FACULDADE ÁREA1

CIÊNCIA DA COMPUTAÇÃO

KAIQUE DIAS PEREIRA

BRUNO SOUSA SILVA

CAUÃ GUILHERME DOS SANTOS

PAULO HENRIQUE LEAL DOS SANTOS

YURI DE JESUS FERNANDES MENDES

DENILSON SANTOS DE JESUS

PEDRO SARMENTO RANGEL SANTOS

**PROJETO DE BANCO DE DADOS BASEADO EM UM SERVIÇO DE STREAMING**

SALVADOR/BA

2023

KAIQUE DIAS PEREIRA

BRUNO SOUSA SILVA

CAUÃ GUILHERME DOS SANTOS

PAULO HENRIQUE LEAL DOS SANTOS

YURI DE JESUS FERNANDES MENDES

DENILSON SANTOS DE JESUS

PEDRO SARMENTO RANGEL SANTOS

**PROJETO DE BANCO DE DADOS BASEADO EM UM SERVIÇO DE STREAMING**

Projeto de Banco de Dados baseado em um Serviço de Streaming apresentado ao Curso de Ciência da Computação da Faculdade Área1

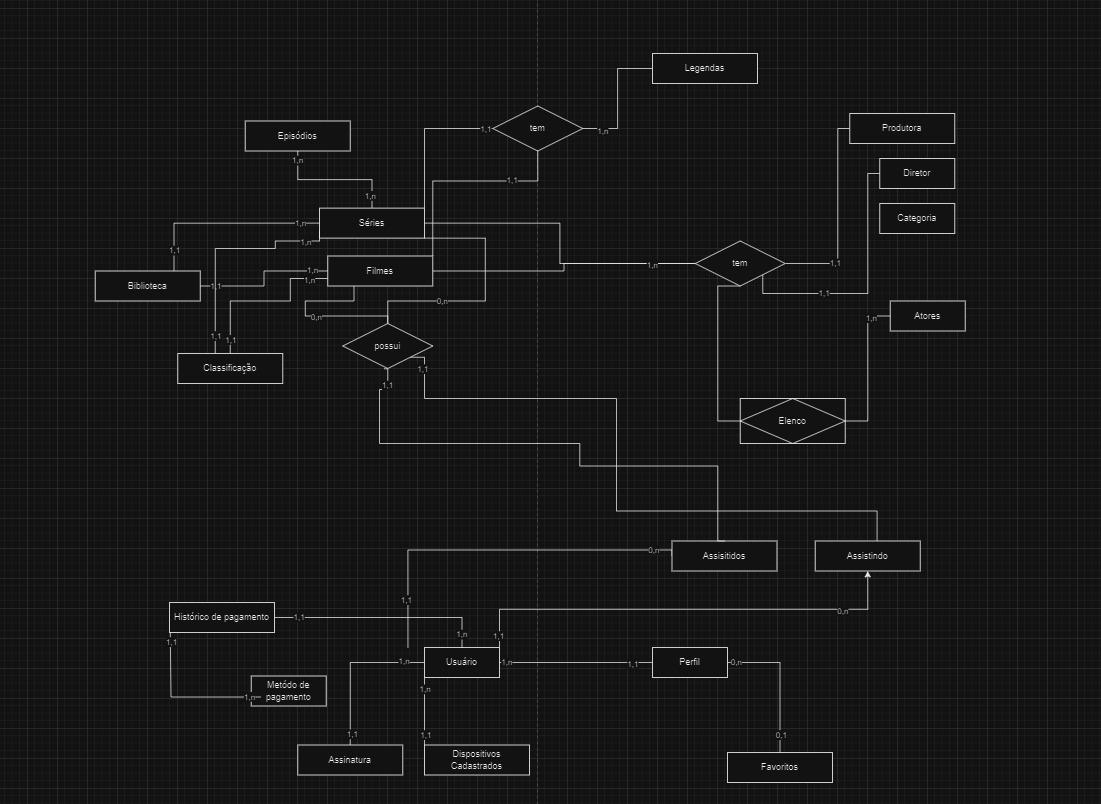
SALVADOR/BA

2023

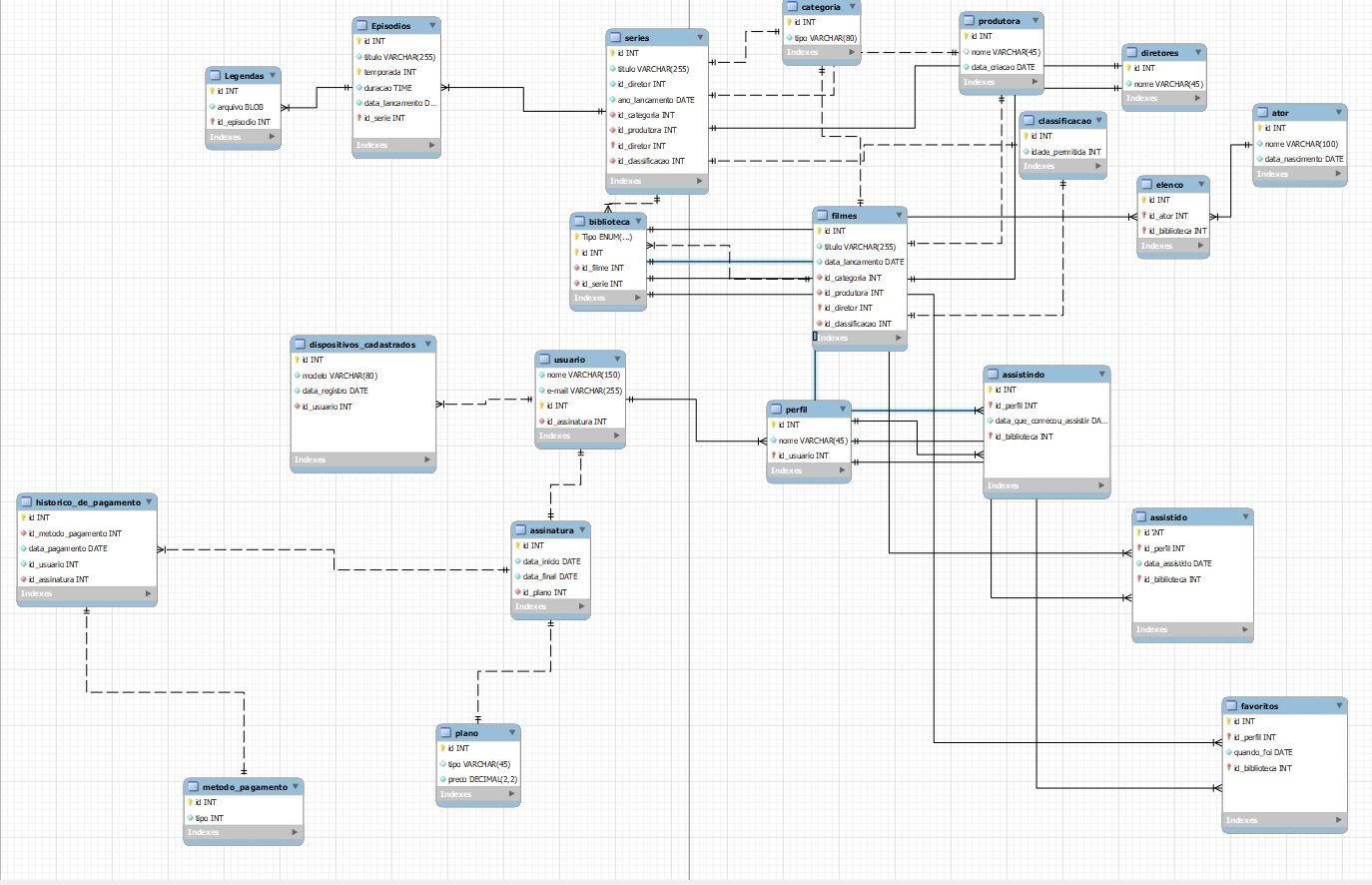
1. **Introdução**

Nosso objetivo é representar de forma conceitual, lógica e física um banco de dados de um serviço de streaming de filmes e séries, prezando o aprendizado do fluxo lógico de como é mantido e desenvolvido um sistema tão robusto e complexo de um serviço de streaming, onde nós esperamos aprender técnicas de consulta, otimização e eficiência dos dados. Para melhor produtividade do nosso grupo, nós nos dividimos em pares para realizar as questões solicitadas do trabalho e manter o nível de qualidade esperado. Focando em entender o funcionamento de um serviço de streaming e identificando quem são as entidades participantes do sistema e identificando seus atributos, tendo a compreensão de como cada uma desempenha seu papel no sistema.

**2. Modelo Conceitual**

****O Modelo de Entidade-Relacionamento (MER) apresentado abaixo descreve a estrutura de dados e os relacionamentos que compõem o Wyden-flix .Este sistema foi pensado desde o início para ser algo voltado ao produto e não a empresa de tal forma a que o usuário é totalmente livre assinar o nosso serviço, para ter diversos perfis e acessá-los em vários dispositivos assim tendo toda liberdade e conforto para assistir seus filmes/series do seu diretor ou produtora preferidos e até mesmo criar sua própria biblioteca de conteúdos favoritos.

