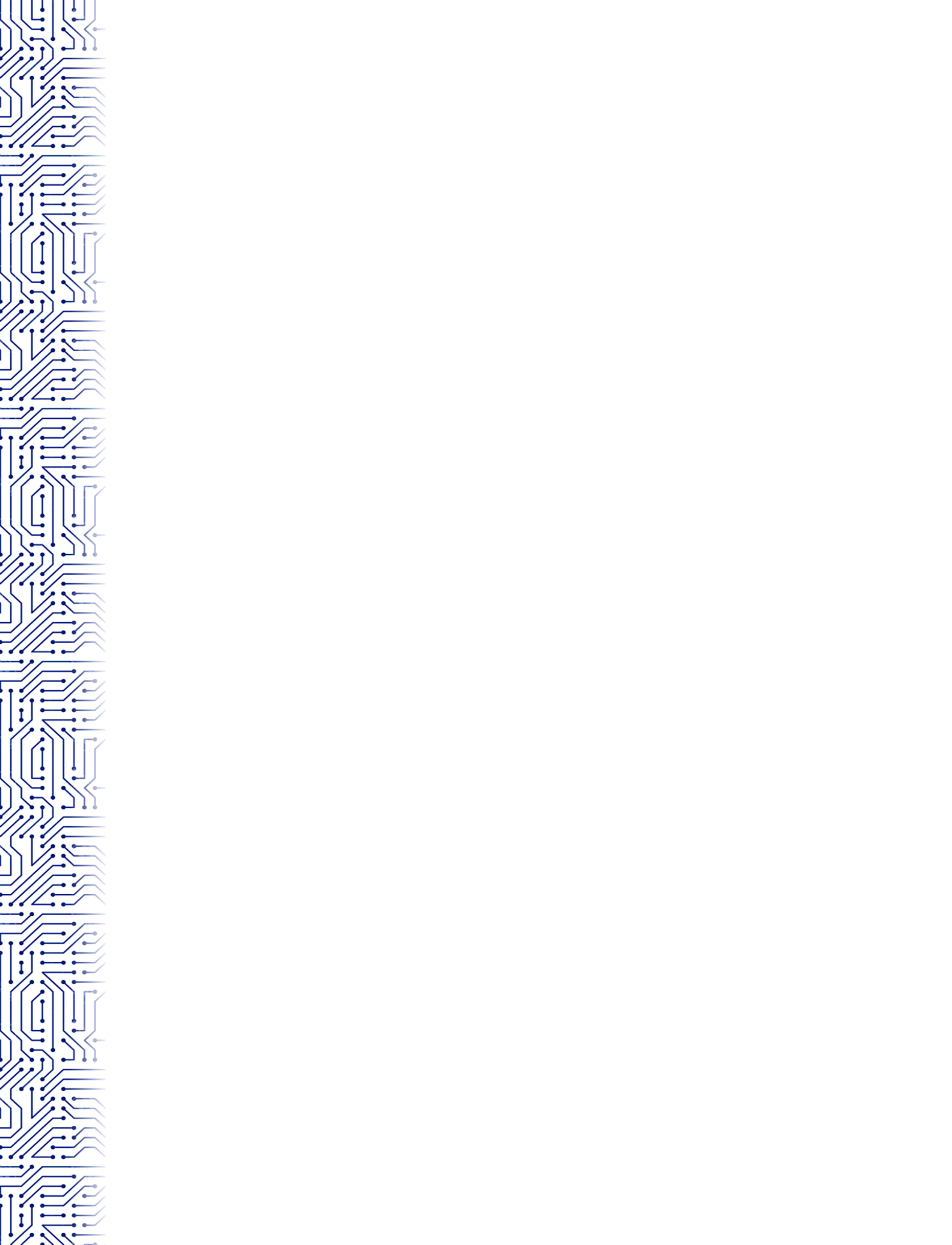
**  
TP BASE DE DATOS**

***Competencia***

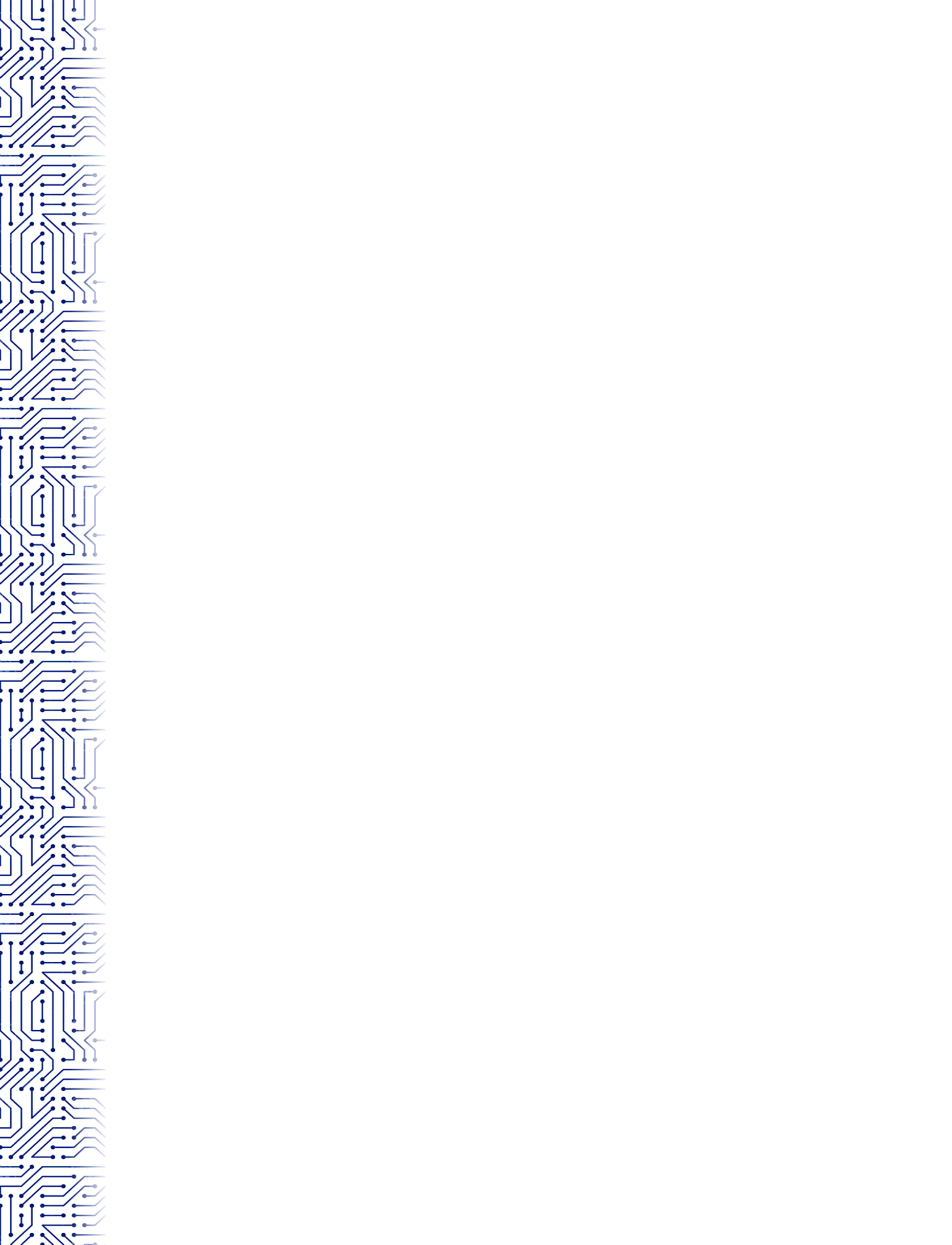
***de Esquiadores***

**Alumnos:**

* **Motta, Marino Juan**
* **Galarza, Jeico Tiziano**
* **Yucra, Ángel Patricio**

**Profesora:**

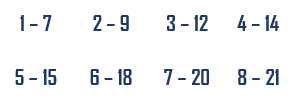
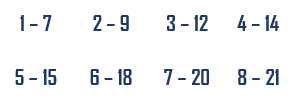
* **Sandra Daujan**

**ÍNDICE**

**Enunciado ………………………………………………………….. 3**

**Estrategia de Resolución ………….…………………… 4**

**MER y Supuestos ……………………………………………… 5**

**Consultas …………………………………………………………… 7**

**Situaciones Lógicas Propias ….……….…………. 23**

**Enunciado de la Base de Datos**

Como parte de la organización de las próximas olimpiadas de invierno, se decide la creación de un sistema de Como parte de la organización de las próximas olimpiadas de invierno, se decide la creación de un sistema de información para realizar la gestión de las pruebas de esquí.

Del análisis realizado se obtiene la siguiente información:

Los juegos se componen de una serie de pruebas, en cada una de las cuales intervienen varios participantes. Cada participante en una prueba puede intervenir a título individual (esquiador individual) o bien formando parte de un equipo en cuyo caso el participante será el equipo (no el esquiador). De cada esquiador (individual o de equipo) se desea tener el *DNI, el nombre completo y la edad*; y en caso de participar de forma individual su *nacionalidad*.

A cada participante en una prueba (esquiador individual o equipo participante) se le asigna un código de participación que identifica a la *sigla del nombre de la prueba junto a un número secuencial*.

De cada equipo participante se conoce un *nombre, un entrenador y los esquiadores que lo componen*. El que un equipo participe en una prueba no significa que todos los esquiadores que lo componen intervengan en la misma. Un esquiador que forma parte de un equipo, no podrá cambiarse a otro ni actuar a título individual mientras duran los Juegos. Tampoco un esquiador individual no podrá pasar a formar parte de un equipo.

