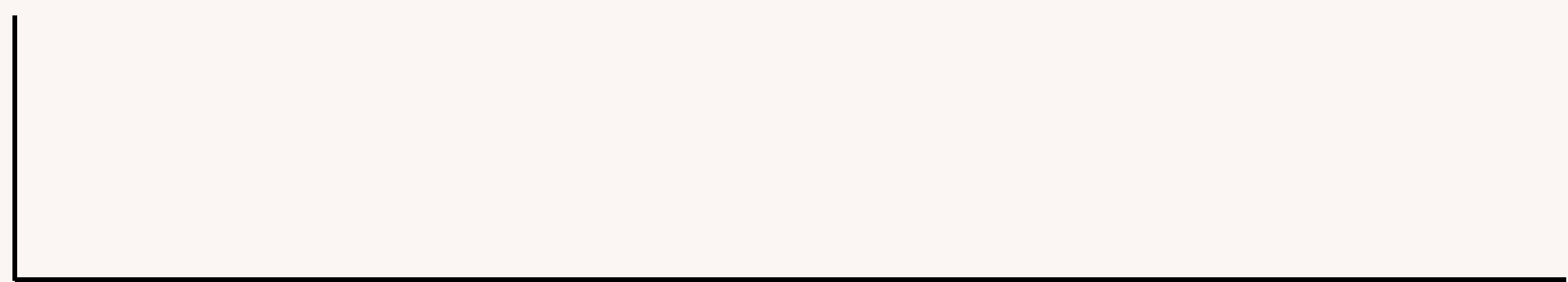


# AGE OF EMPIRES 3

Base de Dados



# INDEX :

---

1. INTRODUÇÃO

2. DER

3. ER

4. DDL

5. DML

A. SELECT

B. INSERT

C. UPDATE

D. DELETE

6. IMPROVEMENTS

A. VIEWS

B. INDEXES

C. STORED PROCEDURES

D. FUNCTIONS

E. TRIGGERS

# INTRODUÇÃO

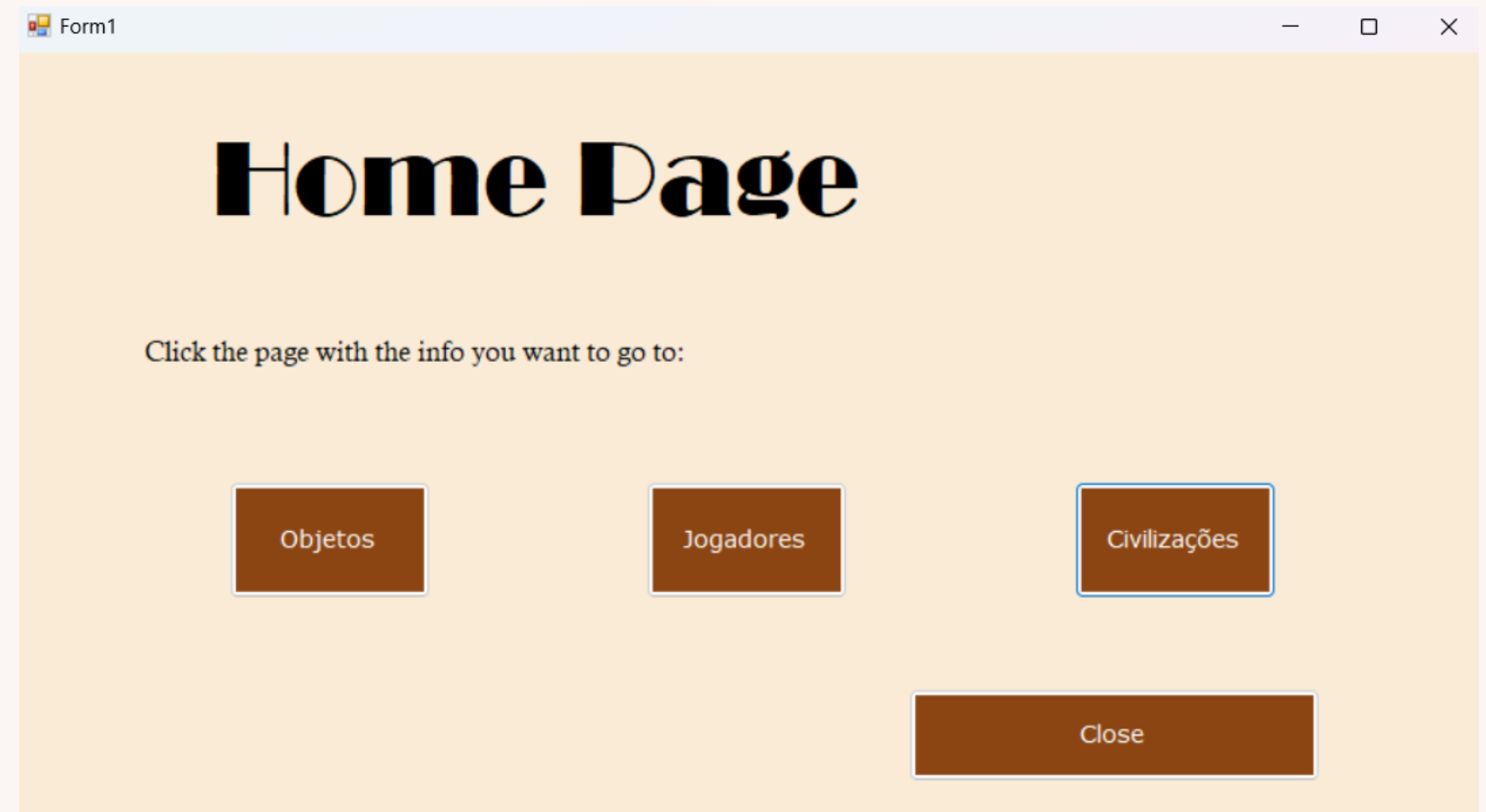
BASE DOS DADOS APÓS UMA BATALHA

COMO GESTOR DESSES DADOS POSSO:

