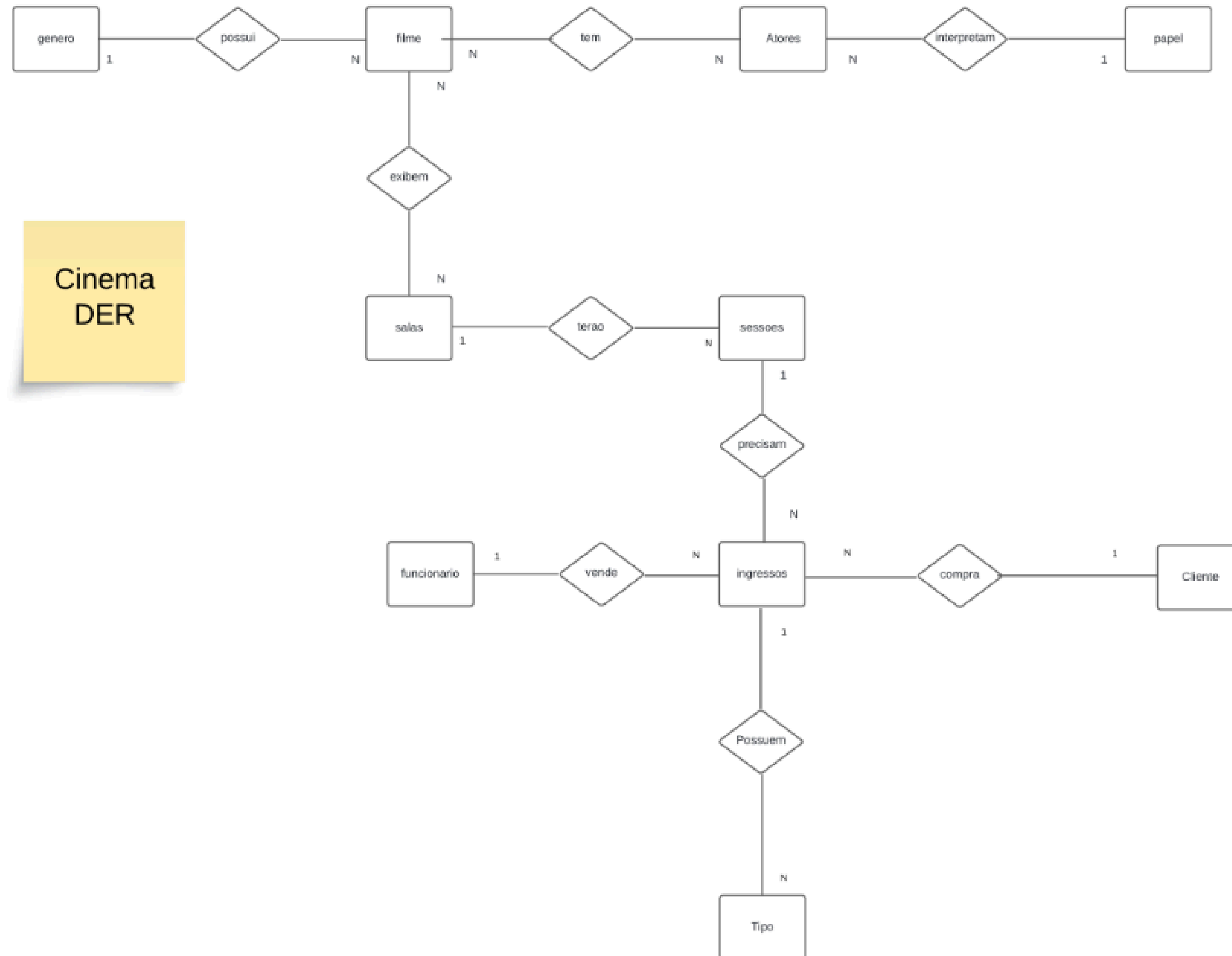


# Cinema

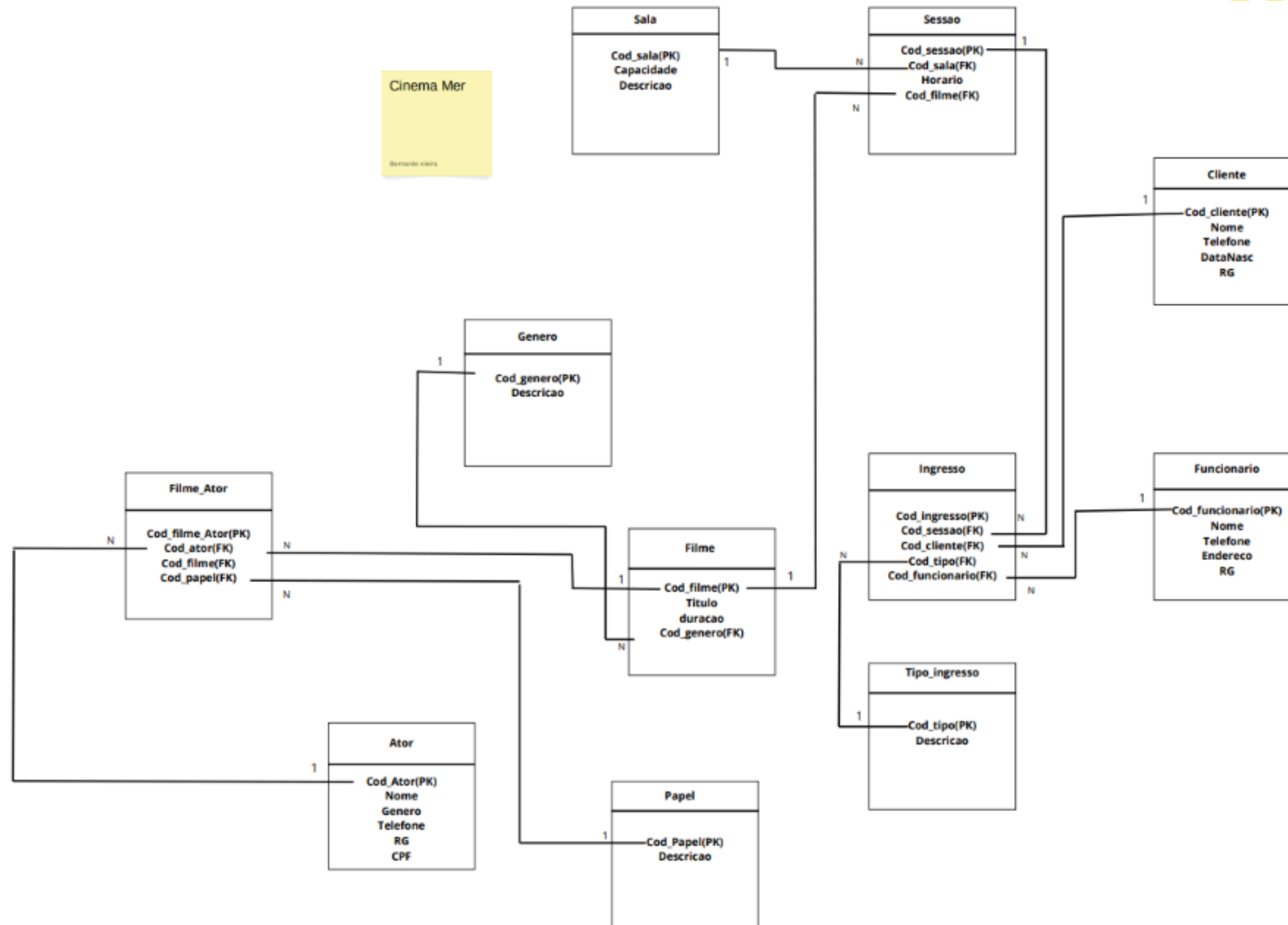
Projeto Banco de dados

# Der

Cinema  
DER



# MER



# Dicionário de Dados

## Ator

Tabela	Ator									
Descrição	Armazenará informações detalhadas sobre o Ator									
Observações	Está tabela vai armazenar as informações do Ator									
Campos										
Nome	Descrição				Tipo de dado		Tamanho		Restrições de dominio (PK...)	
Cod Ator	Identificação do Ator				int				PK, identity	
Nome	Nome do Ator				Varchar		30		Not null	
Genero	Genêro do Ator				Varchar		20		Not null	
Telefone	Telefone do Ator				Varchar		13		Not null	
RG		RG do Ator			Varchar		11		Not null	
CPF	CPF do Ator				Varchar		11		Not null	

