



**DEPARTAMENTO
DE COMPUTACION**

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico N° 1

7 de Mayo de 2017

Bases de Datos

Integrante	LU	Correo electrónico
Diego Gaustein	586/09	diego@gaustein.com.ar
Agustín Cangiani	344/09	cangiani@gmail.com
Francisco Muchinik	238/14	franmuchinik@hotmail.com
Yamil Alis	742/00	yamil.alis@gmail.com



Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Ciudad Universitaria - Pabellón I

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Argentina

Tel/Fax: (54 11) 4576-3359

<http://exactas.uba.ar>

Índice

1. Introducción	3
2. Modelo de Entidad Relación	4
2.1. Restricciones Adicionales	4
2.2. Enums	4
3. Modelo Relacional	5
3.1. Esquema resultante	5
3.2. Supuestos asumidos	9
4. Consultas	10
5. Conclusiones	14

1. Introducción

- Hablar de que elegimos sqlite
- Hablar de que en sqlite no hay enums, pero se los puede simular como constraints Link <http://stackoverflow.com/to-create-enum-type-in-sqlite>

2. Modelo de Entidad Relación

2.1. Restricciones Adicionales

1. Un participante sólo puede coachear alumnos de su misma escuela, a lo sumo 5, y no puede coachearse a sí mismo.
2. Un alumno sólo puede estar inscripto en una modalidad si tiene la graduación adecuada, y si aplicase, el género y el rango de edad correctos.
3. Los titulares y suplentes de un equipo deben pertenecer todos a la misma escuela.
4. Un árbitro sólo puede ser presidente, central, juez o suplente de una modalidad si su graduación es mayor o igual a la graduación de la modalidad (si esta es individual, ya que no aplica para las que son por equipo)
5. La cantidad de suplentes para una modalidad debe ser mayor o igual a 3.
6. Un árbitro no puede cumplir más de un rol por modalidad. Por ejemplo, si es presidente no puede ser central, juez o suplente.
7. Un alumno o equipo sólo puede ganar una modalidad si participó en ella.

2.2. Enums

1. Rango de edad (Juveniles o adultos)
2. Sexo (M o F)
3. Graduación (1er Dan a 6to Dan)
4. Puesto (Primero, segundo o tercero)

3. Modelo Relacional

Basándonos en lo desarrollado del Modelo obtenemos el siguiente Modelo Relacional determinamos que hay que crear las siguientes relaciones:

- Maestro
- Escuela
- Participante
- Alumno
- Ring
- Árbitro
- Es_juez
- Es_suplente
- Modalidad
- ModalidadIndividual
- Combate
- Formas
- Salto
- Equipo
- Ganó equipos
- Participa
- Ganó

de Entidad Relación

3.1. Esquema resultante

Maestro(Nro placa, nombre, graduación, país)

PK = CK = {Nro placa}

Restricciones adicionales:

- Maestro.Nro placa debe estar en Escuela.Nro_placa.

Escuela(idEscuela), Nombre, Nro placa)

PK = CK = {idEscuela}

FK = {Nro placa}

Restricciones adicionales:

- Escuela.Nro placa debe estar en Escuela.Nro_placa.
- Escuela.idEscuela puede no estar en Participante.idEscuela.

Participante(Nro cert grad), idEscuela, nombre, foto, graduación)

PK = CK = {Nro cert grad}

FK = {idEscuela}

Restricciones adicionales:

- Participante.idEscuela debe estar en Escuela.idEscuela.
- Participante.Nro_cert_grad puede no estar en Alumno.Nro_cert_grad.
- Participante.Nro_cert_grad puede no estar en Alumno.coach.

Alumno(Nro cert grad, fecha de nacimiento, dni, sexo, peso, coach)

PK = {Nro cert grad}

CK = {Nro cert grad, dni}

FK = {Nro cert grad, coach}

Restricciones adicionales:

- Alumno.Nro_cert_grad debe estar en Participante.Nro_cert_grad.
- Alumno.coach debe pertenecer a Participante.Nro_cert_grad.
- Alumno.Nro_cert_grad puede no estar en Equipo.titular1, ... Equipo.titular5.
- Alumno.Nro_cert_grad puede no estar en Equipo.suplente1, ... Equipo.suplente3.

