



FACULTAD DE CIENCIAS
FUNDAMENTOS DE BASES DE DATOS

Proyecto Final

Semestre 2024 – 1

Alumno

Marco Silva Huerta

Profesor:

Víctor Manuel Corza Vargas

Ayudantes:

Alexis Hernández Castro

Diana Irma Canchola Hernández

Gibrán Aguilar Zuñiga

Oscar José Hernández Sánchez

13 de Noviembre de 2023

Índice

1. Entregables	2
1.1. Estructura de las carpetas	2
1.2. Estadios	2
1.3. Escuela	2
1.4. Tienda	2
2. Escuela	3
2.1. Lista de Requerimientos	3
2.2. Modelo conceptual	8
2.3. Modelo relacional	9
2.4. Script de creación	9
2.5. Script de Insert	10
2.6. Funcionamiento Restricciones	10
2.7. Funcionamiento Restricciones check	12
2.8. Creación de dominios personalizados	14
2.9. Restricciones para tuplas	15
2.10. Consultas	16
2.11. Vistas	17
3. Escuela	20
3.1. Lista de Requerimientos	20
3.2. Modelo conceptual	20
3.3. Modelo relacional	20
3.4. Script de creación	20
3.5. Script de Insert	20
3.6. Funcionamiento Restricciones	20
3.7. Funcionamiento Restricciones check	22
3.8. Creación de dominios personalizados	23
3.9. Restricciones para tuplas	24
3.10. Consultas	24
3.11. Vistas	25
4. Estadios	27
4.1. Lista de Requerimientos	27
4.2. Modelo conceptual	29
4.3. Modelo relacional	30
4.4. Script de creación	30
4.5. Script de Insert	31
4.6. Funcionamiento restricciones	32
4.7. Funcionamiento Restricciones check	34
4.8. Creación de dominios personalizados	36
4.9. Restricciones para tuplas	38
4.10. Consultas	39
4.11. Vistas	39

1. Entregables

1.1. Estructura de las carpetas

1.2. Estadios

- baseDeDatos/estadios/estadiosInsert.sql
- baseDeDatos/estadios/estadiosConexion.txt

1.3. Escuela

1.4. Tienda

2. Escuela

Saludos desde Hogwarts Escuela de Magia y Hechicería. Después de revisar exhaustivamente sus habilidades y experiencia en el mundo mágico de la programación, nos complace informarles que han sido seleccionados para crear y gestionar la base de datos mágica de nuestra venerable institución.

2.1. Lista de Requerimientos

Tablas para la base de datos

- Tabla Alumnos:
 - *AlumnoID* (Clave primaria)
 - *Nombre*
 - *Apellido*
 - *Casa*
 - *Fecha de Nacimiento*
 - *Género*
 - *CursoID* (Clave foránea referenciando a la tabla Cursos)
 - *UbicacionID* (Clave foránea referenciando a la tabla Ubicacion)
- Tabla Maestros:
 - *maestroID* (Clave primaria)
 - *Nombre*
 - *Apellido*
 - *Especialidad*
 - *Fecha de Nacimiento*
 - *Género*
 - *UbicacionID* (Clave foránea referenciando a la tabla Ubicacion)
- Tabla Ubicacion:
 - *UbicacionID* (Clave primaria)
 - *Localidad*
 - *Pais*
- Tabla Cursos:
 - *CursoID* (Clave primaria)
 - *Nombre del Curso*
 - *gradoAcademico*
 - *ProfesorJefeID* (Clave foránea referenciando a la tabla Maestros)
- Tabla Asignaturas:

- *AsignaturaID* (Clave primaria)
- *Nombre de la Asignatura*
- *maestroID* (Clave foránea referenciando a la tabla Maestros)
- *CursoID* (Clave foránea referenciando a la tabla Cursos)

Principales restricciones

- La clave primaria en cada tabla debe ser única y no nula.
- Las claves foráneas deben hacer referencia a claves primarias existentes en las tablas referenciadas.
- La columna de Fecha de Nacimiento en las tablas Alumnos y Maestros debe contener fechas válidas.
- La columna de Género en las tablas Alumnos y Maestros debe contener valores específicos: 'Masculino' o 'Femenino'.
- La columna de Casa en la tabla Alumnos debe contener valores específicos: 'Gryffindor', 'Hufflepuff', 'Ravenclaw' o 'Slytherin'.
- La columna de Año Académico en la tabla Cursos debe contener valores específicos: '1ro', '2do', '3ro', '4to', '5to', '6to'.
- La columna de Nombre y Apellido en las tablas Alumnos y Maestros debe contener texto no nulo y no vacío.
- La tabla Asignaturas debe tener una relación adecuada entre las claves foráneas y la información del profesor y el curso.
- La columna de Localidad en la tabla Ubicacion debe contener valores no nulos y no vacíos.
- Cada año escolar tiene un curso asociado

Información de la escuela

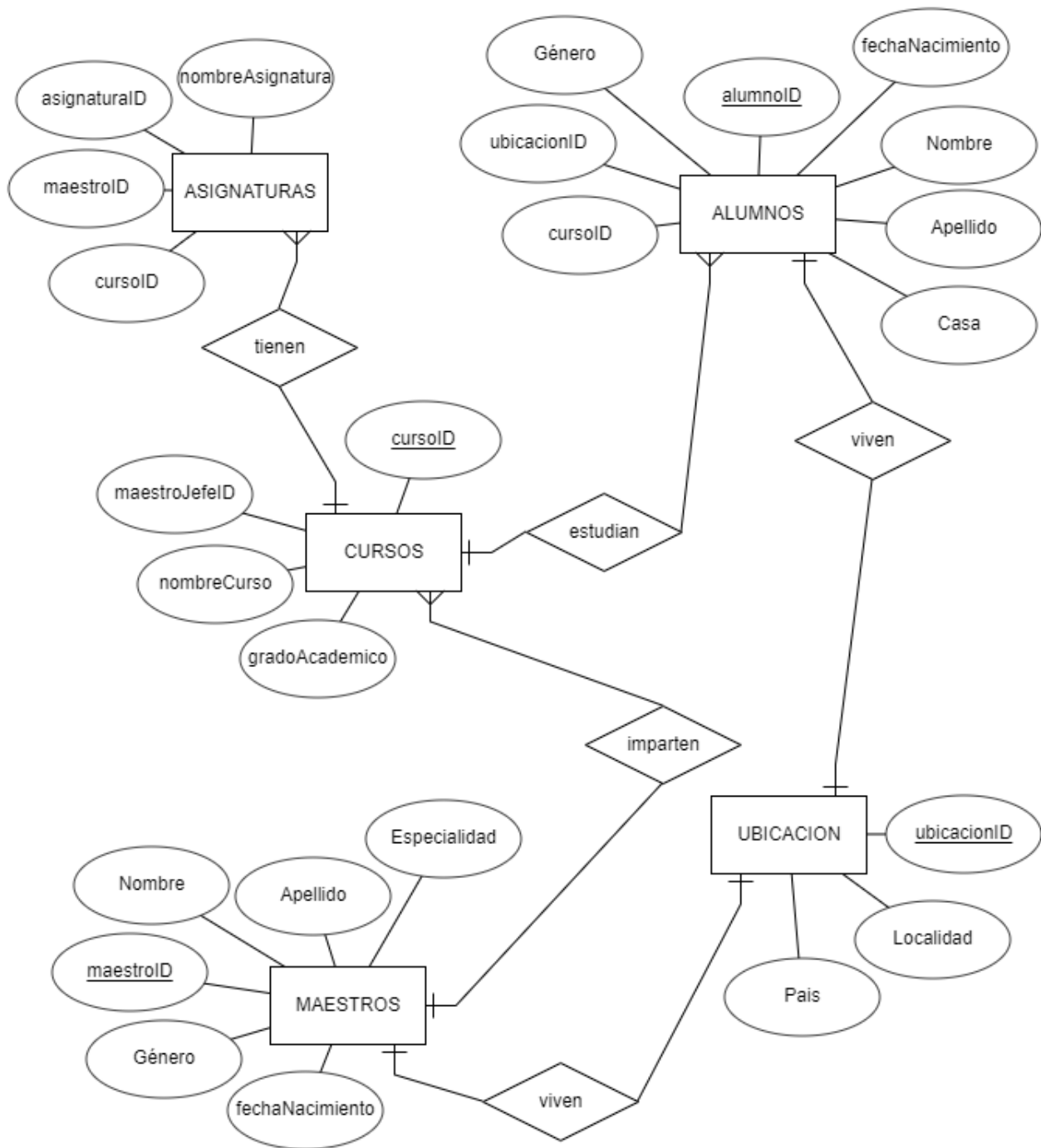
- Curso Primer Año: Iniciación Mágica
 - *Encantamientos Básicos*
 - *Herbología Elemental*
 - *Pociones para Principiantes*
 - *Historia de la Magia Antigua*
 - *Defensa Básica contra las Artes Oscuras*
 - *Vuelo Inicial en Escoba*
 - *Estudios Muggles Introdutorios*
- Curso Segundo Año: Desarrollo Mágico
 - *Transformaciones Intermedias*
 - *Cuidado de Criaturas Mágicas*
 - *Pociones Avanzadas*

