**1. Tablas y su Descripción (CreacionDatabase.sql)**

**1.1. Colegio**

* **Descripción:** Almacena información sobre los colegios.
* **Columnas:**
  + Colegio\_id: Identificador único del colegio.
  + Nombre: Nombre del colegio.
  + Abreviatura: Abreviatura del colegio (ej. POLI).

**1.2. KeyWords**

* **Descripción:** Contiene palabras clave que los estudiantes pueden seleccionar.
* **Columnas:**
  + Keyword\_id: Identificador único de la palabra clave.
  + Palabra: Texto de la palabra clave.

**1.3. Semestre**

* **Descripción:** Representa los semestres académicos.
* **Columnas:**
  + Semestre\_id: Identificador único del semestre.
  + Nombre: Nombre del semestre (ej. "Primer Semestre").

**1.4. Notas**

* **Descripción:** Define las calificaciones disponibles.
* **Columnas:**
  + Nota\_id: Identificador único de la calificación.
  + Nota: Texto que representa la calificación (ej. A, B).

**1.5. Categoria**

* **Descripción:** Clasifica las asignaturas en diferentes categorías.
* **Columnas:**
  + Categoria\_id: Identificador único de la categoría.
  + Nombre: Nombre de la categoría (ej. Arte, Deporte).

**1.6. Profesor**

* **Descripción:** Almacena información sobre los profesores.
* **Columnas:**
  + Profesor\_id: Identificador único del profesor.
  + Nombre: Nombre completo del profesor.
  + Correo: Correo electrónico, verificado para que termine en "@usfq.edu.ec".

**1.7. Dias**

* **Descripción:** Contiene información sobre los días de la semana.
* **Columnas:**
  + Dia\_id: Identificador único del día.
  + Abreviatura: Abreviatura del día (L, M, I, J, V, S, D).
  + Nombre: Nombre completo del día.

**1.8. Horario**

* **Descripción:** Define los intervalos de tiempo para las clases.
* **Columnas:**
  + Horario\_id: Identificador único del horario.
  + Hora: Intervalo de tiempo (ej. '10:00-11:20').

**1.9. Edificio**

* **Descripción:** Almacena información sobre los edificios donde se dictan las clases.
* **Columnas:**
  + Edificio\_id: Identificador único del edificio.
  + Abreviatura: Abreviatura del edificio.
  + Nombre: Nombre completo del edificio.

**1.10. Modalidad**

* **Descripción:** Define el tipo de modalidad en que se dictan los cursos (Presencial, Virtual).
* **Columnas:**
  + Modalidad\_id: Identificador único de la modalidad.
  + Nombre: Nombre de la modalidad.

**1.11. Carrera**

* **Descripción:** Almacena información sobre las carreras disponibles en los colegios.
* **Columnas:**
  + Carrera\_id: Identificador único de la carrera.
  + Colegio\_id: Relaciona con la tabla Colegio.
  + Abreviatura: Abreviatura de la carrera.
  + Nombre: Nombre completo de la carrera.
* **Relaciones:** Depende del Colegio, por lo que se eliminan las carreras si el colegio es eliminado.

**1.12. SubEspecializacion**

* **Descripción:** Almacena información sobre subespecializaciones o "minors".
* **Columnas:**
  + SubEsp\_id: Identificador único de la subespecialización.
  + Colegio\_id: Relaciona con la tabla Colegio.
  + Nombre: Nombre del minor.
* **Relaciones:** Similar a Carrera, depende del Colegio.

**1.13. Estudiante**

* **Descripción:** Almacena información sobre los estudiantes.
* **Columnas:**
  + Estudiante\_id: Identificador único del estudiante.
  + Nombres: Nombres del estudiante.
  + Apellidos: Apellidos del estudiante.
  + Correo: Correo institucional, verificado.
  + Contraseña: Contraseña del estudiante.
  + Carrera\_id: Relación con Carrera.
  + Semestre\_id: Relación con Semestre.
  + SubEsp\_id: Relación opcional con SubEspecializacion.
* **Relaciones:** Depende de Carrera, Semestre, y SubEspecializacion.

**1.14. Palabra\_Estudiante**

* **Descripción:** Establece una relación entre estudiantes y palabras clave.
* **Columnas:**
  + Palabra\_Estudiante\_id: Identificador único de la relación.
  + Keyword\_id: Relación con KeyWords.
  + Estudiante\_id: Relación con Estudiante.

**1.15. Asignatura**

* **Descripción:** Almacena información sobre las asignaturas disponibles.
* **Columnas:**
  + Asignatura\_id: Identificador único de la asignatura.
  + Nombre: Nombre de la asignatura.
  + Descripcion: Descripción de la asignatura.
  + Carrera\_id: Relación con Carrera.
  + Categoria\_id: Relación con Categoria.
  + Nota\_id: Relación con Notas.
  + Numero: Número del curso para la carrera.
  + Creditos: Créditos que tiene la asignatura.

**1.16. Prerrequisito**

* **Descripción:** Almacena información sobre los prerrequisitos de las asignaturas.
* **Columnas:**
  + Prerrequisito\_id: Identificador único del prerrequisito.
  + Asignatura\_id: Asignatura que es un prerrequisito.
  + Dependencia\_id: Asignatura que es un prerrequisito.

**1.17. Malla\_Subespecializacion**

* **Descripción:** Almacena la malla curricular de subespecializaciones.
* **Columnas:**
  + MallaSubEsp\_id: Identificador de la malla del minor.
  + SubEsp\_id: Relación con SubEspecializacion.

**1.18. MallaSubEsp\_Asignatura**

* **Descripción:** Relaciona asignaturas con subespecializaciones.
* **Columnas:**
  + MallaSub\_Asig\_id: Identificador de la relación.
  + MallaSubEsp\_id: Relación con Malla\_Subespecializacion.
  + Asignatura\_id: Relación con Asignatura.

**1.19. Malla\_Carrera**

* **Descripción:** Almacena la malla curricular de las carreras.
* **Columnas:**
  + MallaCarrera\_id: Identificador de la malla de la carrera.
  + Carrera\_id: Relación con Carrera.

