

## CONCEPTOS DE BASES DE DATOS II

Programador Universitario, Ing. y Lic. en Informática  
Fac. de C. Exactas y Tecnología. – UNT



### Trabajo Práctico Nro. 3 - Ciclo 2024

#### Tema: Seguridad, Usuarios y Permisos

##### Ejercicio nro. 1:

Creación de grupos, usuarios y asignación de usuarios a un respectivo grupo:

GRUPOS	
CREATE ROLE pepej_grupo_informes WITH NOLOGIN NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;	CREATE ROLE pepej_grupo_admision WITH NOLOGIN NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;
CREATE ROLE pepej_grupo_rrhh WITH NOLOGIN NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;	CREATE ROLE pepej_grupo_medicos WITH NOLOGIN NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;
CREATE ROLE pepej_grupo_compras WITH NOLOGIN NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;	CREATE ROLE pepej_grupo_facturacion WITH NOLOGIN NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;
CREATE ROLE pepej_grupo_mantenimiento WITH NOLOGIN NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;	CREATE ROLE pepej_grupo_sistemas WITH NOLOGIN NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;
USUARIOS	
CREATE ROLE pepej_user_informes WITH LOGIN NOSUPERUSER NOCREATEDB NOCREATEROLE INHERIT NOREPLICATION CONNECTION LIMIT -1 PASSWORD 'xxxxxx';  GRANT pepej_grupo_informes TO pepej_user_informes;	CREATE ROLE pepej_user_admision WITH LOGIN NOSUPERUSER NOCREATEDB NOCREATEROLE INHERIT NOREPLICATION CONNECTION LIMIT -1 PASSWORD 'xxxxxx';  GRANT pepej_grupo_admision TO pepej_user_admision;

## CONCEPTOS DE BASES DE DATOS II

Programador Universitario, Ing. y Lic. en Informática  
Fac. de C. Exactas y Tecnología. – UNT



### Trabajo Práctico Nro. 3 - Ciclo 2024

#### Tema: Seguridad, Usuarios y Permisos

<pre>CREATE ROLE pepej_user_rrhh WITH LOGIN NOSUPERUSER NOCREATEDB NOCREATEROLE INHERIT NOREPLICATION CONNECTION LIMIT -1 PASSWORD 'xxxxxx';  GRANT pepej_grupo_rrhh TO pepej_user_rrhh;</pre>	<pre>CREATE ROLE pepej_user_medico WITH LOGIN NOSUPERUSER NOCREATEDB NOCREATEROLE INHERIT NOREPLICATION CONNECTION LIMIT -1 PASSWORD 'xxxxxx';  GRANT pepej_grupo_medicos TO pepej_user_medico;</pre>
<pre>CREATE ROLE pepej_user_compra WITH LOGIN NOSUPERUSER NOCREATEDB NOCREATEROLE INHERIT NOREPLICATION CONNECTION LIMIT -1 PASSWORD 'xxxxxx';  GRANT pepej_grupo_compras TO pepej_user_compra;</pre>	<pre>CREATE ROLE pepej_user_facturacion WITH LOGIN NOSUPERUSER NOCREATEDB NOCREATEROLE INHERIT NOREPLICATION CONNECTION LIMIT -1 PASSWORD 'xxxxxx';  GRANT pepej_grupo_facturacion TO pepej_user_facturacion;</pre>
<pre>CREATE ROLE pepej_user_mantenimiento WITH LOGIN NOSUPERUSER NOCREATEDB NOCREATEROLE INHERIT NOREPLICATION CONNECTION LIMIT -1 PASSWORD 'xxxxxx';  GRANT pepej_grupo_mantenimiento TO pepej_user_mantenimiento;</pre>	<pre>CREATE ROLE pepej_user_sistema WITH LOGIN NOSUPERUSER NOCREATEDB NOCREATEROLE INHERIT NOREPLICATION CONNECTION LIMIT -1 PASSWORD 'xxxxxx';  GRANT pepej_grupo_sistemas TO pepej_user_sistema;</pre>