ACRESCENTAR/EDITAR/REMOVER OBJETOS

ACRESCENTAR/EDITAR/REMOVER JOGADORES

ACRESCENTAR/EDITAR/REMOVER CIVILIZAÇÕES



# INTRODUÇÃO

Objetos

ID

32

Nome

Prop31

Localização X

52

Localização Y

11

Era

Discovery Age

Jogador

EAGLEMUT

Jog Eliminado

KaiserKlein

Add

Edit

Remove

Return

Close

Clear

Search

25 Prop24 ( 11, 52 )

26 Prop25 ( 37, 65 )

27 Prop26 ( 48, 42 )

28 Prop27 ( 38, 21 )

29 Prop28 ( 94, 2 )

30 Prop29 ( 1, 96 )

31 Prop30 ( 36, 41 )

32 Prop31 ( 52, 11 )

33 Prop32 ( 23, 20 )

34 Prop33 ( 37, 32 )

35 Prop34 ( 25, 0 )

36 Prop35 ( 6, 85 )

37 Prop36 ( 14, 10 )

38 Prop37 ( 0, 18 )

39 Prop38 ( 45, 25 )

40 Prop39 ( 54, 58 )

41 Prop40 ( 12, 69 )

42 Prop41 ( 81, 93 )

43 Prop42 ( 49, 91 )

44 Prop43 ( 67, 57 )

45 Prop44 ( 16, 22 )

