**UNIVERSIDAD TECNOLÓGICA DE PANAMÁ**



**FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES**

**DEPARTAMENTO DE SISTEMAS DE INFORMACIÓN,**

**CONTROL Y EVALUACIÓN DE RECURSOS INFORMÁTICOS**

**LICENCIATURA EN INGENIERÍA DE SISTEMAS DE INFORMACIÓN**

**SISTEMAS DE BASES DE DATOS ll.**

**LABORATORIO 3**

**Profesor**

**Henry Lezcano**

**Estudiantes**

Solis, Michael 8-958-1219

Sánchez, Ana 8-967-832

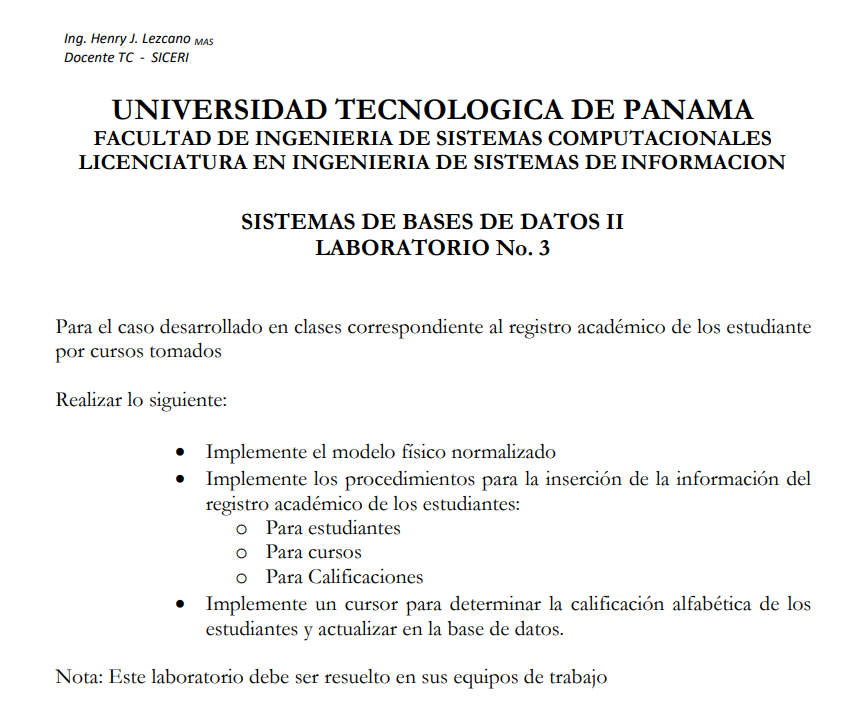
Tejada, Ángel 8-969-974

Samudio, Nedith 8-968-1471

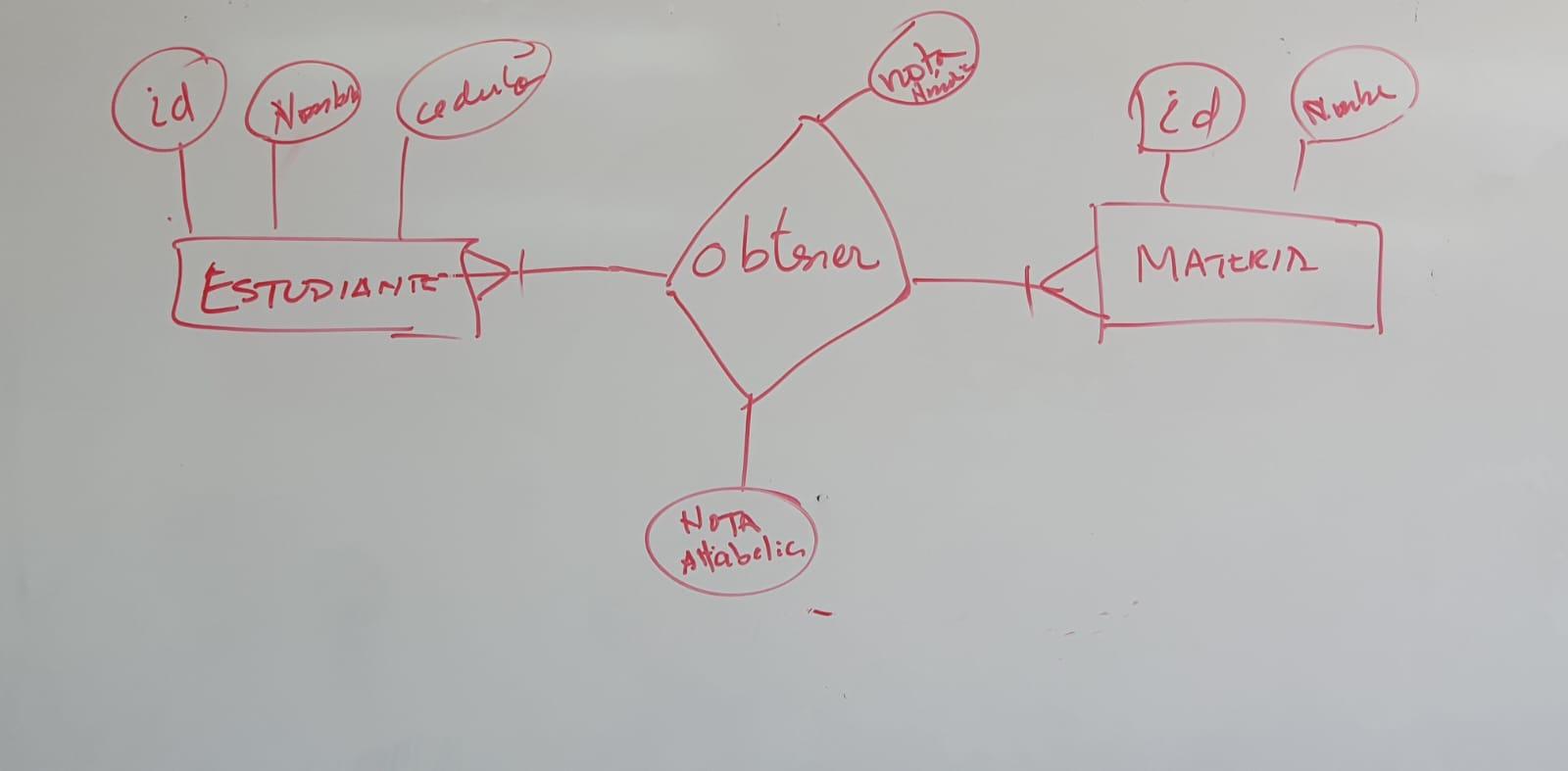
Urriola, Vicente 8-892-2296

**Grupo**

**1IF131**



Modelo Entidad-Relación



|  |  |  |
| --- | --- | --- |
| calificaciones | | |
| PK/FK | e\_id1 | N |
| PK/FK | c\_id1 | N |
|  | nota\_a | S |
|  | nota\_n | N |

Modelo Relacional

|  |  |  |
| --- | --- | --- |
| estudiante | | |
| PK | e\_id | N |
|  | nombre | N |
|  | cedula | N |

|  |  |  |
| --- | --- | --- |
| curso | | |
| PK | c\_id | N |
|  | nombre | N |

Implementación física

create table estudiante (

e\_id number not null,

nombre varchar2(10) not null,

cedula number not null,

constraint pk\_estudiante primary key (e\_id)

);

create table curso (

c\_id number not null,

nombre varchar(20) not null,

constraint pk\_curso primary key (c\_id)

