

# Sistema de gestión académico

## *Entidades y Atributos*

### **Estudiante**

- - id\_estudiante (PK)
- - nombre
- - apellido
- - email
- - anio\_ingreso

### **Profesor**

- - id\_profesor (PK)
- - nombre
- - apellido
- - especialidad

### **Materia**

- - id\_materia (PK)
- - nombre\_materia
- - creditos
- - costo\_curso\_mensual

### **Curso**

- - id\_curso (PK)
- - nombre\_curso
- - descripcion
- - anio
- - id\_profesor (FK)
- - id\_materia (FK)

### **Inscripción**

- - id\_estudiante (FK)
- - id\_curso (FK)
- - fecha\_inscripcion
- - nota\_teorica\_1
- - nota\_teorica\_2
- - nota\_practica
- - nota\_teorica\_recuperatorio
- - nota\_final

### **Cuatrimestre**

- - id\_cuatrimestre (PK)
- - nombre
- - fecha\_inicio
- - fecha\_fin

### **Cuota**

- - id\_cuota (PK)
- - id\_estudiante (FK)
- - id\_cuatrimestre (FK)
- - id\_factura (FK)
- - mes
- - monto
- - fecha\_vencimiento
- - estado\_pago

### **Matriculación**

- - id\_matricula (PK)
- - id\_estudiante (FK)
- - anio
- - fecha\_pago
- - monto
- - estado\_pago

### **Cuenta Corriente**

- - id\_movimiento (PK)
- - id\_estudiante (FK)
- - fecha
- - concepto
- - monto
- - estado

### **Interés por Mora**

- - anio\_carrera (PK)
- - porcentaje\_interes

### **Factura**

- - id\_factura (PK)
- - id\_estudiante (FK)
- - mes
- - anio
- - fecha\_emision
- - fecha\_vencimiento
- - monto\_total
- - estado\_pago

### **ItemFactura**

- - id\_factura (FK)
- - id\_curso (FK)

## **Desarrollo**

En base al trabajo realizado en la actividad 1 (clase 4), modifique la estructura entregada para adaptarla a estos cambios mediante el uso de sentencias alter.

### **Procedimientos almacenados**

Crear un procedimiento almacenado (uno para cada uno) para cargar datos los conceptos de alumnos, materias, cursos, profesores, cuatrimestres e intereses por mora (este solo carga un registro para cada año de la carrera, si el año existe se actualiza.)

Crear un procedimiento que permita dar de baja a un alumno. El mismo debe contemplar que la cuenta corriente este en cero para hacerlo. No debe borrarse el historial del alumno, solo indicar que esta de baja.

Crear un procedimiento que permita volver a dar de alta a un alumno.

Crear un procedimiento que permita matricular un alumno a un año. Solo se acepta una matricula por año por alumno. El procedimiento además de validar los datos ingresados debe generar la factura correspondiente y el cargo en la cuenta corriente.

Crear un procedimiento que permita inscribir a un alumno a un curso. Además de verificar los datos ingresados debe verificar que el alumno no encuentre inscripto es ese u otro curso de la misma materia en ese cuatrimestre.

Crear un procedimiento de le permita cargar nota a un alumno, debe recibir el curso, el alumno, el examen y la nota. Debe validar los datos ingresados. Si la nota corresponde al recuperatorio verificar que al menos una de las instancias anteriores es menor a 4. Si la nota corresponde al recuperatorio verificar que no existan dos o más instancias de evaluaciones anteriores menores a 4.

Crear un procedimiento que permita generar las cuotas de todos los alumnos cada mes del cuatrimestre actual, generando para ello la facturación y el cargo correspondiente a la cuenta corriente.

Crear un procedimiento que permita generar la cuota de un alumno determinado para un mes del cuatrimestre actual, generando para ello la facturación y el cargo correspondiente a la cuenta corriente.

Crear un procedimiento que calcule los intereses por mora para los alumnos que adeudan más de un mes de cuota.

Crear un procedimiento que permita registrar un pago a un alumno determinado.

### **Funciones con devolución de un escalar.**

Crear una función que permita conocer el saldo de la cuenta corriente de un alumno determinado.

Crear una función de devuelva la cantidad de vacantes de un curso, tomando como parámetro que cada curso puede tener como máximo 35 alumnos.

Obtener el nombre completo de un estudiante dado su ID.

Calcular el promedio final de un estudiante en un curso específico.

Determinar el estado de pago de una cuota específica de un estudiante.