Existen una serie de federaciones de esquí, cada una de las cuales tiene un *nombre y un número de federados*, en las federaciones se federan todos los esquiadores.

Por un acuerdo existente entre las distintas federaciones, no se permite que ningún esquiador se federe en dos federaciones distintas. Tampoco se admite que participen esquiadores (ni a título individual ni formando parte de un equipo) que no estén federados.

Cada federación puede o no administrar varias estaciones de esquí y toda estación se administrará **al menos por** una federación, aun cuando puede haber estaciones de esquí administradas **conjuntamente por** varias federaciones.

Una estación de esquí se identifica por un *código, tiene un nombre,* **unas personas** de contacto, *una dirección, un teléfono y un número total de kilómetros esquiables*, así como las **pistas de las que dispone**.

Cada pista se identifica a partir del código de la estación de esquí y un *número secuencial*. Se consideran también como pistas (para la realización de pruebas de largo recorrido) a varias de estas pistas (siempre de la misma estación) que por sus características físicas pudieran enlazarse. Así por ejemplo la *pista diez estaría compuestas por las pistas dos y cuatro*.

Se requiere, para poder planificar las pruebas mantener esta utilización combinada de las pistas. Para cada pista se mantiene también su *longitud en kilómetros y su grado de dificultad (en la escala azul. verde. roja y negra).*

La realización de **cada prueba** se desarrollará a lo largo de varios días utilizando una serie de pistas de una **única estación**. Los equipos o esquiadores individuales podrán competir en diferentes pruebas. Para cada participante en una prueba (equipo o esquiador individual) se registrará la *fecha o fechas en que participa, el tiempo empleado y la posición obtenida*; en el caso de equipos estos datos se obtienen de los *correspondientes a cada uno de los esquiadores del equipo que han intervenido en la prueba*. Cada prueba se identifica por un nombre, será de un *tipo (fondo, slalom, salto.) tendrá unas fechas previstas de realización y se registrará el* participante ganador y el tiempo empleado por este.