# Dicionário de Dados

## Cliente

Tabela	Cliente			
Descrição	Armazenará informações do cliente.			
Observações	Está tabela não possui PK			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de dominio (PK...)
Cod cliente	Identificação do cliente	int		PK, identity
Nome	Nome do cliente	Varchar	30	Not null
Telefone	Telefone do Cliente	Varchar	13	Not null
DataNasc	Data do nascimento	Date		Not null
RG	RG do cliente	Varchar	10	null

# Dicionário de Dados

## Filme

Tabela	Filme			
Descrição	Armazenará informações detalhadas sobre o filme			
Observações	Contm uma FK			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de dominio (PK...)
Cod filme	Identificação do filme	int		PK, identity
Titulo	Titulo do filme	Varchar	30	Not null
Duração	Duração do filme	time		Not null
Cod genero	Identificação do genero	int		FK

# Dicionário de Dados

Filme\_Ator

Tabela	Filme_Ator			
Descrição	Armazenará informações detalhadas sobre o filme_ator			
Observações	Contém 3 FK's			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de domínio (PK...)
Cod filme ator	Identificação do filme ator	int		PK, identity
Cod Ator	Identificação do Ator	int		FK
Cod filme	Identificação do filme	int		FK
Cod papel	Identificação do papel	int		FK



# Dicionário de Dados

## Funcionário

Tabela	Funcionário			
Descrição	Armazenará informações detalhadas sobre os funcionários			
Observações	Esta tabela vai armazenar as informações do funcionário			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de domínio (PK...)
Cod funcionario	Identificação do funcionário	int		PK, identity
Nome	Nome do funcionário	Varchar	30	Not null
Telefone	Telefone do funcionário	Varchar	11	Not null
Endereço	Endereço do funcionário	Varchar	50	Not null
RG	RG do funcionário	Varchar	10	Not null



# Dicionário de Dados

## Genero

Tabela	Genero			
Descrição	Armazenará o gênero do fillme			
Observações	Esta tabela armazenará os dados do gênero			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de dominio (PK...)
Cod_genero	Identificação do gênero	int		PK, identity
Descrição	Descrição do gênero	Varchar	100	Not null

# Dicionário de Dados

## Ingresso

Tabela	Ingresso			
Descrição	Armazenará informações detalhadas sobre os ingressos			
Observações	Está tabela vai armazenar as informações do Ingresso			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de dominio (PK...)
Cod ingresso	Identificação do ingresso	int		PK, identity
Cod sessao	Identificação da sessão	int		FK
Cod cliente	Identificação do cliente	int		FK
Cod tipo	Identificação do tipo do ingresso	int		FK
Cod_funcionario	Identificação do funcionário	int		FK

# Dicionário de Dados

## Papel

Tabela	Papel			
Descrição	Armazenará o Papel			
Observações	Esta tabela armazenará os Pápeis			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de dominio (PK...)
Cod papel	Identificação do papel	int		PK, identity
Descricao	Descricao do papel	Varchar	100	Not null

# Dicionário de Dados

## Sala

	Sala			
Descrição	Armazenará os dados da sala			
Observações	Esta tabela armazenará os dados da sala			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de dominio (PK...)
Cod sala	Identificação do usuario	int		PK, Identity
Capacidade	Capacidade de Pessoas	int		
Descricao	Descrição da Sala	Varchar	30	null