Obtener la especialidad de un profesor dado el nombre del profesor.

Calcular el monto total adeudado por un estudiante pasándole el nombre como parámetro. Si existe más de un estudiante con ese nombre devolver -1.

### ***Funciones con devolución de tabla.***

Listar todos los cursos en los que está inscripto un estudiante.

Obtener todas las cuotas impagas de un estudiante.

Listar los profesores que dictan materias en un cuatrimestre específico.

Mostrar todas las materias con más de 3 cursos activos.

Listar los estudiantes con matrícula activa en un año determinado.

Obtener todas las facturas emitidas en un mes específico.

Listar los cursos con más de 30 estudiantes inscriptos.

Mostrar los movimientos de cuenta corriente de un estudiante.

Listar los cursos dictados por un profesor en un año específico.

Obtener todas las inscripciones con nota final mayor a 8.

### ***Cursor desde procedimiento almacenado (listados)***

Listar todos los estudiantes y sus notas finales por curso.

Mostrar el historial de pagos de cada estudiante.

Listar las materias con sus respectivos profesores y cursos.

Mostrar las inscripciones por cuatrimestre y curso.

Listar los estudiantes con cuotas vencidas.

Mostrar los cursos con su cantidad de inscriptos.

Listar las facturas agrupadas por estado de pago.

Mostrar los intereses por mora aplicados por año de carrera.

Listar los cursos con mayor cantidad de inscripciones.

Mostrar los estudiantes que no tienen matrícula en el año actual.

### **Procedimiento almacenado (SQL Dinámico)**

Listar estudiantes según un campo de búsqueda variable (nombre, apellido, email, año de ingreso), definido por el usuario.

Consultar inscripciones filtrando por una combinación dinámica de notas (por ejemplo, nota\_teorica\_1, nota\_práctica, nota\_final) y operador (menor, mayor, igual).

Listar cursos que tengan más de X inscriptos, donde X es un parámetro, y el campo de agrupación puede ser por año, materia o profesor.

Generar un reporte de facturas agrupadas por un campo dinámico (mes, estado\_pago, estudiante).

Listar cuotas vencidas ordenadas por un campo dinámico (fecha\_vencimiento, monto, estado\_pago).

Mostrar los cursos que cumplen con una condición dinámica (por ejemplo, costo mensual > X, créditos < Y, año = Z).

Listar profesores que dictan cursos en un cuatrimestre específico, con posibilidad de ordenar por nombre, apellido o especialidad.

Consultar movimientos de cuenta corriente filtrando por múltiples conceptos seleccionados por el usuario (por ejemplo, 'matrícula', 'cuota', 'interés').

Listar inscripciones donde el usuario define qué columnas mostrar (por ejemplo, nota\_final, nota\_teorica\_1, nota\_practica).

Generar un listado de estudiantes con filtros dinámicos combinados (por ejemplo, año\_ingreso > X AND apellido LIKE '%Y%').

### **Triggers**

Trigger para actualizar el estado de pago de la cuota al registrar un pago.

Trigger para calcular nota final al insertar una nota de recuperatorio.

Trigger para actualizar el estado de baja de un estudiante al eliminar una inscripción.

Trigger para generar movimiento en cuenta corriente al emitir una factura.

Trigger para validar que no se inscriba dos veces a la misma materia en el mismo cuatrimestre.

Trigger para actualizar el estado de pago de la factura al registrar un pago.

Trigger para registrar automáticamente la cuota mensual al inicio de cada mes.

Trigger para impedir la inscripción si el estudiante está dado de baja.

Trigger para actualizar el monto total de la factura al insertar un ítem de factura.

Trigger para registrar interés por mora al detectar cuota vencida.

### ***Transacciones desde Procedimiento Almacenado***

Registrar matrícula de un estudiante y generar factura y movimiento en cuenta corriente.

Inscribir a un estudiante en un curso y validar disponibilidad de vacantes.

Registrar pago de cuota y actualizar estado de factura y cuenta corriente.

Generar cuotas mensuales para todos los estudiantes y registrar facturas y movimientos.

Dar de baja a un estudiante si su cuenta corriente está en cero.

Registrar nota de examen y actualizar nota final si corresponde.

Generar intereses por mora para cuotas vencidas y actualizar cuenta corriente.

Emitir factura agrupando todas las cuotas impagas del mes.

Reinscribir a un estudiante dado de baja y actualizar su estado.

Registrar inscripción a curso y generar ítem de factura correspondiente.