46 Prop45 ( 30, 27 )

Civilizações

ID

6

Nome

Portuguese

Tipo

Grande Civilização

Lider

Henry the Navig

Capital

Lisbon

Add

Edit

Remove

Return

Close

Todas as Civilizações

Clear

Search

0 Gaia

1 British

2 Dutch

3 French

4 Germans

5 Ottomans

6 Portuguese

7 Russians

8 Spanish

9 Aztecs

10 Carib

11 Cherokee

12 Comanche

13 Cree

14 Quechua

15 Iroquois

16 Lakota

Jogadores

ID

6

Nome

KaiserKlein

Clan

NNN

Cor

Brown

Grande Civ

Dutch

Era

Fortress Age

Equipa

1

Add

Edit

Remove

Return

Close

Clear

Search

1 [ AOEC ] lamgrunt

2 [ AOEC ] EAGLEMUT

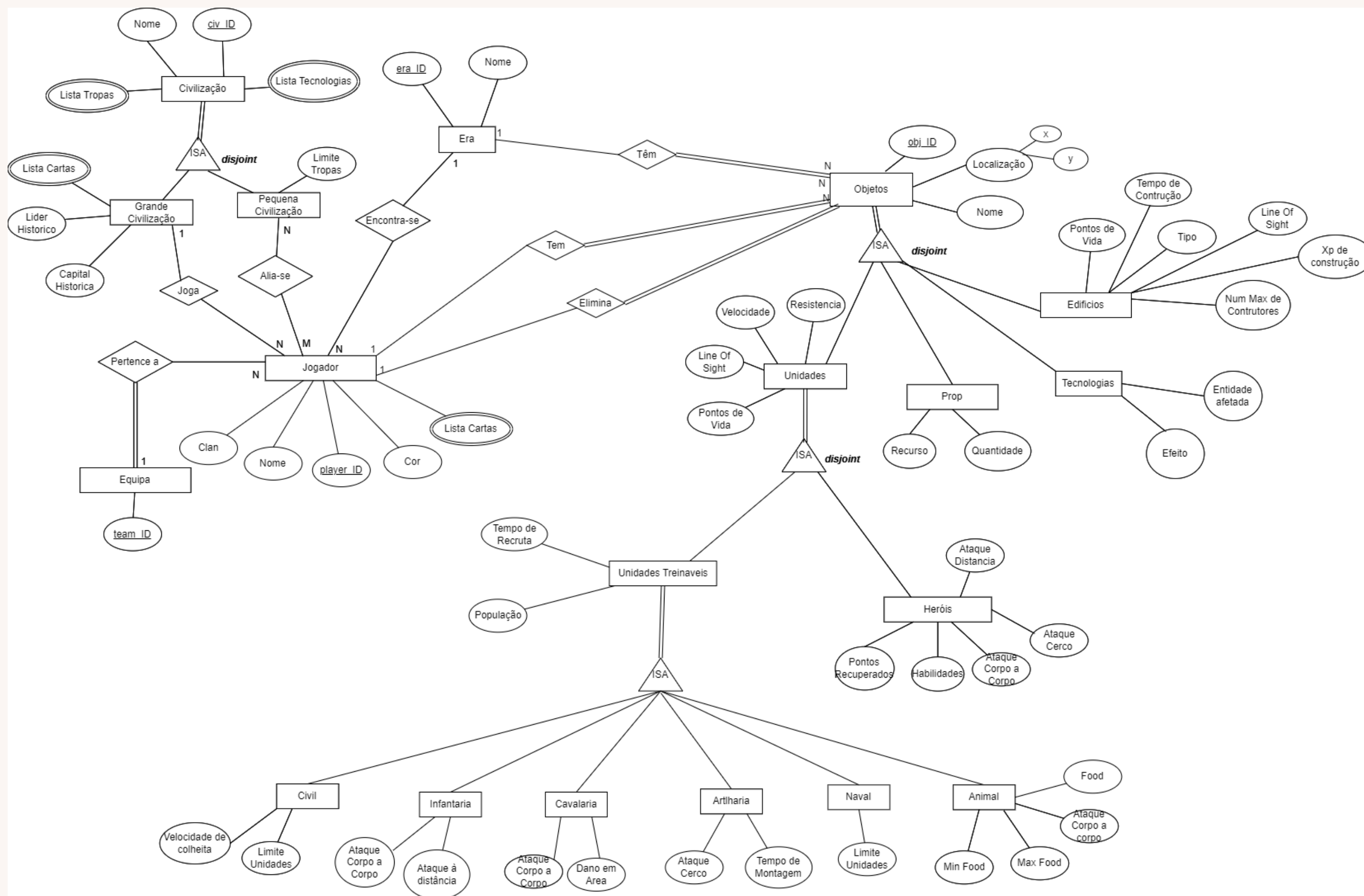
3 MusketJr

4 [ ESOC ] H2O

5 [ NNN ] Aizamk

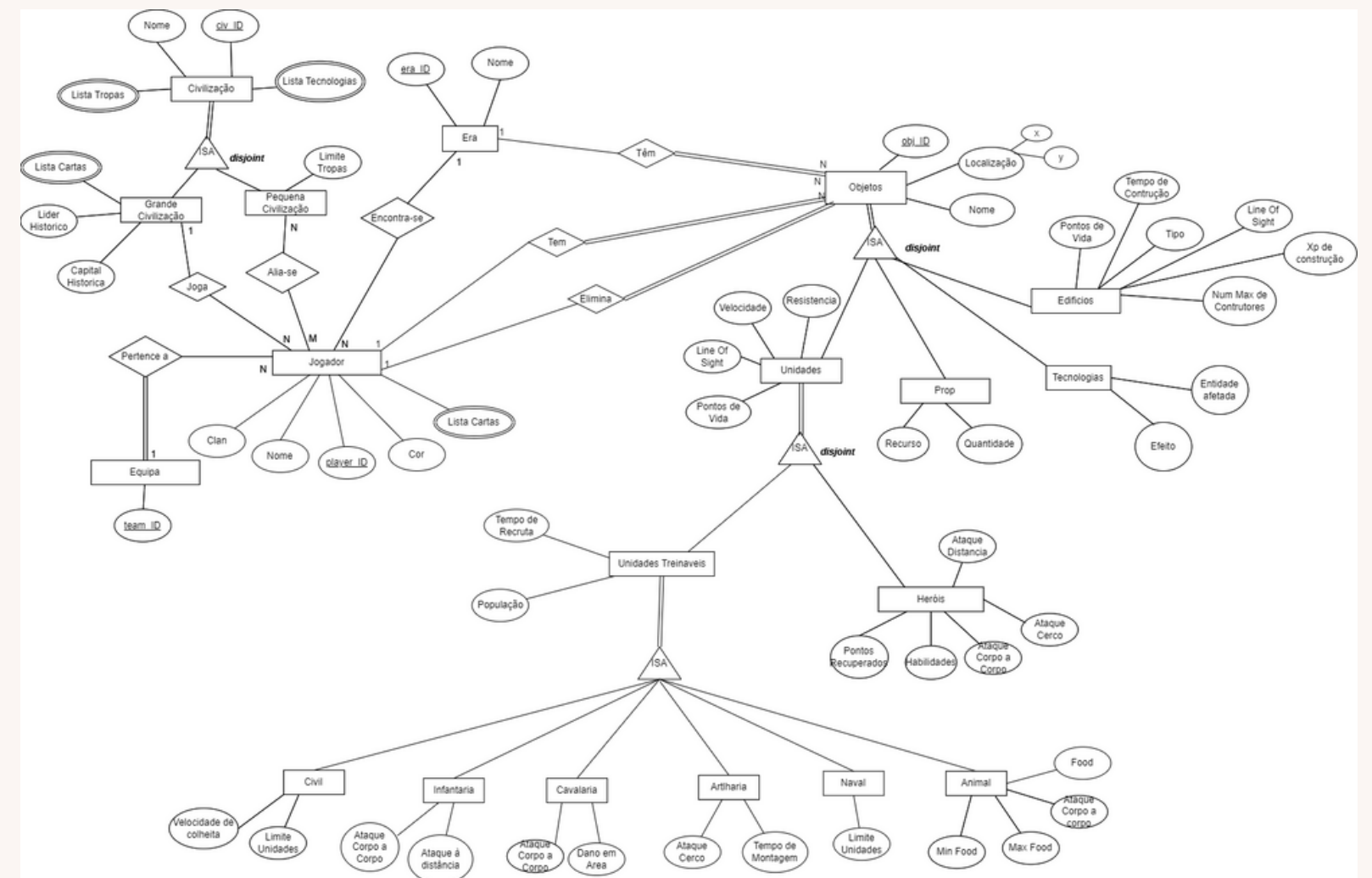
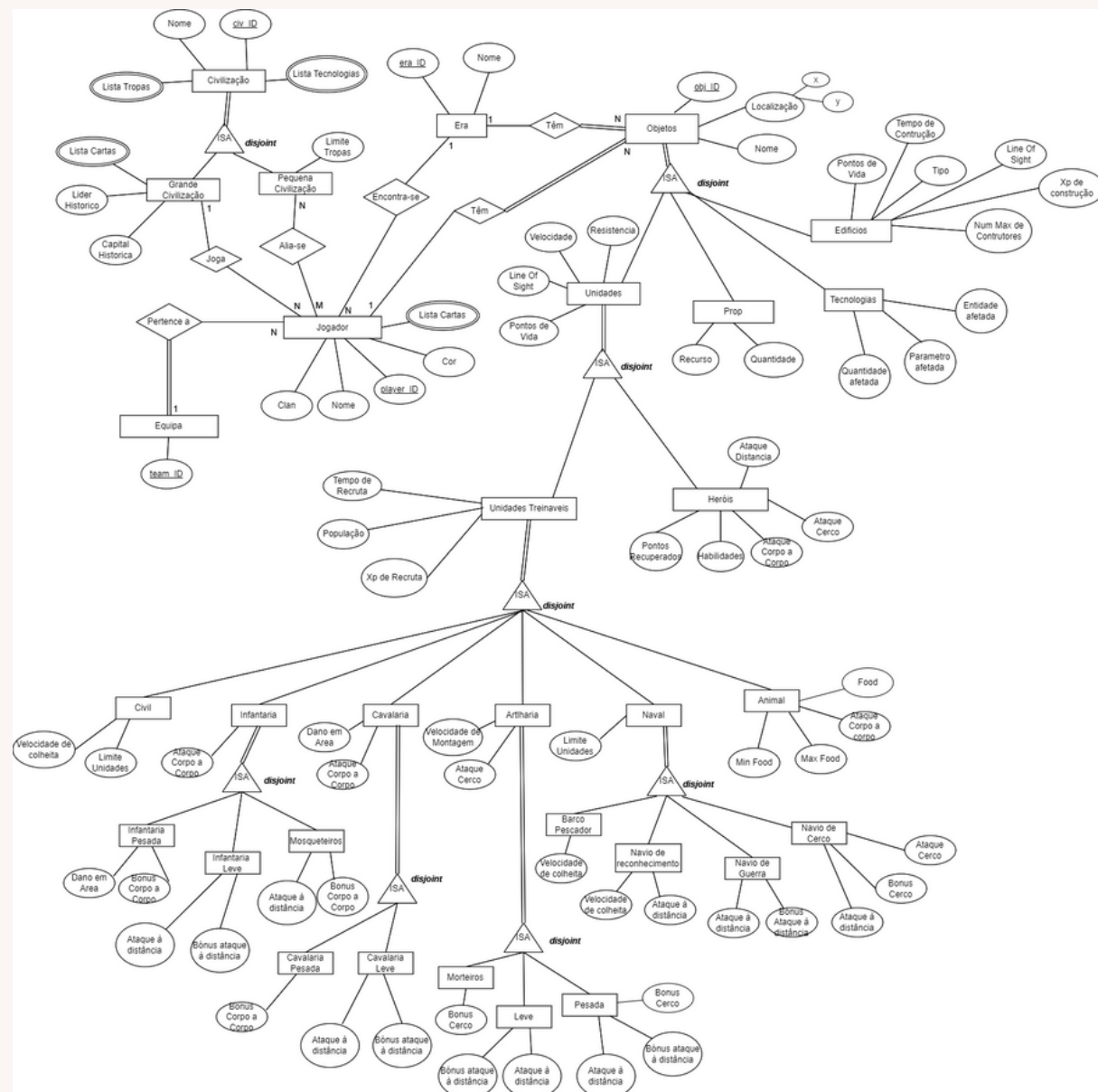
6 [ NNN ] KaiserKlein

