

Universidad de Buenos Aires
Facultad de
Ciencias Exactas y Naturales
Departamento de Computación

Base de Datos

Segundo Cuatrimestre de 2012

Trabajo práctico 1

Campeonato Sudamericano de Básquet.

Grupo 2

Integrante	LU	Correo electrónico
Cammi, Martín	676/02	martincammi@gmail.com
De Sousa, Mariano	389/08	marian_sabianaa@hotmail.com
Méndez, Gonzálo	843/04	gemm83@hotmail.com
Serapio, Noelia	871/03	noeliaserapio@gmail.com

Índice

1. Tecnologías utilizadas	3
2. Modelo Entidad-Relacion	4
3. Modelo Logico Relacional	4
3.1. Entidad Selección	4
3.2. Entidad Posición	4
3.3. Entidad LugarHospedaje	5
3.4. Entidad País	5
3.5. Entidad Integrante	6
3.6. Entidad Jugador	6
3.7. Entidad CuerpoTécnico	7
3.8. Entidad Función	7
3.9. Entidad Equipo	7
3.10. Entidad Partido	8
3.11. Entidad Estadio	9
3.12. Entidad Etapa	9
3.13. Entidad Tanteador	9
3.14. Entidad Arbitro	10
3.15. Entidad Arbitra	11
3.16. Entidad Participación	11
3.17. Entidad Sanción	12
3.18. Entidad TipoSanción	12
4. Supuestos asumidos	13
5. Diseño físico	14
6. Restricciones adicionales al modelo	15
7. Breakers	16
8. Triggers y Store Procedures de restricciones	17
9. Datos de prueba	18

1. Tecnologías utilizadas

Hemos utilizado para la confección del trabajo práctico el motor de base de datos Mysql

2. Modelo Entidad-Relacion

3. Modelo Logico Relacional

3.1. Entidad Selección

SELECCION(idSeleccion, hospedaHospedaje, representaPais, concentraEstadio, ubicaPosicion, cantIntegrantes, fechaArribo, grupo)

PK = { (idSeleccion) }

CC = { (idSeleccion) }

FK = { (hospedaHospedaje), (representaPais), (concentraEstadio), (ubicaPosicion) }

References:

SELECCION.hospedaHospedaje debe estar en LUGARHOSPEDAJE.idHospedaje.

SELECCION.representaPais debe estar en PAIS.idPais.

SELECCION.concentraEstadio debe estar en ESTADIO.idEstadio.

SELECCION.ubicaPosicion debe estar en POSICION.idPosicion.

SELECCION.idSeleccion debe estar en INTEGRANTE.perteneceSeleccion.

SELECCION.idSeleccion debe estar en PARTIDO.equipoSeleccion1.

SELECCION.idSeleccion debe estar en PARTIDO.equipoSeleccion2.

SELECCION.hospedaHospedaje no puede ser nulo.

SELECCION.representaPais no puede ser nulo.

SELECCION.concentraEstadio no puede ser nulo.

SELECCION.ubicaPosicion no puede ser nulo.

Constraints:

SELECCION.grupo == "A" no puede repetirse más de 4 veces.

SELECCION.grupo == "B" no puede repetirse más de 4 veces.

SELECCION.grupo solo puede ser "A" o "B".

SELECCION.fechaArribo <= PARTIDO.fecha.

SELECCION.cantIntegrantes debe ser igual a la cantidad de integrantes relacionados.

3.2. Entidad Posición

POSICION(idPosicion, puntos, partidosJugados, partidosGanados, partidosPerdidos, tantosAFavor, tantosEnContra)

PK = { (idPosicion) }

CC = { (idPosicion) }

FK = {}

References:

POSICION.idPosicion debe estar en SELECCION.ubicaPosicion.

Constraints:

POSICION.puntos ≥ 0 .

POSICION.partidosJugados ≥ 0 .

POSICION.partidosGanados ≥ 0 .

POSICION.partidosPerdidos ≥ 0 .

POSICION.partidosJugados == POSICION.partidosGanados + POSICION.partidosPerdidos.

POSICION.tantosAFavor ≥ 0 .

POSICION.tantosEnContra ≥ 0 .

3.3. Entidad LugarHospedaje

LUGARHOSPEDAJE(idHospedaje, nombreHospedaje)

PK = { (idHospedaje) }

CC = { (idHospedaje) }

FK = {}

References:

LUGARHOSPEDAJE.idHospedaje debe estar en SELECCION.hospedaHospedaje.