#### Informes

- a) Mostrar datos de los pacientes, si tiene obra social, también el nombre de la misma.  
**GRANT SELECT ON** persona, paciente, obra\_social **TO** pepej\_grupo\_informes;

## CONCEPTOS DE BASES DE DATOS II

Programador Universitario, Ing. y Lic. en Informática  
Fac. de C. Exactas y Tecnología. – UNT



### Trabajo Práctico Nro. 3 - Ciclo 2024

#### Tema: Seguridad, Usuarios y Permisos

---

- b) Mostrar las consultas a las que asistió el paciente, también debe mostrar el medico que lo atendió, el diagnóstico y el tratamiento que le suministró.  
`GRANT SELECT ON consulta, diagnostico, tratamiento TO pepej_grupo_informes;`  
`GRANT SELECT(id_medicamento, nombre, presentacion) ON medicamento TO pepej_grupo_informes;`  
-- No hace falta dar permisos en la tabla empleado ya que solo se necesita el nombre y apellido del médico y se puede hacer JOIN entre consulta (o diagnostico) con persona, tabla a la cual ya tiene permiso.
- c) Mostrar las internaciones que tuvo el paciente, debe mostrar la habitación y la cama en la que estuvo.  
`GRANT SELECT(id_paciente, id_cama, fecha_inicio, fecha_alta, hora) ON internacion TO pepej_grupo_informes;`  
`GRANT SELECT(id_habitacion, piso, numero, tipo) ON habitacion TO pepej_grupo_informes;`  
`GRANT SELECT ON cama TO pepej_grupo_informes;`
- d) Mostrar los estudios que le realizaron, los equipos que se utilizaron y el profesional que le realizo el estudio.  
`GRANT SELECT ON equipo TO pepej_grupo_informes;`  
`GRANT SELECT(id_paciente, id_estudio, fecha, id_equipo, resultado, observacion) ON estudio_realizado TO pepej_grupo_informes;`  
`GRANT SELECT(id_estudio, nombre) ON estudio TO pepej_grupo_informes;`

#### Admisión

- a) Agregar, modificar o eliminar un paciente.  
`GRANT INSERT, UPDATE ON persona, paciente TO pepej_grupo_admision;`  
`GRANT DELETE ON paciente TO pepej_grupo_admision;`  
`GRANT SELECT ON persona, obra_social TO pepej_grupo_admision;`  
-- Se da permiso de selección en la tabla persona para poder buscar por sus datos a la hora de eliminar o actualizar, y en obra\_social para buscar la obra social por nombre y obtener su id.
- b) Listar consultas, tratamientos, diagnósticos y estudios realizados de un determinado paciente.  
`GRANT SELECT ON consulta, diagnostico, patología TO pepej_grupo_admision;`  
`GRANT SELECT(id_medicamento, nombre, presentacion) ON medicamento TO pepej_grupo_admision;`  
`GRANT SELECT(id_paciente, id_medicamento, fecha_indicacion, nombre, dosis) ON tratamiento TO pepej_grupo_admision;`  
`GRANT SELECT(id_paciente, id_estudio, fecha, resultado) ON estudio_realizado TO pepej_grupo_admision;`  
`GRANT SELECT(id_estudio, nombre) ON estudio TO pepej_grupo_admision;`  
-- No hace falta dar permisos en la tabla paciente ya que solo se necesita el nombre y apellido del paciente y se puede hacer JOIN entre consulta y persona, tabla a la cual ya tiene permiso.

## CONCEPTOS DE BASES DE DATOS II

Programador Universitario, Ing. y Lic. en Informática  
Fac. de C. Exactas y Tecnología. – UNT



### Trabajo Práctico Nro. 3 - Ciclo 2024

#### Tema: Seguridad, Usuarios y Permisos

---

- c) Agregar consultas.

**GRANT INSERT ON** consulta **TO** pepej\_grupo\_admision;

**GRANT SELECT ON** consultorio, paciente **TO** pepej\_grupo\_admision;

**GRANT SELECT(id\_empleado) ON** empleado **TO** pepej\_grupo\_admision;

-- Se da permiso sobre paciente y empleado solo para hacer los JOIN con persona y poder encontrar las personas que son pacientes y las que son empleados.