Ring(idRing)

PK = CK = {idRing}

Restricciones adicionales:

- Ring.idRing debe estar en Modalidad.idRing.

Árbitro(Nro Placa, nombre país, graduación)

PK = CK = {Nro Placa}

Restricciones adicionales:

- Arbitro.nro_placa puede no estar en Modalidad.nro_placa_presidente.
- Arbitro.nro_placa puede no estar en Modalidad.nro_placa_central.
- Arbitro.nro_placa puede no estar en Es_juez.nro_placa.
- Arbitro.nro_placa puede no estar en Es_suplente.nro_placa.

Es_juez(Nro placa, idModalidad)

PK = CK = {(Nro Placa, idModalidad)}

FK = {Nro Placa, idModalidad}

Restricciones adicionales:

- Es_juez.idModalidad debe estar en Modalidad.idModalidad.
- Es_juez.nro_placa debe estar en Arbitro.nro_placa.

Es_suplente(Nro placa, idModalidad)

PK = CK = {(Nro Placa, idModalidad)}

FK = {Nro Placa, idModalidad}

Restricciones adicionales:

- Es_suplente.idModalidad debe estar en Modalidad.idModalidad.
- Es_suplente.nro_placa debe estar en Arbitro.nro_placa.

Modalidad(idModalidad, sexo, Nro placa presidente, Nro placa central, idRing, individual)

PK = CK = {idModalidad}

FK = {Nro placa presidente, Nro placa central, idRing}

Restricciones adicionales:

- Modalidad.idRing debe estar en Ring.idRing.
- Modalidad.nro_placa_presidente debe estar en Arbitro.nro_placa.
- Modalidad.nro_placa_central debe estar en Arbitro.nro_placa.
- Si hay un participante entonces Modalidad.idModalidad debe estar en ModalidadIndividual.idModalidad.
- Modalidad.idModalidad puede no estar en Gano_equipos.idModalidad.

ModalidadIndividual(idModalidad, graduación, tipo)

PK = CK = FK = {idModalidad}

Restricciones adicionales:

- ModalidadIndividual.idModalidad debe estar el Modalidad.idModalidad.
- ModalidadIndividual.idModalidad debe estar en Combate.idModadalidad, Formas.idModalidad o Salto.idModalidad si tipo no es Rotura_de_potencia, de forma excluyente.
- ModalidadIndividual.idModalidad debe estar en Participa.idModalidad.
- ModalidadIndividual.idModalidad puede no estar en Geno.idModalidad.

Combate(idModalidad, peso, rango de edad)

PK = CK = FK = {idModalidad}

Restricciones adicionales:

- Combate.idModalidad debe estar en ModalidadIndividual.idModalidad.

Formas(idModalidad, rango de edad)

PK = CK = FK = {idModalidad}

Restricciones adicionales:

- Formas.idModalidad debe estar en ModalidadIndividual.idModalidad.

Salto(idModalidad, rango de edad)

PK = CK = FK = {idModalidad}

Restricciones adicionales:

- Salto.idModalidad debe estar en ModalidadIndividual.idModalidad.

Equipo(idEquipo, nombre, titular1, ... titular5, suplente1, ... suplente3, idModalidad)

PK = CK = {idEquipo}

FK = {titular1, ... titular5, suplente1, ... suplente3, idModalidad}

Restricciones adicionales:

- Equipo.idModalidad debe estar en Modadalidad.idModalidad y la modalidad referida debe tener el atributo individual en false.
- Equipo.idEquipo puede no estar en Gano_equipos.idEquipo.
- Equipo.titular1, ... Equipo.titular5 deben estar en Alumno.Nro_cert_grad.

- Equipo.suplente1, ... Equipo.suplente3 deben estar en Alumno.Nro_cert_grad.

Ganó equipos(idModalidad, idEquipo, Puesto)

PK = CK = {(idModalidad, idEquipo, Puesto)}

FK = {idModalidad, idEquipo}

Restricciones adicionales:

- Gano_equipos.idModalidad debe estar en Modalidad.idModalidad.
- Gano_equipos.idEquipo debe estar en Equipo.idEquipo.

Participa(Nro_cert_grad, idModalidad)

PK = CK = {(idModalidad, Nro_cert_grad)}

FK = {idModalidad, Nro cert grad}

Restricciones adicionales:

- Participa.idModalidad debe estar en ModalidadIndividual.idModalidad.
- Participa.Nro_cert_grad debe estar en Alumno.Nro_cert_grad.