- *Historia de la Magia Contemporánea*
- *Defensa Intermedia contra las Artes Oscuras*
- *Aritmancia Elemental*
- *Runas Antiguas*
- Curso Tercer Año: Magia Avanzada
 - *Encantamientos Avanzados*
 - *Herbología Avanzada*
 - *Pociones Maestras*
 - *Historia de la Magia Medieval*
 - *Defensa Avanzada contra las Artes Oscuras*
 - *Adivinación Elemental*
 - *Astronomía Mágica*
- Curso Cuarto Año: Especialización Mágica
 - *Transformaciones Avanzadas*
 - *Cuidado de Criaturas Mágicas Exóticas*
 - *Pociones Maestras Avanzadas*
 - *Historia de la Magia Moderna*
 - *Defensa contra las Artes Oscuras Especializada*
 - *Estudios de Artefactos Mágicos*
 - *Ocultismo*
- Curso Quinto Año: Maestría Mágica
 - *Encantamientos Supremos*
 - *Herbología Aplicada*
 - *Pociones Maestras Maestría*
 - *Historia de la Magia Contemporánea Avanzada*
 - *Defensa Avanzada contra las Artes Oscuras Aplicada*
 - *Aritmancia Avanzada*
 - *Estudio de las Maldiciones*
- Curso Sexto Año: Sabiduría Arcana
 - *Transformaciones Maestras*
 - *Cuidado de Criaturas Mágicas Legendarias*
 - *Pociones Maestras Supremas*
 - *Historia de la Magia Postmoderna*
 - *Defensa Maestra contra las Artes Oscuras*

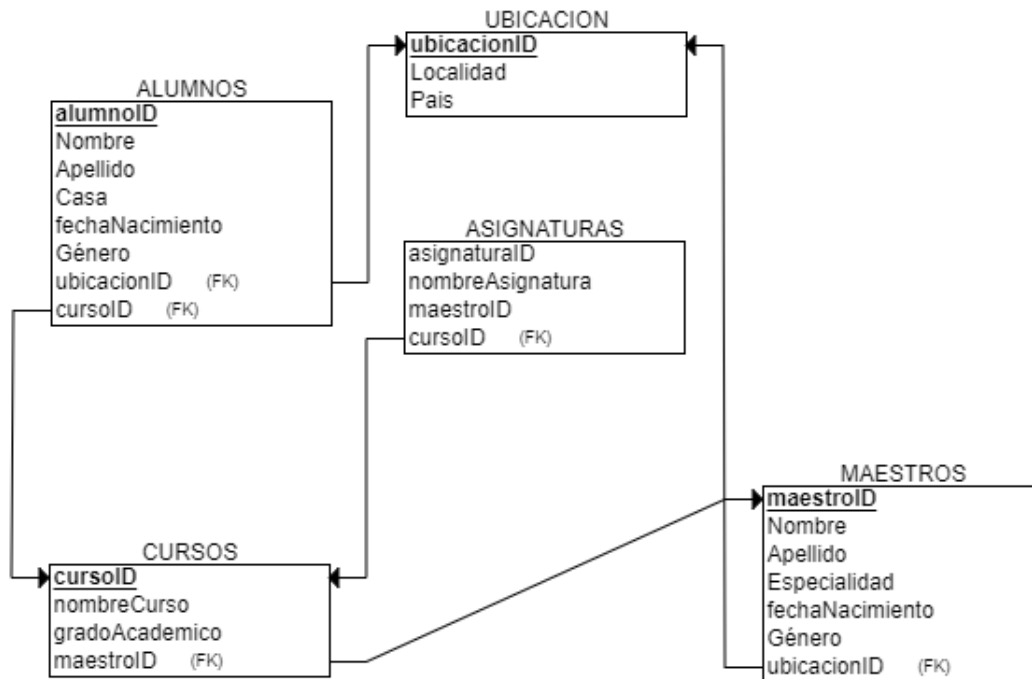
- *Adivinación Avanzada*
- *Astronomía Avanzada*

Curso	Asignatura	Profesor
Primer Año	Iniciación Mágica Encantamientos Básicos Herbología Elemental Pociones para Principiantes Historia de la Magia Antigua Defensa Básica contra las Artes Oscuras Vuelo Inicial en Escoba Estudios Muggles Introdutorios	Profesor Filius Flitwick Profesora Pomona Sprout Profesor Severus Snape Profesora Bathilda Bagshot Profesor Quirinus Quirrell Madame Hooch Profesor Arthur Weasley
Segundo Año	Desarrollo Mágico Transformaciones Intermedias Cuidado de Criaturas Mágicas Pociones Avanzadas Historia de la Magia Contemporánea Defensa Intermedia contra las Artes Oscuras Aritmancia Elemental Runas Antiguas	Profesora Minerva McGonagall Profesor Rubeus Hagrid Profesora Horace Slughorn Profesor Binns (fantasma) Profesora Gilderoy Lockhart Profesora Septima Vector Profesor Bathsheda Babbling
Tercer Año	Magia Avanzada Encantamientos Avanzados Herbología Avanzada Pociones Maestras Historia de la Magia Medieval Defensa Avanzada contra las Artes Oscuras Adivinación Elemental Astronomía Mágica	Profesor Remus Lupin Profesora Neville Longbottom Profesora Horace Slughorn Profesora Adalbert Waffling Profesor Alastor Moody Profesora Sybill Trelawney Profesor Aurora Sinistra
Cuarto Año	Especialización Mágica Transformaciones Avanzadas Cuidado de Criaturas Mágicas Exóticas Pociones Maestras Avanzadas Historia de la Magia Moderna Defensa contra las Artes Oscuras Especializada Estudios de Artefactos Mágicos Ocultismo	Profesora Minerva McGonagall Profesor Rubeus Hagrid Profesora Horace Slughorn Profesor Cuthbert Binns (fantasma) Profesor Dolores Umbridge Profesora Bathilda Bagshot Profesor Gellert Grindelwald
Quinto Año	Maestría Mágica Encantamientos Supremos Herbología Aplicada Pociones Maestras Maestría Historia de la Magia Contemporánea Avanzada Defensa Avanzada contra las Artes Oscuras Aplicada Aritmancia Avanzada Estudio de las Maldiciones	Profesor Filius Flitwick Profesora Neville Longbottom Profesora Horace Slughorn Profesora Bathilda Bagshot Profesora Nymphadora Tonks Profesora Septima Vector Profesor Severus Snape
Sexto Año	Sabiduría Arcana Transformaciones Maestras Cuidado de Criaturas Mágicas Legendarias Pociones Maestras Supremas Historia de la Magia Postmoderna Defensa Maestra contra las Artes Oscuras Adivinación Avanzada Astronomía Avanzada	Profesor Minerva McGonagall Profesor Rubeus Hagrid Profesor Horace Slughorn Profesora Bathilda Bagshot Profesor Alastor Moody Profesora Sybill Trelawney Profesora Aurora Sinistra

2.2. Modelo conceptual



2.3. Modelo relacional



2.4. Script de creación

```

1  -- Solo los titulos de las tablas para escuela
2  CREATE TABLE UBICACION
3  (
4
5  );
6
7  CREATE TABLE MAESTROS
8  (
9
10 );
11
12 CREATE TABLE CURSOS
13 (
14
15 );
16
17 CREATE TABLE ASIGNATURAS
18 (
19
20 );
21
22 CREATE TABLE ALUMNOS
23 (

```

24) ;
25

Listing 1: Tablas para la BdDatos

2.5. Script de Insert

- Se deben generar 100 registros para cada tabla.
- Si para el buen funcionamiento de la base de datos se requieren más de 100 registros o menos de 100 registros en una tabla, se debe explicar claramente la razón, sólo en este caso sí se debe incluir un apartado en el reporte final.

Tablas:

- UBICACION: Si es posible llegar a 100 registros de ubicación
- MAESTROS: No es posible llegar a 100 porque la tabla solo contempla 24 maestros registrados
- CURSOS: No es posible llegar a 100 porque solo se contemplan 6 cursos para la escuela
- ASIGNATURAS: No es posible llegar a 100 porque hay 7 asignaturas por cada curso, son 24 asignaturas
- Alumnos: Si es posible llegar a 100 registros de alumnos

2.6. Funcionamiento Restricciones

Evidencia del funcionamiento de al menos 4 restricciones de integridad referencial.