**1.20. MallaCarreraAsignatura**

* **Descripción:** Relaciona asignaturas con mallas de carrera.
* **Columnas:**
  + MallaCarreraAsignatura\_id: Identificador único.
  + MallaCarrera\_id: Relación con Malla\_Carrera.
  + Asignatura\_id: Relación con Asignatura.
  + Semestre\_id: Relación con Semestre.

**1.21. Campus**

* **Descripción:** Almacena información sobre los diferentes campus.
* **Columnas:**
  + Campus\_id: Identificador del campus.
  + Nombre: Nombre del campus.

**1.22. Kardex**

* **Descripción:** Almacena el kardex académico de los estudiantes.
* **Columnas:**
  + Kardex\_id: Identificador del kardex.
  + Estudiante\_id: Relación con Estudiante.

**1.23. Kardex\_Asignatura**

* **Descripción:** Relaciona asignaturas con el kardex de los estudiantes.
* **Columnas:**
  + Kardex\_Asignatura\_id: Identificador único de la relación.
  + Kardex\_id: Relación con Kardex.
  + Asignatura\_id: Relación con Asignatura.
  + Nota\_id: Relación con Notas.
  + Aprobada: Estado de aprobación de la asignatura.

**1.24. Curso**

* **Descripción:** Almacena información sobre los cursos ofrecidos.
* **Columnas:**
  + Curso\_id: Identificador del curso.
  + Asignatura\_id: Relación con Asignatura.
  + Profesor\_id: Relación con Profesor.
  + Horario\_id: Relación con Horario.
  + Campus\_id: Relación con Campus.
  + Modalidad\_id: Relación con Modalidad.
  + Edificio\_id: Relación con Edificio.
  + Cupo: Número de estudiantes permitidos.

**1.25. Dias\_Asignatura**

* **Descripción:** Define qué días se imparte cada curso.
* **Columnas:**
  + Dias\_Asignatura\_id: Identificador único de la tabla.
  + Curso\_id: Relación con Curso.
  + Dia\_id: Relación con Dias.
* **Relaciones:** Permite establecer en qué días se dicta cada curso, ayudando en la organización del horario académico.

**1.26. Correquisito**

* **Descripción:** Almacena información sobre los correquisitos de las asignaturas.
* **Columnas:**
  + Correquisito\_id: Identificador único del correquisito.
  + Asignatura\_id: Asignatura que tiene un correquisito.
  + Correquisito\_Asignatura\_id: Asignatura que es correquisito.
* **Relaciones:** Permite identificar asignaturas que deben ser tomadas simultáneamente para facilitar el aprendizaje.

**1.27. Restricciones**

* **Descripción:** Define restricciones en la inscripción a asignaturas.
* **Columnas:**
  + Restriccion\_id: Identificador único de la restricción.
  + Asignatura\_id: Asignatura a la cual pertenecen las restricciones.
  + Carrera\_id: Carrera que se permite.
  + Colegio\_id: Colegio que se permite.
  + Semestre\_id: Semestre desde el cual se aplica la aceptación.
* **Relaciones:** Permite aplicar restricciones específicas para la inscripción en función de la carrera, colegio o semestre.

**1.28. Audit\_Log**

* **Descripción:** Almacena registros de auditoría de cambios en la base de datos.
* **Columnas:**
  + Audit\_id: Identificador de la auditoría.
  + Fecha: Fecha y hora del cambio.
  + Usuario: Usuario que realizó el cambio.
  + Tabla\_Modificada: Nombre de la tabla modificada.
  + Descripcion: Descripción del cambio realizado.

**1.29. Malla\_Extras**

* **Descripción:** Almacena información sobre asignaturas optativas y electivas en las mallas de carrera.
* **Columnas:**
  + MallaExtras\_id: Identificador único de la tabla.
  + MallaCarrera\_id: Relación con Malla\_Carrera.
  + Categoria\_id: Relación con Categoria.
  + Optativa: Booleano que indica si es optativa.
  + Electiva: Booleano que indica si es electiva.
  + Semestre\_id: Relación con Semestre.
* **Relaciones:** Facilita la gestión de cursos adicionales que los estudiantes pueden tomar, permitiendo una mayor flexibilidad en sus estudios.

**2. Conteo de elementos en las tablas (Ingresos.sql)**

En esta sección, se presenta el número de elementos insertados en cada una de las tablas de la base de datos. Estos conteos permiten tener una visión clara de la magnitud de los datos almacenados y facilitan el análisis de la carga de información que soporta el sistema.

|  |  |
| --- | --- |
| **Tabla** | **Cantidad de Elementos** |
| **Asignatura** | 186 |
| **Campus** | 3 (Cumbaya, Galápagos, Tiputini) |
| **Carrera** | 57 |
| **Categoría** | 15 |
| **Colegio** | 11 |
| **Correquisito** | 3 |
| **Curso** | 930 (5 por cada Asignatura) |
| **Días** | 7 |
| **Días Asignatura** | 1816 (Cada asignatura tiene una probabilidad de tener 1 o 2 días) |
| **Edificio** | 22 |
| **Estudiante** | 101 (00999999 es de prueba) |
| **Horario** | 10 (De 7:00 a 8:20 hasta 8:30 a 9:50 pm) |
| **Kardex** | 5 |
| **Kardex Asignatura** | 125 |
| **Keywords** | 50 (Ninguna funcional) |
| **Malla Carrera** | 5 (Compu, Gastronomía, Educación, Administración, Artes Visuales) |
| **Malla Extras** | 48 |
| **Malla Subespecialización** | 5 (Videojuegos, Cine, Danza, Ing. Industrial, Matemáticas) |
| **Modalidad** | 4 (Presencial, Híbrido, Virtual, En línea) |
| **Notas** | 6 (A, B, C, D, F, P) |
| **Palabra Estudiante** | 505 (5 por estudiante) |
| **Prerrequisito** | 64 |
| **Profesor** | 212 |
| **Restricciones** | 49 |
| **Semestre** | 12 |
| **Subespecialización** | 74 |

**3. Stored procedures (Procedures.sql)**

### Stored Procedure: ObtenerAsignaturasRestantes

**Descripción**: Este procedimiento almacena las materias que el estudiante aún no ha cursado o aprobado, agrupándolas por su categoría. Además, incluye el manejo de optativas y electivas, mostrando si ya han sido tomadas o no. Se filtra por las asignaturas aún no aprobadas por el estudiante y se ordenan por semestre.