# Dicionário de Dados

## Sessão

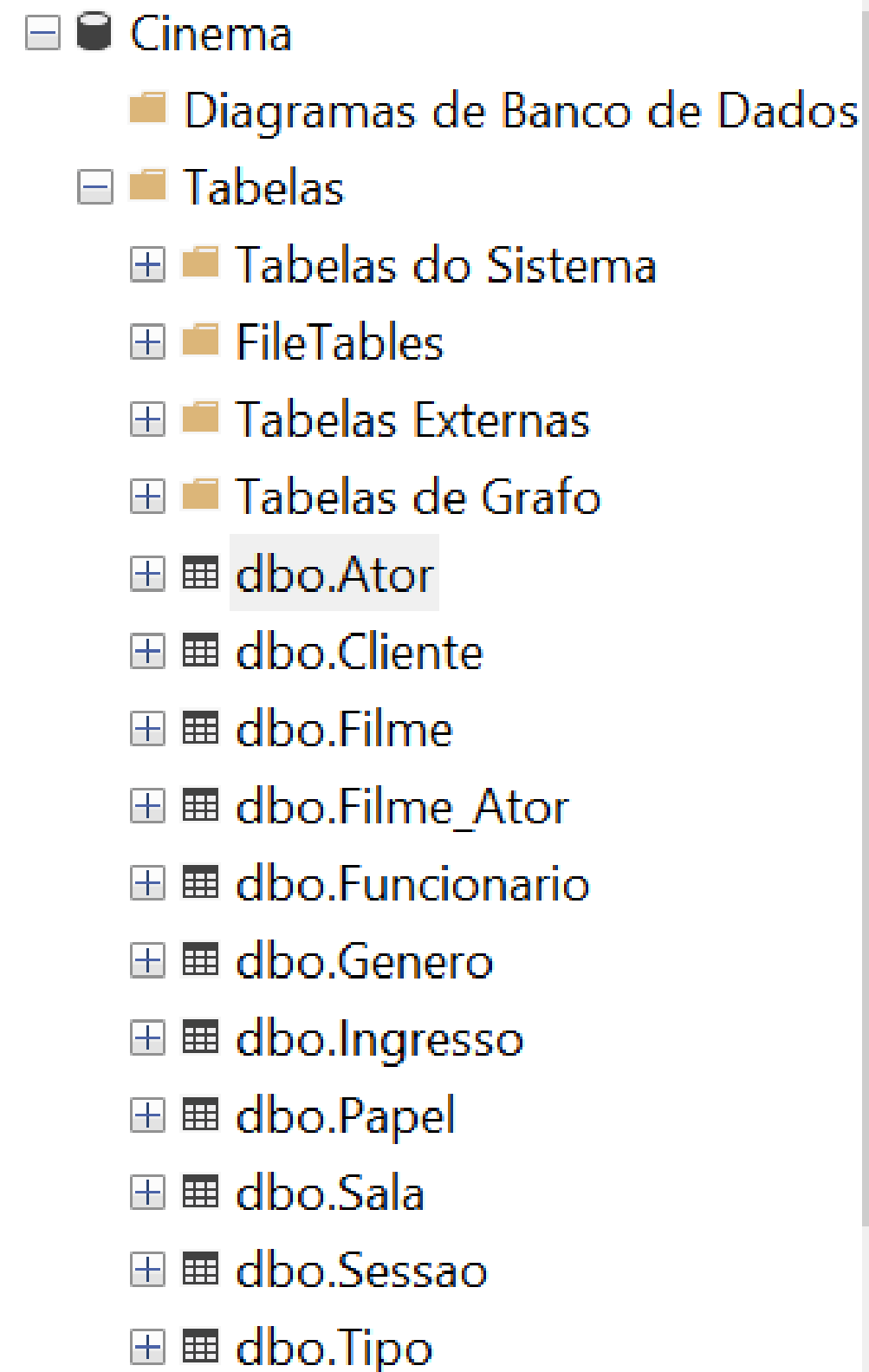
Tabela	Sessão			
Descrição	Armazenará todos os registros da Sessão			
Observações	Esta tabela armazenará os registros da sala			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de dominio (PK...)
Cod_sessao	Identificação da sessão	int		PK, identity
Cod_sala	Identificação da sala	int		FK
Horario	Horario da sessao	time		Not null
Cod_filme	Identificação do filme	int		FK

# Dicionário de Dados

Tipo\_Ingresso

Tabela	Tipo_ingresso			
Descrição	Armazenará o tipo de ingresso			
Observações	Esta tabela armazenará os tipos de ingresso			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de dominio (PK...)
Cod tipo	Identificação do tipo	int		PK, identity
Descrição	Descrição do tipo	Varchar	25	Not null