# DER

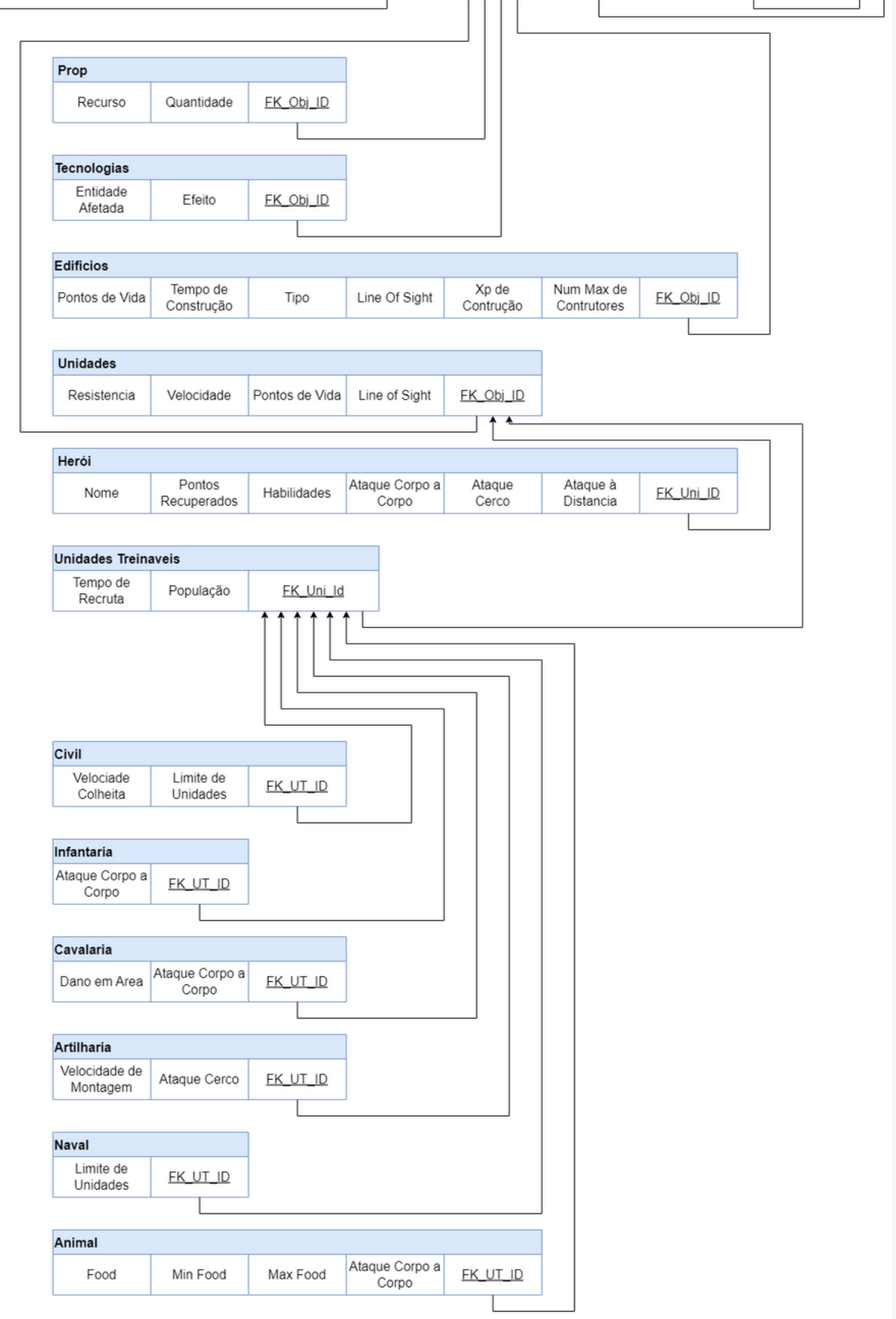
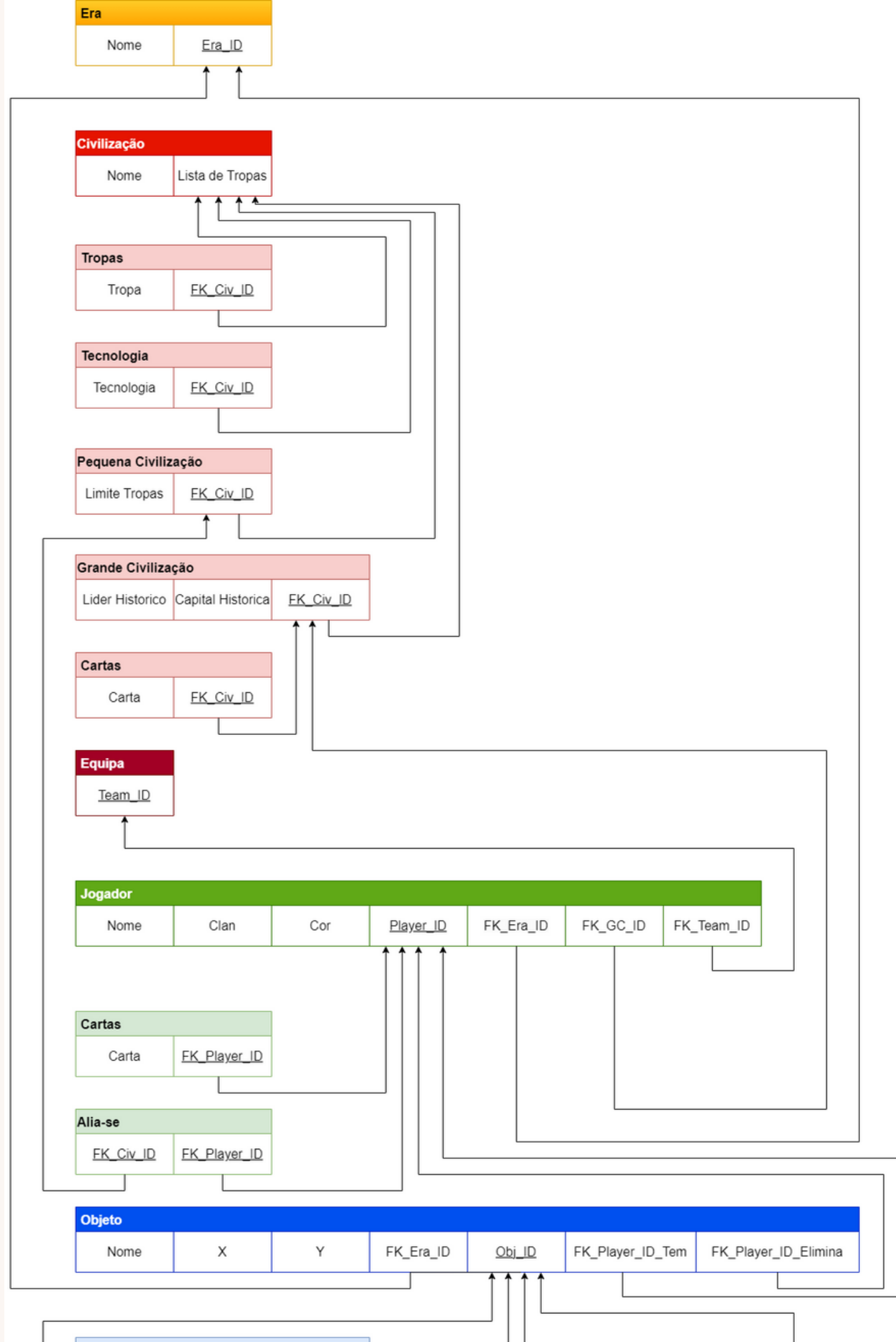