- d) Agregar estudios realizados.

**GRANT INSERT ON** estudio\_realizado **TO** pepej\_grupo\_admision;

**GRANT SELECT ON** equipo **TO** pepej\_grupo\_admision;

-- El resto de los permisos necesarios para las FK ya los tiene de los apartados anteriores.

- e) Listar, agregar, modificar internaciones.

**GRANT SELECT, INSERT, UPDATE ON** internacion **TO** pepej\_grupo\_admision;

**GRANT SELECT ON** cama **TO** pepej\_grupo\_admision;

**GRANT SELECT(id\_habitacion, numero, tipo) ON** habitacion **TO** pepej\_grupo\_admision;

-- El resto de los permisos necesarios para las FK ya los tiene de los apartados anteriores.

#### RRHH

- a) Agregar, modificar o eliminar empleados.

**GRANT SELECT, INSERT, UPDATE ON** persona, empleado **TO** pepej\_grupo\_rrhh;

**GRANT DELETE ON** empleado **TO** pepej\_grupo\_rrhh;

**GRANT SELECT ON** cargo, especialidad **TO** pepej\_grupo\_rrhh;

- b) Modificar los datos de los empleados, especialidad, cargo, horarios que cumplen.

**GRANT INSERT, UPDATE ON** trabajan **TO** pepej\_grupo\_rrhh;

**GRANT SELECT ON** turno **TO** pepej\_grupo\_rrhh;

-- El resto de los permisos ya los tiene del apartado anterior.

#### Médicos

- a) Agregar consultas.

**GRANT INSERT ON** consulta **TO** pepej\_grupo\_medicos;

**GRANT SELECT ON** persona, paciente, consultorio **TO** pepej\_grupo\_medicos;

**GRANT SELECT(id\_empleado) ON** empleado **TO** pepej\_grupo\_medicos;

- b) Agregar, modificar o eliminar tratamientos.

**GRANT INSERT, UPDATE, DELETE ON** tratamiento **TO** pepej\_grupo\_medicos;

**GRANT SELECT(id\_medicamento, nombre, presentacion) ON** medicamento **TO** pepej\_grupo\_medicos;

## CONCEPTOS DE BASES DE DATOS II

Programador Universitario, Ing. y Lic. en Informática  
Fac. de C. Exactas y Tecnología. – UNT



### Trabajo Práctico Nro. 3 - Ciclo 2024

#### Tema: Seguridad, Usuarios y Permisos

---

- c) Agregar, modificar o eliminar diagnósticos.

**GRANT INSERT, UPDATE, DELETE ON** diagnostico **TO** pepej\_grupo\_medicos;  
**GRANT SELECT ON** consulta, patologia **TO** pepej\_grupo\_medicos;

- d) Agregar, modificar o eliminar estudios realizados.

**GRANT INSERT, UPDATE, DELETE ON** estudio\_realizado **TO** pepej\_grupo\_medicos;  
**GRANT SELECT ON** equipo **TO** pepej\_grupo\_medicos;  
**GRANT SELECT(id\_estudio, nombre) ON** estudio **TO** pepej\_grupo\_medicos;  
-- El resto de los permisos necesarios para las FK ya los tiene de los apartados anteriores.

- e) Puede realizar todas las consultas que se realizan en Informes y Admisión.

**GRANT** pepej\_grupo\_informes **TO** pepej\_grupo\_medicos;  
**GRANT** pepej\_grupo\_admision **TO** pepej\_grupo\_medicos;

#### Compras

- a) Listar compras, mostrando proveedores, clasificación y laboratorio de cada insumo adquirido.

**GRANT SELECT ON** compra, proveedor, medicamento, clasificacion, laboratorio **TO** pepej\_grupo\_compras;

- b) Agregar laboratorios, clasificaciones, proveedores y medicamentos.

**GRANT INSERT ON** laboratorio, clasificacion, proveedor, medicamento **TO** pepej\_grupo\_compras;

- c) Modificar laboratorios, clasificaciones, proveedores y medicamentos.