Restricción 01

- Tablas involucradas en la restricción: ALUMNOS y UBICACION
- FK de la tabla que referencia y PK de la tabla referenciada: ubicacionID en ALUMNOS es una clave foránea que referencia a ubicacionID en UBICACION
- Justificación del trigger de integridad referencial elegido: Esta restricción asegura que cada alumno esté asociado con una ubicación válida en la tabla UBICACION. Si se intenta eliminar una ubicación que está siendo referenciada por un alumno, o se intenta actualizar el ubicacionID de una ubicación a un valor que no existe en la tabla ALUMNOS, la operación será rechazada.
- Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.

```
DELETE FROM UBICACION WHERE ubicacionID = 'OXFU';
```

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

```
mysql> DELETE FROM UBICACION WHERE ubicacionID = 'OXFU';
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails ('escuela'. 'alumnos', CONSTRAINT 'alumnos_ibfk_1' FOREIGN KEY ('ubicacionID') REFERENCES 'ubicacion' ('ubicacionID'))
```

Restricción 02

- Tablas involucradas en la restricción: CURSOS y MAESTROS
- FK de la tabla que referencia y PK de la tabla referenciada: maestroID en CURSOS es una clave foránea que referencia a maestroID en MAESTROS
- Justificación del trigger de integridad referencial elegido: Esta restricción asegura que cada curso esté asociado con un maestro válido en la tabla MAESTROS. Si se intenta eliminar un maestro que está siendo referenciado por un curso, o se intenta actualizar el maestroID de un maestro a un valor que no existe en la tabla CURSOS, la operación será rechazada.
- Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.

```
DELETE FROM MAESTROS WHERE maestroID = 'GG001';
```

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

```
mysql> DELETE FROM MAESTROS WHERE maestroID = 'GG001';  
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails ('escuela'. 'cursos', CONSTRAINT 'cursos_ibfk_1' FOREIGN KEY ('maestroID') REFERENCES 'maestros' ('maestroID'))
```

Restricción 03

- Tablas involucradas en la restricción: ASIGNATURAS y CURSOS
- FK de la tabla que referencia y PK de la tabla referenciada: cursoID en ASIGNATURAS es una clave foránea que referencia a cursoID en CURSOS
- Justificación del trigger de integridad referencial elegido: Esta restricción asegura que cada asignatura esté asociada con un curso válido en la tabla CURSOS. Si se intenta eliminar un curso que está siendo referenciado por una asignatura, o se intenta actualizar el cursoID de un curso a un valor que no existe en la tabla ASIGNATURAS, la operación será rechazada.
- Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.

```
DELETE FROM CURSOS WHERE cursoID = 'A1IM';
```

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

```
mysql> DELETE FROM CURSOS WHERE cursoID = 'A1IM';  
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails ('escuela'. 'alumnos', CONSTRAINT 'alumnos_ibfk_2' FOREIGN KEY ('cursoID') REFERENCES 'cursos' ('cursoID'))
```

Restricción 03

- Tablas involucradas en la restricción: ALUMNOS y CURSOS
- FK de la tabla que referencia y PK de la tabla referenciada: cursoID en ALUMNOS es una clave foránea que referencia a cursoID en CURSOS
- Justificación del trigger de integridad referencial elegido: Esta restricción asegura que cada alumno esté asociado con un curso válido en la tabla CURSOS. Si se intenta eliminar un curso que está siendo referenciado por un alumno, o se intenta actualizar el cursoID de un curso a un valor que no existe en la tabla ALUMNOS, la operación será rechazada.
- Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.

```
UPDATE ALUMNOS
  SET cursoID = 'A7SA'
  WHERE alumnoID = 'S29';
```

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.



```
mysql> UPDATE ALUMNOS
  -> SET cursoID = 'A7SA'
  -> WHERE alumnoID = 'S29';
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails ('escuela'. 'alumnos', CONSTRAINT 'alumnos_ibfk_2' FOREIGN KEY ('cursoID') REFERENCES 'cursos' ('cursoID'))
```

2.7. Funcionamiento Restricciones check

Evidencia del funcionamiento de al menos 3 restricciones check para “atributos” de varias tablas.

- Tabla elegida: ALUMNOS
- Atributo elegido: casa
- Descripción: Esta restricción asegura que el valor de la columna Casa sea uno de los cuatro nombres de las casas de Hogwarts: 'Hufflepuff', 'Ravenclaw', 'Gryffindor', 'Slytherin'.
- Instrucción para la creación de la restricción:

```
ALTER TABLE ALUMNOS
ADD CONSTRAINT casaCasa CHECK (Casa IN ('Hufflepuff',
                                         'Ravenclaw',
                                         'Gryffindor',
                                         'Slytherin'));
```

- Instrucción que permita evidenciar que la restricción esta funcionando:

```
INSERT INTO ALUMNOS (alumnoID, Nombre, Apellido, Casa, fechaNacimiento,
                    Genero, ubicacionID, cursoID)
VALUES ('nuevoID', 'Nombre', 'Apellido', 'CasaIncorrecta',
        '2000-01-01', 'Genero', 'ubicacionID', 'cursoID');
```

- Captura de pantalla

```
mysql> ALTER TABLE ALUMNOS
-> ADD CONSTRAINT casaCasa CHECK (Casa IN ('Hufflepuff', 'Ravenclaw', 'Gryffindor', 'Slytherin'));
ERROR 3822 (HY000): Duplicate check constraint name 'casaCasa'.
```

- Tabla elegida: MAESTROS
- Atributo elegido: fechaNacimiento
- Descripción: Esta restricción asegura que el valor de la columna fechaNacimiento sea una fecha válida y no sea en el futuro.
- Instrucción para la creación de la restricción:

```
ALTER TABLE MAESTROS
ADD CONSTRAINT fechaMaestro CHECK (fechaNacimiento <= CURRENT_DATE);
```

- Instrucción que permita evidenciar que la restricción esta funcionando:

```
INSERT INTO MAESTROS
    (maestroID, Nombre, Apellido, Especialidad, fechaNacimiento,
     Genero, ubicacionID)
VALUES ('nuevoID', 'Nombre', 'Apellido', 'Especialidad',
        '3000-01-01', 'Genero', 'ubicacionID');
```

- Tabla elegida: ALUMNOS
- Atributo elegido: fechaNacimiento
- Breve descripción de la restricción: Esta restricción asegura que el valor de la columna fechaNacimiento sea una fecha válida y no sea en el futuro.
- Instrucción para la creación de la restricción:

```
ALTER TABLE ALUMNOS
ADD CONSTRAINT fechaAlumno CHECK (fechaNacimiento <= CURRENT_DATE);
```

- Instrucción que permita evidenciar que la restricción esta funcionando:

```
INSERT INTO ALUMNOS (alumnoID, Nombre, Apellido, Casa, fechaNacimiento,
                     Genero, ubicacionID, cursoID)
VALUES ('nuevoID', 'Nombre', 'Apellido', 'Hufflepuff', '3000-01-01',
        'Genero', 'ubicacionID', 'cursoID');
```

2.8. Creación de dominios personalizados

Evidencia de la creación de al menos tres dominios personalizados. Se deben utilizar restricciones check en la creación de los tres dominios.

- Tabla elegida: ALUMNOS
- Atributo elegido: Casa
- Descripción: El dominio Casa representa las cuatro casas de la escuela Hogwarts de Magia y Hechicería. Las cuatro casas son: Hufflepuff Ravenclaw Gryffindor Slytherin
- Instrucción para la creación del dominio personalizado.

```
1      CREATE DOMAIN Casa
2      AS CHAR(20)
3      CHECK (
4          Casa IN ('Hufflepuff', 'Ravenclaw',
5                  'Gryffindor', 'Slytherin')
6      );
```

Listing 2: Tablas para la Bddatos

- Tabla elegida: MAESTROS
- Atributo elegido: genero
- Descripción: El dominio Genero representa el género de un maestro. Los valores permitidos son: Masculino, Femenino
- Instrucción para la creación del dominio personalizado.

```
1      CREATE DOMAIN Genero
2      AS CHAR(20)
3      CHECK (
4          Genero IN ('Masculino', 'Femenino')
5      );
```

Listing 3: Tablas para la Bddatos

- Tabla elegida: MAESTROS
- Atributo elegido: especialidad
- Descripción: El dominio Especialidad representa la especialidad de un maestro. La restricción check propuesta indica que el valor de este atributo debe ser una cadena de caracteres de al menos 3 caracteres y no más de 50 caracteres.
- Instrucción para la creación del dominio personalizado.

```
1      USE [EscuelaDeMagia]
2      GO
3
4      CREATE DOMAIN [Especialidad] AS VARCHAR(50)
5      CHECK (LENGTH(VALUE) >= 3 AND LENGTH(VALUE) <= 50)
6      GO
```

Listing 4: Tablas para la BdDatos

2.9. Restricciones para tuplas

Evidencia del funcionamiento de al menos 2 restricciones para “tuplas” en diferentes tablas (Unidad 8 Integridad, tema Specifying Constraints on Tuples Using CHECK)

- Tabla elegida: ALUMNOS
- Breve descripción de la restricción: Esta restricción asegura que los alumnos no puedan ser mayores de 100 años. Para ello, se verifica que la fecha de nacimiento sea posterior al año actual menos 100.
- Instrucción para la creación de la restricción:

```
ALTER TABLE ALUMNOS
ADD CONSTRAINT edadAlumno
CHECK (fechaNacimiento > DATE_SUB(CURRENT_DATE, INTERVAL 100 YEAR));
```

- Instrucción Insert o Update que permita evidenciar que la restricción esta funcionando:

```
INSERT INTO ALUMNOS (alumnoID, Nombre, Apellido, Casa, fechaNacimiento, Genero, ubic
VALUES ('nuevoID', 'Nombre', 'Apellido', 'Hufflepuff', '1900-01-01', 'Genero', 'ubic
```

- Tabla elegida: ‘MAESTROS’
- Breve descripción de la restricción: Esta restricción asegura que los maestros no puedan ser menores de 25 años. Para ello, se verifica que la fecha de nacimiento sea anterior al año actual menos 25.
- Instrucción para la creación de la restricción:

```
ALTER TABLE MAESTROS
ADD CONSTRAINT edadMaestro
CHECK (fechaNacimiento < DATE_SUB(CURRENT_DATE, INTERVAL 25 YEAR));
```

- Instrucción Insert o Update que permita evidenciar que la restricción esta funcionando:

```
INSERT INTO MAESTROS (maestroID, Nombre, Apellido, Especialidad, fechaNacimiento, Gene
VALUES ('nuevoID', 'Nombre', 'Apellido', 'Especialidad', '2000-01-01', 'Genero', 'ubic
```


