Primer Parcial DNI: _____ Nombre y Apellido: _____ TEMA 1
Universidad Nacional de La Matanza - Base de datos **09/11/2023**

SQL - PRÁCTICA

1) Dada la siguiente base de datos: Proveedor (CODPROV, NOMBRE), Provee(***CODMAT, CODPROV***), y la siguiente consulta, indique a cuál enunciado se corresponde:
Referencias: PK, **FK**, ***PK+FK***

```
SELECT CODPROV  FROM  PROVEE PM1 WHERE CODMAT=35
AND CODPROV NOT IN  (SELECT  CODPROV FROM PROVEE PM2 WHERE CODMAT<>35)
```

a) Materiales cuyo código no es 35
b) Proveedores que proveen solamente el material 35
c) Proveedores que no proveen el material 35
d) El resultado de la consulta siempre dará vacío
e) Ninguna de las anteriores

2) Dada la siguiente base de datos Almacen (Nro, nombre), Tiene (***Nro_almacen, CodArt***), y la siguiente consulta, indique la afirmación correcta:
Referencias: PK, **FK**, ***PK+FK***

```
SELECT t.* from TIENE t where t.CodArt= 1
INTERSECT
SELECT t.* from TIENE t where t.CodArt= 2
```

a) La consulta lista los almacenes que tienen los artículos 1 y los artículos 2 (ambos)
b) La consulta lista los Almacenes que tienen solamente el artículo 1 y el artículo 2
c) El resultado de la consulta es vacío
d) La consulta da error
e) Ninguna de las anteriores

Dadas las siguientes tablas: Artículos (codart, descrip, precio), Vende (id, **vendedor, articulo,** cant) y Vendedor (codvend, nombre, apellido)
Referencias: PK, **FK**, ***PK+FK***
Escribir el código SQL que da solución a los siguientes requerimientos:

3) Listar los códigos de los vendedores que vendieron todos los artículos

```
OPCION 1
select v.*
from vendedor v
where not exists (
                        select 1 from articulo a
                        where not exists (
                                                select 1 from vende ven
                                                where ven.articulo = a.id and
ven.vendedor=v.id))

OPCION 2
select distinct ven.vendedor
from vende ven
group by ven.vendedor
having count (distinct ven.articulo) = (select count(a.id) from articulo a)
```

4) Listar los códigos de los vendedores que no vendieron el artículo 95

```
– OPCION con EXCEPT. Pueden usar NOT IN, NOT EXISTS…
select v.id
from vendedor v
```

```
except
select v.vendedor
from vende v
where v.articulo = 95
```

5) Listar los códigos de los vendedores que no vendieron ningun articulo

```
– OPCION con EXCEPT. Pueden usar NOT IN, NOT EXISTS…
select v.id
from vendedor v
except
select v.vendedor
from vende v
```

6) Listar los códigos de los vendedores y la cantidad total de artículos vendidos por cada uno

```
select v.vendedor, sum (v.cant) as cantidad
from vende v
group by v.vendedor
```

7) Listar los códigos de los artículos de mayor precio

```
select *
from articulo where precio =
(select max(precio)
from articulo )
```

8) Listar los códigos de los artículos más vendidos

```
create view art_y_ventas as
select v.articulo, sum(cant) as cantidad from vende v group by v.articulo

select a.articulo from art_y_ventas a
where a.cantidad =
(select max(cantidad) from art_y_ventas)
```

SQL TEORÍA

9) Cuando utilizamos la cláusula GROUP BY en una consulta:

a) Siempre podremos usar en el select cualquier campo de las tablas que aparecen en el FROM, independientemente de los campos que se usen en el GROUP BY.
b) Todos los campos que se usen en el GROUP BY, obligatoriamente deben estar en el SELECT
c) Los campos listados en el SELECT, que no sean funciones agregadas, deben estar en el GROUP BY
d) Todos los campos que se usen enel GROUP BY, obligatoriamente deben aparecer en el HAVING
e) Ninguna de las anteriores