**Estrategia para la Resolución**

Para hacer la base de datos, primero analizamos el enunciado, y como se puede ver más arriba, está completamente marcado con diferentes cosas, como por ejemplo las partes *así* son simplemente los atributos que de primera mano supimos que deben estar en las tablas, lo que está marcado así hace referencia a algunas de las tablas que identificamos en las primeras lecturas, etc.

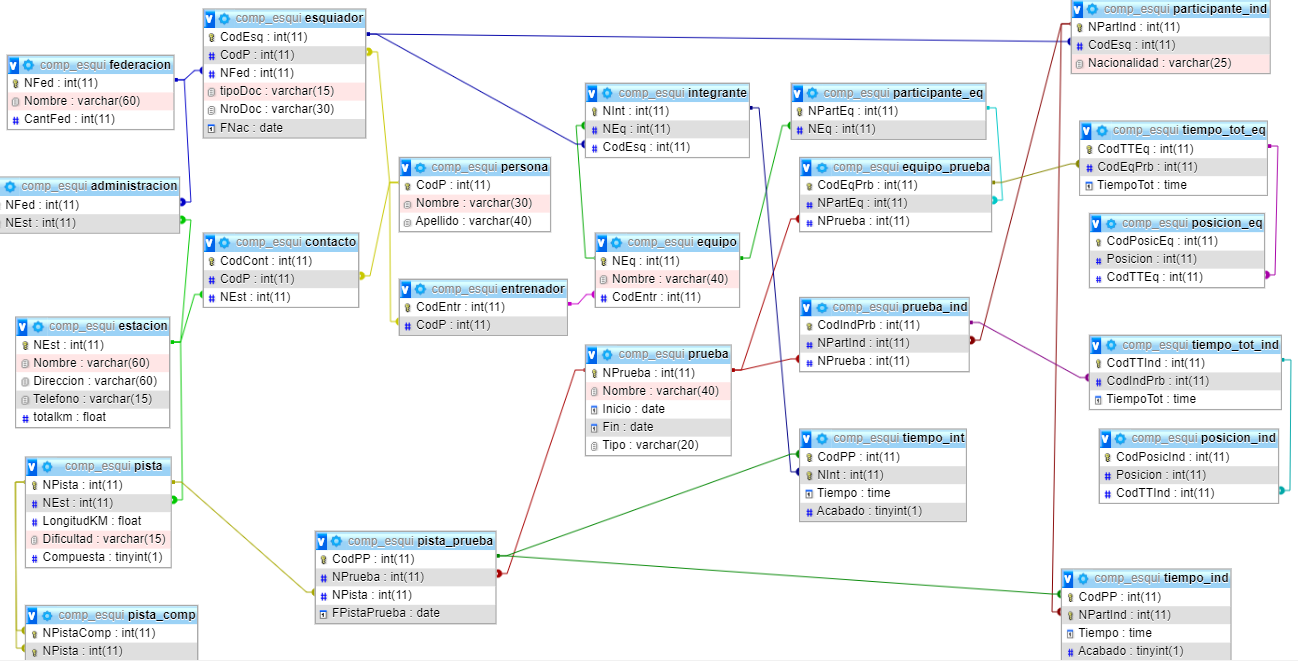
Tuvimos que pensar cuidadosamente cada estructura y frecuentemente estuvimos debatiendo sobre separar o no toda la parte de los esquiadores relacionada con “Equipo” e “Individual”, finalmente quedamos en que no hay forma de que queden juntos y cumplan la función que necesitamos, así que separados quedaron.

Las pistas también dudamos de separarlas o no, y al final decidimos poner un booleano que diga si son o no compuestas, en el caso de que sí, estarán vinculadas a una tabla dedicada a pistas compuestas.

Para hacer el MER inicial, utilizamos draw.io, y fue la parte más extensa del trabajo por todas las veces que rehicimos ciertas partes. Luego, armamos la base de datos y todo el resto del código (inserts) con el editor de texto Visual Studio Code, ya que estamos más familiarizados y le podemos sacar provecho a algunas funciones que no están en Notepad+ + (un ejemplo sería el poder copiar solo ciertas partes de varias filas al mismo tiempo, útil sobre todo en la parte de los inserts para copiar PK y pasarlos como FK a otras tablas). Para comunicarnos durante la resolución utilizamos mayormente la aplicación Discord que tiene chats de texto y de voz, además de que creamos un repositorio en GitHub para poder compartir siempre la última versión del código *(más info en* [*https://github.com/Marin37/Comp\_Esqui-BD*](https://github.com/Marin37/Comp_Esqui-BD)*).*

Luego, una vez terminada la base de datos CompEsqui, graficamos el MER desde la página de phpmyadmin, y lo ordenamos para que sea legible, puesto que si no se ordena quedan todos encimados uno sobre otro y es imposible trabajarlo. Más adelante mostramos el gráfico del MER.

Durante la parte de los inserts, en las primeras 10 tablas insertamos los datos en conjunto, pero luego ya nos dividimos formalmente uno en toda la parte de Equipos, uno en las tablas de Individual, y uno que avance con el documento para entregar, obviamente todo consultándolo entre nosotros, pero el “trabajo duro” (escribir cada cosa en realidad) quedó dividido de esa manera.