Ganó(idModalidad, Nro_cert_grad, Puesto)

PK = CK = {(idModalidad, Nro_cert_grad, Puesto)}

FK = {idModalidad, Nro_cert_grad}

Restricciones adicionales:

- Gano.idModalidad debe estar en ModalidadIndividual.idModalidad.
- Gano.Nro_cert_grad debe estar en Alumno.Nro_cert_grad.

3.2. Supuestos asumidos

4. Consultas

1. El listado de inscriptos en cada categoría para el armado de llaves:

```
CREATE VIEW modalidades_individuales_por_nombre AS
  SELECT c.idModalidad, "Combate_" || c.peso || "_" || c.rango_de_edad
    || "_" || mod.i.graduacion || "_" || mod.sexo as Nombre
FROM combate c
INNER JOIN modalidadindividual mod_i ON c.idModalidad = mod_i.
    idModalidad
INNER JOIN modalidad mod ON mod.idModalidad = mod_i.idModalidad

UNION

  SELECT f.idModalidad, "Formas_" || f.rango_de_edad || "_" ||
    mod.i.graduacion || "_" || mod.sexo as Nombre
FROM formas f
INNER JOIN modalidadindividual mod_i ON f.idModalidad = mod_i.
    idModalidad
INNER JOIN modalidad mod ON mod.idModalidad = mod_i.idModalidad

UNION

  SELECT s.idModalidad, "Salto_" || s.rango_de_edad || "_" || mod.i.
    graduacion || "_" || mod.sexo as Nombre
FROM salto s
INNER JOIN modalidadindividual mod_i ON s.idModalidad = mod_i.
    idModalidad
INNER JOIN modalidad mod ON mod.idModalidad = mod_i.idModalidad

UNION

  SELECT mod_i.idModalidad, "Rotura_de_potencia_" || mod_i.graduacion
    || "_" || mod.sexo as Nombre
FROM modalidadindividual mod_i
INNER JOIN modalidad mod ON mod.idModalidad = mod_i.idModalidad
  WHERE tipo = "rotura_de_potencia"

SELECT a.nro_cert_grad, mod.nombre
FROM alumno a
INNER JOIN participa p ON a.nro_cert_grad = p.nro_cert_grad
INNER JOIN modalidades_individuales_por_nombre mod ON p.idModalidad = mod.
    idModalidad

UNION

  SELECT e.nombre, 'Combate_por_equipos_' || mod.sexo as Nombre
FROM equipo e
INNER JOIN modalidad mod ON mod.idModalidad = e.idModalidad
```

2. El país que obtuvo mayor cantidad de medallas de oro, plata y bronce:

```
SELECT m.pais
FROM (
  SELECT m.pais, gano.puesto
FROM gano
INNER JOIN alumno a ON a.nro_cert_grad = gano.nro_cert_grad
```

```

INNER JOIN participante p ON p.nro_cert_grad = a.nro_cert_grad
INNER JOIN escuela e ON e.idEscuela = p.idEscuela
INNER JOIN maestro m ON m.nro_placa = e.nro_placa
UNION ALL
SELECT m.pais , ganoequipos.puesto
FROM ganoequipos
INNER JOIN equipo eq ON eq.idEquipo = ganoequipos.idEquipo
INNER JOIN alumno a ON a.nro_cert_grad = eq.titular1
INNER JOIN participante p ON p.nro_cert_grad = a.nro_cert_grad
INNER JOIN escuela e ON e.idEscuela = p.idEscuela
INNER JOIN maestro m ON m.nro_placa = e.nro_placa) as m
GROUP BY m.pais
ORDER BY COUNT(*) DESC
LIMIT 1