# Estrutura SSMS





# Estrutura 2

  **dbo.Ator**

 Colunas

- 🔑 Cod\_ator (PK, int, não nulo)
- 📄 Nome (varchar(30), não nulo)
- 📄 Genero (varchar(20), não nulo)
- 📄 Telefone (varchar(13), não nulo)
- 📄 RG (varchar(11), não nulo)
- 📄 CPF (varchar(11), não nulo)

# Estrutura 2

  dbo.Cliente

  Colunas

 Cod\_cliente (PK, int, não nulo)

 Nome (varchar(30), não nulo)

 Telefone (varchar(30), não nulo)

 DataNasc (date, nulo)

 RG (varchar(10), nulo)

# Estrutura 2

  dbo.Filme

 Colunas

🔑 Cod\_filme (PK, int, não nulo)

📄 Titulo (varchar(30), não nulo)





📄 duracao (time(7), nulo)

🔑 Cod\_genero (FK, int, nulo)

# Estrutura 2

  dbo.Filme\_Ator






 Colunas

-  Cod\_filme\_ator (PK, int, não nulo)
-  Cod\_ator (FK, int, nulo)
-  Cod\_filme (FK, int, nulo)
-  Cod\_papel (FK, int, nulo)

# Estrutura 2

  **dbo.Funcionario**

 **Colunas**

-  **Cod\_funcionario** (PK, int, não nulo)
-  **Nome** (varchar(30), não nulo)
-  **Telefone** (varchar(11), não nulo)
-  **Endereco** (varchar(50), não nulo)
-  **RG** (varchar(10), não nulo)

# Estrutura 2



  `dbo.Genero`

  `Colunas`

 `Cod_genero (PK, int, não nulo)`

 `Descricao (varchar(100), nulo)`

# Estrutura 2

  dbo.Ingresso

 Colunas

 Cod\_ingresso (PK, int, não nulo)

 Cod\_sessao (FK, int, nulo)

 Cod\_cliente (FK, int, nulo)

 Cod\_tipo (FK, int, nulo)

 Cod\_funcionario (FK, int, nulo)



# Estrutura 2

  dbo.Papel

 Colunas

 Cod\_papel (PK, int, não nulo)

 Descricao (varchar(100), não nulo)

# Estrutura 2

  dbo.Sala

 Colunas

 Cod\_sala (PK, int, não nulo)

 Capacidade (int, não nulo)

 Descricao (varchar(30), nulo)

# Estrutura 2

  **dbo.Sessao**

 **Colunas**

🔑 **Cod\_sessao** (PK, int, não nulo)

🔑 **Cod\_sala** (FK, int, nulo)

📅 **Horario** (time(7), nulo)

🔑 **Cod\_filme** (FK, int, nulo)