10) Indicar cuál de las siguientes es una CONSTRAINT que puede utilizarse en un CREATE TABLE:

a) UNIQUE
b) FOREIGN KEY
c) CHECK
d) NOT NULL
e) Todas las anteriores

11) Dadas las siguientes tablas: Empleado (legajo int, sueldo float), Asignado (***legajo_emp int, id_proyecto int, fecha date***), Proyecto (id int, descripción char(30)) y el siguiente Trigger:
Referencias: PK, **FK**, ***PK+FK***

```
CREATE TRIGGER examen ON Proyecto INSTEAD OF delete
AS
BEGIN
        DELETE FROM Asignado
        WHERE Asignado.id_proyecto IN (SELECT id FROM inserted)
END
```

Seleccione la respuesta correcta

a) El trigger no podría crearse porque tiene errores de sintaxis
b) El trigger elimina las tuplas de la tabla Asignado para los proyectos que se han eliminado de la tabla Proyecto.
c) El trigger elimina las tuplas de la tabla Asignado para los proyectos que se han intentado eliminar de la tabla Proyecto.
d) Luego de la ejecución del trigger, no se han eliminado tuplas ni de la tabla Proyecto ni de la tabla Asignado.
e) Ninguna de las anteriores

12) Indicar cuál de las siguientes es una sentencia DML de SQL:

a) INSERT INTO
b) UPDATE
c) DELETE
d) SELECT
e) Todas las anteriores

13) Seleccione la afirmación correcta:

a) Los Stored Procedures siempre deben finalizar con una sentencia de RETURN
b) Las Funciones escalares pueden o no recibir parámetros pero siempre deben retornar un valor.
c) Los Triggers siempre deben recibir un parámetro para su invocación y puedo acceder a los mismos haciendo un SELECT en las tablas inserted o deleted.
d) Las Funciones que devuelven una tabla deben invocarse con EXEC
e) Ninguna de las anteriores

14) Dada la siguiente función, la cual retorna la cantidad de días de diferencia entre la fecha actual y la recibida por parámetro:

```
create function f_mi_funcion (@fecha date) RETURNS int
begin
        RETURN DATEDIFF(DAY,@fecha,getdate())
end
```

La forma correcta de invocarla es:

a) SELECT fecha FROM dbo.f_mi_funcion
b) SELECT dbo.f_mi_funcion
c) EXEC dbo.f_mi_funcion
d) EXEC dbo.f_mi_funcion '01/01/2022'
e) Ninguna de las anteriores

15) ¿Cuál de las siguientes sentencias SQL se debe utilizar si desea modificar la tabla denominada "mi_tabla" de forma de agregar una clave primaria sobre los campos "campo_clave_uno" y "campo_clave_dos"?

a) ALTER TABLE mi_tabla ADD pk_mi_tabla PRIMARY KEY (campo_clave_uno+campo_clave_dos)
b) ALTER TABLE mi_tabla ADD PRIMARY KEY ON (campo_clave_uno and campo_clave_dos)
c) ALTER TABLE mi_tabla ADD CONSTRAINT pk_mi_tabla PRIMARY KEY (campo_clave_uno, campo_clave_dos)
d) MODIFY TABLE mi_tabla ADD PRIMARY KEY (campo_clave_uno, campo_clave_dos)
e) Ninguna de las opciones es correcta

TRANSACCIONES - TEORÍA

16) Sucede una lectura sucia cuando:

a) Aparecen nuevas filas que cumplen una determinada condición de búsqueda y que inicialmente no fueron vistas cuando se consultaron las filas que cumplen ese criterio.
b) Una transacción tiene acceso a las mismas filas varias veces y lee datos distintos en cada ocasión.
c) Una transacción lee un dato que ha sido modificado por otra transacción pero aún ese cambio no fue confirmado.
d) Una transacción no puede leer datos modificados por otras transacciones.
e) Ninguna de las anteriores

17) El nivel de aislamiento que previene la lectura fantasma es:
a) Read Uncommitted
b) Read Commited
c) Repeatable Read
d) Serializable
e) Ninguno de los anteriores

18) De acuerdo a las propiedades de una transacción, cuando decimos que: “la ejecución de una transacción no debe interferir con la ejecución de otra transacción.”, nos estamos refiriendo a:

a) Atomicidad
b) Durabilidad
c) Conservación de la consistencia
d) Aislamiento
e) Ninguna de las anteriores