* **Parámetro de entrada**:
  + EstudianteId (INT): El ID del estudiante para obtener las asignaturas restantes.
* **Consultas**:
  + Crea una tabla temporal con el estado de las asignaturas aprobadas.
  + Usa CTEs para calcular las asignaturas pendientes (materias restantes), las electivas y optativas tomadas.
  + Devuelve la lista de asignaturas que el estudiante aún no ha cursado, junto con su estado.
* **Consideraciones**:
  + Las variables @tomadasElec y @tomadasOpt controlan cuántas electivas y optativas han sido tomadas.

### Stored Procedure: ObtenerKardexEst

**Descripción**: Devuelve el historial académico (kardex) de un estudiante, incluyendo el nombre de la materia, el código, la nota obtenida, el estado (aprobada o no) y la categoría de la asignatura.

* **Parámetro de entrada**:
  + student\_id (INT): El ID del estudiante para obtener su historial académico.
* **Consultas**:
  + Selecciona las asignaturas cursadas por el estudiante, junto con la nota y el estado de aprobación.

### Stored Procedure: GetMallaCarrera

**Descripción**: Proporciona la malla curricular de una carrera, mostrando todas las asignaturas obligatorias, electivas y optativas, junto con su información relevante.

* **Parámetro de entrada**:
  + carrera\_id (INT): El ID de la carrera para obtener la malla curricular.
* **Consultas**:
  + Realiza un UNION ALL para combinar las asignaturas obligatorias, optativas y electivas, ordenando los resultados por semestre.

### Stored Procedure: ObtenerDetallesAsignatura

**Descripción**: Proporciona los detalles de una asignatura, incluyendo su nombre, código y los prerrequisitos que debe cumplir el estudiante para cursarla. Los prerrequisitos se muestran concatenados.

* **Parámetro de entrada**:
  + asignaturaId (INT): El ID de la asignatura para obtener sus detalles.
* **Consultas**:
  + Devuelve el nombre de la asignatura y sus prerrequisitos, concatenados en un solo campo.

### Stored Procedure: VerificarPrerrequisitos

**Descripción**: Este procedimiento verifica si un estudiante cumple con los prerrequisitos necesarios para cursar una asignatura específica. Devuelve un estado que indica si el estudiante puede o no inscribirse en la materia deseada.

* **Parámetros de entrada**:
  + p\_Estudiante\_id (INT): El ID del estudiante cuya elegibilidad se va a verificar.
  + p\_Asignatura\_id (INT): El ID de la asignatura para la cual se están verificando los prerrequisitos.
* **Consulta**:
  + Realiza una selección que cuenta los prerrequisitos asociados a la asignatura especificada.
  + Comprueba si alguno de los prerrequisitos ya ha sido aprobado por el estudiante.
  + Si el estudiante no ha aprobado ningún prerrequisito requerido, devuelve 'Cumple con los prerrequisitos'; de lo contrario, devuelve 'No cumple con los prerrequisitos'.
* **Consideraciones**:
  + El procedimiento es útil para asegurar que los estudiantes solo puedan inscribirse en asignaturas para las cuales tienen la preparación adecuada, evitando conflictos académicos y asegurando una trayectoria educativa fluida.

### Stored Procedure: ObtenerCorrequisitos

**Descripción**: Este procedimiento recupera los correquisitos asociados a una asignatura específica. Un correquisito es una asignatura que debe ser cursada simultáneamente con otra.

* **Parámetros de entrada**:
  + p\_asignatura\_id (INT): El ID de la asignatura para la cual se desean obtener los correquisitos.
* **Consulta**:
  + Se realiza una selección que identifica la asignatura y su correquisito relacionado.
  + Utiliza un JOIN entre la tabla de asignaturas y la tabla de correquisitos para obtener la información relevante.
  + Devuelve el nombre de la asignatura y su correquisito correspondiente.

### Stored Procedure: ObtenerInformacionAsignatura

**Descripción**: Este procedimiento obtiene las restricciones de la asignatura, la carrera asociada, el colegio y el semestre en el que se restringe.

* **Parámetros de entrada**:
  + asignaturaID (INT): El ID de la asignatura de la cual se desea obtener información.
* **Consulta**:
  + Utiliza INNER JOIN para conectar la tabla de asignaturas con la tabla de restricciones, y LEFT JOIN para obtener información adicional de las tablas de carrera, colegio y semestre.
* **Consideraciones**:
  + Ayuda a garantizar que los estudiantes estén informados sobre las condiciones asociadas a cada asignatura y facilita la planificación académica.

### Stored Procedure: VerificarRestriccionesEstudiante

**Descripción**: Este procedimiento verifica si un estudiante cumple con las restricciones necesarias para tomar una asignatura específica. Las restricciones pueden estar basadas en la carrera, el colegio y el semestre del estudiante.

* **Parámetros de entrada**:
  + estudianteID (INT): El ID del estudiante cuyo cumplimiento de restricciones se va a verificar.
  + asignaturaID (INT): El ID de la asignatura para la cual se están verificando las restricciones.
* **Variables**:
  + cumple (BOOLEAN): Indica si el estudiante cumple o no con las restricciones (inicialmente TRUE).
  + estudianteCarrera (INT): Almacena la carrera del estudiante.
  + estudianteColegio (INT): Almacena el colegio del estudiante.
  + estudianteSemestre (INT): Almacena el semestre del estudiante.
* **Proceso**:
  + **Obtener información del estudiante**: Se recupera la carrera, colegio y semestre del estudiante a partir de su ID.
  + **Verificar restricciones**:
    - Si la asignatura tiene restricciones, se realizan las siguientes verificaciones:
      * **Carrera**: Comprueba si la carrera del estudiante cumple con la restricción.
      * **Colegio**: Verifica si el colegio del estudiante cumple con la restricción.
      * **Semestre**: Verifica si el semestre del estudiante es adecuado para la asignatura.
  + **Resultado**: Se selecciona un mensaje indicando si el estudiante cumple o no con las restricciones.
* **Consideraciones**:
  + Este procedimiento es útil para los administradores académicos y estudiantes para garantizar que se sigan las normas establecidas antes de inscribirse en una asignatura.
  + Facilita la toma de decisiones informadas en la planificación académica y la inscripción en cursos.