# Estrutura 2

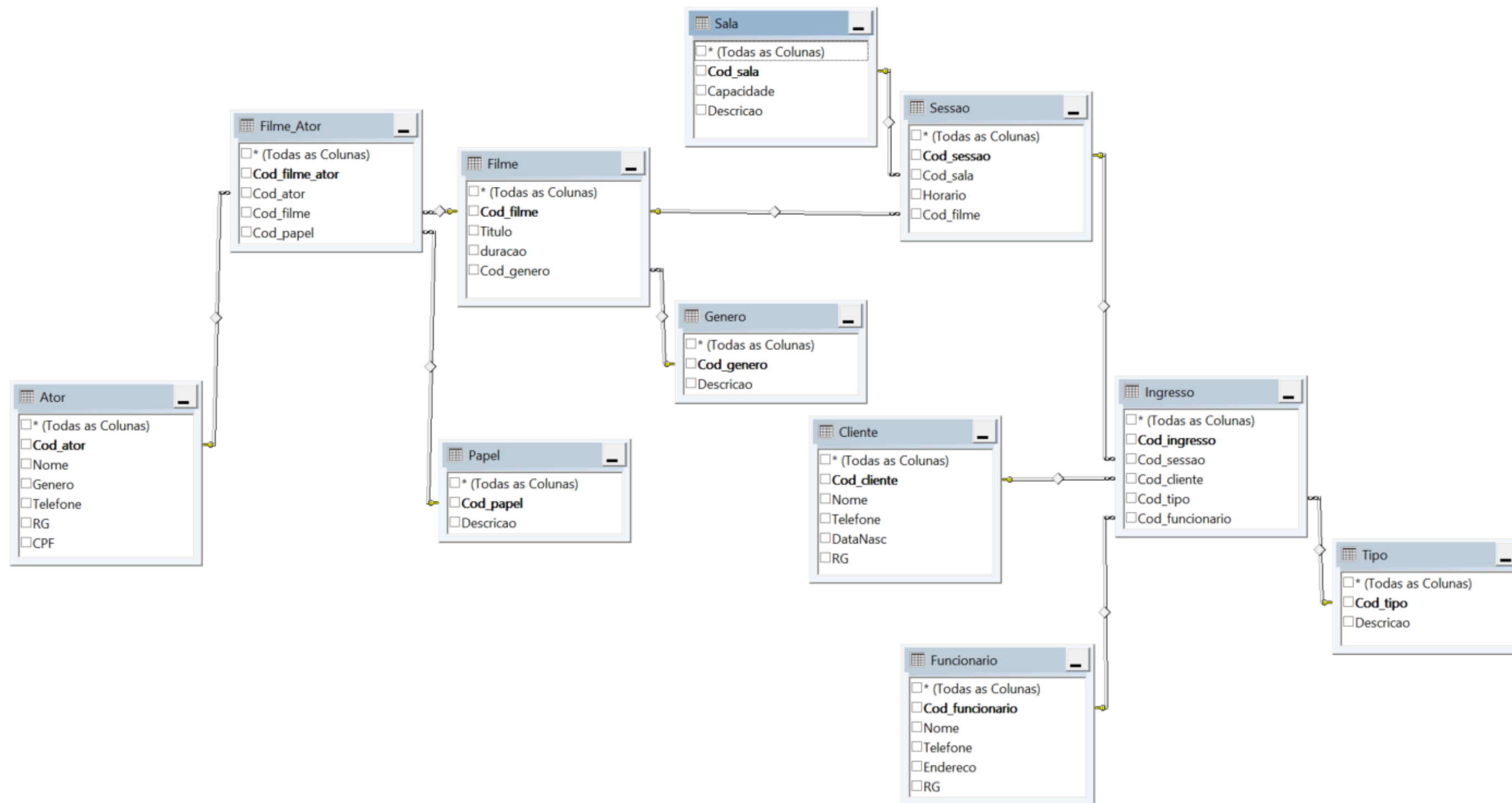
  dbo.Tipo

 Colunas

 Cod\_tipo (PK, int, não nulo)

 Descricao (varchar(25), não nulo)

# Mer (SSMS)



# Código gerado na Query

```
--Criando o Banco com o nome Cinema
CREATE DATABASE Cinema;

-- Abrir o banco cinema
use Cinema;

SELECT
FROM Papel INNER JOIN
    Ator INNER JOIN
        Filme_Ator ON Ator.Cod_ator = Filme_Ator.Cod_ator INNER JOIN
        Filme ON Filme_Ator.Cod_filme = Filme.Cod_filme INNER JOIN
        Genero ON Filme.Cod_genero = Genero.Cod_genero ON Papel.Cod_papel = Filme_Ator.Cod_papel INNER JOIN
        Sessao ON Filme.Cod_filme = Sessao.Cod_filme INNER JOIN
        Ingresso INNER JOIN
            Funcionario ON Ingresso.Cod_funcionario = Funcionario.Cod_funcionario INNER JOIN
            Cliente ON Ingresso.Cod_cliente = Cliente.Cod_cliente ON Sessao.Cod_sessao = Ingresso.Cod_sessao INNER JOIN
            Sala ON Sessao.Cod_sala = Sala.Cod_sala INNER JOIN
            Tipo ON Ingresso.Cod_tipo = Tipo.Cod_tipo

--Criando a tabela sala
CREATE TABLE Sala (
    Cod_sala int identity,
    Capacidade int NOT NULL,
    Descricao varchar (30) NULL,
    PRIMARY KEY (Cod_sala) --Definição de PK
);SELECT
FROM Papel INNER JOIN
    Ator INNER JOIN
        Filme_Ator ON Ator.Cod_ator = Filme_Ator.Cod_ator INNER JOIN
        Filme ON Filme_Ator.Cod_filme = Filme.Cod_filme INNER JOIN
```