**GRANT UPDATE ON** laboratorio, clasificacion, proveedor, medicamento **TO** pepej\_grupo\_compras;

- d) Eliminar laboratorios, clasificaciones, proveedores y medicamentos.

**GRANT DELETE ON** laboratorio, clasificacion, proveedor, medicamento **TO** pepej\_grupo\_compras;

#### Facturación

- a) Listar las facturas, mostrando los pacientes.

**GRANT SELECT ON** factura **TO** pepej\_grupo\_facturacion ;  
**GRANT SELECT(id\_persona, nombre, apellido, dni) ON** persona **TO** pepej\_grupo\_facturacion;

- b) Agregar, modificar y eliminar facturas.

**GRANT INSERT, UPDATE, DELETE ON** factura **TO** pepej\_grupo\_facturacion;

- c) Listar los pagos, mostrando el paciente.

**GRANT SELECT ON** pago **TO** pepej\_grupo\_facturacion;  
-- El resto de los permisos ya los tiene de los apartados anteriores.

## CONCEPTOS DE BASES DE DATOS II

Programador Universitario, Ing. y Lic. en Informática  
Fac. de C. Exactas y Tecnología. – UNT



### Trabajo Práctico Nro. 3 - Ciclo 2024

#### Tema: Seguridad, Usuarios y Permisos

---

- d) Agregar modificar y eliminar pagos.

**GRANT INSERT, UPDATE, DELETE ON** pago **TO** pepej\_grupo\_facturacion;

#### Mantenimiento

- a) Listar los equipos y el estado de los mismos.

**GRANT SELECT ON** equipo, mantenimiento\_equipo **TO** pepej\_grupo\_mantenimiento;

- b) Listar las camas y el estado de las mismas.

**GRANT SELECT ON** cama, mantenimiento\_cama **TO** pepej\_grupo\_mantenimiento;

- c) Agregar nuevos equipos y camas.

**GRANT INSERT ON** equipo, cama **TO** pepej\_grupo\_mantenimiento;

**GRANT SELECT(id\_habitacion, numero, tipo) ON** habitacion **TO** pepej\_grupo\_mantenimiento;

#### Sistemas

- a) Agregar, modificar o eliminar estudios.

**GRANT INSERT, UPDATE, DELETE ON** estudio **TO** pepej\_grupo\_sistemas;

**GRANT SELECT ON** tipo\_estudio **TO** pepej\_grupo\_sistemas;

- b) Agregar, modificar o eliminar cargos.

**GRANT INSERT, UPDATE, DELETE ON** cargo **TO** pepej\_grupo\_sistemas;

- c) Agregar, modificar o eliminar especialidades.

**GRANT INSERT, UPDATE, DELETE ON** especialidad **TO** pepej\_grupo\_sistemas;

- d) Agregar, modificar o eliminar tipos de estudios.

**GRANT INSERT, UPDATE, DELETE ON** tipo\_estudio **TO** pepej\_grupo\_sistemas;

- e) Agregar, modificar o eliminar consultorios.

**GRANT INSERT, UPDATE, DELETE ON** consultorio **TO** pepej\_grupo\_sistemas;

- f) Agregar, modificar o eliminar obras sociales.

**GRANT INSERT, UPDATE, DELETE ON** obra\_social **TO** pepej\_grupo\_sistemas;

- g) Agregar, modificar o eliminar turnos.

**GRANT INSERT, UPDATE, DELETE ON** turno **TO** pepej\_grupo\_sistemas;

- h) Listar todas las tablas antes mencionadas.

**GRANT SELECT ON** estudio, cargo, especialidad, consultorio, obra\_social, turno **TO** pepej\_grupo\_sistemas;

## CONCEPTOS DE BASES DE DATOS II

Programador Universitario, Ing. y Lic. en Informática  
Fac. de C. Exactas y Tecnología. – UNT



### Trabajo Práctico Nro. 3 - Ciclo 2024

#### Tema: Seguridad, Usuarios y Permisos

---

##### Ejercicio nro. 2:

Determine e indique con que usuario (de los creados en el ejercicio anterior) se pueden llevar a cabo las siguientes consultas y realice el código SQL que lo resuelva.