2.10. Consultas

Plantea 3 consultas que consideres relevantes para la base de datos propuesta. Para cada consulta planteada, incluir en el reporte los siguientes incisos:

Consulta 01

- Redacción clara de la consulta: Obtener una lista de todos los alumnos que están en la casa Gryffindor.
- Código en lenguaje SQL de la consulta.

```
1  SELECT Nombre, Apellido
2  FROM ALUMNOS
3  WHERE Casa = 'Slytherin';
```

Listing 5: Tablas para la BdDatos

```
mysql> SELECT Nombre, Apellido
->      FROM ALUMNOS
->      WHERE Casa = 'Slytherin';
+-----+-----+
| Nombre | Apellido |
+-----+-----+
| Oliver | Bennett |
| Ella   | Ward    |
| Logan  | Perez   |
| Zoe    | Taylor  |
| Aiden | Hill    |
| Hudson | Bennett |
| Eva    | Ward    |
| Zane   | Perez   |
| Aria   | Taylor  |
| Oliver | Hill    |
| Hudson | Bennett |
| Eva    | Ward    |
| Zane   | Perez   |
| Aria   | Taylor  |
| Oliver | Hill    |
| Hudson | Bennett |
| Eva    | Ward    |
| Zane   | Perez   |
| Aria   | Taylor  |
| Oliver | Hill    |
| Hudson | Bennett |
| Alicia | Silva   |
| Zane   | Perez   |
| Aria   | Taylor  |
| Oliver | Hill    |
+-----+-----+
25 rows in set (0.00 sec)
```

Consulta 02

- Redacción clara de la consulta: Obtener una lista de todos los cursos que son impartidos por un maestro específico.
- Código en lenguaje SQL de la consulta.

```
1  SELECT nombreCurso
2  FROM CURSOS
3  WHERE maestroID = 'GG001';
```

Listing 6: Tablas para la BdDatos

```
mysql> SELECT nombreCurso
->      FROM CURSOS
->      WHERE maestroID = 'GG001';
+-----+
| nombreCurso |
+-----+
| Sabiduría Arcana |
+-----+
1 row in set (0.03 sec)
```

Consulta 03

- Redacción clara de la consulta: Obtener una lista de todos los maestros que viven en una ubicación específica.
- Código en lenguaje SQL de la consulta.

```
1  SELECT Nombre, Apellido
2  FROM MAESTROS
3  WHERE ubicacionID = 'NEWC';
```

Listing 7: Tablas para la Bddatos

```
mysql> SELECT Nombre, Apellido
->      FROM MAESTROS
->      WHERE ubicacionID = 'NEWC';
+-----+-----+
| Nombre | Apellido |
+-----+-----+
| Arthur | Weasley  |
| Minerva | McGonagall |
+-----+-----+
2 rows in set (0.01 sec)
```

2.11. Vistas

Plantea 3 vistas que consideres relevantes para la base de datos propuesta. Para cada vista planteada, incluir en el reporte los siguientes incisos:

Vista 01

- Redacción clara de la vista planteada: Una vista que muestre todos los alumnos junto con la información de su ubicación.
- Código en lenguaje SQL que permita crear la vista solicitada.

```
1  CREATE VIEW AlumnosUbicacion AS
2  SELECT ALUMNOS.Nombre, ALUMNOS.Apellido, UBICACION.Localidad,
        UBICACION.Pais
3  FROM ALUMNOS
4  JOIN UBICACION ON ALUMNOS.ubicacionID = UBICACION.ubicacionID
        ;
```

Listing 8: Tablas para la Bddatos

- USO:

```
1 SELECT * FROM AlumnosUbicacion;
```

Listing 9: Tablas para la BdDatos

```
mysql> CREATE VIEW AlumnosUbicacion AS
--> SELECT ALUMNOS.Nombre, ALUMNOS.Apellido, UBICACION.Localidad, UBICACION.Pais
--> FROM ALUMNOS
--> JOIN UBICACION ON ALUMNOS.ubicacionID = UBICACION.ubicacionID;
Query OK, 0 rows affected (0.06 sec)

mysql> SELECT * FROM AlumnosUbicacion;
+-----+-----+-----+-----+
| Nombre | Apellido | Localidad | Pais |
+-----+-----+-----+-----+
| Ava | Ross | Bangor | Reino Unido |
| Eva | Hill | Bangor | Reino Unido |
| Caleb | Collins | Bangor | Reino Unido |
| Sophia | Collins | Birmingham | Reino Unido |
| Jackson | Anderson | Birmingham | Reino Unido |
| Olivia | Garcia | Birmingham | Reino Unido |
| Sophie | Hill | Chelmsford | Reino Unido |
| Carter | Collins | Chelmsford | Reino Unido |
| Zoe | Jones | Chelmsford | Reino Unido |
| Sophie | Hill | Chelmsford | Reino Unido |
| Carter | Collins | Chelmsford | Reino Unido |
| Zoe | Jones | Chelmsford | Reino Unido |
| Levi | Ferguson | Chelmsford | Reino Unido |
| Mila | Ross | Chelmsford | Reino Unido |
| Mason | Lopez | Exeter | Reino Unido |
| Sophia | Collins | Exeter | Reino Unido |
| Olivia | Garcia | Exeter | Reino Unido |
| Eva | Martin | Exeter | Reino Unido |
| Mason | Lopez | Exeter | Reino Unido |
| Evelyn | Barnes | Fort William | Escocia |
| Mason | Jones | Fort William | Escocia |
| Scarlett | Collins | Fort William | Escocia |
| Lucas | Ferguson | Fort William | Escocia |
| Mason | Taylor | Hogsmeade | Reino Unido |
| Ava | Hill | Hogsmeade | Reino Unido |
| Jackson | Wright | Hogsmeade | Reino Unido |
| Sophia | Fisher | Hogsmeade | Reino Unido |
| Liam | Simmons | Hogsmeade | Reino Unido |
| Eli | Martinez | Hogsmeade | Reino Unido |
| Aria | Fisher | Hogsmeade | Reino Unido |
| Carter | Turner | Hogsmeade | Reino Unido |
| Luna | Cooper | Hogsmeade | Reino Unido |
| Owen | Ward | Hogsmeade | Reino Unido |
| Aubrey | Simmons | Hogsmeade | Reino Unido |
+-----+-----+-----+-----+
```

Vista 02

- Redacción clara de la vista planteada: Una vista que muestre todos los cursos junto con el nombre del maestro que los imparte.
- Código en lenguaje SQL que permita crear la vista solicitada.

```
1 CREATE VIEW CursosMaestros AS
2 SELECT CURSOS.nombreCurso, MAESTROS.Nombre, MAESTROS.Apellido
3 FROM CURSOS
4 JOIN MAESTROS ON CURSOS.maestroID = MAESTROS.maestroID;
```

Listing 10: Tablas para la BdDatos

- USO:

```
1 SELECT * FROM CursosMaestros;
```

Listing 11: Tablas para la BdDatos

```
mysql> CREATE VIEW CursosMaestros AS
--> SELECT CURSOS.nombreCurso, MAESTROS.Nombre, MAESTROS.Apellido
--> FROM CURSOS
--> JOIN MAESTROS ON CURSOS.maestroID = MAESTROS.maestroID;
Query OK, 0 rows affected (0.03 sec)

mysql> SELECT * FROM CursosMaestros;
+-----+-----+-----+
| nombreCurso | Nombre | Apellido |
+-----+-----+-----+
| Iniciación Mágica | Minerva | McGonagall |
| Desarrollo Mágico | Filius | Flitwick |
| Magia Avanzada | Remus | Lupin |
| Especialización Mágica | Septima | Vector |
| Maestría Mágica | Aurora | Sinistra |
| Sabiduría Arcana | Gellert | Grindelwald |
+-----+-----+-----+
6 rows in set (0.03 sec)
```

Vista 03

- Redacción clara de la vista planteada: Una vista que muestre todas las asignaturas junto con el nombre del curso al que pertenecen.
- Código en lenguaje SQL que permita crear la vista solicitada.

```

1 CREATE VIEW AsignaturasCursos AS
2 SELECT ASIGNATURAS.nombreAsignatura, CURSOS.nombreCurso
3 FROM ASIGNATURAS
4 JOIN CURSOS ON ASIGNATURAS.cursoID = CURSOS.cursoID;
```