Constraints:

Ninguna.

3.4. Entidad País

PAIS(idPais, nombrePais)

PK = { (idPais) }

CC = { (idPais) }

FK = {}

References:

PAIS.idPais puede no estar en SELECCION.representaPais.

PAIS.idPais puede no estar en ARBITRO.pertenecePais.

Constraints:

Ninguna.

3.5. Entidad Integrante

INTEGRANTE(idIntegrante, perteneceSeleccion, nroPasaporte, nombreIntegrante, apellido, fechaNacimiento, tipoIntegrante)

PK = { (idIntegrante) }

CC = { (idIntegrante), (nroPasaporte) }

FK = { (perteneceSeleccion) }

References:

INTEGRANTE.perteneceSeleccion debe estar en SELECCION.idSeleccion.

INTEGRANTE.idIntegrante puede estar en JUGADOR.idJugador o (exclusivo) CUERPO-TECNICO.idCuerpoTecnico.

INTEGRANTE.perteneceSeleccion no puede ser nulo.

Constraints:

INTEGRANTE.tipoIntegrante IN { "Jugador", "CuerpoTecnico" } .

AÑO(SYSDATE) – AÑO(INTEGRANTE.fechaNacimiento) >= 18.

3.6. Entidad Jugador

JUGADOR(idJugador, estaEnEquipo)

PK = { (idJugador) }

CC = { (idJugador) }

FK = { (estaEnEquipo), (idJugador) }

References:

JUGADOR.idJugador debe estar en INTEGRANTE.idIntegrante.

JUGADOR.estaEnEquipo debe estar en EQUIPO.idEquipo.

JUGADOR.idJugador debe estar en PARTICIPACION.participaJugador.

JUGADOR.estaEnEquipo no puede ser nulo.

Constraints:

Por jugador, tiene que haber una sola participación en un partido.

3.7. Entidad CuerpoTécnico

CUERPOTECNICO(idCuerpoTecnico, cumpleFuncion)

PK = { (idCuerpoTecnico) }

CC = { (idCuerpoTecnico) }

FK = { (cumpleFuncion), (idCuerpoTecnico) }

References:

CUERPOTECNICO.idCuerpoTecnico debe estar en INTEGRANTE.idIntegrante.

CUERPOTECNICO.cumpleFuncion debe estar en FUNCION.idFuncion.

CUERPOTECNICO.cumpleFuncion no puede ser nulo.

Constraints:

Ninguna.

3.8. Entidad Función

FUNCION(idFuncion, nombreFuncion)

PK = { (idFuncion) }

CC = { (idFuncion) }

FK = { }

References:

FUNCION.idFuncion puede no estar en CUERPOTECNICO.cumpleFuncion.

Constraints:

Ninguna.

3.9. Entidad Equipo

EQUIPO(idEquipo, nombreEquipo)

PK = { (idEquipo) }

CC = { (idEquipo) }

FK = { }

References:

EQUIPO.idEquipo debe estar en JUGADOR.estaEnEquipo.

Constraints:

Ninguna

3.10. Entidad Partido

PARTIDO(idPartido, juegaEnEtapa, equipoSeleccion1, equipoSeleccion2, juegaEnEstadio, duracion, fecha, horario)

PK = { (idPartido) }

CC = { (idPartido) }

FK = { (juegaEnEtapa), (equipoSeleccion1), (equipoSeleccion2), (juegaEnEstadio) }

References:

PARTIDO.juegaEnEtapa debe estar en ETAPA.idEtapa.

PARTIDO.equipoSeleccion1 debe estar en SELECCION.idSeleccion.

PARTIDO.equipoSeleccion2 debe estar en SELECCION.idSeleccion.

PARTIDO.juegaEnEstadio debe estar en ESTADIO.idEstadio.

PARTIDO.idPartido debe estar en ARBITRA.idPartidoArb.

PARTIDO.idPartido puede no estar en PARTICIPACION.jugoPartido.

PARTIDO.idPartido debe estar en TANTEADOR.idPartido.

PARTIDO.juegaEnEtapa no puede ser nulo.

PARTIDO.equipoSeleccion1 no puede ser nulo.

PARTIDO.equipoSeleccion2 no puede ser nulo.

PARTIDO.juegaEnEstado no puede ser nulo.

