

Instituto Tecnológico de Costa Rica

Tarea 04

Curso:

Bases de Datos II

Profesor:

Alberto Shum Chan

Alumnos:

Alberto Zumbado Abarca

Jonathan Quesada Salas

NOTA = 87

Grupo 20

II Semestre

2021

```
-- Borrar las tablas por aquello  
drop table hospital;  
drop table medico_hospital;  
drop table medico;  
drop table especialidad;  
drop table medico_especialidad;  
drop table temporal;
```

-- Creacion de tablas

-4 pts

```
CREATE TABLE hospital (  
    hospital_id NUMBER,  
    nombre VARCHAR2(40),  
    provincia VARCHAR2(20),  
    PRIMARY KEY ( hospital_id )  
);
```

Deben ser NOT NULL

```
CREATE TABLE medico (  
    medico_id NUMBER,  
    cedula VARCHAR2(15),  
    nombre VARCHAR2(25),  
    primer_apellido VARCHAR2(25),  
    direccion_provincia VARCHAR2(15),  
    PRIMARY KEY ( medico_id )  
);
```

```
CREATE TABLE medico_hospital  
(medico_id number,  
    hospital_id number,  
    PRIMARY KEY (hospital_id, medico_id)
```

Faltaron las FKs

);

```
CREATE TABLE especialidad (  
    especialidad_id    NUMBER,  
    nombre VARCHAR2(20),  
    PRIMARY KEY ( especialidad_id )  
);
```

```
CREATE TABLE medico_especialidad (  
    medico_id    NUMBER,  
    especialidad_id NUMBER,  
    PRIMARY KEY(medico_id, especialidad_id)  
);
```

Faltaron las FKs

-- Secuencias

-- Para que el id vaya autoincrementado segun las interacciones que vaya teniendo

```
drop sequence especialidad_sq;
```

```
create SEQUENCE especialidad_sq increment by 1;
```

```
create or replace trigger especialidad_para_insertar before
```

```
    insert on especialidad
```

```
    for each row
```

```
begin
```

```
    select especialidad_sq.NEXTVAL into :new.especialidad_id from dual;
```

```
end;
```

```
/
```

```
drop sequence medico_sq;
```

```
create SEQUENCE medico_sq increment by 1;
```

```
create or replace trigger medico_para_insertar before
```

```
    insert on medico
```

```

        for each row
begin
    select medico_sq.NEXTVAL into :new.medico_id from dual;
end;
/

drop sequence hospital_sq;
create SEQUENCE hospital_sq increment by 1;
create or replace trigger hospital_para_insertar before
    insert on hospital
    for each row
begin
    select hospital_sq.NEXTVAL into :new.hospital_id from dual;
end;
/

-- Insertar datos

insert into hospital values(1,'Hospital Max Peralta', 'Cartago');
insert into hospital values(2,'Hospital San Rafael', 'Alajuela');
insert into hospital values(3,'Hospital San Vicente de Paul', 'Heredia');

select * from hospital;

insert into medico_hospital values(1, 2);
insert into medico_hospital values(1, 3);
insert into medico_hospital values(4, 3);

insert into medico values(1,'4-0071-0076', 'Gloria', 'Morales', 'Alajuela');
insert into medico values(2,'1-0071-0656', 'Andrea', 'Porrás', 'Heredia');
insert into medico values(3,'4-0071-6535', 'Aurelio', 'Sanabria', 'Alajuela');