Listing 12: Tablas para la BdBdatos

- USO:

```

1 SELECT * FROM AsignaturasCursos;
```

Listing 13: Tablas para la BdBdatos

```

mysql> CREATE VIEW AsignaturasCursos AS
-> SELECT ASIGNATURAS.nombreAsignatura, CURSOS.nombreCurso
-> FROM ASIGNATURAS
-> JOIN CURSOS ON ASIGNATURAS.cursoID = CURSOS.cursoID;
Query OK, 0 rows affected (0.02 sec)

mysql> SELECT * FROM AsignaturasCursos;
+-----+-----+
| nombreAsignatura | nombreCurso |
+-----+-----+
| Defensa Básica contra las Artes Oscuras | Iniciación Mágica |
| Encantamientos Básicos | Iniciación Mágica |
| Estudios Muggles Introdutorios | Iniciación Mágica |
| Herbología Elemental | Iniciación Mágica |
| Historia de la Magia Antigua | Iniciación Mágica |
| Pociones para Principiantes | Iniciación Mágica |
| Vuelo Inicial en Escoba | Iniciación Mágica |
| Aritmancia Elemental | Desarrollo Mágico |
| Cuidado de Criaturas Mágicas | Desarrollo Mágico |
| Defensa Intermedia contra las Artes Oscuras | Desarrollo Mágico |
| Historia de la Magia Contemporánea | Desarrollo Mágico |
| Pociones Avanzadas | Desarrollo Mágico |
| Runas Antiguas | Desarrollo Mágico |
| Transformaciones Intermedias | Desarrollo Mágico |
| Adivinación Elemental | Magia Avanzada |
| Astronomía Mágica | Magia Avanzada |
| Defensa Avanzada contra las Artes Oscuras | Magia Avanzada |
| Encantamientos Avanzados | Magia Avanzada |
| Herbología Avanzada | Magia Avanzada |
| Historia de la Magia Medieval | Magia Avanzada |
| Pociones Maestras | Magia Avanzada |
| Cuidado de Criaturas Mágicas Exóticas | Especialización Mágica |
| Defensa contra las Artes Oscuras Especializada | Especialización Mágica |
| Estudios de Artefactos Mágicos | Especialización Mágica |
| Historia de la Magia Moderna | Especialización Mágica |
| Ocultismo | Especialización Mágica |
| Pociones Maestras Avanzadas | Especialización Mágica |
| Transformaciones Avanzadas | Especialización Mágica |
| Aritmancia Avanzada | Maestría Mágica |
| Defensa Avanzada contra las Artes Oscuras Aplicada | Maestría Mágica |
| Estudio de las Maldiciones | Maestría Mágica |
| Encantamientos Supremos | Maestría Mágica |
| Historia de la Magia Contemporánea Avanzada | Maestría Mágica |
| Herbología Aplicada | Maestría Mágica |
| Pociones Maestras Maestría | Maestría Mágica |
| Adivinación Avanzada | Sabiduría Arcana |
| Astronomía Avanzada | Sabiduría Arcana |
| Cuidado de Criaturas Mágicas Legendarias | Sabiduría Arcana |
| Defensa Maestra contra las Artes Oscuras | Sabiduría Arcana |
| Historia de la Magia Postmoderna | Sabiduría Arcana |
| Pociones Maestras Supremas | Sabiduría Arcana |
| Sabiduría Arcana | Sabiduría Arcana |
+-----+-----+
```

3. Escuela

Saludos desde Hogwarts Escuela de Magia y Hechicería. Después de revisar exhaustivamente sus habilidades y experiencia en el mundo mágico de la programación, nos complace informarles que han sido seleccionados para crear y gestionar la base de datos mágica de nuestra venerable institución.

3.1. Lista de Requerimientos

Tablas para la base de datos

Principales restricciones

- TABLAS

3.2. Modelo conceptual

MAPA

3.3. Modelo relacional

MAPA

3.4. Script de creación

```
1  -- Solo los titulos de las tablas para naves
```

Listing 14: Tablas para la BdDatos

3.5. Script de Insert

- Se deben generar 100 registros para cada tabla.
- Si para el buen funcionamiento de la base de datos se requieren más de 100 registros o menos de 100 registros en una tabla, se debe explicar claramente la razón, sólo en este caso sí se debe incluir un apartado en el reporte final.

Tablas:

- TABLAS

3.6. Funcionamiento Restricciones

Evidencia del funcionamiento de al menos 4 restricciones de integridad referencial.

Restricción 01

- Tablas involucradas en la restricción:
 - FK de la tabla que referencia y PK de la tabla referenciada:
 - Justificación del trigger de integridad referencial elegido:
 - Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.
-
- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

CAPTURA

Restricción 02

- Tablas involucradas en la restricción:
 - FK de la tabla que referencia y PK de la tabla referenciada:
 - Justificación del trigger de integridad referencial elegido:
 - Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.
-
- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

CAPTURA

Restricción 03

- Tablas involucradas en la restricción:
 - FK de la tabla que referencia y PK de la tabla referenciada:
 - Justificación del trigger de integridad referencial elegido:
 - Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.
-
- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

CAPTURA

Restricción 04

- Tablas involucradas en la restricción:
 - FK de la tabla que referencia y PK de la tabla referenciada:
 - Justificación del trigger de integridad referencial elegido:
 - Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.
-
- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

CAPTURA

3.7. Funcionamiento Restricciones check

Evidencia del funcionamiento de al menos 3 restricciones check para “atributos” de varias tablas.

Restricción 01

- Tabla elegida:
 - Atributo elegido:
 - Descripción:
 - Instrucción para la creación de la restricción:
 -
 - Instrucción que permita evidenciar que la restricción esta funcionando:
-
- Captura de pantalla

CAPTURA

Restricción 02

- Tabla elegida:
- Atributo elegido:
- Descripción:
- Instrucción para la creación de la restricción:
-

- Instrucción que permita evidenciar que la restricción esta funcionando:
- Captura de pantalla

CAPTURA

Restricción 03

- Tabla elegida:
- Atributo elegido:
- Descripción:
- Instrucción para la creación de la restricción:
-
- Instrucción que permita evidenciar que la restricción esta funcionando:
- Captura de pantalla

CAPTURA

3.8. Creación de dominios personalizados

Evidencia de la creación de al menos tres dominios personalizados. Se deben utilizar restricciones check en la creación de los tres dominios.

Dominio 01

- Tabla elegida:
- Atributo elegido:
- Descripción:
- Instrucción para la creación del dominio personalizado.

Listing 15: Tablas para la BdDatos

Dominio 02

- Tabla elegida:
- Atributo elegido:
- Descripción:
- Instrucción para la creación del dominio personalizado.

Listing 16: Tablas para la Bddatos

Dominio 03

- Tabla elegida:
- Atributo elegido:
- Descripción:
- Instrucción para la creación del dominio personalizado.

Listing 17: Tablas para la Bddatos

3.9. Restricciones para tuplas

Evidencia del funcionamiento de al menos 2 restricciones para tuplas en diferentes tablas (Unidad 8 Integridad, tema Specifying Constraints on Tuples Using CHECK)

Dominio 01

- Tabla elegida:
- Breve descripción de la restricción:
- Instrucción para la creación de la restricción:
- Instrucción Insert o Update que permita evidenciar que la restricción esta funcionando:

Dominio 02

- Tabla elegida:
- Breve descripción de la restricción:
- Instrucción para la creación de la restricción:
- Instrucción Insert o Update que permita evidenciar que la restricción esta funcionando:

3.10. Consultas

Plantea 3 consultas que consideres relevantes para la base de datos propuesta. Para cada consulta planteada, incluir en el reporte los siguientes incisos:

Consulta 01

- Redacción clara de la consulta:
- Código en lenguaje SQL de la consulta.

Listing 18: Tablas para la Bddatos

CAPTURA

Consulta 02

- Redacción clara de la consulta:
- Código en lenguaje SQL de la consulta.

Listing 19: Tablas para la Bddatos

CAPTURA

Consulta 03

- Redacción clara de la consulta:
- Código en lenguaje SQL de la consulta.

Listing 20: Tablas para la Bddatos

CAPTURA

3.11. Vistas

Plantea 3 vistas que consideres relevantes para la base de datos propuesta. Para cada vista planteada, incluir en el reporte los siguientes incisos:

Vista 01

- Redacción clara de la vista planteada:
- Código en lenguaje SQL que permita crear la vista solicitada.

Listing 21: Tablas para la Bddatos

- USO:

Listing 22: Tablas para la Bddatos

CAPTURA

Vista 02

- Redacción clara de la vista planteada:
- Código en lenguaje SQL que permita crear la vista solicitada.

Listing 23: Tablas para la Bddatos

- USO:

Listing 24: Tablas para la Bddatos

CAPTURA

Vista 03

- Redacción clara de la vista planteada:
- Código en lenguaje SQL que permita crear la vista solicitada.

Listing 25: Tablas para la Bddatos

- USO:

Listing 26: Tablas para la Bddatos

CAPTURA

4. Estadios

La FIFA nos ha encargado el desarrollo de un sistema para la administración de la venta de boletos en estadios de fútbol para los partidos del Mundial de 2026. El sistema deberá cubrir los cuatro estadios de fútbol de México que albergarán partidos del torneo: el Estadio Azteca, el Estadio Corregidora, el Estadio Hidalgo y el Estadio Jalisco.