**Gráfico Modelo-Entidad-Relación**

**Supuestos**

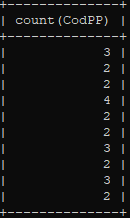
* Tengamos en cuenta que los contactos solo pueden servir en una única estación a la vez.
* Como una Federación puede administrar varias estaciones y una estación puede ser administrada en conjunto, la tabla “Administración” vincula ambas tablas.
* El “tipo” en pistas dice si la pista es compuesta o no. Si es compuesta, ese número de pista estará reflejado en la tabla PistaComp en NPistaComp, y todas las pistas que contenga estarán en NPista de esa tabla.
* Las pruebas se desarrollarán a lo largo de varios días, y cada pista de la prueba tiene en Pistas\_Prueba su FPruebaPista correspondiente.
* En Tiempo\_Integrante se registrarán los tiempos de cada *Integrante del equipo* (NInt) que participen en la *pista N* de la *prueba N* (CodPistaPrueba). Si un integrante hace más de una prueba o corre en más de una pista, sus tiempos tendrán distintos CodPistaPrueba, y si en una misma pista hay más de un integrante (relevos u otros equipos), habrá tuplas con distintos NInt.
* El Tiempo\_Ind se aplica de la misma manera que Tiempo\_Int, habrá un CodPP en cada de *Pista N en Prueba N*, y cada participante tendrá su propio NPartInd para los tiempos.
* El Acabado en Tiempo\_Int y Tiempo\_Ind es un booleano que se pone en 1 cuando se inserta el último tiempo de un integrante del equipo//participante individual en una misma prueba. Mientras alguien lo tenga en esa prueba, nos sirve para poder identificar si ese equipo completó la prueba en su totalidad, y si nadie el equipo lo tiene en esa prueba, no se debe contar para los resultados (porque si no el equipo tendría menos tiempo porque nunca hizo X pista y ganaría sin completar la prueba).
* Cada Pista en la que se deba competir POR PRUEBA va a ser transitada un solo día, es decir que en la misma prueba no habrá varios días para la misma pista y los esquiadores individuales o en equipo participarán solo ese día. Cabe destacar que si OTRA PRUEBA requiere correr en esa pista, no hay problema en que se programe otra fecha para la misma pista siempre que estemos en otra prueba.
* Para buscar a los ganadores (calculable) buscamos a los que tengan posición 1.

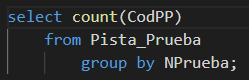
**Consultas**

***Consulta 1***

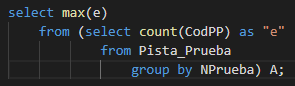
*Informar ganador y tiempo utilizado de la prueba que se desarrolló en la mayor cantidad de jornadas.*

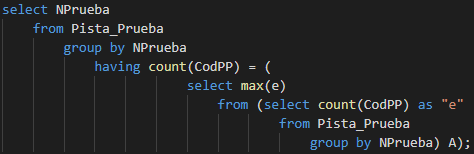
Antes de proyectar los ganadores, hay que identificar la “mayor cantidad de jornadas”, y luego la prueba que cumpla con las condiciones.



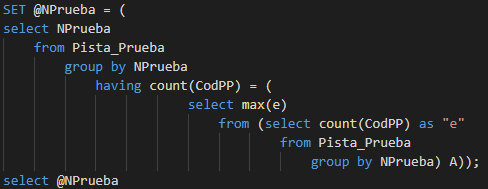


Con la cantidad de jornadas, ahora hace falta obtener el máximo y a que prueba pertenece



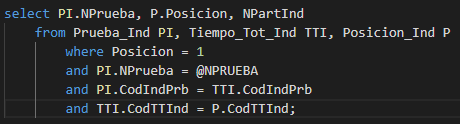
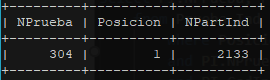


Y para un código más limpio guardaremos el resultado en una variable

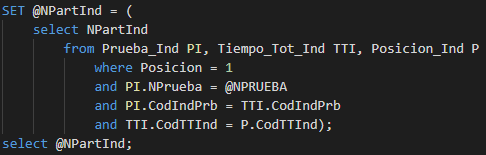


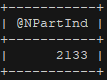


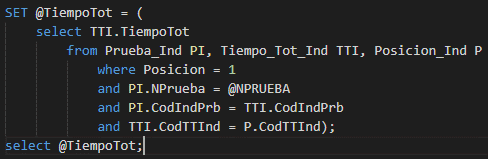
Ahora buscaremos el ganador de esa prueba, vinculando varias tablas a la vez

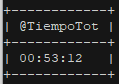


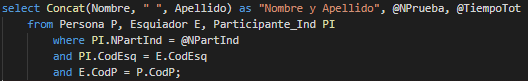
Con ese resultado, podemos simplemente sacar el NPartInd a una variable para facilitar buscar los datos del ganador, y lo mismo para el tiempo

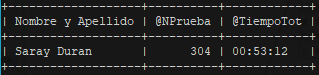








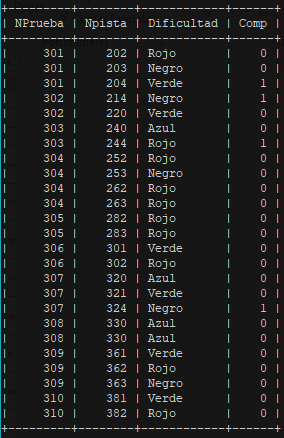
El resultado final queda entonces:

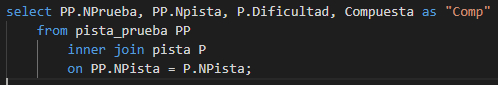


Eso serviría para la parte individual, pero si quisiéramos sacar los equipos si bien hay que cambiar los nombres de las tablas y de los atributos, la lógica se aplica de la misma manera. De todas formas, en esta prueba solo participaron esquiadores individuales y no hay equipos compitiendo en la misma, por lo tanto, la resolución finaliza aquí.