### Stored Procedure: getOptativas

**Descripción**: Este procedimiento obtiene una lista de asignaturas optativas que un estudiante puede inscribir, basándose en varios criterios, incluyendo el kardex, la malla curricular y las restricciones.

* **Parámetro de entrada**:
  + EstudianteID (INT): El ID del estudiante para quien se están buscando las optativas.
* **Proceso**:
  + **Seleccionar asignaturas**:
    - Se seleccionan las asignaturas que cumplen con los siguientes criterios:
      * No están ya inscritas en el kardex del estudiante.
      * No están incluidas en la malla curricular de la carrera del estudiante.
      * Cumplen con las restricciones de carrera, colegio y semestre.
      * Cumplen con los prerrequisitos establecidos.
      * El número de la asignatura está en el rango de 2 a 6.
      * Pertenecen al colegio del estudiante
  + **Resultados**:
    - Se devuelven los IDs de asignaturas, su número, nombre, carrera y colegio.

### Stored Procedure: GetElectivas

**Descripción**: Este procedimiento obtiene una lista de asignaturas electivas que un estudiante puede inscribir, excluyendo las asignaturas ya inscritas y las de su carrera y colegio.

* **Parámetro de entrada**:
  + EstudianteID (INT): El ID del estudiante para quien se están buscando las electivas.
* **Proceso**:
  + **Seleccionar asignaturas**:
    - Se seleccionan las asignaturas que cumplen con los siguientes criterios:
      * No están ya inscritas en el kardex del estudiante.
      * No están incluidas en la malla curricular de la carrera del estudiante.
      * Cumplen con las restricciones de carrera, colegio y semestre.
      * Cumplen con los prerrequisitos establecidos.
      * No pertenecen a la carrera del estudiante.
      * No pertenecen al colegio del estudiante.
  + **Resultados**:
    - Se devuelven los IDs de asignaturas, su número, nombre, carrera y colegio.

### Stored Procedure: VerCursos

**Descripción**: Este procedimiento devuelve información sobre los cursos asociados a una asignatura específica, incluyendo detalles como el código, nombre, NRC, modalidad, días, horario, profesor, campus y ubicación.

* **Parámetro de entrada**:
  + asignaturaId (INT): El ID de la asignatura para la cual se desea obtener información de los cursos.
* **Proceso**:
  + **Seleccionar detalles de cursos**:
    - Se obtienen datos de la asignatura, curso, modalidad, horario, profesor, campus y ubicación.
    - Se agrupan los días en los que se imparten las clases.
  + **Resultados**:
    - Se devuelve una lista con el código de la asignatura, nombre, NRC, modalidad, días, horario, nombre del profesor, campus, ubicación y créditos.

### Stored Procedure: getSubEspecializacion

**Descripción**: Este procedimiento obtiene información sobre las asignaturas pertenecientes a una subespecialización específica.

* **Parámetro de entrada**:
  + p\_SubEsp\_id (INT): El ID de la subespecialización.
* **Proceso**:
  + **Seleccionar asignaturas de la subespecialización**:
    - Se obtienen el código, nombre y créditos de las asignaturas asociadas a la subespecialización.
  + **Resultados**:
    - Se devuelve una lista con la clave, nombre y créditos de las asignaturas.

### Stored Procedure: verSubEspRestantes

**Descripción**: Este procedimiento devuelve las asignaturas que un estudiante aún no ha cursado dentro de su subespecialización.

* **Parámetro de entrada**:
  + p\_Estudiante\_id (INT): El ID del estudiante.
* **Proceso**:
  + **Seleccionar asignaturas restantes**:
    - Se obtienen las asignaturas que forman parte de la subespecialización del estudiante, excluyendo las que ya han sido aprobadas.
  + **Resultados**:
    - Se devuelve una lista con la clave, nombre y créditos de las asignaturas restantes.

### Stored Procedure: verSubEspTomadas

**Descripción**: Este procedimiento devuelve las asignaturas que un estudiante ha aprobado dentro de su subespecialización.

* **Parámetro de entrada**:
  + p\_Estudiante\_id (INT): El ID del estudiante.
* **Proceso**:
  + **Seleccionar asignaturas tomadas**:
    - Se obtienen las asignaturas que forman parte de la subespecialización del estudiante y que han sido aprobadas.
  + **Resultados**:
    - Se devuelve una lista con la clave, nombre y créditos de las asignaturas aprobadas.

**4. Creación de Índices (Index.sql)**

#### **Índice: idx\_kardex\_asignatura\_estudiante\_asignatura**

* **Descripción**: Este índice se utiliza para optimizar las consultas relacionadas con la tabla Kardex\_Asignatura, especialmente aquellas que filtran por la columna Aprobada.
* **Tabla**: Kardex\_Asignatura
* **Columna(s)**:
  + Aprobada

#### **Índice: idx\_asignatura**

* **Descripción**: Este índice mejora la eficiencia de las consultas que buscan información sobre las asignaturas, utilizando la columna Nombre como criterio de búsqueda.
* **Tabla**: Asignatura
* **Columna(s)**:
  + Nombre

#### **Índice: idx\_carrera\_abrv**

* **Descripción**: Este índice está diseñado para acelerar las consultas que filtran por la abreviatura del colegio, lo que permite una recuperación más rápida de la información relacionada.
* **Tabla**: Colegio
* **Columna(s)**:
  + Abreviatura

#### **Índice: idx\_asignatura\_num**

* **Descripción**: Este índice facilita las consultas que buscan asignaturas utilizando la columna Numero, mejorando la velocidad de búsqueda.
* **Tabla**: Asignatura
* **Columna(s)**:
  + Numero

**5. Creación de Triggers (triggers.sql)**

Para asegurar la integridad y trazabilidad de las modificaciones realizadas en la base de datos, se han implementado triggers en cada tabla. Cada trigger se activa después de una operación de eliminación (AFTER DELETE), inserción (AFTER INSERT) o actualización (AFTER UPDATE).

