

Nombre de la práctica	PRACTICA DE EVALUACIÓN-VISTAS			No.	3
Asignatura:	ADMINISTRACION DE BASE DE DATOS	Carrera:	INGENIERÍA EN SISTEMAS COMPUTACIONALES-3501	Duración de la práctica (Hrs)	2

Nombre del alumno: Jesús Navarrete Martínez

Grupo: 3601

PRACTICA 3 (Evaluación):

Un club dicta cursos de distintos deportes. Almacena la información en varias tablas. El director no quiere que los empleados de administración conozcan la estructura de las tablas ni algunos datos de los profesores y socios, por ello se crean vistas a las cuales tendrán acceso.

1- Elimine las tablas y créelas nuevamente:

```
-- Eliminamos las tablas si ya existen
if object_id('inscriptos', 'U') is not null
    drop table inscriptos;
if object_id('socios', 'U') is not null
    drop table socios;
if object_id('profesores', 'U') is not null
    drop table profesores;
if object_id('cursos', 'U') is not null
    drop table cursos;
```



```
-- Creamos la tabla "socios"
create table socios(
    documento char(8) not null,
    nombre varchar(40),
    domicilio varchar(30),
    constraint PK_socios_documento
    primary key (documento)
);
GO

-- Creamos la tabla "profesores"
create table profesores(
    documento char(8) not null,
    nombre varchar(40),
    domicilio varchar(30),
    constraint PK_profesores_documento
    primary key (documento)
);
GO
```

```
-- Creamos la tabla "cursos"
create table cursos(
    numero tinyint identity,
    deporte varchar(20),
    dia varchar(15),
    constraint CK_cursos_dia check (dia in ('lunes',
    'martes', 'miercoles', 'jueves', 'viernes', 'sabado')),
    documentoprofesor char(8),
    constraint PK_cursos_numero
    primary key (numero)
);
GO

-- Creamos la tabla "inscriptos"
create table inscriptos(
    documentosocio char(8) not null,
    numero tinyint not null,
    matricula char(1),
    constraint CK_inscriptos_matricula check (matricula in ('s', 'n')),
    constraint PK_inscriptos_documento_numero
    primary key (documentosocio, numero)
);
GO
```



2- Ingrese algunos registros para todas las tablas:

```
-- Insertamos datos en la tabla "socios"
insert into socios values ('30000000', 'Fabian Fuentes', 'Caseros 987');
insert into socios values ('31111111', 'Gaston Garcia', 'Guemes 65');
insert into socios values ('32222222', 'Hector Huerta', 'Sucre 534');
insert into socios values ('33333333', 'Ines Irala', 'Bulnes 345');
GO

-- Insertamos datos en la tabla "profesores"
insert into profesores values ('22222222', 'Ana Acosta', 'Avellaneda 231');
insert into profesores values ('23333333', 'Carlos Caseres', 'Colon 245');
insert into profesores values ('24444444', 'Daniel Duarte', 'Sarmiento 987');
insert into profesores values ('25555555', 'Esteban Lopez', 'Sucre 1204');
GO

-- Insertamos datos en la tabla "cursos"
insert into cursos values ('tenis', 'lunes', '22222222');
insert into cursos values ('tenis', 'martes', '22222222');
insert into cursos values ('natacion', 'miercoles', '22222222');
insert into cursos values ('natacion', 'jueves', '23333333');
insert into cursos values ('natacion', 'viernes', '23333333');
insert into cursos values ('futbol', 'sabado', '24444444');
insert into cursos values ('futbol', 'lunes', '24444444');
insert into cursos values ('basquet', 'martes', '24444444');
GO

-- Insertamos datos en la tabla "inscriptos"
insert into inscriptos values ('30000000', 1, 's');
insert into inscriptos values ('30000000', 3, 'n');
insert into inscriptos values ('30000000', 6, 'n');
insert into inscriptos values ('31111111', 1, 's');
insert into inscriptos values ('31111111', 4, 's');
insert into inscriptos values ('32222222', 8, 's');
GO
```

3- Elimine la vista "vista_club" si existe:

```
-- Eliminamos la vista "vista_club" si existe
if object_id('vista_club', 'V') is not null
    drop view vista_club;
GO
```

4- Cree una vista en la que aparezca el nombre y documento del socio, el deporte, el día y el nombre del profesor:

```
-- Creamos la vista "vista_club"
create view vista_club as
select
    s.nombre as socio,
    s.documento as docsocio,
    s.domicilio as domsocio,
    c.deporte,
    c.dia,
    p.nombre as profesor,
    i.matricula
from socios as s
full join inscriptos as i
on s.documento = i.documentosocio
full join cursos as c
on i.numero = c.numero
full join profesores as p
on c.documentoprofesor = p.documento;
GO
```

5- Muestre la información contenida en la vista:

```
-- Consultamos la información contenida en la vista
select * from vista_club;
GO
```

6-

	socio	docsocio	deporte	dia	profesor	matricula
1	Gaston Garcia	311111111	tenis	lunes	Ana Acosta	s
2	Gaston Garcia	311111111	natacion	jueves	Carlos Caseres	s
3	Hector Huerta	322222222	basquet	martes	Daniel Duarte	s
4	Ines Irala	333333333	NULL	NULL	NULL	NULL
5	NULL	NULL	tenis	lunes	Ana Acosta	s
6	NULL	NULL	natacion	miercoles	Ana Acosta	n
7	NULL	NULL	futbol	sabado	Daniel Duarte	NULL
8	NULL	NULL	futbol	lunes	Daniel Duarte	NULL
9	NULL	NULL	natacion	viernes	Carlos Caseres	NULL
10	NULL	NULL	tenis	martes	Ana Acosta	NULL
11	NULL	NULL	NULL	NULL	Esteban Lopez	NULL