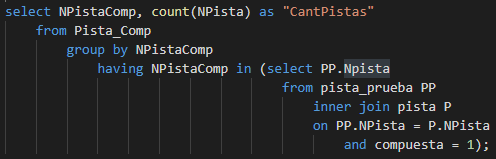
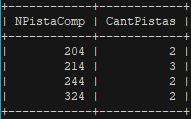
***Consulta 2***

*Listar cada una de las pruebas junto el identificador de pista, la dificultad y en caso de estar compuesta por otras pistas la cantidad de pistas que la componen.*

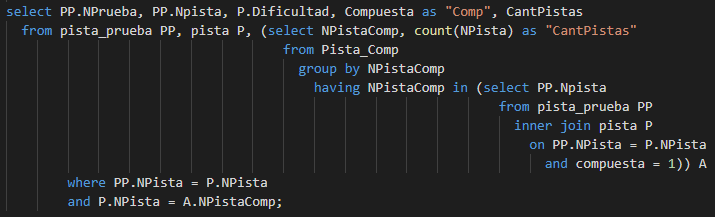
Primero obtenemos una tabla con las pruebas y los demás datos

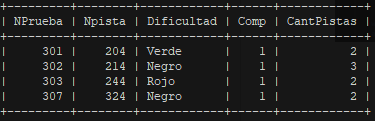


Luego, averiguamos cuantas pistas forman las compuestas



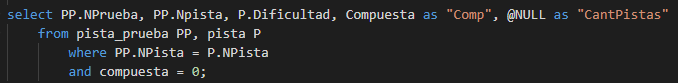
Ahora agregamos la cantidad de pistas con el mismo formato que el primer paso, agregando la cantidad de pistas

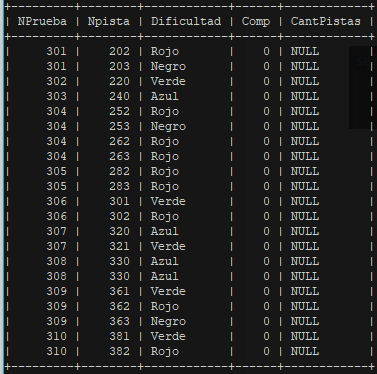




Llamaremos a esta tabla

“**CONSULTA 2.1**”

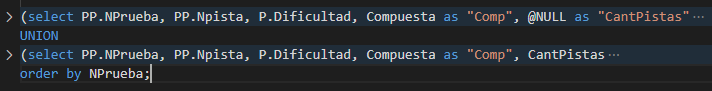
Ahora, unimos la tabla anterior con la del primer paso, pero filtrando las compuestas ya que las incorporamos en la segunda tabla. Talvez es medio confuso y las subconsultas son bastante grandes, así que lo explicamos mediante las imágenes



PRIMERA TABLA  
FILTRADA Y MODIFICADA

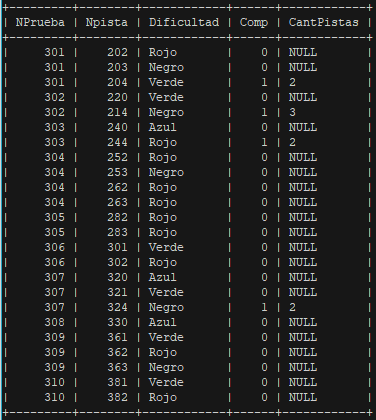
la llamaremos **“CONSULTA 2.2”**

Finalmente, utilizamos el operador UNION para juntar los datos de las dos tablas y los ordenamos por el Número de Prueba.



**CONSULTA 2.2**

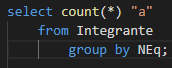
**CONSULTA 2.1**

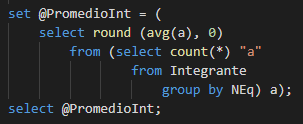


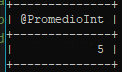
***Consulta 3***

*Informar los datos de los esquiadores que pertenecen a los equipos que superen el promedio de integrantes de esquiadores de todos los equipos.*

Primero sacamos la cantidad de integrantes por cada equipo, y luego el promedio de esos datos lo guardamos en una variable

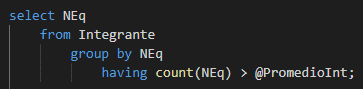




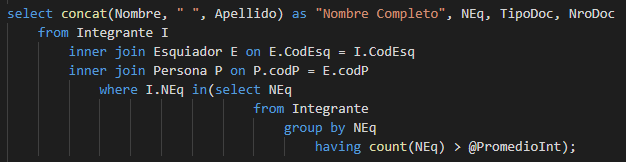
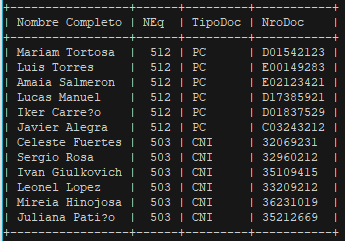


Ahora buscamos los equipos que superen ese promedio





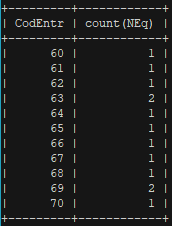
Y por último volcamos la información pertinente de los esquiadores pertenecientes a esos equipos

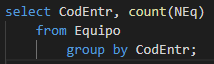


***Consulta 4***

*Mostrar a cada entrenador junto a la cantidad de equipos que entrena en las olimpiadas.*

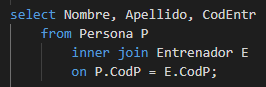
Primero vamos a la tabla de equipo y como los entrenadores ya están vinculados directamente a los equipos en esta tabla, solamente hay que hacer un count desde la tabla Equipo.