Cada vez que se realiza una modificación en una tabla, el trigger correspondiente ejecuta las siguientes acciones:

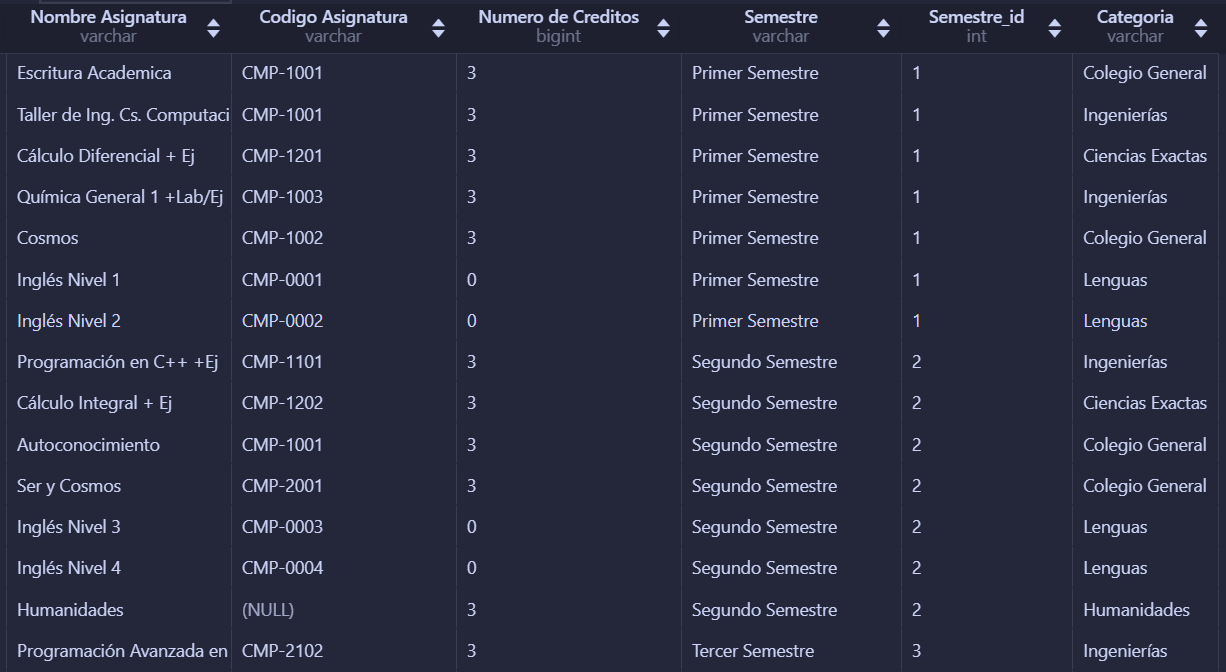
1. **Tipo de Trigger**:
   * AFTER DELETE
   * AFTER INSERT
   * AFTER UPDATE
2. **Acciones Realizadas**:
   * Los triggers capturan la información relevante de la operación realizada, incluyendo:
     + **Usuario**: El usuario que realizó la modificación.
     + **Tabla**: El nombre de la tabla afectada.
     + **Descripción**: Un breve resumen de la operación.
3. **Registro en la Tabla audit\_log**:
   * Los datos recolectados por los triggers se almacenan en la tabla audit\_log, lo que permite llevar un registro completo de todas las modificaciones realizadas en la base de datos. Esto es esencial para auditorías y para garantizar la trazabilidad de los cambios.

Esta implementación asegura que cualquier cambio en la base de datos sea fácilmente rastreable, promoviendo así la transparencia y la seguridad en la gestión de datos.

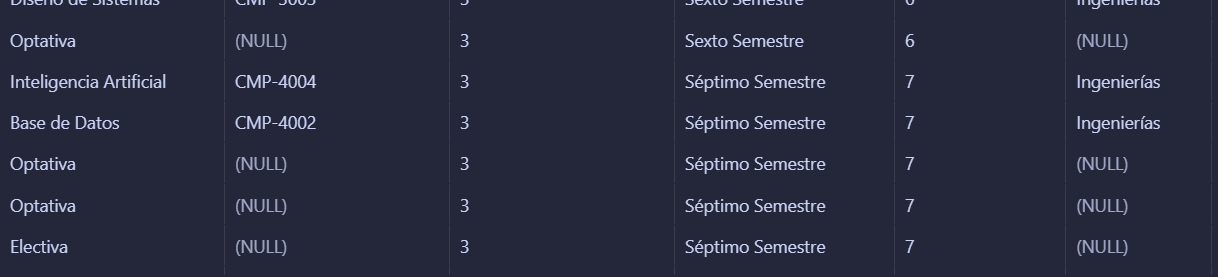
**6. Pruebas de los Procedures (CallProcedures.sql)**

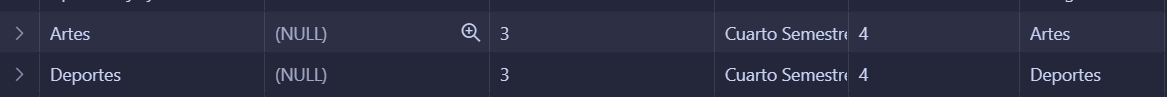
1. **Call GetMallaCarrera(31)**:

**El id\_Carrera = 31 es la carrera de Ingeniería en Ciencias de la computación y al realizar el procedure se tiene lo siguiente.**