4.1. Lista de Requerimientos

Datos para la base de datos:

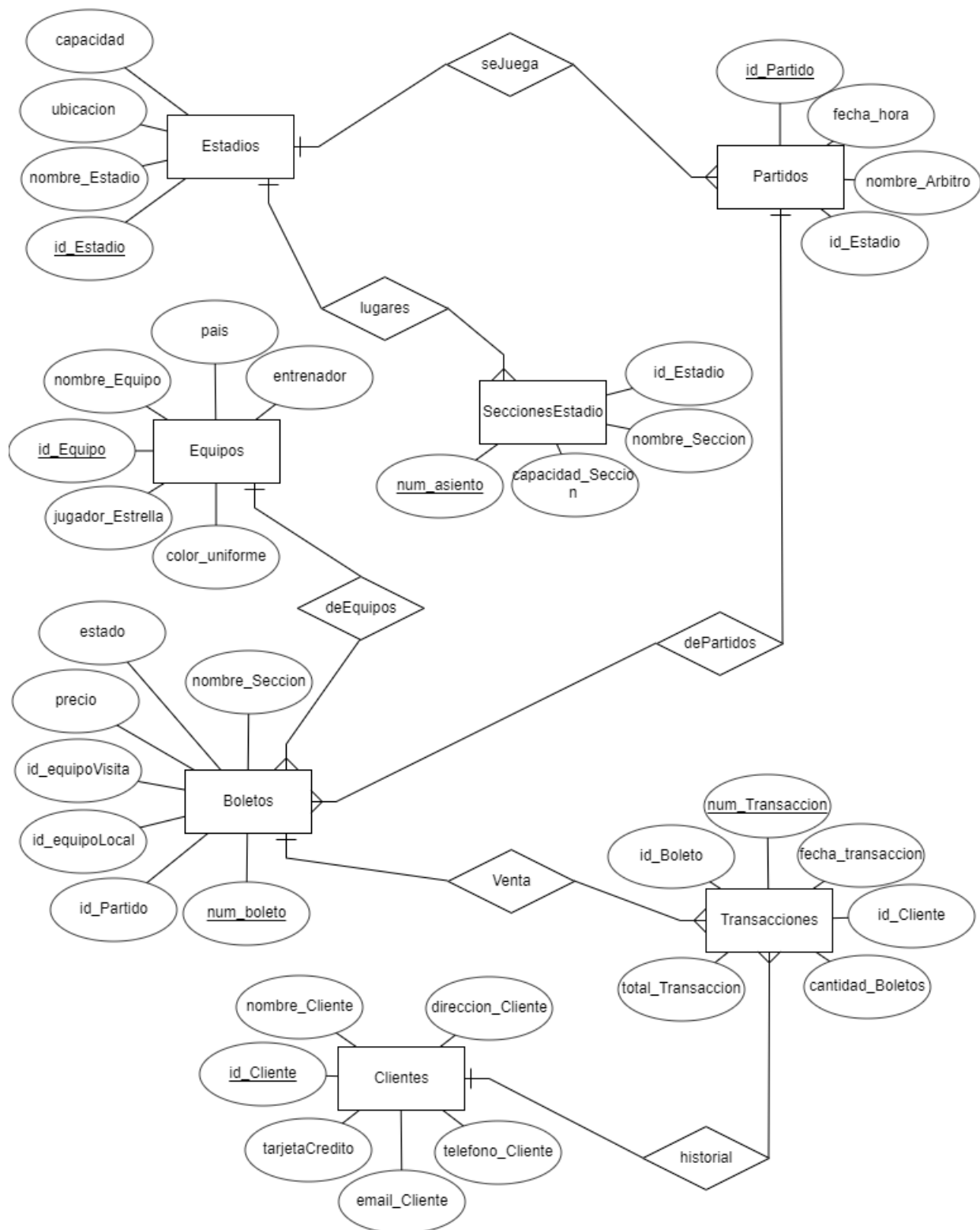
- Nombre de los 4 estadios
 - Estadio Azteca { Capacidad: 83,000,
Ubicación: Calzada de Tlalpan 3465, Sta. Úrsula Coapa, Coyoacán, 04650 Ciudad de México, CDMX,
Precio de las secciones: Cabecera Local: 300, Cabecera Visita: 300, Lateral Visita: 400, Lateral Local: 450, Palcos: 1500 Asientos de las secciones: Cabecera Local: 20,750, Cabecera Visita: 16,600, Lateral Visita: 12,450, Lateral Local: 20,750, Palcos: 12,450 }
 - Estadio Corregidora { Capacidad: 34,000,
Ubicación: Avenidas de las Torres S/N, Centro Sur, 76090 Santiago de Querétaro, Qro.,
Precio de las secciones: Cabecera Local: 250, Cabecera Visita: 250, Lateral Visita: 300, Lateral Local 300, Palcos 900 Asientos de las secciones: Cabecera Local: 8,500, Cabecera Visita: 6,800, Lateral Visita: 5,100, Lateral Local: 8,500, Palcos: 5,100 }
 - Estadio Hidalgo { Capacidad: 30,000,
Ubicación: 2da B Juárez 102, Los Jales, Ex Hacienda de Coscotitlán, 42064 Pachuca de Soto, Hgo.,
Precio de las secciones: Cabecera Local: 250, Cabecera Visita: 250, Lateral Visita: 300, Lateral Local 300, Palcos 1000 Asientos de las secciones: Cabecera Local: 7,500, Cabecera Visita: 6,000, Lateral Visita: 4,500, Lateral Local: 7,500, Palcos: 4,500 }
 - Estadio Jalisco { Capacidad: 55,000,
Ubicación: C. Siete Colinas 1772, Independencia, 44290 Guadalajara, Jal.,
Precio de las secciones: Cabecera Local: 270, Cabecera Visita: 270, Lateral Visita: 350, Lateral Local 400, Palcos 1200 Asientos de las secciones: Cabecera Local: 13,750, Cabecera Visita: 11,000, Lateral Visita: 8,250, Lateral Local: 13,750, Palcos: 8,250 }

Restricciones de los datos

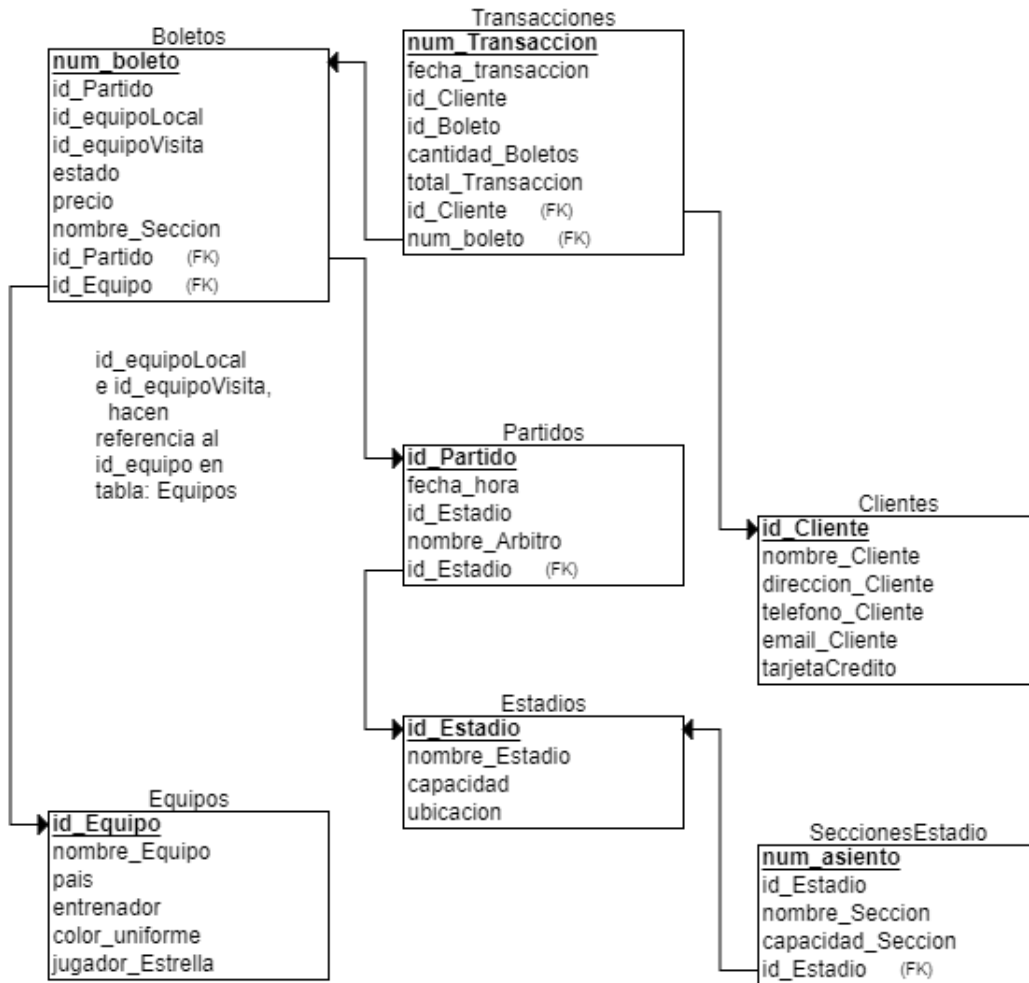
- El número de boleto, el nombre del estadio, el nombre de la sección (dentro de un estadio), el número de transacción, el nombre del equipo y el id del partido deben ser únicos.
- Las fechas del partido y de la transacción deben ser válidas.
- La fecha y hora del partido deben estar en el futuro.
- El estadio debe ser uno de los cuatro estadios de México.
- La sección y el asiento deben ser válidos dentro del estadio.

- El precio y la capacidad del estadio y de la sección deben ser números positivos.
- El estado de venta debe ser Vendido o Disponible.
- El nombre y la dirección del cliente, así como el país del equipo, deben ser cadenas no vacías.
- El teléfono del cliente y el número de tarjeta de crédito deben ser válidos.
- El correo electrónico del cliente debe ser una dirección de correo electrónico válida.
- La ubicación del estadio debe ser una cadena que describa la ciudad o lugar donde se encuentra.
- Debe haber al menos una sección asociada a cada estadio y al menos dos equipos participando en cada partido.
- El estadio asociado al partido y el cliente y el boleto asociados a la transacción deben existir en sus respectivas tablas.
- El precio en la transacción debe ser igual al precio del boleto multiplicado por la cantidad de boletos en la transacción.
- El id del equipo en el partido debe hacer referencia a un equipo existente en la tabla Equipos.
- El id del partido en el equipo partido debe hacer referencia a un partido existente en la tabla Partido.
- La combinación única de id del equipo y id del partido debe asegurar que un equipo no participe más de una vez en el mismo partido.