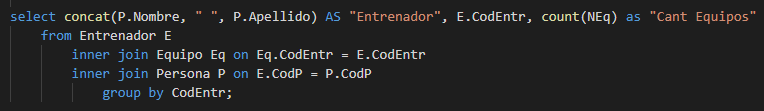


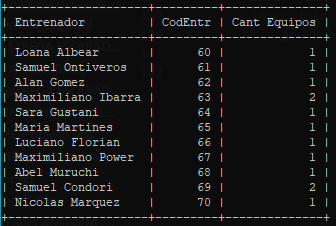


Ya que tenemos los códigos de los entrenadores y a cuantos equipos entrena cada uno, necesitamos los datos personales del entrenador, por lo tanto, recurrimos a Entrenador y a Persona:





Para finalizar, simplemente juntamos esas consultas en una sola y vinculamos los resultados:

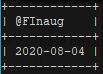
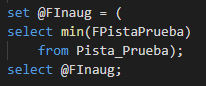


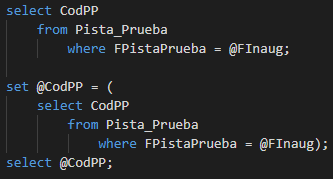
***Consulta 5***

*Listar nombre y federación de todos los participantes (individual o en equipo) que compitieron en la inauguración de las olimpiadas.*

Analizando la base de datos, podemos deducir que la fecha de inauguración de las olimpiadas aparece tanto en Prueba como en Pista\_Prueba, pero a nosotros nos sirve directamente apuntar a Pista\_Prueba para saber quienes son los que corrieron el primer día en esa pista, ya que si utilizamos solo la prueba nos daría los resultados de los que corrieron en toda la jornada de la misma.

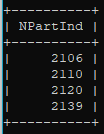
Entonces busquemos la menor fecha y el identificador de le pista que se corrió ese mismo día

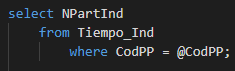


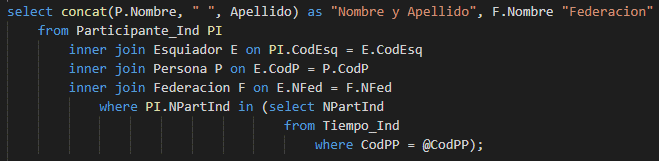


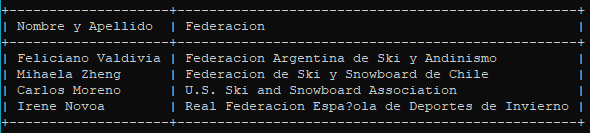


Ahora solo hay que ver quienes son los que corrieron allí, y puesto que en cada pista (por prueba) se corre en un solo día, solo necesitamos los tiempos (prueba de que corrieron) de esa pista que los obtendremos desde Tiempo\_Ind y Tiempo\_Int, pero como esta Pista en esa fecha es solamente utilizada por los participantes individuales, no es necesario buscar los tiempos de los integrantes de equipos (de todas formas si lo hiciéramos simplemente seguimos el mismo procedimiento con distintos nombres/tablas y al final juntamos ambos resultados con union).

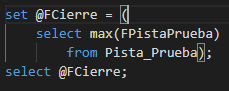
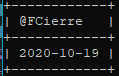


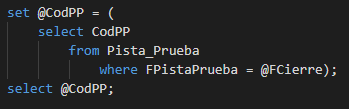


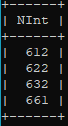
Ahora ya con la identificación de los participantes, simplemente debemos vincular a las demás tablas y mostrar los datos pertinentes

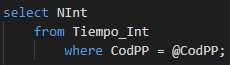


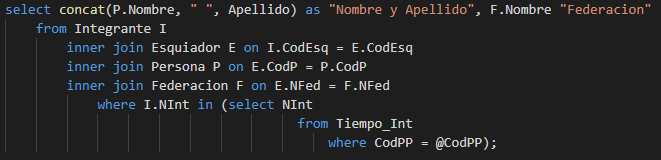
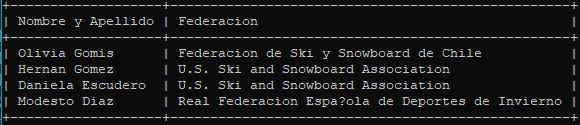
Como dijimos previamente, ningún integrante de equipo participó en esa prueba ni mucho menos en esa pista el primer día, así que acá terminaría la resolución de la consulta. Sin embargo, como extra, podemos realizar el mismo procedimiento para los integrantes de equipos que hayan participado en el cierre (último día) de la competencia (por dar un ejemplo con equipos).









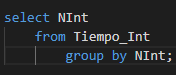


***Consulta 6***

*Identificar los esquiadores que al final de la competencia no participaron en ninguna prueba y formaban parte de un equipo, junto al nombre del equipo; ordenado por equipo.*

Para identificar a los esquiadores que no participaron y estaban en un equipo, primero debemos buscar los que sí participaron, y luego usar ese resultado como filtro para los integrantes.

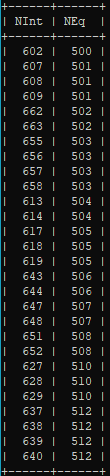
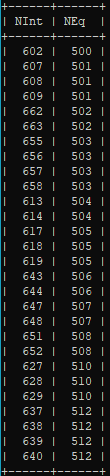
Así que busquemos a todos los integrantes que tengan tiempos registrados (compitieron).