# DER

## BEFORE AND AFTER



# ER





# DDL

```
/* ***** SCHEMA ***** */
-- CREATE THE SCHEMA
IF NOT EXISTS (SELECT * FROM SYS.SCHEMAS WHERE NAME =
'EMPIRES')
BEGIN
EXEC('CREATE SCHEMA EMPIRES')
END

GO

/* ***** TABLES ***** */

-- ERA
CREATE TABLE EMPIRES.ERA (
ERA_ID INT,
NOME VARCHAR(50) NOT NULL,
PRIMARY KEY (ERA_ID),
);

GO

-- CIVILIZACAO
CREATE TABLE EMPIRES.CIVILIZACAO (
-- ENUM??
CIV_ID INT,
NOME VARCHAR(50) NOT NULL,

PRIMARY KEY (CIV_ID),
);

GO

CREATE TABLE EMPIRES.CIVILIZACAO_TROPAS (
TROPA VARCHAR(50),
FK_CIV_ID INT,

CONSTRAINT PK_CIVILIZACAO_TROPAS PRIMARY KEY
(FK_CIV_ID, TROPA),
FOREIGN KEY (FK_CIV_ID) REFERENCES
EMPIRES.CIVILIZACAO(CIV_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

-- PEQUENA CIVILIZACAO
CREATE TABLE EMPIRES.PEQUENA_CIVILIZACAO (
LIMITE_TROPAS INT NOT NULL,
FK_CIV_ID INT,

PRIMARY KEY (FK_CIV_ID),
FOREIGN KEY (FK_CIV_ID) REFERENCES
EMPIRES.CIVILIZACAO(CIV_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

-- GRANDE CIVILIZACAO
CREATE TABLE EMPIRES.GRANDE_CIVILIZACAO (
LIDER VARCHAR(50) NOT NULL,
CAPITAL VARCHAR(50) NOT NULL,
FK_CIV_ID INT,
PRIMARY KEY (FK_CIV_ID),
FOREIGN KEY (FK_CIV_ID) REFERENCES
EMPIRES.CIVILIZACAO(CIV_ID)
ON DELETE CASCADE ON UPDATE CASCADE
);

GO
```

```
CREATE TABLE EMPIRES.CARTAS_CIVILIZACAO (
CARTA VARCHAR(50) NOT NULL,
FK_CIV_ID INT,

CONSTRAINT PK_CARTAS_CIVILIZACAO PRIMARY KEY (FK_CIV_ID, CARTA),
FOREIGN KEY (FK_CIV_ID) REFERENCES EMPIRES.CIVILIZACAO(CIV_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

-- EQUIPA
CREATE TABLE EMPIRES.EQUIPA (
EQUIPA_ID INT
PRIMARY KEY (EQUIPA_ID)
);

GO

-- JOGADOR
CREATE TABLE EMPIRES.JOGADOR (
JOGADOR_ID INT,
NOME VARCHAR(50) NOT NULL,
CLAN VARCHAR(50),
COR VARCHAR(50),
FK_ERA_ID INT NOT NULL,
FK_GRANDE_ID INT NOT NULL,
FK_EQUIPA_ID INT NOT NULL,

PRIMARY KEY (JOGADOR_ID),
FOREIGN KEY (FK_ERA_ID) REFERENCES EMPIRES.ERA(ERA_ID),
FOREIGN KEY (FK_GRANDE_ID) REFERENCES
EMPIRES.GRANDE_CIVILIZACAO(FK_CIV_ID),
FOREIGN KEY (FK_EQUIPA_ID) REFERENCES EMPIRES.EQUIPA(EQUIPA_ID),
);

GO

CREATE TABLE EMPIRES.CARTAS_JOGADOR (
CARTA VARCHAR(50) NOT NULL,
FK_JOGADOR_ID INT,

CONSTRAINT PK_CARTAS_JOGADOR PRIMARY KEY (FK_JOGADOR_ID, CARTA),
FOREIGN KEY (FK_JOGADOR_ID) REFERENCES EMPIRES.JOGADOR(JOGADOR_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.ALIA (
FK_CIV_ID INT,
FK_JOGADOR_ID INT,

PRIMARY KEY (FK_CIV_ID, FK_JOGADOR_ID),
FOREIGN KEY (FK_CIV_ID) REFERENCES EMPIRES.CIVILIZACAO(CIV_ID),
FOREIGN KEY (FK_JOGADOR_ID) REFERENCES EMPIRES.JOGADOR(JOGADOR_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.OBJETO (
NOME VARCHAR(50) NOT NULL,
LOCALIZACAO_X INT,
LOCALIZACAO_Y INT,
OBJ_ID INT,
FK_ERA_ID INT NOT NULL,
FK_JOGADOR_ID_TEM INT,
FK_JOGADOR_ID_ELIMINA INT,

PRIMARY KEY (OBJ_ID),
FOREIGN KEY (FK_ERA_ID) REFERENCES EMPIRES.ERA (ERA_ID),
FOREIGN KEY (FK_JOGADOR_ID_TEM) REFERENCES EMPIRES.JOGADOR
(JOGADOR_ID),
FOREIGN KEY (FK_JOGADOR_ID_ELIMINA) REFERENCES EMPIRES.JOGADOR
(JOGADOR_ID),
);

GO
```

```
CREATE TABLE EMPIRES.PROP (
RECURSO VARCHAR(50),
QUANTIDADE INT,
FK_OBJ_ID INT,
PRIMARY KEY (FK_OBJ_ID),
FOREIGN KEY (FK_OBJ_ID) REFERENCES EMPIRES.OBJETO
(OBJ_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.TECNOLOGIA (
ENTIDADE VARCHAR(50) NOT NULL,
EFEITO VARCHAR(50) NOT NULL,
FK_OBJ_ID INT,
PRIMARY KEY (FK_OBJ_ID),
FOREIGN KEY (FK_OBJ_ID) REFERENCES EMPIRES.OBJETO
(OBJ_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.EDIFICIOS (
PONTOS_DE_VIDA INT NOT NULL,
TEMPO_DE_CONSTRUCAO INT NOT NULL,
TIPO VARCHAR(50) NOT NULL,
LINE_OF_SIGHT INT,
XP_DE_CONSTRUCAO INT NOT NULL,
N_MAX_DE_CONSTRUTORES INT,
FK_OBJ_ID INT,
PRIMARY KEY (FK_OBJ_ID),
FOREIGN KEY (FK_OBJ_ID) REFERENCES EMPIRES.OBJETO
(OBJ_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.UNIDADES (
PONTOS_DE_VIDA INT NOT NULL,
VELOCIDADE INT NOT NULL,
RESISTENCIA INT,
LINE_OF_SIGHT INT,
FK_OBJ_ID INT NOT NULL,
PRIMARY KEY (FK_OBJ_ID),
FOREIGN KEY (FK_OBJ_ID) REFERENCES EMPIRES.OBJETO
(OBJ_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.HEROI (
PONTOS_RECUPERADOS INT,
HABILIDADES VARCHAR(50),
ATAQUE_CORPO_A_CORPO INT,
ATAQUE_CERCO INT,
ATAQUE_A_DISTANCIA INT,
FK_UNI_ID INT NOT NULL,
PRIMARY KEY (FK_UNI_ID),
FOREIGN KEY (FK_UNI_ID) REFERENCES EMPIRES.UNIDADES
(FK_OBJ_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.UNIDADES_TREINAVEIS (
POPULACAO INT NOT NULL,
TEMPO_DE_RECRUTA INT NOT NULL,
FK_UNI_ID INT,
PRIMARY KEY (FK_UNI_ID),
FOREIGN KEY (FK_UNI_ID) REFERENCES EMPIRES.UNIDADES
(FK_OBJ_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO
```

```
CREATE TABLE EMPIRES.CIVIL (
VELOCIDADE_COLHEITA INT NOT NULL,
LIMITE_DE_UNIDADES INT NOT NULL,
FK_UT_ID INT,
PRIMARY KEY (FK_UT_ID),
FOREIGN KEY (FK_UT_ID) REFERENCES
EMPIRES.UNIDADES_TREINAVEIS (FK_UNI_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.INFANTARIA (
ATAQUE_CORPO_A_CORPO INT NOT NULL,
ATAQUE_A_DISTANCIA INT,
FK_UT_ID INT,
PRIMARY KEY (FK_UT_ID),
FOREIGN KEY (FK_UT_ID) REFERENCES
EMPIRES.UNIDADES_TREINAVEIS (FK_UNI_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.CAVALARIA (
DANO_EM_AREA INT NOT NULL,
ATAQUE_CORPO_A_CORPO INT NOT NULL,
FK_UT_ID INT,
PRIMARY KEY (FK_UT_ID),
FOREIGN KEY (FK_UT_ID) REFERENCES
EMPIRES.UNIDADES_TREINAVEIS (FK_UNI_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.ARTILHARIA (
VELOCIDADE_MONTAGEM INT NOT NULL,
ATAQUE_CERCO INT NOT NULL,
FK_UT_ID INT,
PRIMARY KEY (FK_UT_ID),
FOREIGN KEY (FK_UT_ID) REFERENCES
EMPIRES.UNIDADES_TREINAVEIS (FK_UNI_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.NAVAL (
LIMITE_DE_UNIDADES INT NOT NULL,
FK_UT_ID INT,
PRIMARY KEY (FK_UT_ID),
FOREIGN KEY (FK_UT_ID) REFERENCES
EMPIRES.UNIDADES_TREINAVEIS (FK_UNI_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO

CREATE TABLE EMPIRES.ANIMAL (
FOOD INT,
MIN_FOOD INT,
MAX_FOOD INT,
ATAQUE_CORPO_A_CORPO INT,
FK_UT_ID INT,
PRIMARY KEY (FK_UT_ID),
FOREIGN KEY (FK_UT_ID) REFERENCES
EMPIRES.UNIDADES_TREINAVEIS (FK_UNI_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
);

GO
```