```

3. Sabiendo que las medallas de oro suman 3 puntos, las de plata 2 y las de bronce 1 punto, se quiere realizar un ranking de puntaje por país y otro por escuela:

```

SELECT idEscuela , sum(puntos)
FROM
    (SELECT p.idEscuela idEscuela , SUM(CASE
        WHEN g.puesto = 1 THEN 3
        WHEN g.puesto = 2 THEN 2
        ELSE 1
    END) puntos
    FROM gano g, participante p
    WHERE p.nro_cert_grad = g.nro_cert_grad
    GROUP BY idEscuela
    UNION
    SELECT p.idEscuela idEscuela , SUM(CASE
        WHEN g.puesto = 1 THEN 3
        WHEN g.puesto = 2 THEN 2
        ELSE 1
    END) puntos
    FROM ganoequipos g, equipo eq, participante p
    WHERE g.idEquipo = eq.idEquipo
    AND eq.titular1 = p.nro_cert_grad
    GROUP BY idEscuela)
GROUP BY idEscuela
ORDER BY puntos DESC;

```

```

/* por pais */
SELECT pais , SUM(puntos) puntos
FROM
    (SELECT m.pais pais , SUM(CASE
        WHEN g.puesto = 1 THEN 3
        WHEN g.puesto = 2 THEN 2
        ELSE 1
    END) puntos
    FROM gano g, participante p,
        escuela es, maestro m
    WHERE p.nro_cert_grad = g.nro_cert_grad
    AND p.idEscuela = es.idEscuela
    AND es.nro_placa = m.nro_placa
    GROUP BY m.pais
    UNION

```

```

SELECT m.pais pais, SUM(CASE
    WHEN g.puesto = 1 THEN 3
    WHEN g.puesto = 2 THEN 2
    ELSE 1
END) puntos
FROM gano_equipos g, equipo eq, participante p, maestro m, escuela es
WHERE g.idEquipo = eq.idEquipo
AND eq.titular1 = p.nro_cert_grad
AND p.idEscuela = es.idEscuela
AND es.nro_placa = m.nro_placa
GROUP BY m.pais)
GROUP BY pais
ORDER BY puntos DESC;

```

4. Dado un competidor, la lista de categorías donde haya participado y el resultado obtenido:

```

SELECT p.idmodalidad, g.puesto
FROM participa p
LEFT JOIN gano g
    ON g.nro_cert_grad = p.nro_cert_grad
    AND g.idmodalidad = p.idmodalidad
WHERE p.nro_cert_grad = '19856G';

UNION

SELECT eq.idmodalidad, g.puesto
FROM equipo eq
LEFT JOIN gano_equipos g
    ON g.idEquipo = eq.idEquipo
    AND g.idmodalidad = eq.idmodalidad
WHERE eq.titular1 = '19856G' OR eq.titular2 = '19856G' OR eq.titular3 = '
19856G' OR eq.titular4 = '19856G' OR eq.titular5 = '19856G' OR eq.
suplente1 = '19856G' OR eq.suplente2 = '19856G' OR eq.suplente3 = '
19856G';

```

5. El medallero por escuela:

```

SELECT p.idEscuela, g.idModalidad, g.puesto
FROM gano g, participante p, alumno a
WHERE a.nro_cert_grad = g.nro_cert_grad
AND a.coach = p.nro_cert_grad

UNION

SELECT p.idEscuela, g.idModalidad, g.puesto
FROM gano_equipos g, equipo eq, participante p, alumno a
WHERE g.idEquipo = eq.idEquipo
AND eq.titular1 = a.nro_cert_grad
AND a.coach = p.nro_cert_grad

```

6. El listado de los árbitros por país:

```

SELECT nombre, pais
FROM arbitro
ORDER BY pais

```

7. La lista de todos los árbitros que actuaron como árbitro central en las modalidades de combate:

```

SELECT a.nro_placa , a.nombre
FROM combate c, modalidad m, arbitro a
WHERE c.idModalidad = m.idModalidad
AND m.nro_placa_central = a.nro_placa

```

UNION

```

SELECT a.nro_placa , a.nombre
FROM modalidad m, arbitro a
WHERE m.participantes > 1
AND m.nro_placa_central = a.nro_placa

```

8. La lista de equipos por país:

```

SELECT m.pais , eq.idEquipo
FROM equipo eq, alumno a, participante p, escuela es, maestro m
WHERE eq.titular1 = a.nro_cert_grad
AND a.coach = p.nro_cert_grad
AND p.idEscuela = es.idEscuela
AND es.nro_placa = m.nro_placa
ORDER BY m.pais

```

5. Conclusiones

Idea 1: Pregunta: no me molesta eso, solo tener que hacer 5 joins para sacar los datos de las modalidades

Respuesta: es la desventaja del modelado de esa jerarquía