- a) Muestre el nombre, apellido y la obra social de todos los pacientes (incluso de aquellos que no tienen obra social).
- ```
SELECT p.nombre, apellido, COALESCE(o.nombre, 'SIN OBRA SOCIAL') AS obra_social FROM persona p
INNER JOIN paciente pa ON p.id_persona = pa.id_paciente
LEFT JOIN obra_social o USING(id_obra_social)
-- Filas devueltas: 113.910
-- Users: pepej_user_informes, pepej_user_admision, pepej_user_medicos
```
- b) Liste el nombre, apellido, cargo, especialidad y sueldo de todos los empleados activos.
- ```
SELECT nombre, apellido, cargo, especialidad, sueldo FROM persona
INNER JOIN empleado ON id_persona = id_empleado
INNER JOIN cargo USING(id_cargo)
INNER JOIN especialidad USING(id_especialidad)
WHERE fecha_baja IS NULL
-- Filas devueltas: 928
-- Users: pepej_user_rrhh
```
- c) Mostrar los tratamientos cuyo número de dosis sea mayor que 3, es decir, el nombre del medicamento, presentación, clasificación y descripción del tratamiento.
- ```
SELECT m.nombre, presentacion, clasificacion, descripcion, dosis FROM tratamiento t
INNER JOIN medicamento m USING(id_med medicamento)
INNER JOIN clasificacion USING(id_clasificacion)
WHERE dosis > 3
-- Filas devueltas: 158.423
-- Users: NINGUNO
```
- d) Mostrar todas las consultas realizadas en el primer trimestre del 2022 (muestre el nombre completo del paciente y del médico).
- ```
SELECT (pac.nombre || ' ' || pac.apellido) AS paciente, (emp.nombre || ' ' || emp.apellido) AS medico,
fecha, hora, resultado FROM consulta c
INNER JOIN persona pac ON c.id_paciente = pac.id_persona
INNER JOIN persona emp ON c.id_empleado = emp.id_persona
WHERE fecha BETWEEN '2022-01-01' AND '2022-03-31'
-- Filas devueltas: 140.769
-- Users: pepej_user_informes, pepej_user_admision, pepej_user_medicos
```

## CONCEPTOS DE BASES DE DATOS II

Programador Universitario, Ing. y Lic. en Informática  
Fac. de C. Exactas y Tecnología. – UNT



### Trabajo Práctico Nro. 3 - Ciclo 2024

#### Tema: Seguridad, Usuarios y Permisos

---

- e) Mostrar todas las facturas que han sido pagadas parcialmente.

```
SELECT * FROM factura
WHERE pagada = 'N' AND saldo <> monto
-- Filas devueltas: 1.064.020
-- Users: pepej_user_facturacion
```

- f) Mostrar todas las consultas que atendió el médico “ANGELA MENDOZA” ordenado por fecha y hora.

```
SELECT (pac.nombre || ' ' || pac.apellido) AS paciente, fecha, hora, con.nombre, resultado
FROM consulta c
INNER JOIN persona pac ON c.id_paciente = pac.id_persona
INNER JOIN persona emp ON c.id_empleado = emp.id_persona
INNER JOIN consultorio con USING(id_consultorio)
WHERE emp.apellido LIKE 'MENDOZA' AND emp.nombre LIKE 'ANGELA'
-- Filas devueltas: 2.878
-- Users: pepej_user_admision, pepej_user_medicos
```