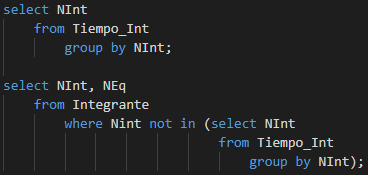


*(es bastante largo por eso hay varias columnas)*



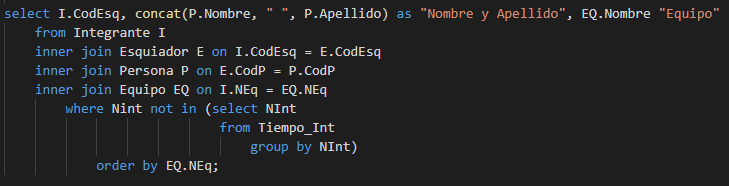
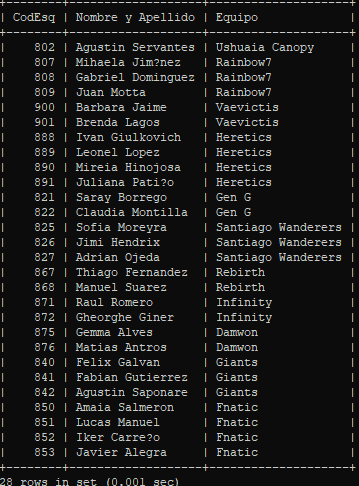
Ahora averigüemos todos los integrantes que NO aparezcan en esa lista, utilizándola como filtro







Ahora ya teniendo identificados a los esquiadores junto con el Número de equipo en el que están, con un par de inner joins y un ordenar por NEquipo obtendremos el resultado esperado.

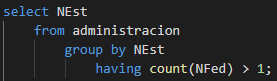




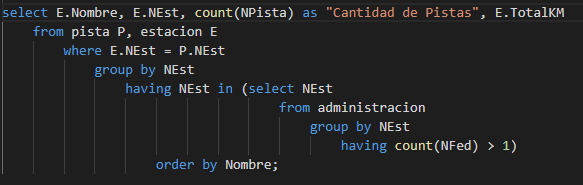
***Consulta 7***

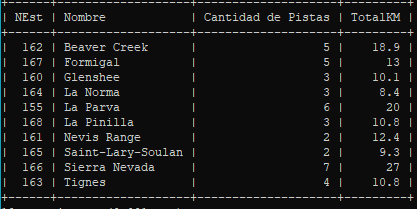
*Identificar a las estaciones de esquí que sean administradas por más de una federación, indicando nombre, km esquiables y cantidad de pistas, ordenadas alfabéticamente.*

Para encontrar las estaciones deseadas, hay que filtrar las que sean administradas por una sola federación y usar las restantes.



Ahora teniendo identificadas a las estaciones que necesitamos, hace falta proyectar los demas datos pertinentes, vinculandolos en base a los Números de Estación de la consulta anterior

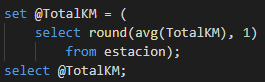


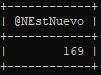
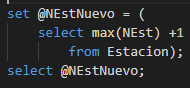


***Consulta 8 – Situación Lógica***

*Dada la cantidad de participantes en las olimpiadas se decidió habilitar una nueva estación de esquí llamada “Sur/Norte”. Para administrar se pensó en la federación que tiene asignada una sola estación junto a la federación que tiene la mayor cantidad de estaciones asignadas. Para determinar cuántos km esquiables tendrá se pensó en tomar el promedio de km totales que tiene toda la competencia. Las pistas aún no serán asignadas porque deben evaluar si hay algún recorte de las ya existentes.*

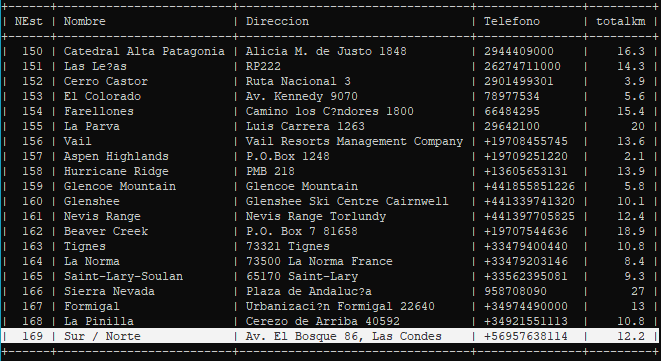
Primero averigüaremos el promedio de km esquiables de todas las estaciones (que coinciden con los de todas las pistas) para luego poner los datos en la estación mediante una variable. Con otra variable, necesitaremos saber cual es el identificador de la última estación registrada y sumarle uno.



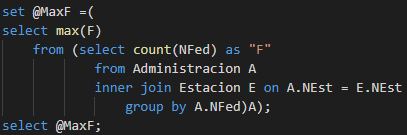


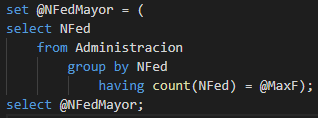
Ahora creamos la estación mediante un insert (los demás datos son inventados)

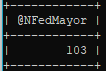




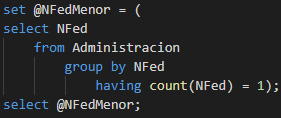
Ahora para determinar quienes estarán a cargo de esa estación, hay que buscar la federación que administre una sola y la que administre más. Primero buscamos el máximo, y luego que federación tiene esa cantidad.





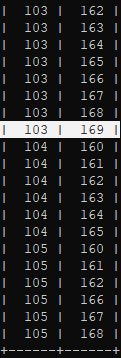
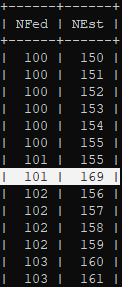


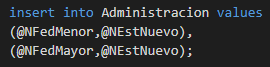
Y ahora la federación que administre una sola estación





Ahora que tenemos todos los datos, simplemente hay que cargarlos a la tabla Administración

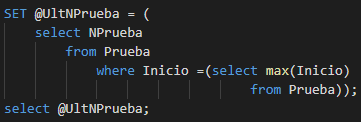


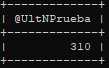


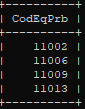
***Situación Lógica A***

*Debido a los decepcionantes resultados de la ultima prueba, los administradores del equipo que quedó en ultimo puesto han decidido cambiar su entrenador, y contratar al entrenador del equipo que resultó ganador, y de esta manera en un futuro conseguir mejores resultados*