```

```
insert into medico values(4,'3-7879-8765', 'Jaime', 'Vargas', 'Cartago');
```

```
insert into medico_especialidad values (1,1);
```

```
insert into medico_especialidad values (1,2);
```

```
insert into medico_especialidad values (2,3);
```

```
insert into medico_especialidad values (2,4);
```

```
insert into medico_especialidad values (3,4);
```

```
insert into medico_especialidad values (4,4);
```

```
insert into especialidad values (1, 'Cardiologo');
```

```
insert into especialidad values (2, 'Alergologo');
```

```
insert into especialidad values (3, 'Pediatra');
```

```
insert into especialidad values (4, 'Nutricionista');
```

```
-----  
-----
```

```
CREATE OR REPLACE FUNCTION lista_especialidades (c_medico  
VARCHAR2)
```

```
RETURN VARCHAR2 IS
```

```
especialidades VARCHAR2(100);
```

```
l_especialidades VARCHAR2(100);
```

```
n_medico VARCHAR2(25);
```

```
a_medico VARCHAR2(25);
```

```
BEGIN
```

```
SELECT m.nombre, m.primer_apellido INTO n_medico, a_medico FROM  
medico m
```

```
WHERE m.cedula = c_medico;
```

```
FOR i IN (SELECT e.nombre as n_especialidad FROM especialidad e
          INNER JOIN medico_especialidad me ON e.especialidad_id =
me.especialidad_id
```

```
          INNER JOIN medico m ON me.medico_id = m.medico_id
```

```
          WHERE m.cedula = c_medico order by e.nombre) LOOP
```

```
    especialidades := TRIM(i.n_especialidad);
```

```
    especialidades := CONCAT(especialidades,', ');
```

```
    l_especialidades := CONCAT(l_especialidades,especialidades);
```

```
END LOOP;
```

```
    l_especialidades := 'Especialidades del medico/a ' || n_medico || ' ' ||
a_medico || ': ' || l_especialidades;
```

```
    l_especialidades := TRIM(l_especialidades);
```

```
    l_especialidades := SUBSTR(l_especialidades, 0,
LENGTH(l_especialidades) - 1);
```

```
    RETURN l_especialidades;
```

```
END;
```

```
/
```

```
-----
-----
```

```
create table temporal (
```

```
    medico_cedula varchar2(20),
```

```
    medico_nombre varchar2(20),
```

```
    medico_apellido varchar2(20),
```

```
    medico_provincia varchar2(20),
```

```
    especialidades varchar2(100),
```

```
hospital varchar2(50)
);
```

```
insert into temporal values ('3-0098-8768', 'Marta', 'Morales', 'Cartago',
'Alergologo, Pediatra, Nutricionista, Odontologo', 'Hospital Max Peralta');
```

```
insert into temporal values ('2-0876-4527', 'Flor', 'Flores', 'Heredia',
'Nutricionista, Cardiologa, Medico general', 'Hospital San Vicente de Paul');
```

```
insert into temporal values ('1-9976-0442', 'Kevin', 'Moraga',
'Alajuela', 'Cardiologo, Pediatra, Hepatologo', 'Hospital San Rafael');
```

```
select *from temporal;
```

```
create or replace procedure procesa_temporal as
```

```
m_cedula varchar2(30);
```

```
m_nombre varchar2(30);
```

```
m_apellido varchar2(30);
```

```
m_provincia varchar2(30);
```

```
m_id_hospital number;
```

```
m_especialidades varchar(50);
```

```
m_especialidad varchar(50);
```

```
auxiliar number;
```

```
begin
```

```
-- Se obtiene todos los datos de temporal
```

```
for especialidades_medico in (select * from temporal)
```

```
LOOP
```

```
-- Se quitan los espacios
```

```
-- PEEEEERO EN TODAS LAS VARIABLES QUE OCUPE MEDICO  
PARA QUE NO ESTE VACIAS LAS FILAS
```

```
m_cedula := trim(especialidades_medico.medico_cedula);
```

```
m_nombre := trim(especialidades_medico.medico_nombre);
```

En lugar de ellas se puede usar directamente
ESPECIALIDADES_MEDICO.medico_cedula
ESPECIALIDADES_MEDICO.mecico_nombre
...

```
m_apellido := trim(especialidades_medico.medico_apellido);  
m_provincia := trim(especialidades_medico.medico_provincia);  
m_especialidades := trim(especialidades_medico.especialidades);
```

-- Se cuenta la cantidad de medicos segun la cedulas que tengan

```
select count(*) into auxiliar from medico where medico.cedula =  
m_cedula;
```

El COUNT no es eficiente para validar la existencia

-- Cuando auxiliar es 0 se insertan los valores de medicos

if auxiliar < 1 then

```
insert into medico values (medico_sq.currval, m_cedula, m_nombre,  
m_apellido, m_provincia);
```

end if;

-- Si aun hay especialidades que siga adelante

if length(m_especialidades) > 0 then

LOOP

-- Recorre especialidades si para ver cuando hay alguna coma

if instr(m_especialidades, ',') > 0 then

```
m_especialidad := substr(m_especialidades,1,  
instr(m_especialidades,',')-1);
```

-- Se quita el espacio de la coma y se coloca junto a la proxima
especialidad

```
m_especialidades := trim(substr(m_especialidades,  
instr(m_especialidades,',')+1));
```

else

-- Y si no se retorna la especialidad vacia

```
m_especialidad := m_especialidades;
```

```
m_especialidades := '';
```

end if;

-- Se van contando la cantidad de especialidades que hay


```

        select count(*) into auxiliar from especialidad where
especialidad.nombre = m_especialidad;

        if auxiliar = 0 then

            insert into especialidad values (especialidad_sq.currval,
m_especialidad);

            auxiliar := especialidad_sq.currval;

        else

            -- Se selecciona la especialidad segun el id junto con el nombre

            select especialidad.especialidad_id into auxiliar from especialidad
where especialidad.nombre = m_especialidad;

            end if;

            -- Se inserta la especialidad del medico segun su valor

            insert into medico_especialidad values (medico_sq.currval, auxiliar);


            -- Si no hay especialidades retorne null

            exit when m_especialidades is null;

            END LOOP;

        end if;

        -- Selecciona el hospital id cuando el nombre sea igual a la especialdad
del medico

        select hospital.hospital_id into m_id_hospital from hospital where
hospital.nombre = especialidades_medico.hospital;

        insert into medico_hospital values (medico_sq.currval, m_id_hospital);

        END LOOP;

    end;

/

```