# Código gerado na Query

```
Filme ON Filme_Ator.Cod_filme = Filme.Cod_filme INNER JOIN
Genero ON Filme.Cod_genero = Genero.Cod_genero ON Papel.Cod_papel = Filme_Ator.Cod_papel INNER JOIN
Sessao ON Filme.Cod_filme = Sessao.Cod_filme INNER JOIN
Ingresso INNER JOIN
Funcionario ON Ingresso.Cod_funcionario = Funcionario.Cod_funcionario INNER JOIN
Cliente ON Ingresso.Cod_cliente = Cliente.Cod_cliente ON Sessao.Cod_sessao = Ingresso.Cod_sessao INNER JOIN
Sala ON Sessao.Cod_sala = Sala.Cod_sala INNER JOIN
Tipo ON Ingresso.Cod_tipo = Tipo.Cod_tipo

CREATE TABLE Genero (
    Cod_genero int identity,
    Descricao varchar (100),
    PRIMARY KEY (Cod_genero) --Definição de PK
);

CREATE TABLE Filme(
    Cod_filme int identity,
    Titulo varchar (30) NOT NULL,
    duracao time,
    Cod_genero int,
    PRIMARY KEY (Cod_filme), --Definição de PK
    foreign key (Cod_genero) references Genero --Definição de FK
);

CREATE TABLE Sessao (
    Cod_sessao int identity,
    Cod_sala int,
    Horario time.
```



# Código gerado na Query

```
Horario time,  
Cod_filme int,  
PRIMARY KEY (Cod_sessao),  
foreign key(Cod_sala) references Sala, --Definição de FK  
foreign key(Cod_filme) references Filme  
);  
  
CREATE TABLE Cliente (  
    Cod_cliente int identity,  
    Nome varchar (30) NOT NULL,  
    Telefone varchar (13) NOT NULL,  
    DataNasc date,  
    RG varchar (10) NULL,  
    PRIMARY KEY (Cod_cliente)  
);  
  
CREATE TABLE Funcionario(  
    Cod_funcionario int identity,  
    Nome varchar (30) NOT NULL,  
    Telefone varchar (11) NOT NULL,  
    Endereco varchar (50) NOT NULL,  
    RG varchar (10) NOT NULL,  
    PRIMARY KEY (Cod_funcionario)  
);
```

# Código gerado na Query

```
CREATE TABLE Tipo ( --Tipo de Ingresso
    Cod_tipo int identity,
    Descricao varchar (25) NOT NULL,
    PRIMARY KEY (Cod_tipo)
);

CREATE TABLE Ingresso (
    Cod_ingresso int identity,
    Cod_sessao int,
    Cod_cliente int,
    Cod_tipo int,
    Cod_funcionario int,
    PRIMARY KEY (Cod_ingresso), --Definição de PK
    --Definições de FK
    foreign key (Cod_sessao) references Sessao,
    foreign key (Cod_cliente) references Cliente,
    foreign key (Cod_tipo) references Tipo, --Tipo de Ingresso
    foreign key (Cod_funcionario) references Funcionario
);
```

# Código gerado na Query

```
CREATE TABLE Papel (  
    Cod_papel int identity ,  
    Descricao varchar(100) NOT NULL,  
    PRIMARY KEY (Cod_papel)  
);  
  
CREATE TABLE Ator (  
    Cod_ator int identity,  
    Nome varchar (30) NOT NULL,  
    Genero varchar (20) NOT NULL,  
    Telefone varchar (13) NOT NULL,  
    RG varchar(11) NOT NULL,  
    CPF varchar(11) NOT NULL,  
    PRIMARY KEY (Cod_ator)  
);  
  
CREATE TABLE Filme_Ator (  
    Cod_filme_ator int identity,  
    Cod_ator int,  
    Cod_filme int,  
    Cod_papel int,  
    PRIMARY KEY (Cod_filme_ator), --Definição de PK  
    foreign key (Cod_ator) references Ator,  
    foreign key (Cod_filme) references Filme,  
    foreign key (Cod_papel) references Papel  
);
```