Constraints:

PARTIDO.equipoSeleccion1 <> PARTIDO.equipoSeleccion2.

Si PARTIDO.juegaEnEtapa == "FASE_GRUPOS" \Rightarrow PARTIDO.equipoSeleccion1.grupo == PARTIDO.equipoSeleccion2.grupo.

Si PARTIDO.juegaEnEtapa == "FASE_GRUPOS" \Rightarrow # PARTIDO \leq 6.

Si PARTIDO.juegaEnEtapa == "5TO_PUESTO" \Rightarrow # PARTIDO \leq 1.

Si PARTIDO.juegaEnEtapa == "3ER_PUESTO" \Rightarrow # PARTIDO \leq 1.

Si PARTIDO.juegaEnEtapa == "SEMIFINAL" \Rightarrow # PARTIDO \leq 2.

Si PARTIDO.juegaEnEtapa == "FINAL" \Rightarrow # PARTIDO \leq 1.

No puede haber dos partidos en un mismo horario.

Validar que al insertar un partido, esten todos los anteriores de fase.

Las fechas de los partidos tienen que estar ordenado por la etapa (FASE_GRUPO < 5TO_PUESTO < SEMIFINAL < 3ER_PUESTO < FINAL).

La duración de los partidos tiene que ser > 0.

La hora del partido tiene que estar entre 0 y 23.

Dos equipos no pueden enfrentarse en la misma etapa dos veces.

3.11. Entidad Estadio

ESTADIO(idEstadio, nombreEstadio)

PK = { (idEstadio) }

CC = { (idEstadio) }

FK = {}

References:

ESTADIO.idEstadio puede no estar en PARTIDO.juegaEnEstadio.

ESTADIO.idEstadio puede no estar en SELECCION.concentraEstadio.

Constraints:

Ninguna.

3.12. Entidad Etapa

ETAPA(idEtapa, nombreEtapa)

PK = { (idEtapa) }

CC = { (idEtapa) (nombreEtapa) }

FK = {}

References:

ETAPA.idEtapa puede no estar en PARTIDO.juegaEnEtapa.

Constraints:

ETAPA.nombreEtapa debe ser o bien FASE_GRUPOS o 5TO_PUESTO, o 3ER_PUESTO o SEMIFINAL, o FINAL.

3.13. Entidad Tanteador

TANTEADOR(nroCuarto, idPartido, scoreEquip1, scoreEquip2)

PK = { (nroCuarto, idPartido) }

CC = { (nroCuarto, idPartido) }

FK = { (idPartido) }

References:

TANTEADOR.idPartido debe estar en PARTIDO.idPartido.

TANTEADOR.idPartido no puede ser nulo.

Constraints:

Los valores posibles de TANTEADOR.nroCuatro son {1, 2, 3, 4}.

TANTEADOR.scoreEquip1 \geq 0.

TANTEADOR.scoreEquip2 \geq 0.

Si TANTEADOR.nroCuarto == 4 \Rightarrow TANTEADOR.scoreEquip1 \neq TANTEADOR.scoreEquip2 (no hay empates).

Constraints Adicionales:

TANTEADOR.nroCuarto no puede aparecer más de 4 veces por TANTEADOR.idPartido (el tanteador se genera con los 4 cuartos cuando se genera un partido).

3.14. Entidad Arbitro

ARBITRO(idArbitro, pertenecePais, nombreArbitro)

PK = { (idArbitro) }

CC = { (idArbitro) }

FK = (pertenecePais) }

References:

ARBITRO.pertenecePais debe estar en PAIS.idPais.

ARBITRO.idArbitro debe estar en ARBITRA.idArbitroArb.

ARBITRO.idArbitro puede no estar en SANCION.sancionadaPorArbitro.

ARBITRO.pertenecePais no puede ser nulo.

Constraints:

Ninguna.

Constraints Adicionales:

Un árbitro no puede dirigir un partido si:

- Dirigió a alguno de los equipos 2 o más veces, y en todos los partidos el equipo obtuvo el mismo resultado (ganó o perdió, no hay empate).
- Dirigió una sola vez a cada equipo con resultados opuestos.
- Está asignado a dirigir un partido en la misma fecha.