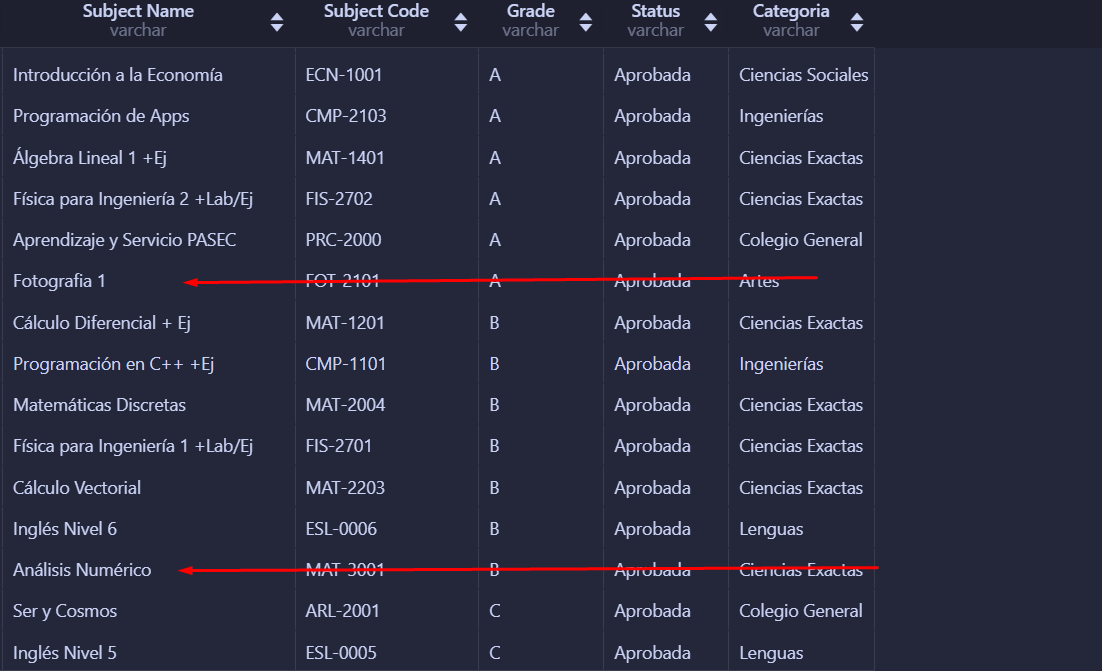
La tabla incluye las materias no específicas, como “Humanidades” y Optativas y Electivas





1. **Call ObtenerKardexEst(00999999)**

El Estudiante\_id = 00999999 es un estudiante de prueba que está entrando a 5to semestre de computación.

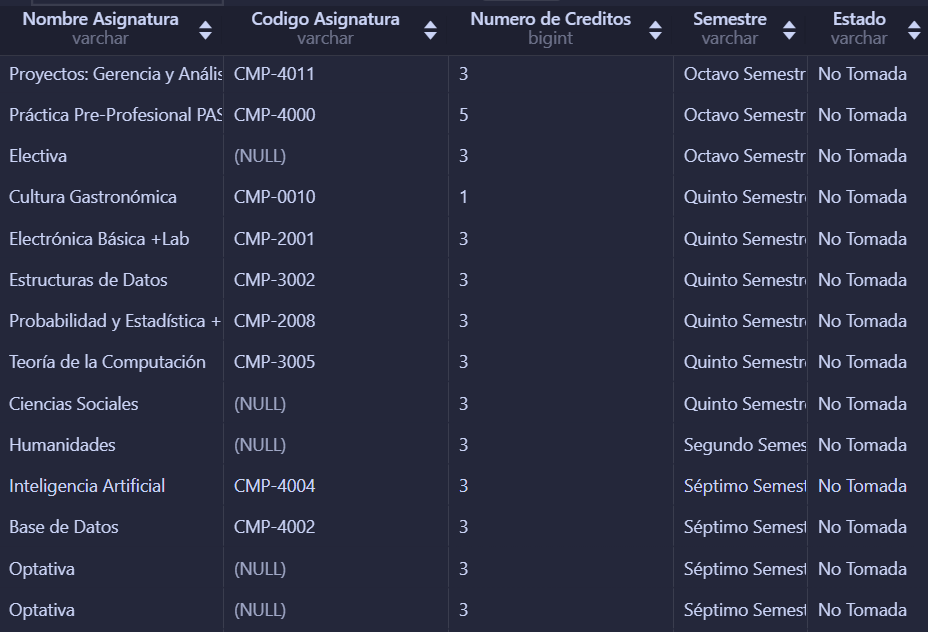


Se muestra el kardex académico del estudiante. Es importante mencionar que el estudiante ha tomado una materia de artes que no consta en su malla, porque lo que cumple con una materia electiva y una materia de artes.

Además, ha tomado una materia de su colegio, que no consta en su malla, por lo que se considera que tiene una materia optativa.

1. **Call ObtenerAsignaturasRestantes (31)**

Se presentan las asignaturas restantes para la malla del estudiante

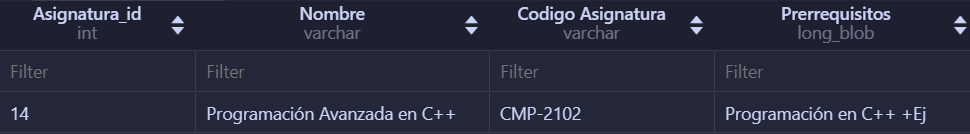


Se observa que la materia de Artes, una optativa y una electiva ya no están presentes, pues ya se verifico que se tomaron.

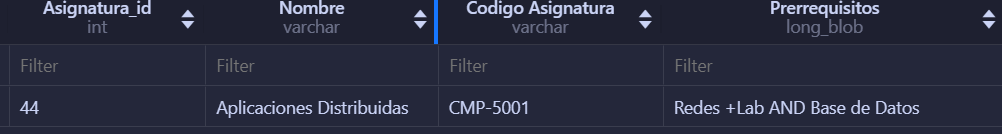
1. **Call ObtenerDetallesAsignatura(186)**

Asignatura\_id = 186 hace referencia a una asignatura con prerrequisitos



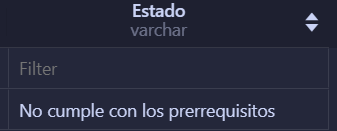


Para una asignatura con varios prerrequisitos se imprime de la sig manera.

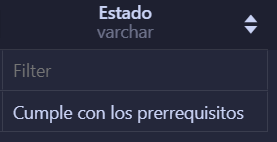


1. **VerificarPrerrequisto(00999999, 186)**

Se verifica si es estudiante puede tomar la materia.



