

PARCIALITO DE SQL

Forma de entrega

Para la automaización de la corrección se pide que la entrega de este parcialito se deberá hacer con una archivo .sql por ejercicio. Cada archivo deberá tener el siguiente formato:

`<padron>_<NombreCompletoSinEspacios>_<nroEjercicio>.sql`

```
12345_NombreCompletoSinEspacios_1.sql
12345_NombreCompletoSinEspacios_2.sql
12345_NombreCompletoSinEspacios_3.sql
12345_NombreCompletoSinEspacios_4.sql
12345_NombreCompletoSinEspacios_5.sql
12345_NombreCompletoSinEspacios_6.sql
```

Además, se deberá agregar como comentario en cada archivo el resultado generado de la ejecución de dicha consulta. Por lo tanto, el contenido de cada sql deberá ser:

`12345_ApellidoCompleto_1.sql`

```
SELECT ...
FROM ...

-- Resultados --
-- <Texto del resultado>
```

Importante, se espera que el alumno entregue estos 6 ejercicios cada uno en un archivo distinto con los nombres correctos, dentro de un ZIP (para poder dejarlo en el campus). De no ser así, se dará por no corregido.

`<padron>_<NombreCompletoSinEspacios>.zip`

`12345_NombreCompletoSinEspacios.zip`

Considerando los esquemas del dataset (**OJO NO ES IGUAL AL DE CLASE**), resuelva los siguientes puntos escribiendo para cada uno de ellos una única consulta SQL que devuelva el resultado pedido:

Nota: En la ultima página de este documento puede encontrar una descripción del dataset.

1. Obtener el padrón y apellido de el/los alumno/s que tenga/n la mayor cantidad de materias aprobadas.

Nota: Considerar la posibilidad de que sean más de uno.

2. Obtener el padrón y apellido de aquellos alumnos que tienen nota en las materias 71.14 y 71.15 y no tienen nota ni en la materia 75.01 ni en 75.15.

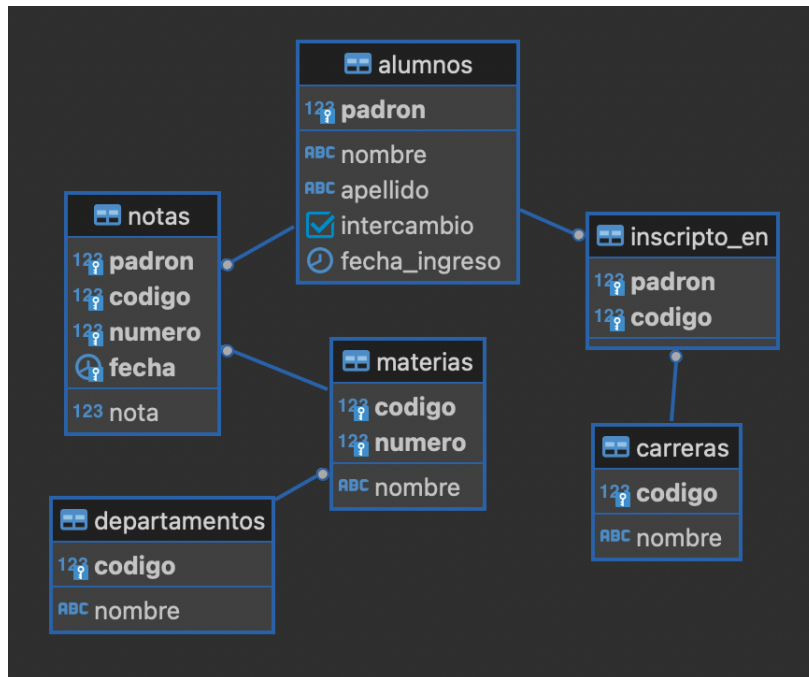
3. Se necesita saber el promedio general para cada carrera y cada departamento.
Devolviendo código de carrera, código de departamento, y el promedio de notas para todos los alumnos de cada departamento, ordenados por carrera y departamento.
Nota: no considerar combinaciones de departamento / carrera tal que no haya alumnos de dicha carrera con notas en dicho departamento.
Nota 2: Si un alumno está inscripto en más de una carrera, sus notas aportan a los promedios de todas esas carreras.
4. Mostrar el padrón, apellido y promedio para aquellos alumnos que tienen nota en más de 3 materias y un promedio de al menos 5.
5. Para cada nota del alumno más antiguo, mostrar su padrón, código de departamento, número de materia y el valor de la nota.
Nota: En caso de que sean más de uno los alumnos más antiguos, mostrar dichos datos para todos esos alumnos.
*Nota 2: Si bien en la realidad puede darse que los valores de padrón sean estrictamente crecientes en el tiempo. **NO utilizar este criterio para determinar al alumno más antiguo.***
6. Listar el padrón de aquellos alumnos que, por lo menos, tienen nota en todas las materias que aprobó el alumno de padrón 71000.

En el Campus en la sección de Exámenes encontrará el dataset que puede utilizar para entender el contenido de la tablas y validar sus respuestas.

ANEXO: SIMPLE EXPLICACIÓN DEL DATASET

A continuación una pequeña descripción de las tablas pertenecientes al dataset. El mismo puede ser descargado desde el campus en este link.

- **departamentos:** Código y nombre de los departamentos de la facultad.
- **materias:** Código (de departamento), número y nombre de las materias en las que puede tener nota cada alumno.
- **alumnos:** Varios datos de cada alumno de la facultad, tengan o no notas registradas. Se tiene la fecha de ingreso a la facultad y un indicador de si es o no de intercambio.
- **notas:** Registra todas las notas que hayan tenido los alumnos y la fecha de cada una de ellas. Puede haber varias notas de un mismo alumno y materia en caso de que no haya aprobado en la primer instancia.
- **carreras:** Código y nombre de las carreras de la facultad.
- **inscripto_en:** Vincula a cada alumno con la o las carreras en las que esté inscripto.



Nota

El set de datos de prueba permite detectar si una consulta es incorrecta, ya sea por devolver datos incorrectos o por no devolver datos que debiera. En cambio, si los datos devueltos son correctos para este set de datos esto no garantiza que la consulta sea correcta, ya que para otro set de datos podría fallar. Es necesario que cada consulta hecha funcione no sólomente con el set de datos dado sino con cualquier otro posible set de datos con el mismo esquema.

“No es posible comprobar *TODAS* las entradas posibles para un programa dado”