

Contenidos: Procedimientos. Variables. **Condicionales.**

Ciclo: Módulo:

- ✓ Las inserciones de los datos se entregan en un archivo SQL aparte.

Diagrama de bases de datos en UML para un sistema de gestión de una escuela. El diagrama muestra las entidades: persona, profesor, departamento, asignatura, alumno_se_matricula_asignatura, curso_escolar y grado, con sus atributos y relaciones.

Entidad persona:

- id INT AI (clave primaria)
- dni VARCHAR(9)
- nombre VARCHAR(25) NN
- apellido1 VARCHAR(50) NN
- apellido2 VARCHAR(50)
- ciudad VARCHAR(25) NN
- direccion VARCHAR(50) NN
- telefono VARCHAR(9)
- fecha_nacimiento DATE NN
- sexo ENUM('H', 'M') NN
- tipo ENUM('profesor', 'alum no') NN

Entidad profesor:

- id_profesor INT
- id_departamento INT

Entidad departamento:

- id INT AI (clave primaria)
- nombre VARCHAR(50) NN

Entidad asignatura:

- id INT AI (clave primaria)
- nombre VARCHAR(100) NN
- creditos FLOAT NN
- tipo ENUM('básica', 'obligatoria', 'optativa') NN
- curso TINYINT NN
- cuatrimestre TINYINT NN
- id_profesor INT
- id_grado INT NN

Entidad alumno_se_matricula_asignatura:

- id_alumno INT
- id_asignatura INT
- id_curso_escolar INT
- nota DECIMAL(4,2)

Entidad curso_escolar:

- id INT AI (clave primaria)
- anyo_inicio YEAR NN
- anyo_fin YEAR NN

Entidad grado:

- id INT AI (clave primaria)
- nombre VARCHAR(100) NN

Relaciones:

- persona (1) a profesor (1): Relación 1:1.
- persona (1) a alumno_se_matricula_asignatura (∞): Relación 1:∞.
- profesor (∞) a departamento (1): Relación ∞:1.
- asignatura (1) a alumno_se_matricula_asignatura (∞): Relación 1:∞.
- asignatura (1) a curso_escolar (1): Relación 1:1.
- asignatura (1) a grado (1): Relación 1:1.
- alumno_se_matricula_asignatura (∞) a curso_escolar (1): Relación ∞:1.
- alumno_se_matricula_asignatura (∞) a grado (1): Relación ∞:1.
- departamento (1) a asignatura (∞): Relación 1:∞ (línea punteada).

id_alumno, id_asignatura e id_curso_escolar conforman una clave primaria combinada

Consejo: Primero escribe en un script aparte las consultas necesarias y, una vez que funcionan correctamente, intégralas en los procedimientos.

- Debes usar la plantilla y cada uno de los procedimientos debe poder ejecutarse sin errores y debe contar con TODAS las ejecuciones (los CALLs) para cada uno de los casos que figuran en las capturas.
- Cada procedimiento debe contar con su DROP PROCEDURE IF EXISTS.
- Es decir, **queremos que al cargar vuestro fichero en MySQL WorkBench tan solo debamos ejecutar el fichero completo y aparezcan todos los resultados en el orden en el que están las capturas** (o los resultados que hayáis conseguido hacer). Si habéis incluido código que no funciona, pero queréis que cuente el código que habéis escrito comentarlo para que no de error. **Si figuran errores se penalizará.**

- El procedimiento debe hacer las siguientes comprobaciones de errores en los parámetros de entrada en el siguiente orden:

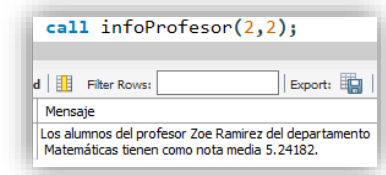
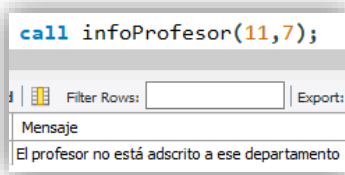
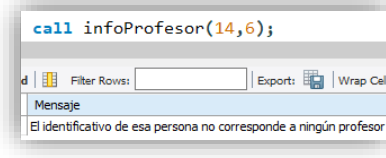
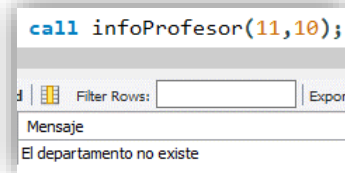
- Compruebe si existe el departamento y, en caso de no ser así, mostrará el aviso por pantalla.
- Compruebe si existe el profesor y, en caso de no ser así, mostrará el aviso por pantalla.
- Compruebe que el profesor pertenece al departamento y, de no ser así, mostrará el aviso por pantalla.

Actividad evaluable exhaustiva: Universidad

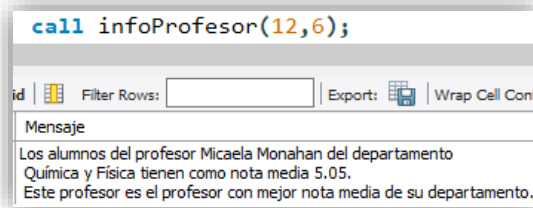
Contenidos: Procedimientos. Variables. **Condicionales.**

Nombre estudiante: Ciclo: Módulo:

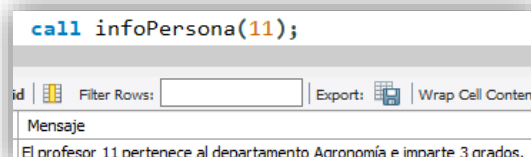
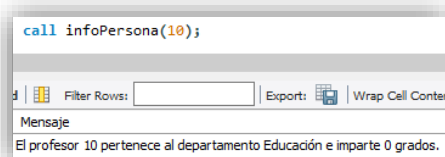
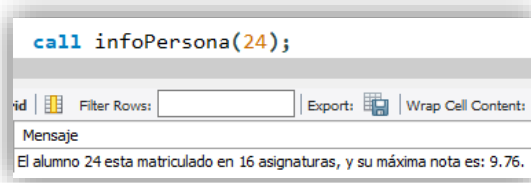
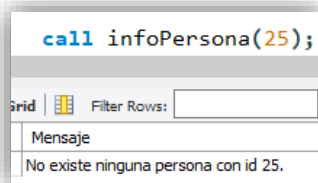
Si el profesor y el departamento existen y el profesor se halla adscrito al departamento, se deberá mostrar el nombre y primer apellido del profesor, el del departamento y la nota media obtenida por todos sus alumnos del profesor.



Reto opcional: En caso de ser el profesor con mejor nota media de su departamento, también se informará por pantalla de ello. Sino no se hará referencia a este aspecto.



2. (5 puntos) Implementa un procedimiento llamado “infoPersona” que reciba como parámetro un código de persona. El procedimiento comprobará que existe la persona y, en caso contrario, mostrará un mensaje de aviso por pantalla. Si la persona existe mostrará lo siguiente:
 - a. En caso de ser un alumno se mostrará su nota máxima y el número de asignaturas distintas en las que está matriculado.
 - b. En caso de ser un profesor se mostrará su departamento y el Nº de grados en los que imparte docencia.



Actividad evaluable exhaustiva: Universidad

Contenidos: Procedimientos. Variables. **Condicionales.**

Nombre estudiante: Ciclo: Módulo:

- Pista 1: Recuerda que para contar el número de filas existentes con un id de grado **diferente** debemos usar **COUNT(DISTINCT grado.id)**.
- Pista 2: Recuerda que para poder obtener información de un profesor que no imparte ningún grado deberás usar **LEFT JOIN**.