

# EJERCICIO RESUELTO

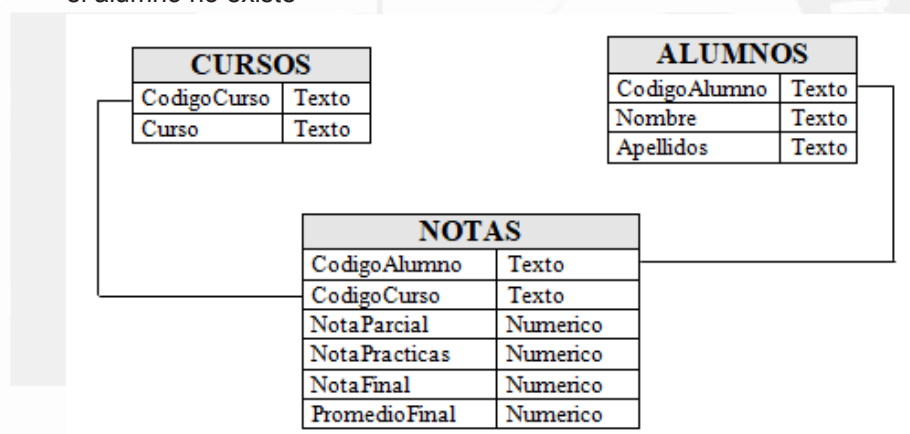
## Módulo: BASES DE DATOS

### Cursores

#### Descripción:

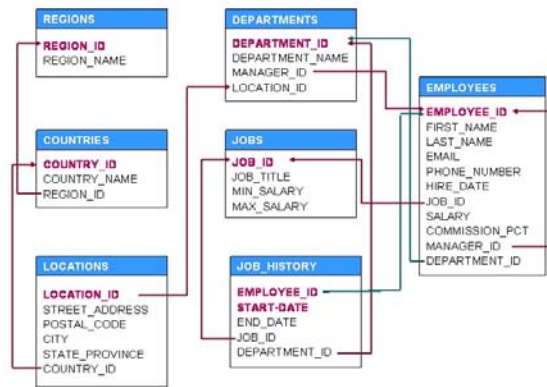
El siguiente caso práctico se basará en esta base de datos, sobre el cual se solicitará crear un cursor.

- Crear un cursor, basado en el primer esquema, que muestre la relación de cursos y notas obtenidas por curso (Código alumno, Nombre, Apellidos, Curso, Nota Parcial, Practicas, Nota Final y Promedio Final). Ingresar como parámetro el código del alumno. En caso de no encontrar registros, indicar que el alumno no existe



Y el siguiente caso práctico se realizará sobre la base de datos del diagrama HR creando un segundo cursor.

- Crear un cursor, basado en el segundo esquema, que muestre los datos del empleado y su respectivo salario, todo ello dentro de un procedimiento.



### Objetivos:

- Identificar la sintaxis de creación de cursores.
- Verificar los diferentes métodos para ejecutar cursores.

### Recursos:

- Acceso a Internet.
- Software ofimático.
- Oracle.

## Resolución:

Crear un cursor, basado en el primer esquema, que muestre la relación de cursos y notas obtenidas por curso (Código alumno, Nombre, Apellidos, Curso, Nota Parcial, Practicas, Nota Final y Promedio Final). Ingresar como parámetro el código del alumno. En caso de no encontrar registros, indicar que el alumno no existe.

--Declaramos el cursor c\_cursos

```
DECLARE CURSOR c_cursos (cod_alumno IN VARCHAR2)
IS
```

--Seleccionamos los datos del alumno

```
SELECT a.codigoalumno, a.nombre, a.apellidos, c.curso, n.notaparcial,
       n.notapracticas, n.notafinal, n.promediofinal
FROM alumnos a inner join notas n
      on a.codigoalumno = n.codigoalumno inner join cursos c
      on c.codigocurso = n.codigocurso
WHERE a.codigoalumno = cod_alumno;
```

--Declaramos el registro donde almacenaremos lo datos

```
ccurs_registro c_cursos%ROWTYPE;
```

```
BEGIN
```

```
  DBMS_OUTPUT.PUT_LINE('Relación de cursos con notas ' ||
                        ccurs_registro.curso);
```

```
  FOR ccurs_registro IN c_cursos(&cod_alumno)
```

```
  LOOP
```

```
    DBMS_OUTPUT.PUT_LINE (ccurs_registro.codigoalumno || ' ' ||
                          ccurs_registro.nombre || ' ' ||
                          ccurs_registro.curso || ' ' ||
                          ccurs_registro.notaparcial || ' ' ||
                          ccurs_registro.notafinal || ' ' ||
                          ccurs_registro.promediofinal);
```

```
  END LOOP;
```

--Controlamos que no exista el alumno

```
EXCEPTION
```

```
WHEN NO_DATA_FOUND THEN
```

```
  DBMS_OUTPUT.PUT_LINE ('El alumno no existe');
```

```
END;
```

Crear un cursor, basado en el segundo esquema, que muestre los datos del empleado y su respectivo salario, todo ello dentro de un procedimiento.

--Declaramos el procedimiento P\_empleado

```
CREATE OR REPLACE PROCEDURE P_empleado
IS
```

--Declaramos el cursor cur\_emp

```
CURSOR cur_emp IS
```

```
  SELECT employee_id, first_name, last_name, salary
  FROM employees
```

--Declaramos las variables para almacenar el cursor

```
V_Cod_Empleado    NUMBER(6);
```

```
V_Nombre          VARCHAR2(20);
```

```
V_Apellido      EMPLOYEES.LAST_NAME%Type;
V_Salario       EMPLOYEES.SALARY%Type;

BEGIN
  --Se declara implícitamente el registro para recuperar datos del cursor
  FOR reg_cur_emp IN cur_emp
  LOOP
    --Almacenamos los datos de cada registro
    V_Cod_Empleado:= reg_cur_emp.employee_id;
    V_Nombre:= reg_cur_emp.first_name;
    V_Apellido:= reg_cur_emp.last_name;
    V_Salario:= reg_cur_emp.salary;

    --Mostramos el resultado
    DBMS_OUTPUT.PUT_LINE
    ('Codigo : '|| V_Cod_Empleado ||
    ' Nombre : '|| V_Nombre ||
    ' Apellido : '|| V_Apellido ||
    'Salario : '|| V_Salario);
  END LOOP;
END;
```