);

create table calificaciones (

e\_id1 number not null,

c\_id1 number not null,

nota\_a char(1) null,

nota\_n number not null,

constraint fk\_estudiante\_cal foreign key (e\_id1) references estudiante(e\_id),

constraint fk\_curso\_cal foreign key (c\_id1) references curso(c\_id),

constraint pk\_todo primary key (e\_id1,c\_id1)

);

-- Procedimiento de la tabla estudiante

create or replace procedure agregar(

p\_estudiante\_id estudiante.e\_id%type,

p\_estudiante\_nom estudiante.nombre%type,

p\_estudiante\_ced estudiante.cedula%type)

as

begin

insert into estudiante (e\_id, nombre, cedula)

values (p\_estudiante\_id, p\_estudiante\_nom, p\_estudiante\_ced);

end agregar;

-- Procedimiento de la tabla curso

create or replace procedure agregar\_curso(

p\_curso\_id curso.c\_id%type,

p\_curso\_nombre curso.nombre%type) as

begin

insert into curso(c\_id, nombre)

values (p\_curso\_id,p\_curso\_nombre);

end agregar\_curso;

-- Procedimiento para calificiones

create or replace procedure agregar\_cal(

p\_calificaciones\_id calificaciones.e\_id1%type,

p\_calificaciones\_c\_id calificaciones.c\_id1%type,

p\_calificaciones\_nota\_a calificaciones.nota\_a%type,

p\_calificaciones\_nota\_n calificaciones.nota\_n%type)

as

begin

insert into calificaciones (e\_id1,c\_id1,nota\_a,nota\_n)

values(p\_calificaciones\_id,p\_calificaciones\_c\_id,p\_calificaciones\_nota\_a,p\_calificaciones\_nota\_n);

end agregar\_cal;

-- ejecutando procedimiento de la tabla curso

begin

agregar\_curso(16,'programacion');

agregar\_curso(1,'matematicas');

agregar\_curso(2,'historia');

agregar\_curso(3,'contabilidad');

agregar\_curso(4,'geografia');

end;

-- ejecutando procedimiento de la tabla estudiante

begin

agregar(1,'javier',8-958-1219);

agregar(2,'miguel',8-972-5269);

agregar(3,'carina',8-777-9987);

agregar(4,'nedith',8-777-5555);

agregar(5,'angel',8-777-7814);

end;

-- ejecutando procedimiento de tabla calificaciones

begin

agregar\_cal(1,16,null,99);

agregar\_cal(2,16,null,75);

agregar\_cal(3,16,null,42);

end;

/\* Cursor \*/

set serveroutput on;

declare

v\_nota\_a calificaciones.nota\_a%type;

v\_nota\_n calificaciones.nota\_n%type;

cursor c\_cursor2 is select e\_id1,nota\_n,c\_id1 from calificaciones;

begin

for v\_nota in c\_cursor2 loop

case

when (v\_nota.nota\_n >= 91) then v\_nota\_a := 'A';

when (v\_nota.nota\_n >= 81) THEN v\_nota\_a := 'B';

when (v\_nota.nota\_n >= 71) THEN v\_nota\_a := 'C';

when (v\_nota.nota\_n >= 61) THEN v\_nota\_a := 'D';

when (v\_nota.nota\_n <= 60) THEN v\_nota\_a := 'F';

end case;

dbms\_output.put\_line('nota numérica: ' || v\_nota.nota\_n || chr(10)||'nota alfanumérica: ' || v\_nota\_a);

update calificaciones

set nota\_a = v\_nota\_a

where c\_id1 = 16

and nota\_n = v\_nota.nota\_n

and e\_id1 = v\_nota.e\_id1;

end loop;

exception

when no\_data\_found then

dbms\_output.put\_line('Error: Datos no encontrados');

end;

-- Creación de la vista

create view vista\_estudiante as select e.nombre, k.nota\_a

from estudiante e inner join calificaciones k

on e.e\_id = k.e\_id1

inner join curso c

on c.c\_id = k.c\_id1;

select \* from vista\_estudiante;