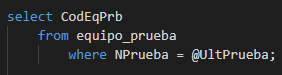
Primero buscaremos cual es la última prueba, teniendo en cuenta que sea la que comenzó más tarde



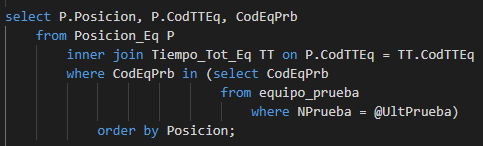


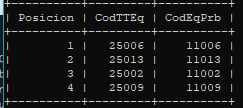


Luego buscamos los equipos que participaron en esa prueba

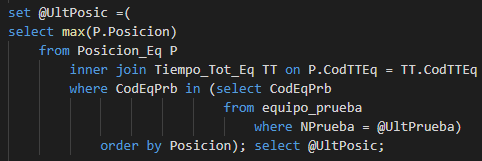


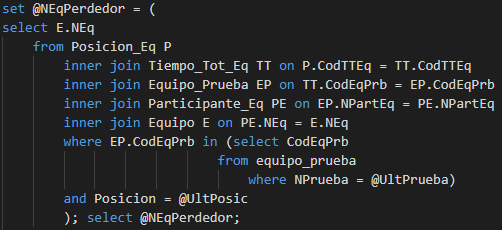
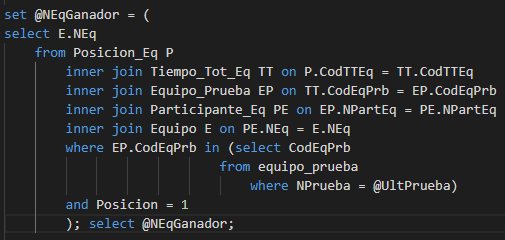
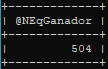
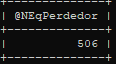
Con los equipos identificados, ahora hace falta ver quien es el ganador y quien es el perdedor que despide a su entrenador, por eso ahora mostraremos las posiciones

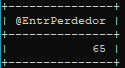
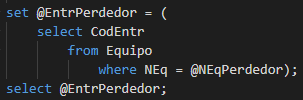
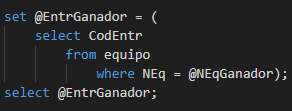
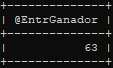




Una vez encontrados, solo tenemos que buscar cual es la última posición y con eso sabremos que equipo perdió (el que ganó ya se sabe que quedó en 1ra posición)

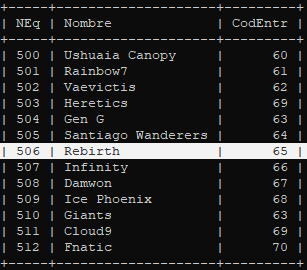
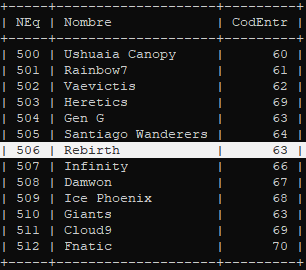


Ahora hay que identificar a esos equipos

Una vez que identificamos a los equipos, solo nos queda averiguar los entrenadores y luego hacer el update correspondiente

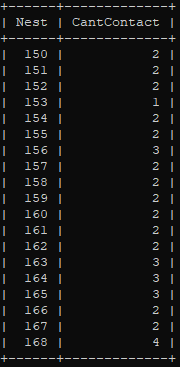
Ahora con todos los datos hacemos el update de los equipos, y listo





***Situación lógica B***

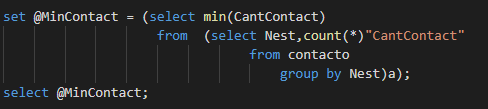
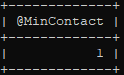
*Debido que hay entrenadores que por un largo tiempo no lograron estar en un equipo, han estado en una situación economica bastante inestable y se les presentó la oportunidad de trabajar como contactos, laburando en la estación con la menor cantidad de contactos.*



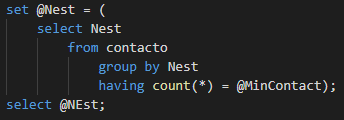
Primero hay que ver cuantos contactos hay en cada estación



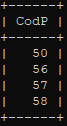
Y luego buscamos el mínimo

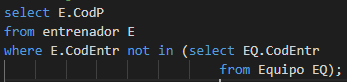


Ahora identificamos la Estación



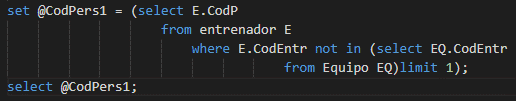
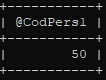


Posteriormente debemos conseguir a los entrenadores que NO tienen equipo, así que utilizaremos como filtro los que ya están en un equipo.

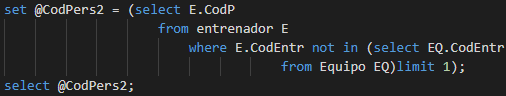
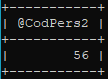


En un principio utilizaríamos el operador IF para procesar esos cuatro códigos e insertarlos directamente, pero como el InnoDB no lo soporta, debemos hacerlo uno por uno…

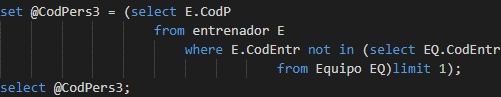
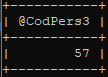
Para cada uno guardaremos el CodP en una variable y luego lo borramos de entrenador



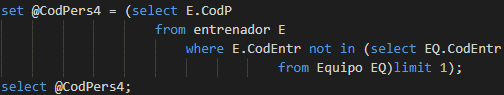


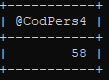














Y ahora solo queda insertarlos, como la tabla de contactos ya tiene su auto\_increment, no hace falta saber quién es el último contacto, se suma uno automáticamente.