4.2. Modelo conceptual



4.3. Modelo relacional



4.4. Script de creación

- Script completo y sin errores para la creación de todos los elementos que conforman el esquema de la base de datos.
- El Script debe estar diseñado para la versión 14 de Postgres.
- Deben estar contempladas todas las llaves primarias, llaves candidatas y llaves foráneas; todas las llaves foráneas deben contar con un trigger de integridad referencial (SET NULL, CASCADE o SET DEFAULT).

```

1  -- Solo los titulos de las tablas
2  CREATE TABLE Clientes(
3
4  );
5
6  CREATE TABLE Estadios(
7

```

```
8      );
9
10     CREATE TABLE SeccionesEstadio(
11
12     );
13
14     CREATE TABLE Equipos(
15
16     );
17
18     CREATE TABLE Partidos(
19
20     );
21
22     CREATE TABLE Boletos (
23
24     );
25
26     CREATE TABLE Transacciones(
27
28     );
```

Listing 27: Tablas para la BdDatos

4.5. Script de Insert

- Se deben generar 100 registros para cada tabla.
- Si para el buen funcionamiento de la base de datos se requieren más de 100 registros o menos de 100 registros en una tabla, se debe explicar claramente la razón, sólo en este caso sí se debe incluir un apartado en el reporte final.

Tablas:

- Clientes: Si es posible llegar a 100 registros
- Estadios: La tabla únicamente contiene la información de los 4 estadios que albergan el mundial, por lo que no es posible llegar a 100 registros
- SeccionesEstadio: Si es posible llegar a 100 registros
- Equipos: La tabla únicamente contiene la información de las 32 selecciones clasificadas al mundial por lo que no es posible llegar a 100 registros
- Partidos: La tabla solo contiene los 24 partidos de fase de grupos por lo que no es posible llegar a 100 registros
- Boletos: Si es posible llegar a 100 registros
- Transacciones: Si es posible llegar a 100 registros

4.6. Funcionamiento restricciones

Evidencia del funcionamiento de al menos 4 restricciones de integridad referencial.

Restricción 01

- Tablas involucradas en la restricción: Partidos y Estadios
- FK de la tabla que referencia y PK de la tabla referenciada: FK: id Estadio en la tabla Partidos. PK id Estadio en la tabla Estadios
- Justificación del trigger de integridad referencial elegido: : Esta restricción asegura que no puedes tener un partido en un estadio que no exista en la tabla Estadios.
- Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.

```
DELETE FROM Estadios WHERE id_Estadio = 1;
```

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.



```
| M2026C06 | 2026-06-18 17:00:00 | 2 | John Smith |
| M2026D01 | 2026-06-11 14:00:00 | 3 | John Smith |
| M2026D02 | 2026-06-11 17:00:00 | 4 | Alexander Hall |
| M2026D03 | 2026-06-15 14:00:00 | 1 | Christopher Brown |
| M2026D04 | 2026-06-15 17:00:00 | 2 | Michael Davis |
| M2026D05 | 2026-06-19 14:00:00 | 3 | Michael Davis |
| M2026D06 | 2026-06-19 17:00:00 | 4 | Sophia Harris |
+-----+-----+-----+-----+
24 rows in set (0.00 sec)

mysql> DELETE FROM Estadios WHERE id_Estadio = 1;
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails ('pEstadio`.`Partidos'
, CONSTRAINT 'Partidos_ibfk_1' FOREIGN KEY ('id_Estadio') REFERENCES 'Estadios' ('id_Estadio'))
mysql>
```

Restricción 02

- Tablas involucradas en la restricción: Boletos y Equipos
- FK de la tabla que referencia y PK de la tabla referenciada: Las claves foráneas en Boletos que hacen referencia a Equipos son id equipoLocal y id equipoVisita, y la clave primaria en Equipos es id Equipo.
- Justificación del trigger de integridad referencial elegido: Se utiliza la opción ON DELETE SET NULL. Esto significa que si se elimina un registro en la tabla Equipos, entonces el id equipoLocal y/o id equipoVisita correspondiente en la tabla Boletos se establecerá en NULL. Esto asegura la integridad de los datos.
- Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.

```
DELETE FROM Equipos WHERE id_Equipo = 'SEN';
```

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

```
mysql> SELECT * FROM Boletos;
```

num_boleto	id_Partido	id_equipoLocal	id_equipoVisita	estado	precio	nombre_Seccion
1	M2026A01	QAT	ECU	Vendido	450	Lateral Local
2	M2026A01	QAT	ECU	Disponible	300	Cabecera Visita
3	M2026A02	SEN	NED	Vendido	1200	Palcos
4	M2026A02	SEN	NED	Disponible	600	Lateral Visita
5	M2026B01	IRN	USA	Vendido	400	Lateral Visita
6	M2026B01	IRN	USA	Disponible	450	Lateral Local
7	M2026B02	WAL	ARG	Vendido	1200	Palcos
8	M2026B02	WAL	ARG	Disponible	600	Lateral Visita
9	M2026C01	KSA	MEX	Vendido	450	Lateral Local

```
mysql> DELETE FROM Equipos WHERE id_Equipo = 'SEN';
Query OK, 1 row affected (0.03 sec)

mysql> SELECT * FROM Boletos;
```

num_boleto	id_Partido	id_equipoLocal	id_equipoVisita	estado	precio	nombre_Seccion
1	M2026A01	QAT	ECU	Vendido	450	Lateral Local
2	M2026A01	QAT	ECU	Disponible	300	Cabecera Visita
3	M2026A02	NULL	NED	Vendido	1200	Palcos
4	M2026A02	NULL	NED	Disponible	600	Lateral Visita
5	M2026B01	IRN	USA	Vendido	400	Lateral Visita
6	M2026B01	IRN	USA	Disponible	450	Lateral Local
7	M2026B02	WAL	ARG	Vendido	1200	Palcos

Restricción 03

- Tablas involucradas en la restricción: Transacciones y Clientes
- FK de la tabla que referencia y PK de la tabla referenciada: La clave foránea en Transacciones que hace referencia a Clientes es id Cliente, y la clave primaria en Clientes es id Cliente.
- Justificación del trigger de integridad referencial elegido: Se utiliza la opción ON DELETE CASCADE. Esto significa que si se elimina un registro en la tabla Clientes, entonces todos los registros correspondientes en la tabla Transacciones también se eliminarán. Esto asegura la integridad de los datos.
- Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.

```
DELETE FROM Clientes WHERE id_Cliente = 'CLI100';
```

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

```
diego.santos@email.com | 3456789012345678 |
CLI099 | Camila Costa | Avenida 4, Bairro Oeste, Salvador | 55-71-4567-8901
camila.costa@email.com | 4567890123456789 |
CLI100 | Lucas Pereira | Rua 5, Bairro Leste, Belo Horizonte | 55-31-5678-9012
lucas.pereira@email.com | 5678901234567890 |

100 rows in set (0.04 sec)

mysql> DELETE FROM Clientes WHERE id_Cliente = 'CLI100';
Query OK, 1 row affected (0.04 sec)

mysql>
```

97	2026-05-22 00:00:00	CLI097	97	1	600
98	2026-05-22 00:00:00	CLI098	98	3	3600
99	2026-05-22 00:00:00	CLI099	99	2	1200

```
99 rows in set (0.00 sec)

mysql>
```

Restricción 04

- Tablas involucradas en la restricción: Transacciones y Boletos
- FK de la tabla que referencia y PK de la tabla referenciada: La clave foránea en Transacciones que hace referencia a Boletos es id Boleto, y la clave primaria en Boletos es num boleto.
- Justificación del trigger de integridad referencial elegido: Se utiliza la opción ON DELETE CASCADE. Esto significa que si se elimina un registro en la tabla Boletos, entonces todos los registros correspondientes en la tabla Transacciones también se eliminarán. Esto asegura la integridad de los datos.
- Instrucción UPDATE o DELETE que permita evidenciar que la restricción está funcionando.

```
DELETE FROM Boletos WHERE num_boleto = 2;
```

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

```
mysql> SELECT * FROM Transacciones;
```

num_Transaccion	fecha_transaccion	id_Cliente	id_Boleto	cantidad_Boletos	total_Transaccion
2	2026-01-24 00:00:00	CLI002	2	3	900
3	2026-01-24 00:00:00	CLI003	3	1	1200
4	2026-01-29 00:00:00	CLI004	4	4	2400
5	2026-02-03 00:00:00	CLI005	5	2	800

```
mysql> DELETE FROM Boletos WHERE num_boleto = 2;
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM Transacciones;
```

num_Transaccion	fecha_transaccion	id_Cliente	id_Boleto	cantidad_Boletos	total_Transaccion
3	2026-01-24 00:00:00	CLI003	3	1	1200
4	2026-01-29 00:00:00	CLI004	4	4	2400

4.7. Funcionamiento Restricciones check

Evidencia del funcionamiento de al menos 3 restricciones check para *atributos* de varias tablas.