- g) Mostrar la historia clínica de la paciente “SOFIA TELLO” (todas las consultas, tratamientos, estudios, internaciones). Ordene los resultados por fecha.

```
SELECT * FROM (
    SELECT p.nombre, p.apellido, 'Consulta' AS tipo, fecha, resultado AS cuestion
    FROM persona p
    INNER JOIN consulta c ON p.id_persona = c.id_paciente
    WHERE p.nombre LIKE 'SOFIA' AND p.apellido LIKE 'TELLO'
    UNION
    SELECT p.nombre, p.apellido, 'Tratamiento' AS tipo, fecha_indicacion, t.nombre
    FROM persona p
    INNER JOIN tratamiento t ON p.id_persona = t.id_paciente
    WHERE p.nombre LIKE 'SOFIA' AND p.apellido LIKE 'TELLO'
    UNION
    SELECT p.nombre, p.apellido, 'Internación' AS tipo, fecha_inicio, 'Alta: ' || fecha_alta::varchar
    FROM persona p
    INNER JOIN internacion i ON p.id_persona = i.id_paciente
    WHERE p.nombre LIKE 'SOFIA' AND p.apellido LIKE 'TELLO'
    UNION
    SELECT p.nombre, p.apellido, 'Estudio' AS tipo, fecha, e.nombre || ' - Resultado: ' || resultado
    FROM persona p
    INNER JOIN estudio_realizado er ON p.id_persona = er.id_paciente
    INNER JOIN estudio e USING(id_estudio)
    WHERE p.nombre LIKE 'SOFIA' AND p.apellido LIKE 'TELLO'
) AS historia
ORDER BY fecha
```



# CONCEPTOS DE BASES DE DATOS II

Programador Universitario, Ing. y Lic. en Informática  
Fac. de C. Exactas y Tecnología. – UNT



## Trabajo Práctico Nro. 3 - Ciclo 2024

### Tema: Seguridad, Usuarios y Permisos

---

-- Filas devueltas: 12

-- Users: pepej\_user\_informes, pepej\_user\_admision, pepej\_user\_medicos

- h) Mostrar todos los pagos realizados por "SERGIO DANIEL PEREZ".

```
SELECT nombre, apellido, id_factura, pago.fecha, pago.monto FROM pago
INNER JOIN factura f USING(id_factura)
INNER JOIN persona p ON p.id_persona = f.id_paciente
WHERE nombre LIKE 'SERGIO DANIEL' AND apellido LIKE 'PEREZ'
```

-- Filas devueltas: 21

-- Users: pepej\_user\_facturacion

- i) Listar todos los equipos que aún están en mantenimiento.

```
SELECT nombre, marca, fecha_ingreso, observacion, estado
FROM equipo
INNER JOIN mantenimiento_equipo USING(id_equipo)
WHERE fecha_egreso IS NULL
```

-- Filas devueltas: 44

-- Users: pepej\_user\_mantenimiento

- j) Muestre todas las compras realizadas al laboratorio "ROEMMERS" indicando la fecha de compra, el medicamento, el proveedor y el empleado que realizó la compra.

```
SELECT m.nombre, proveedor, fecha, (p.nombre || ' ' || p.apellido) AS empleado, precio_unitario,
cantidad FROM compra c
INNER JOIN medicamento m USING(id_medimento)
INNER JOIN proveedor USING(id_proveedor)
INNER JOIN persona p ON p.id_persona = c.id_empleado
INNER JOIN laboratorio USING(id_laboratorio)
WHERE laboratorio LIKE 'ROEMMERS'
```

-- Filas devueltas: 3.885

-- Users: NINGUNO

- k) Agregar un nuevo Laboratorio (206, "LABUNT", "AV. INDEPENDENCIA 1800", "54-381-123-5555").

```
INSERT INTO laboratorio VALUES(206, 'LABUNT', 'AV. INDEPENDENCIA 1800', '54-381-123-5555')
```

-- Users: pepej\_user\_compra

- l) Ingresar una cama a mantenimiento (52, "2023-02-02", "A la espera de repuestos", "En reparacion", null, 26, 210).

```
INSERT INTO cama VALUES(52, '2023-02-02', 'A la espera de repuestos', 'En reparacion', null, 26, 210)
```

-- No se pudo realizar porque viola una restricción de unicidad (ya existe esa PK)

-- Users: pepej\_user\_mantenimiento

## CONCEPTOS DE BASES DE DATOS II

Programador Universitario, Ing. y Lic. en Informática  
Fac. de C. Exactas y Tecnología. – UNT