- 6- Realice una consulta a la vista donde muestre la cantidad de socios inscritos en cada deporte ordenados por cantidad:

```
-- 6. Mostrar la cantidad de socios inscritos en cada deporte, ordenados por cantidad
select
    deporte,
    count(socio) as cantidad_socios
from vista_club
group by deporte
order by cantidad_socios desc;
GO
```

	deporte	cantidad
1	natacion	2
2	tenis	2
3	NULL	1
4	basquet	1
5	futbol	1

7- Muestre (consultando la vista) los cursos (deporte y día) para los cuales no hay inscritos:

```
-- 7. Mostrar los cursos (deporte y día) para los cuales no hay inscritos
select
    deporte,
    dia
from cursos c
left join inscriptos i on c.numero = i.numero
where i.numero is null;
GO
```

93 %

	deporte	dia
1	tenis	martes
2	natacion	viernes
3	futbol	lunes
4	NULL	NULL

8- Muestre los nombres de los socios que no se han inscrito en ningún curso (consultando la vista):

```
-- 8. Mostrar los nombres de los socios que no se han inscrito en ningún curso
select
    s.nombre
from socios s
left join inscriptos i on s.documento = i.documentosocio
where i.documentosocio is null;
GO
```

	nombre
1	Ines Irala

9- Muestre (consultando la vista) los profesores que no tienen asignado ningún deporte aún:

```
-- 9. Mostrar los profesores que no tienen asignado ningún deporte aún
```

```
select
  p.nombre as profesor
from profesores p
left join cursos c on p.documento = c.documentoprofesor
where c.documentoprofesor is null;
GO
```

Results		Messages
	nombre	
1	Esteban Lopez	

10- Muestre (consultando la vista) el nombre y documento de los socios que deben matrículas:

```
-- 10. Mostrar el nombre y documento de los socios que deben matrículas
```

```
select
  s.documento,
  s.nombre
from socios s
join inscriptos i on s.documento = i.documentosocio
where i.matricula = 'n';
GO
```

	socio	docsocio
1	Fabian Fuentes	30000000
2	Ines Irala	33333333
3	NULL	NULL
4	NULL	NULL
5	NULL	NULL
6	NULL	NULL

11- Consulte la vista y muestre los nombres de los profesores y los días en que asisten al club para dictar sus clases:

```
-- 11. Mostrar los nombres de los profesores y los días en que asisten al club para dictar sus clases
```

```
select distinct
  profesor,
  dia
from vista_club;
GO
```



	profesor	dia
1	Ana Acosta	lunes
2	Ana Acosta	lunes
3	Ana Acosta	martes
4	Ana Acosta	miercoles
5	Carlos Caseres	jueves
6	Carlos Caseres	viernes
7	Daniel Duarte	sabado
8	Daniel Duarte	lunes
9	Daniel Duarte	martes
10	NULL	NULL
11	Esteban Lopez	NULL

12- Muestre la misma información anterior pero ordenada por día:

```
-- 12. Mostrar lo anterior, pero ordenado por día
select distinct
    profesor,
    dia
from vista_club
order by dia;
GO
```

	profesor	dia
1	NULL	NULL
2	Esteban Lopez	NULL
3	Carlos Caseres	jueves
4	Daniel Duarte	lunes
5	Ana Acosta	lunes
6	Ana Acosta	lunes
7	Ana Acosta	martes
8	Daniel Duarte	martes
9	Ana Acosta	miercoles
10	Daniel Duarte	sabado
11	Carlos Caseres	viernes

13- Muestre todos los socios que son compañeros en tenis los lunes:

```
-- 13. Mostrar todos los socios que son compañeros en tenis los lunes
select
  s1.nombre as socio1,
  s2.nombre as socio2
from socios s1
join inscriptos i1 on s1.documento = i1.documentosocio
join cursos c on i1.numero = c.numero
join inscriptos i2 on c.numero = i2.numero
join socios s2 on i2.documentosocio = s2.documento
where c.deporte = 'tenis' and c.dia = 'lunes' and s1.documento < s2.documento;
GO
```

Results		Messages
	socio	
1	Fabian Fuentes	
2	Gaston Garcia	

14- Elimine la vista "vista_inscriptos" si existe y créela para que muestre la cantidad de inscriptos por curso, incluyendo el número del curso, el nombre del deporte y el día:

```
-- 14. Eliminamos la vista "vista_inscriptos" si existe
if object_id('vista_inscriptos', 'V') is not null
  drop view vista_inscriptos;
GO

-- Creamos la vista "vista_inscriptos" para mostrar la cantidad de inscriptos por curso
create view vista_inscriptos as
select
  c.numero as curso_numero,
  c.deporte,
  c.dia,
  count(i.documentosocio) as cantidad_inscriptos
from cursos c
left join inscriptos i on c.numero = i.numero
group by c.numero, c.deporte, c.dia;
GO
```

15- Consulte la vista:

```
-- 15. Consultamos la vista "vista_inscriptos"
select * from vista_inscriptos;
GO
```

93 %

Results Messages

	numero_curso	nombre_deporte	dia_curso	cantidad_inscriptos
1	1	tenis	lunes	2
2	2	tenis	martes	0
3	3	natacion	miercoles	1
4	4	natacion	jueves	1
5	5	natacion	viernes	0
6	6	futbol	sabado	1
7	7	futbol	lunes	0
8	8	basquet	martes	1