Check 01

- Tabla elegida: Clientes
- Atributo elegido: tarjetaCredito
- Breve descripción de la restricción: La restricción garantiza que el número de tarjeta de crédito tenga exactamente 16 dígitos.
- Instrucción para la creación de la restricción.

```
1 ALTER TABLE Clientes
2   ADD CONSTRAINT CHK_TarjetaCredito
3   CHECK (LENGTH(tarjetaCredito) = 16);
```

Listing 28: Tablas para la BdDatos

- Instrucción que permita evidenciar que la restricción esta funcionando.

```

1  INSERT INTO Clientes
2      (id_Cliente, nombre_Cliente, direccion_Cliente,
3       telefono_Cliente, email_Cliente, tarjetaCredito)
4  VALUES ('CIL120', 'Juan_Luna', 'Calle_Falsa_123',
5          '55555555', 'juan.perez@example.com', '
          123456789012345');

```

Listing 29: Tablas para la BdDatos

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

```

| diego.santos@email.com | 3456789012345678 |
| CLI099 | Camila Costa | Avenida 4, Bairro Oeste, Salvador | 55-71-4567-8901
| camila.costa@email.com | 4567890123456789 |
+-----+-----+
+-----+-----+
99 rows in set (0.02 sec)

mysql> INSERT INTO Clientes (id_Cliente, nombre_Cliente, direccion_Cliente, telefono_Cliente, email_Cliente, ta
rjetaCredito) VALUES ('CLI123', 'Juan Perez', 'Calle Falsa 123', '55555555', 'juan.perez@example.com', '1234567
89012345');
ERROR 3819 (HY000): Check constraint 'Clientes_chk_1' is violated.
mysql>

```

Check 02

- Tabla elegida: Estadios
- Atributo elegido: capacidad
- Breve descripción de la restricción: La restricción garantiza que la capacidad del estadio sea mayor a 0.
- Instrucción para la creación de la restricción.

```

1  ALTER TABLE Estadios
2  ADD CONSTRAINT CHK_Capacidad CHECK (capacidad > 0);

```

Listing 30: Tablas para la BdDatos

- Instrucción que permita evidenciar que la restricción esta funcionando.

```

1  INSERT INTO Estadios (id_Estadio, nombre_Estadio, capacidad,
2  ubicacion) VALUES
(5, 'Anfield', 0, 'San_Paulo');

```

Listing 31: Tablas para la BdDatos

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

```

mysql> INSERT INTO Estadios (id_Estadio, nombre_Estadio, capacidad, ubicacion) VALUES
(5, -> (5, 'Maracan ', 0, 'San Paulo');
ERROR 3819 (HY000): Check constraint 'Estadios_chk_1' is violated.
mysql>

```

Check 03

- Tabla elegida: Boletos
- Atributo elegido: estado
- Breve descripción de la restricción: Garantiza que el estado del boleto sea Vendido o Disponible.
- Instrucción para la creación de la restricción.

```

1 ALTER TABLE Boletos
2 ADD CONSTRAINT CHK_Estado CHECK (estado IN ('Vendido', '
    Disponible'));
```

Listing 32: Tablas para la BdDatos

- Instrucción que permita evidenciar que la restricción esta funcionando.

```

1 INSERT INTO Boletos (num_boleto, id_Partido, id_equipoLocal,
2 id_equipoVisita,
3 estado, precio, nombre_Seccion)
4 VALUES (1, 'P123', 'E123', 'E124', 'Reservado',
    100, 'Seccion_1');
```

Listing 33: Tablas para la BdDatos

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

```

mysql> INSERT INTO Boletos (num_boleto, id_Partido, id_equipoLocal, id_equipoVisita, estado, precio, nombre_Seccion)
-> VALUES (1, 'P123', 'E123', 'E124', 'Reservado', 100, 'Seccion 1'); -- Esto debería fallar porque el estado es 'Reservado'
ERROR 3819 (HY000): Check constraint 'Boletos_chk_1' is violated.
mysql>
```

4.8. Creación de dominios personalizados

Evidencia de la creación de al menos tres dominios personalizados. Se deben utilizar restricciones check en la creación de los tres dominios.

Dominio 01

- Tabla elegida: Clientes
- Atributo elegido: tarjetaCredito
- Breve descripción del dominio y de la restricción check propuesta: Un dominio personalizado para el número de tarjeta de crédito que debe tener exactamente 16 dígitos.
- Instrucción para la creación del dominio personalizado.

```

1 CREATE DOMAIN TarjetaCredito AS VARCHAR(16)
2 CHECK (LENGTH(VALUE) = 16);
```

Listing 34: Tablas para la BdDatos

Instrucción para usar el dominio personalizado en la tabla:

```
1 ALTER TABLE Clientes
2 ALTER COLUMN tarjetaCredito TYPE TarjetaCredito;
```

Listing 35: Tablas para la Bddatos

- Captura de pantalla de la estructura de la tabla donde se muestre el dominio personalizado en uso.

Dominio 02

- Tabla elegida: Equipos
- Atributo elegido: pais
- Breve descripción del dominio y de la restricción check propuesta: Un dominio personalizado para el país del equipo que no debe estar vacío.
- Instrucción para la creación del dominio personalizado.

```
1 CREATE DOMAIN PaisEquipo AS VARCHAR(255)
2 CHECK (VALUE <> '');
```

Listing 36: Tablas para la Bddatos

Instrucción para usar el dominio personalizado en la tabla:

```
1 ALTER TABLE Equipos
2 ALTER COLUMN pais TYPE PaisEquipo;
```

Listing 37: Tablas para la Bddatos

- Captura de pantalla de la estructura de la tabla donde se muestre el dominio personalizado en uso.

Dominio 03

- Tabla elegida: Partidos
- Atributo elegido: id Estadio
- Breve descripción del dominio y de la restricción check propuesta: Un dominio personalizado para el id del estadio que debe estar entre 1 y 4.
- Instrucción para la creación del dominio personalizado.

```
1 CREATE DOMAIN IdEstadio AS INT
2 CHECK (VALUE IN (1, 2, 3, 4));
```

Listing 38: Tablas para la Bddatos

Instrucción para usar el dominio personalizado en la tabla:

```

1 ALTER TABLE Partidos
2 ALTER COLUMN id_Estadio TYPE IdEstadio;

```

Listing 39: Tablas para la BdDatos

- Captura de pantalla de la estructura de la tabla donde se muestre el dominio personalizado en uso.

4.9. Restricciones para tuplas

Evidencia del funcionamiento de al menos 2 restricciones para “tuplas” en diferentes tablas (Unidad 8 Integridad, tema Specifying Constraints on Tuples Using CHECK)

Restricción 01

- Tabla elegida: Partidos
- Breve descripción de la restricción: La restricción garantiza que el id del estadio sea válido solo si la fecha y hora del partido están en el futuro.
- Instrucción para la creación de la restricción.

```

1 ALTER TABLE Partidos
2 ADD CONSTRAINT
3     CHK_FechaEstadio CHECK ((fecha_hora > CURRENT_TIMESTAMP AND
4     id_Estadio IN (1, 2, 3, 4)) OR (fecha_hora <=
        CURRENT_TIMESTAMP));

```

Listing 40: Tablas para la BdDatos

- Instrucción Insert o Update que permita evidenciar que la restricción esta funcionando.

```

1 INSERT INTO Partidos (id_Partido, fecha_hora, id_Estadio,
2     nombre_Arbitro)
3 VALUES ('P124', '2022-12-31 20:00:00', 5, 'Arbitro Prueba');
4 -- falla porque la fecha y hora estan en el futuro pero
   -- el id del estadio no es valido

```

Listing 41: Tablas para la BdDatos

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

Restricción 02

- Tabla elegida: Transacciones
- Breve descripción de la restricción: La restricción garantiza que el total de la transacción sea igual al precio del boleto multiplicado por la cantidad de boletos.

```
1 ALTER TABLE Transacciones
2 ADD CONSTRAINT CHK_TotalPrecioCantidad CHECK (total_Transaccion
   =
3     (SELECT precio
4     FROM Boletos
5     WHERE num_boleto = id_Boleto) *
      cantidad_Boletos);
```

Listing 42: Tablas para la BdDatos

- Instrucción Insert o Update que permita evidenciar que la restricción esta funcionando.

```
1 INSERT INTO Transacciones (num_Transaccion, fecha_transaccion,
   id_Cliente,
2                               id_Boleto, cantidad_Boletos,
                               total_Transaccion)
3 VALUES (2, CURRENT_TIMESTAMP, 'C123', 1, 2, 150);
```

Listing 43: Tablas para la BdDatos

- Captura de pantalla con el resultado de la instrucción que muestre que la restricción está funcionando.

4.10. Consultas

Plantea 3 consultas que consideres relevantes para la base de datos propuesta. Para cada consulta planteada, incluir en el reporte los siguientes incisos:

- Redacción clara de la consulta.
- Código en lenguaje SQL de la consulta.
- Ejecutar la consulta en Postgres e incluir una captura de pantalla con el resultado de la consulta.

4.11. Vistas

Plantea 3 vistas que consideres relevantes para la base de datos propuesta. Para cada vista planteada, incluir en el reporte los siguientes incisos:

- Redacción clara de la vista planteada.
- Código en lenguaje SQL que permita crear la vista solicitada.
- Ejecutar el código para la creación de la vista en Postgres e incluir una captura de pantalla con la vista creada satisfactoriamente.
- Incluir un ejemplo que los evaluadores puedan ejecutar para verificar el funcionamiento de las vistas.