**Para asignatura\_id = 14**

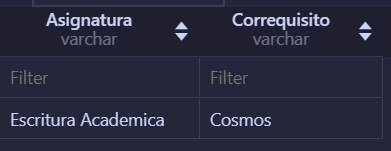


1. **ObtenerCorrequisitos(5)**

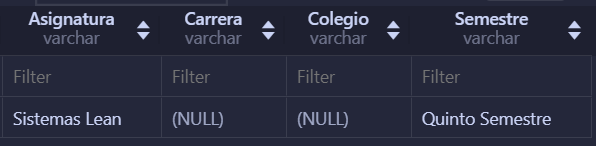
Se obtiene los correquisitos de la asignatura\_id = 5

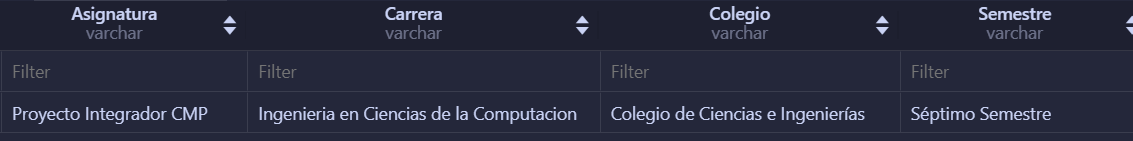


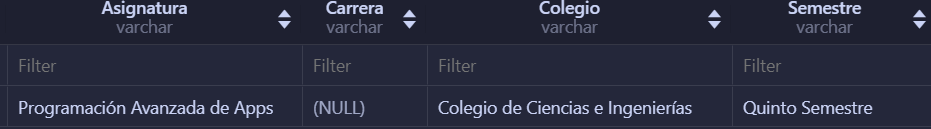
Si se buscara el id de “Escritura Académica”



1. **Call ObtenerInformacionAsignatura(186)**

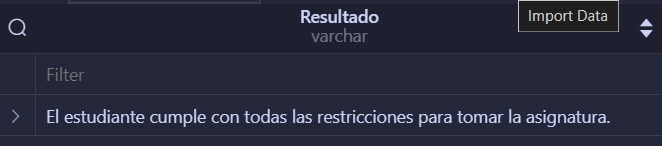




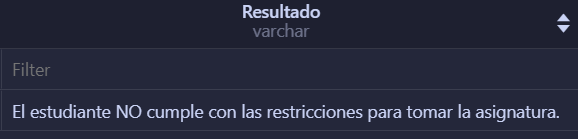


1. **Call VerificarRestriccionesEstudiantes(00999999, 186)**

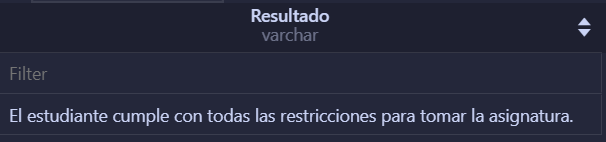
**Para Sistemas Lean se cumplen las restricciones, el estudiante está pasando a quinto semestre**



**Para Proyecto Integrador CMP no se cumple las restricciones, el estudiante si pertenece al colegio y carrera pero aun no llega al semestre mínimo**