### Trabajo Práctico Nro. 3 - Ciclo 2024

#### Tema: Seguridad, Usuarios y Permisos

---

- m) Modificar el teléfono del laboratorio “LABUNT”, por el “54-381-456-5555”.

```
UPDATE laboratorio
SET telefono = '54-381-456-5555'
WHERE laboratorio LIKE 'LABUNT'
-- Users: pepej_user_compra
```

- n) El empleado “PABLO ALEJANDRO MEDRANO” pasa de trabajar los sábados a la mañana a los domingos a la mañana.

```
UPDATE trabajan t
SET fin = current_date
WHERE id_employado = (SELECT id_persona FROM persona
                      WHERE nombre LIKE 'PABLO ALEJANDRO' AND apellido LIKE 'MEDRANO')
AND fin IS NULL;

INSERT INTO trabajan VALUES((SELECT id_turno FROM turno WHERE turno = 'Domingo'),
                             (SELECT id_persona FROM persona
                              WHERE nombre LIKE 'PABLO ALEJANDRO' AND apellido LIKE 'MEDRANO'),
                             current_date, null)
-- Users: pepej_user_rrhh
```

- o) Eliminar el laboratorio “MERRELL”.

```
DELETE FROM laboratorio
WHERE laboratorio LIKE 'MERRELL'
-- No se pudo realizar porque viola una restricción de FK
-- Users: pepej_user_compra
```

#### Ejercicio nro. 3:

Implemente la seguridad a través de esquemas, para ello:

- a) Cree un nuevo esquema en la base de datos Hospital, con el nombre de “facturacion”.
- ```
CREATE SCHEMA facturacion AUTHORIZATION postgres;
```
- b) Mueva las tablas pago y factura al nuevo esquema “facturacion” (lo puede hacer desde la pestaña general de las propiedades de tabla).
- ```
ALTER TABLE pago SET SCHEMA facturacion;
ALTER TABLE factura SET SCHEMA facturacion;
```

## CONCEPTOS DE BASES DE DATOS II

Programador Universitario, Ing. y Lic. en Informática  
Fac. de C. Exactas y Tecnología. – UNT



### Trabajo Práctico Nro. 3 - Ciclo 2024

#### Tema: Seguridad, Usuarios y Permisos

---

- c) Vuelva a realizar la consulta del apartado H del ejercicio anterior.

```
SELECT nombre, apellido, id_factura, pago.fecha, pago.monto
FROM facturacion.pago
INNER JOIN facturacion.factura f USING(id_factura)
INNER JOIN persona p ON p.id_persona = f.id_paciente
WHERE nombre LIKE 'SERGIO DANIEL' AND apellido LIKE 'PEREZ'
-- Filas devueltas: 21
-- Users: pepej_user_facturacion
```

Hubo que indicar el nombre del schema antes del nombre de tabla y, además, dar permiso de uso sobre el schema facturación al usuario pepej\_user\_facturacion

```
GRANT USAGE ON SCHEMA facturacion TO pepej_grupo_facturacion
```

- d) Cree un nuevo usuario del área facturación que tenga los permisos para realizar:

- Listar las facturas, mostrando los pacientes.
- Agregar, modificar y eliminar facturas.
- Listar los pagos, mostrando el paciente.
- Agregar modificar y eliminar pagos.

Pero esta vez estableciendo permisos sobre el nuevo esquema “facturacion”.

Al mover las tablas pago y factura, los permisos que tenía el grupo pepej\_grupo\_facturacion no pueden aplicar ya que el grupo solo tienen permiso sobre el schema public. Por lo tanto, se debe dar permiso de uso sobre el schema facturacion para que los permisos sobre las tablas apliquen. En el apartado anterior se dio permiso de uso sobre el schema facturacion por lo que solo es necesario crear el nuevo usuario y unirlo al grupo:

```
CREATE ROLE user_facturador WITH
LOGIN
NOSUPERUSER
NOCREATEDB
NOCREATEROLE
INHERIT
NOREPLICATION
CONNECTION LIMIT -1
PASSWORD '1234';
```

```
GRANT pepej_grupo_facturacion TO user_facturador;
```