# DML

Form3

## Jogadores

Clear Search

ID

Nome

Clan

Cor

Grande Civ

Era

Equipa

Add Edit Remove

Return Close

Form4

## Objetos

ID

Nome

Localização X

Localização Y

Era

Jogador

Jog Eliminado

Add Edit Remove

Return Close

Form2

## Civilizações

Todas as Civilizações

Clear Search

ID

Nome

Tipo

Lider

Capital

Add Edit Remove

Return Close

2 Dutch

4 Germans

5 Ottomans

6 Portuguese

8 Spanish

9 Estaues

10 Carib

11 Cherokee

13 Cree

14 Quechua

16 Lakota

18 Nootka

19 Seminole

# DML

## SELECTION

The screenshot shows a Windows application window titled "Form2". The main area is titled "Civilizações". It contains several input fields and buttons. The "ID" field has the value "5". The "Nome" field has the value "Ottomans". The "Tipo" dropdown menu is set to "Grande Civilização". The "Lider" field has the value "Sultan Suleimar". The "Capital" field has the value "Istanbul". There are buttons for "Add", "Edit", "Remove", "Return", and "Close". A dropdown menu on the right shows a list of civilizations: 2 Dutch, 4 Germans, 5 Ottomans (selected), 6 Portuguese, 8 Spanish, 9 Estaves, 10 Carib, 11 Cherokee, 13 Cree, 14 Quechua, 16 Lakota, 18 Nootka, and 19 Seminole. Above this list is a search bar with the text "Todas as Civilizações" and buttons for "Clear" and "Search".

```
SELECT *  
FROM Empires.Civilizacao INNER JOIN  
Empires.Grande_Civilizacao  
ON Civilizacao.civ_id = Grande_Civilizacao.FK_civ_id;
```

```
SELECT *  
FROM Empires.Civilizacao INNER JOIN  
Empires.Pequena_Civilizacao  
ON Civilizacao.civ_id = Pequena_Civilizacao.FK_civ_id;
```

```
SELECT *  
FROM (Empires.Civilizacao LEFT OUTER JOIN  
Empires.Grande_Civilizacao  
ON Civilizacao.civ_id = Grande_Civilizacao.FK_civ_id)  
LEFT OUTER JOIN Empires.Pequena_Civilizacao  
ON Civilizacao.civ_id = Pequena_Civilizacao.FK_civ_id ;
```

# DML

## INSERTION

Form2

Civilizações

ID: 5 Nome: Ottomans

Tipo: Grande Civilização

Lider: Sultan Suleimar Capital: Istanbul

Clear Search

2 Dutch  
4 Germans  
5 Ottomans  
6 Portuguese  
8 Spanish  
9 Estaves  
10 Carib  
11 Cherokee  
13 Cree  
14 Quechua  
16 Lakota  
18 Nootka  
19 Seminole

Add Edit Remove

Return Close

```
INSERT INTO Empires.Civilizacao  
    (civ_id, nome)  
VALUES (@civ_id, @nome)
```

```
INSERT INTO Empires.Grande_Civilizacao  
    (FK_civ_id, lider, capital)  
VALUES (@FK_civ_id, @lider, @capital)
```

```
INSERT INTO Empires.Pequena_Civilizacao  
    (FK_civ_id, limite_tropas)  
VALUES (@FK_civ_id, @limite_tropas)
```

