por nombre
- 6) Calcular el salario mínimo de los empleados del departamento "VENTAS"
- 7) Visualizar la cantidad de empleados de cada departamento.
- 8) Para cada oficio obtener la suma de salarios
- 9) Mostrar los datos de los empleados cuyo salario sea mayor que la media de todos los salarios
- 10) Seleccionar el nombre del empleado que tiene máximo salario (resolver con subconsulta)
- 11) Mostrar los datos del empleado que tiene el salario más bajo en el departamento de "VENTAS" (resolver con subconsulta).
- 12) Incrementar en un 10% el sueldo de los empleados del departamento 10.
- 13) En la tabla EMPLEADO borrar todos los empleados que sean del departamento 20 y sean "ANALISTAS".
- 14) Crear una vista que devuelva un listado completo de todos los empleados incluyendo el nombre de los departamentos.

/\*1) Seleccionar el nombre, el oficio y la localizacion de los departamentos donde trabajan los vendedores.\*/

```

SELECT e.nombre, e.oficio, d.localizacion
FROM empleado e
JOIN departamento d
ON e.depto_nro = d.depto_nro
WHERE e.oficio = 'Vendedor';

```

/\*2) Listar los nombres de los empleados cuyo nombre termine con la letra "o"\*/

```

SELECT nombre
FROM empleado
WHERE nombre LIKE '%o';

```

/\*3) Seleccionar el nombre, salario y localidad donde trabajan los empleados que tengan un salario entre 10000 y 13000\*/

```

SELECT e.nombre, e.salario, d.localizacion
FROM empleado e
JOIN departamento d
ON e.depto_nro = d.depto_nro

```

```
WHERE e.salario BETWEEN 10000 AND 13000;
```

```
/* 4) Visualizar el nombre de los departamentos con más de 5 empleados*/
```

```
SELECT d.nombre_depto  
FROM departamento d  
JOIN empleado e ON d.depto_nro = e.depto_nro  
GROUP BY d.nombre_depto  
HAVING COUNT(e.codigo_c) > 5;
```

```
/*5) Mostrar los datos de los empleados que trabajan en el departamento de "contabilidad", ordenados por nombre*/
```

```
SELECT e.*  
FROM empleado e  
JOIN departamento d ON e.depto_nro = d.depto_nro  
WHERE d.nombre_depto = 'contabilidad'  
ORDER BY e.nombre;
```

```
/*6) Calcular el salario mínimo de los empleados del departamento "VENTAS"*/
```

```
SELECT MIN(e.salario) AS salario_minimo  
FROM empleado e  
JOIN departamento d ON e.depto_nro = d.depto_nro  
WHERE d.nombre_depto = 'ventas';
```

```
/*7) Visualizar la cantidad de empleados de cada departamento.*/
```

```
SELECT d.nombre_depto AS departamento, COUNT(e.codigo_c) AS cantidad_empleados  
FROM departamento d  
LEFT JOIN empleado e  
ON d.depto_nro = e.depto_nro  
GROUP BY d.depto_nro, d.nombre_depto;
```

```
/*8) Para cada oficio obtener la suma de salarios*/
```

```
SELECT oficio, SUM(salario) AS suma_salarios  
FROM empleado  
GROUP BY oficio;
```

```
/*9) Mostrar los datos de los empleados cuyo salario sea mayor que la media de todos los salarios*/
```

```
SELECT *  
FROM empleado  
WHERE salario > (SELECT AVG(salario) FROM empleado);
```

```
/*10) Seleccionar el nombre del empleado que tiene máximo salario (resolver con subconsulta)*/
```

```
SELECT nombre  
FROM empleado  
WHERE salario = (  
    SELECT MAX(salario)  
    FROM empleado  
);
```

```
/*11) Mostrar los datos del empleado que tiene el salario más bajo en el departamento de "VENTAS" (resolver con subconsulta).*/
```

```
SELECT *  
FROM empleado  
WHERE salario = (  
SELECT MIN(salario)  
FROM empleado  
WHERE depto_nro = (  
SELECT depto_nro  
FROM departamento  
WHERE nombre_depto = 'ventas')) );
```

/\*12) Incrementar en un 10% el sueldo de los empleados del departamento 10.\*/

```
UPDATE empleado  
SET salario = salario * 1.10  
WHERE depto_nro = 10;
```

/\*13) En la tabla EMPLEADO borrar todos los empleados que sean del departamento 20 y sean "ANALISTAS".\*/

```
SELECT *  
FROM empleado  
WHERE depto_nro = 20 AND oficio = 'Analista';
```

```
DELETE FROM empleado  
WHERE depto_nro = 20 AND oficio = 'Analista';
```

/\*14) Crear una vista que devuelva un listado completo de todos los empleados incluyendo el nombre de los departamentos.\*/

```
CREATE VIEW VistaEmpleadosDepartamentos  
AS SELECT e.codigo_c, e.nombre, e.edad, e.oficio, e.dir, e.fecha_alt, e.salario, e.comision, d.nombre_depto  
FROM empleado e  
JOIN departamento d  
ON e.depto_nro = d.depto_nro;
```

```
SELECT * FROM VistaEmpleadosDepartamentos;
```

```
 RodrigoPintoComisionD.sql
```

Comentario:  
punto 11 incorrecto

**Pregunta 7**

Correcta

Se puntúa 0,50 sobre 0,50

¿Qué afirmación es VERDADERA sobre la cláusula GROUP BY?

Seleccione una:

- ☒ a. Para agrupar filas que tengan los mismos valores para un subconjunto de columnas ✓
- ☐ b. GROUP BY requiere el uso de HAVING
- ☐ c. Para recuperar filas, se utiliza la cláusula GROUP BY.
- ☐ d. GROUP BY requiere el uso de alguna función agregada o "de agregación" en el SELECT o en el HAVING

**Respuesta correcta**

La respuesta correcta es: Para agrupar filas que tengan los mismos valores para un subconjunto de columnas