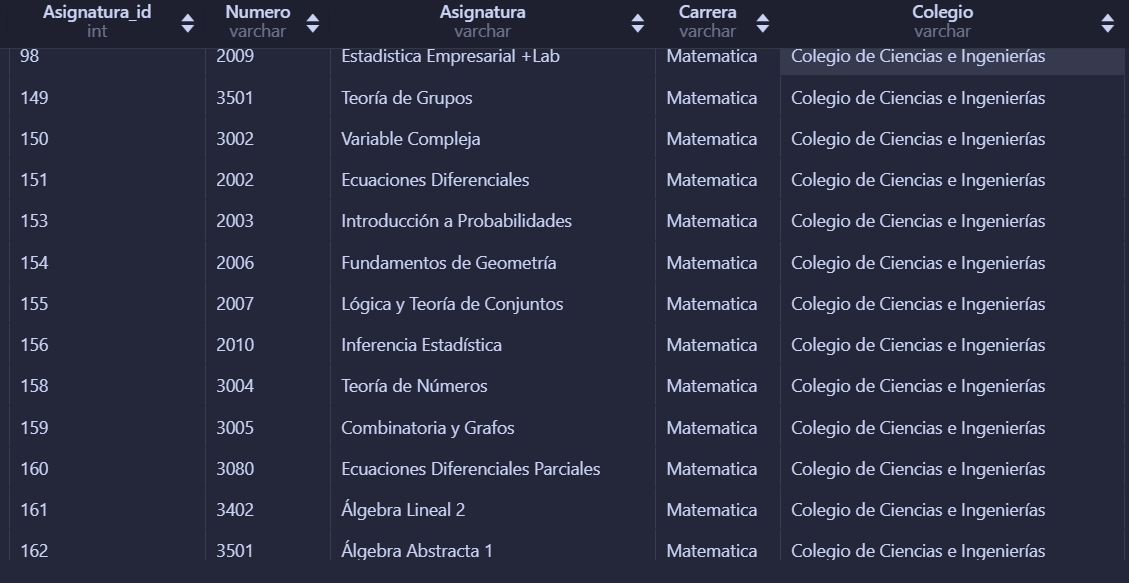


**Para programación avanzada de apps se cumple las restricciones.**



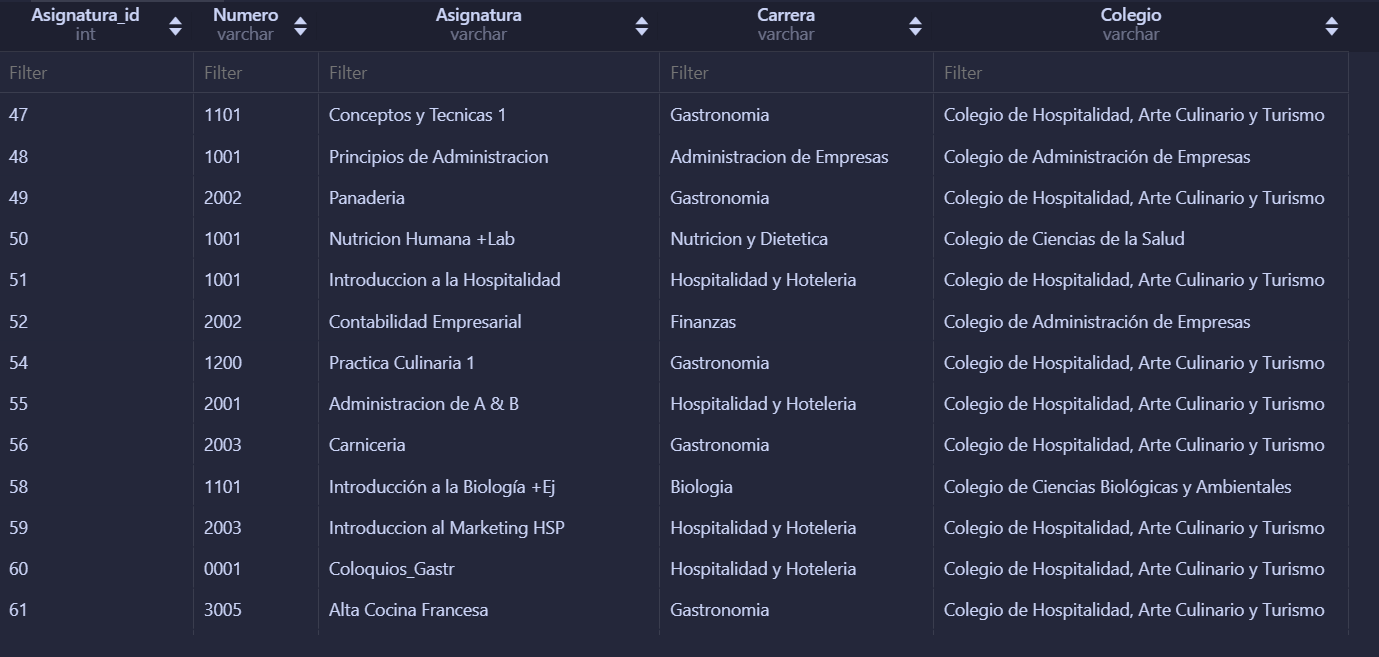
1. **Call getOptativas(00999999)**

Se consideran optativas las asignaturas del mismo colegio que la carrera del estudainte, que no pertenezcan a su malla y tengan un número 2000 en adelante. Se revisan previamente que se cumplan con prerrequisitos y las restricciones.



1. **Call getElectivas(00999999)**

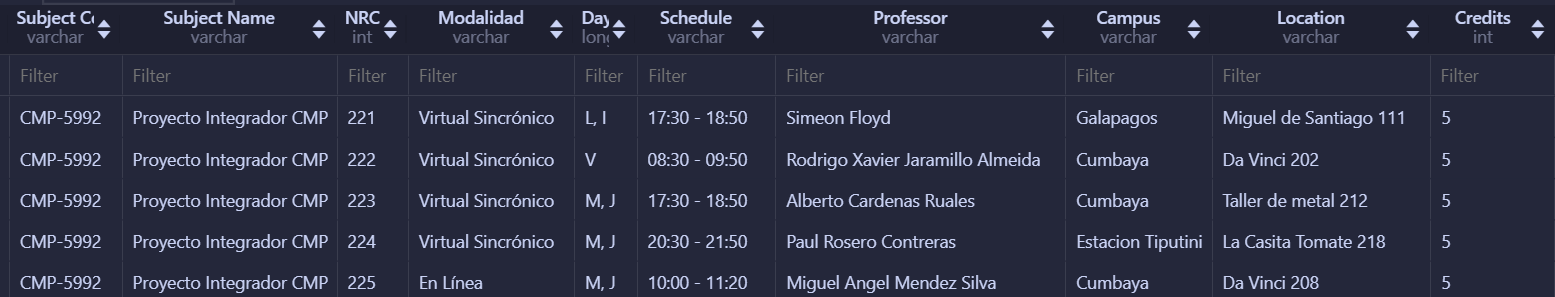
Se consideran electivas aquellas asignaturas que no formen parte de la malla curricular ni del colegio del estudiante. Se verifican prerrequisitos y restricciones previo a entregar la lista.



1. **Call VerCursos(45)**

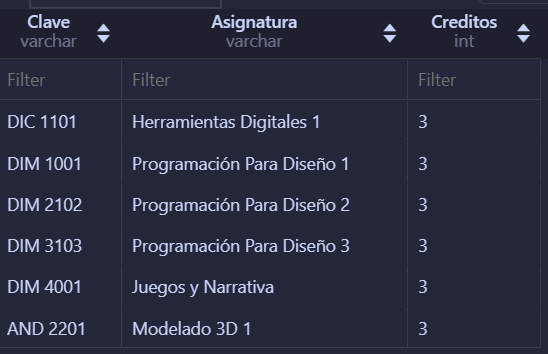
Asignatura\_id = 45 es Proyecto Integrador CMP.

Cada curso se creó de forma al azar, tanto su campus, profesor días, horario, modalidad y localización

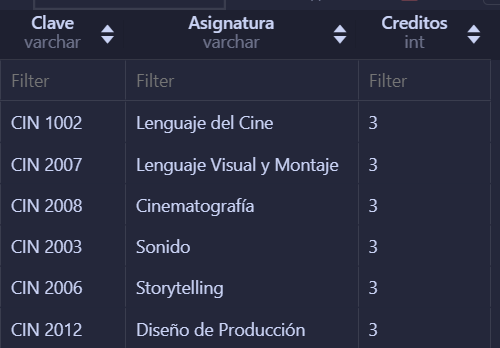


1. **Call getSubEspecializacion(17)**

SubEsp\_id = 17 hace referencia a la subespecializacion ‘Videojuegos’

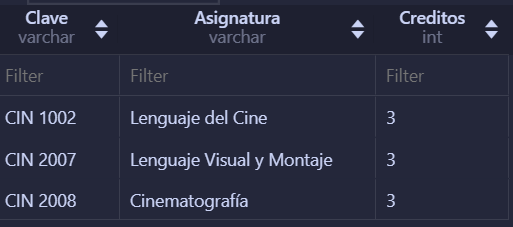


Para Cine

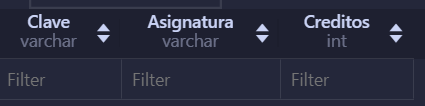


1. **Call verSubEspTomadas(00777152)**

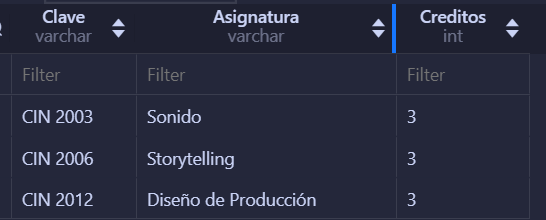
El estudiante\_id = 00777152 está tomando la subespecialización de Cine



Si un estudiante no toma ninguna subespecializacion, la tabla esta vacia.



1. **Call verSubEspRestantes(00777152)**



Si un estudiante no toma ninguna subespecializacion, la tabla esta vacia.