# DML

## ALTERATION

Form2

### Civilizações

ID: 5 Nome: Ottomans

Tipo: Grande Civilização

Lider: Sultan Suleimar Capital: Istanbul

Clear Search

2 Dutch  
4 Germans  
5 Ottomans  
6 Portuguese  
8 Spanish  
9 Estaves  
10 Carib  
11 Cherokee  
13 Cree  
14 Quechua  
16 Lakota  
18 Nootka  
19 Seminole

Add Edit Remove

Return Close

```
UPDATE Empires.Civilizacao  
SET nome = @novo_nome  
WHERE civ_ID = @civ_id
```

# DML

## DELETION

Form2

### Civilizações

ID: 5 Nome: Ottomans

Tipo: Grande Civilização

Lider: Sultan Suleimar Capital: Istanbul

Clear Search

2 Dutch  
4 Germans  
5 Ottomans  
6 Portuguese  
8 Spanish  
9 Estaves  
10 Carib  
11 Cherokee  
13 Cree  
14 Quechua  
16 Lakota  
18 Nootka  
19 Seminole

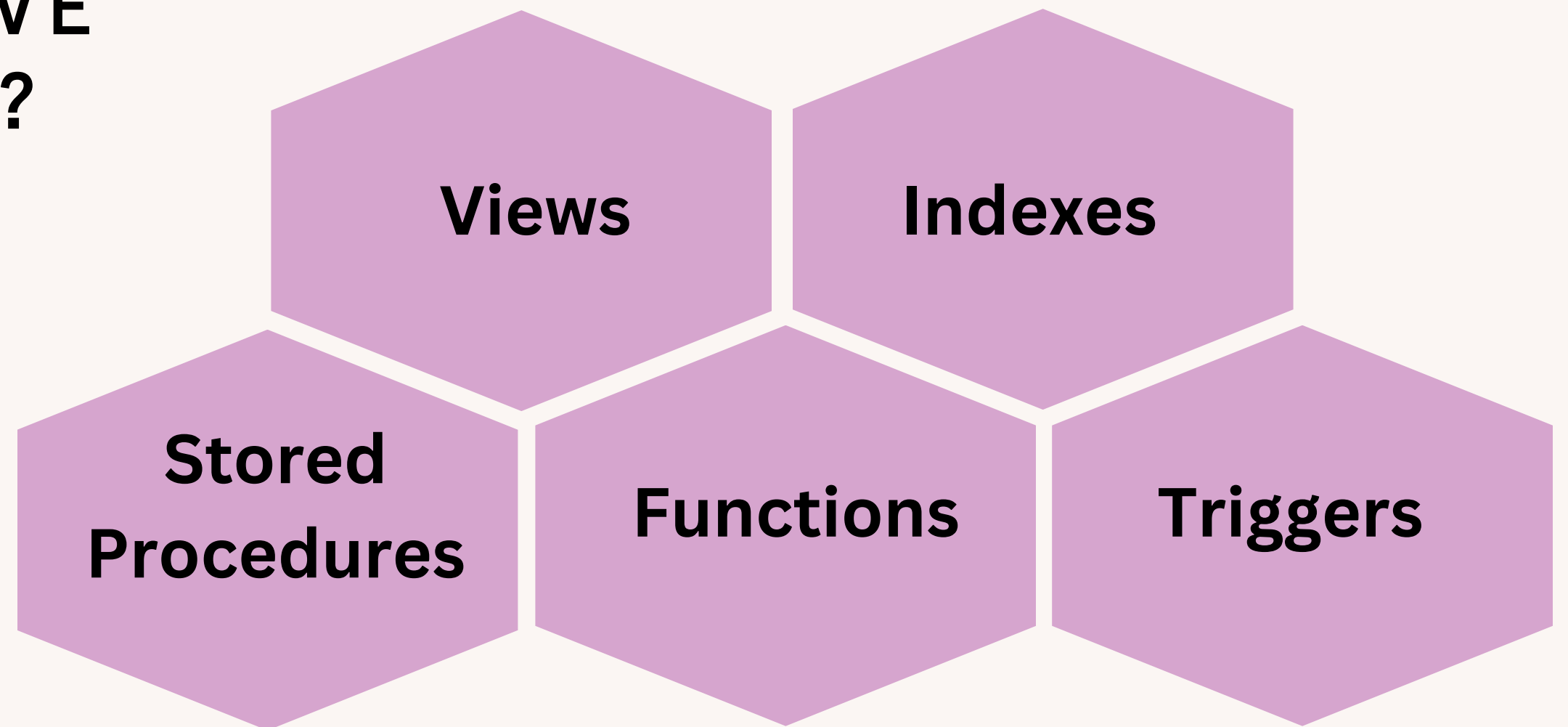
Add Edit Remove

Return Close

**DELETE FROM Empires.Civilizacao**  
**WHERE civ\_id = @civ\_id**

# IMPROVEMENTS

HOW CAN WE  
FURTHER IMPROVE  
OUR DATABASE?





# VIEWS

```
CREATE VIEW Empires.Todas_Civilizacao AS
SELECT civ_id AS ID, nome AS Nome, lider AS Lider, capital
AS Capital, limite_tropas AS 'Limite Tropas'
FROM (Empires.Civilizacao LEFT OUTER JOIN
Empires.Grande_Civilizacao
ON Civilizacao.civ_id = Grande_Civilizacao.FK_civ_id)
LEFT OUTER JOIN Empires.Pequena_Civilizacao
ON Civilizacao.civ_id = Pequena_Civilizacao.FK_civ_id;
```

```
SELECT *
FROM (Empires.Civilizacao LEFT OUTER JOIN
Empires.Grande_Civilizacao
ON Civilizacao.civ_id = Grande_Civilizacao.FK_civ_id)
LEFT OUTER JOIN Empires.Pequena_Civilizacao
ON Civilizacao.civ_id = Pequena_Civilizacao.FK_civ_id ;
```

```
SELECT *
FROM Empires.Todas_Civilizacao
```

# VIEWS

```
CREATE VIEW Empires.Jogadores_Objects AS  
SELECT jogador_id AS ID, Objeto.Nome AS OBJETO, Objeto.Obj_Id AS  
OBJ_ID  
FROM Empires.Jogador LEFT OUTER JOIN Empires.Objeto  
ON Jogador.jogador_id = Objeto.FK_jogador_id_tem  
GROUP BY jogador_id, Objeto.Nome, Objeto.Obj_Id
```

```
CREATE VIEW Empires.Jogadores_Full_Info AS  
SELECT jogador_id AS ID, Jogador.nome Nome, clan AS Clan, cor  
AS Cor, Era.nome AS Era, Civilizacao.nome AS Civilizacao  
FROM (Empires.Jogador LEFT OUTER JOIN Empires.Era  
ON Jogador.FK_era_id = Era.era_id)  
LEFT OUTER JOIN Empires.Civilizacao  
ON Jogador.FK_grande_id = Civilizacao.civ_id
```