3.15. Entidad Arbitra

ARBITRA(idArbitroArb, idPartidoArb)

PK = { (idArbitroArb, idPartidoArb) }

CC = { (idArbitroArb, idPartidoArb) }

FK = { (idArbitroArb), (idPartidoArb) }

References:

ARBITRA.idArbitroArb debe estar en ARBITRO.idArbitro.

ARBITRA.idPartidoArb debe estar en PARTIDO.idPartido.

Constraints:

(ARBITRO.idArbitro == ARBITRA.idArbitroArb) and (ARBITRA.idPartidoArb == PARTIDO.idPartido) \Rightarrow (ARBITRO.pertenecePais \neq PARTIDO.seleccionEquipo1.representsPais and ARBITRO.pertenecePais \neq PARTIDO.seleccionEquipo2.representsPais).

3.16. Entidad Participación

PARTICIPACION(idParticipacion, jugoPartido, participaJugador, asistencias, rebotes, posicion, puntos, esTitular)

PK = { (idParticipacion) }

CC = { (idParticipacion) }

FK = { (jugoPartido), (participaJugador) }

References:

PARTICIPACION.jugoPartido debe estar en PARTIDO.idPartido.

PARTICIPACION.participaJugador debe estar en JUGADOR.idIntegrante.

PARTICIPACION.idParticipacion puede no estar en SANCION.aplicaParticipacion.

PARTICIPACION.posicion puede ser o bien BASE o ESCOLTA o ALERO o ALA-PIVOT o PIVOT o nulo.

PARTICIPACION.jugoPartido no puede ser nulo.

PARTICIPACION.participaJugador no puede ser nulo.

Constraints:

PARTICIPACION.jugoPartido.equipoSeleccion1 puede aparecer a lo sumo 5 veces con PARTICIPACION.esTitular == true.

PARTICIPACION.jugoPartido.equipoSeleccion2 puede aparecer a lo sumo 5 veces con PARTICIPACION.esTitular == true.

No puede haber dos PARTICIPACION.participaJugador tal que PARTICIPACION.esTitular == true y PARTICIPACION.participaJugador.IdIntegrante.perteneceSeleccion sean iguales y

PARTICIPACION.posicion sean iguales.

PARTICIPACION.rebotes ≥ 0 .

PARTICIPACION.asistencias ≥ 0 .

PARTICIPACION.puntos ≥ 0 .

Si PARTICIPACION.esTitular == false \Rightarrow PARTICIPACION.posicion is null.

3.17. Entidad Sanción

SANCION(idSancion, aplicaParticipacion, sancionadaPorArbitro, esDeTipo)

PK = { (idSancion) }

CC = { (idSancion) }

FK = { (aplicaParticipacion) , (sancionadaPorArbitro), (esDeTipo) }

References:

SANCION.aplicaParticipacion debe estar en PARTICIPACION.idParticipacion.

SANCION.sancionadaPorArbitro debe estar en ARBITRO.idArbitro.

SANCION.esDeTipo debe estar en TIPOSANCION.idTipoSancion.

SANCION.aplicaParticipacion no puede ser nulo.

SANCION.sancionadaPorArbitro no puede ser nulo.

SANCION.esDeTipo no puede ser nulo.

Constraints:

ARBITRA.idArbitroArb == SANCION.sancionadaPorArbitro and ARBITRA.idPartidoArb == SANCION.aplicaParticipacion.jugoPartido.

3.18. Entidad TipoSanción

TIPOSANCION(idTipoSancion, nombreSancion) PK = { (idTipoSancion) }

CC = { (idTipoSancion) }

FK = { }

References:

TIPOSANCION.idTipoSancion puede no estar en SANCION.esDeTipo.

Constraints:

Ninguna.

4. Supuestos asumidos

Asumimos que sólo vamos a guardar los hospedajes que nos indiquen de los jugadores, no otros.

Asumimos que sólo vamos a querer guardar los equipos en que jueguen los jugadores de la selección.

5. Diseño físico

Detalle grafico o escrito de las tablas de mysql

6. Restricciones adicionales al modelo

Las 1000 restricciones que pusimos.

7. Breakers

Los breakers son sentencias de inserts diseñadas específicamente para poner a prueba las restricciones del modelo. El archivo `breakers.sql` permite ejecutarlas y ver como todas ellas se verifican.

8. Triggers y Store Procedures de restricciones

9. Datos de prueba

Imagen de datos.sql y el fixture de los partidos con los puntajes

Todo el trabajo práctico cuenta con