# INDEXES

```
CREATE INDEX idxEra    ON Empires.Era(era_id);    -- Primary Key
CREATE INDEX idxCiv    ON Empires.Civilizacao(civ_id);    -- Primary Key
CREATE INDEX idxCivTroops  ON Empires.Civilizacao_Tropas(FK_civ_id); -- Foreign Key
CREATE INDEX idxSmallCiv  ON Empires.Pequena_Civilizacao(FK_civ_id); -- Foreign Key
CREATE INDEX idxBigCiv    ON Empires.Grande_Civilizacao(FK_civ_id); -- Foreign Key
CREATE INDEX idxCivCards  ON Empires.Cartas_Civilizacao(FK_civ_id); -- Foreign Key
CREATE INDEX idxTeam     ON Empires.Equipa(equipa_id);    -- Primary Key
CREATE INDEX idxPlayer   ON Empires.Jogador(jogador_id);  -- Primary Key
CREATE INDEX idxPlayersEra ON Empires.Jogador(FK_era_id);  -- Foreign Key
CREATE INDEX idxPlayerCards ON Empires.Cartas_Jogador(FK_jogador_id); -- Foreign Key
CREATE INDEX idxAllyCiv   ON Empires.Alia(FK_civ_id);     -- Foreign Key
CREATE INDEX idxAllyPlayer ON Empires.Alia(FK_jogador_id); -- Foreign Key
```

# INDEXES

```
CREATE INDEX idxObject  ON Empires.Objeto(Obj_Id);    -- Primary Key
CREATE INDEX idxObjectsEra  ON Empires.Objeto(FK_Era_Id);    -- Foreign Key
CREATE INDEX idxObjectsOwner  ON Empires.Objeto(FK_jogador_id_tem); -- Foreign Key
CREATE INDEX idxObjectsKiller  ON Empires.Objeto(FK_jogador_id_elimina); -- Foreign Key
CREATE INDEX idxProp  ON Empires.Prop(FK_Obj_Id);    -- Primary Key
CREATE INDEX idxTech  ON Empires.Tecnologia(FK_Obj_Id);    -- Primary Key
CREATE INDEX idxBuild  ON Empires.Edificios(FK_Obj_Id);    -- Primary Key
CREATE INDEX idxUnit  ON Empires.Unidades(FK_Obj_Id);    -- Primary Key
CREATE INDEX idxHero  ON Empires.Heroi(FK_Uni_Id);    -- Primary Key
CREATE INDEX idxTrain  ON Empires.Unidades_Treinaveis(FK_Uni_Id); -- Primary Key
CREATE INDEX idxCivil  ON Empires.Civil(FK_UT_Id);    -- Primary Key
CREATE INDEX idxInf  ON Empires.Infantaria(FK_UT_Id);    -- Primary Key
CREATE INDEX idxCav  ON Empires.Cavalaria(FK_UT_Id);    -- Primary Key
CREATE INDEX idxArt  ON Empires.Artilharia(FK_UT_Id);    -- Primary Key
CREATE INDEX idxNav  ON Empires.Naval(FK_UT_Id);    -- Primary Key
CREATE INDEX idxAni  ON Empires.Animal(FK_UT_Id);    -- Primary Key
```

# STORED PROCEDURES

```
-- PROCEDURE PARA INSERIR UMA  
NOVA CIVILIZACAO NA BASE DE DADOS
```

```
CREATE PROC Empires.InsertCivilizacao  
@Nome VARCHAR(50),  
@FK_grande_id INT  
AS  
BEGIN  
    SET NOCOUNT ON;  
    INSERT INTO Empires.Civilizacao (Nome, civ_id)  
    VALUES (@Nome, @FK_grande_id);  
END
```

# STORED PROCEDURES

```
-- PROCEDURE PARA ATUALIZAR  
UM JOGADOR NA BASE DE DADOS
```

```
CREATE PROC Empires.UpdateJogador  
@Nome VARCHAR(50),  
@FK_civ_id INT,  
@FK_equipa_id INT,  
@jogador_id INT  
AS  
BEGIN  
SET NOCOUNT ON;  
UPDATE Empires.Jogador  
SET Nome = @Nome,  
FK_grande_id = @FK_civ_id,  
FK_equipa_id = @FK_equipa_id  
WHERE jogador_id = @jogador_id;  
END
```



# FUNCTIONS

```
CREATE FUNCTION Empires.GetCivilizacaoCount()  
RETURNS INT  
AS  
BEGIN  
    DECLARE @Count INT;  
    SELECT @Count = COUNT(*) FROM Empires.Civilizacao;  
    RETURN @Count;  
END
```

```
CREATE FUNCTION Empires.GetTotalObjectsPerPlayer(@jogador_id INT)  
RETURNS INT  
AS  
BEGIN  
    DECLARE @TotalObjects INT;  
    SELECT @TotalObjects = COUNT(*)  
    FROM Empires.Objeto  
    WHERE FK_jogador_id_tem = @jogador_id OR FK_jogador_id_elimina = @jogador_id;  
    RETURN @TotalObjects;  
END
```

# FUNCTIONS

```
CREATE FUNCTION Empires.GetTotalCardsPerCivilizacao(@civ_id INT)
RETURNS INT
AS
BEGIN
    DECLARE @TotalCards INT;
    SELECT @TotalCards = COUNT(*)
    FROM Empires.Cartas_Civilizacao
    WHERE FK_civ_id = @civ_id;
    RETURN @TotalCards;
END
```

```
CREATE FUNCTION Empires.GetTotalCardsPerPlayer(@jogador_id INT)
RETURNS INT
AS
BEGIN
    DECLARE @TotalCards INT;
    SELECT @TotalCards = COUNT(*)
    FROM Empires.Cartas_Jogador
    WHERE FK_jogador_id = @jogador_id;
    RETURN @TotalCards;
END
```

# TRIGGERS

```
-- DELETAR JOGADORES RELACIONADOS  
À GRANDE CIVILIZACAO
```

```
CREATE TRIGGER trg_DeleteGrandeCiv  
ON Empires.Grande_Civilizacao  
AFTER DELETE  
AS  
BEGIN  
    SET NOCOUNT ON;  
  
    DELETE FROM Empires.Jogador  
    WHERE FK_grande_id IN (SELECT FK_civ_id FROM deleted);  
END;
```

# TRIGGERS

```
-- DELETAR OBJETOS  
RELACIONADOS AO JOGADOR
```

```
-- DELETAR O JOGADOR
```

```
CREATE TRIGGER trg_DeleteJogador  
ON Empires.Jogador  
INSTEAD OF DELETE  
AS  
BEGIN  
    SET NOCOUNT ON;  
  
    DELETE FROM Empires.Objeto  
    WHERE FK_jogador_id_tem IN (SELECT jogador_id FROM  
deleted)  
    OR FK_jogador_id_elimina IN (SELECT jogador_id FROM  
deleted);  
  
    DELETE FROM Empires.Jogador  
    WHERE jogador_id IN (SELECT jogador_id FROM deleted);  
END;
```