**3.**  **Modelo Lógico**

****

Usuário: É figura principal para o funcionamento autenticação e personalização do sistema, o mesmo possui como atributos uma id única para identificação, endereço de e-mail, a id da assinatura para identificar se é usuário pagante ou não e um nome.

Assinatura: É o principal método de validação de acesso ao aplicativo por parte do usuário, ela se relaciona também diretamente ao histórico de pagamento e o plano escolhido pelo usuário tem como atributos uma id para identificação da assinatura uma data de início e término.

Dispositivos cadastrados: É a forma pela qual é possível ver a quantidade de aparelhos que estão vinculados a uma assinatura.

Perfil: Onde é possível ver as informações do usuário, como nome e a identificação dele.

Biblioteca: É onde serão armazenados os filmes assistidos, favoritos e em reprodução.

Filmes: Onde será exibido as opções a serem assistidas contendo categorias,elenco, classificação indicativa e diretor.

Séries: Onde será exibido as opções a serem assistidas contendo categorias,elenco, classificação indicativa e diretor.

**4. Modelo físico**

-- MySQL Script generated by MySQL Workbench

-- Wed Nov 8 21:36:11 2023

-- Model: New Model Version: 1.0

-- MySQL Workbench Forward Engineering

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

-- -----------------------------------------------------

-- Schema wydenflix

-- -----------------------------------------------------

-- -----------------------------------------------------

-- Schema wydenflix

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `wydenflix` DEFAULT CHARACTER SET utf8 ;

USE `wydenflix` ;

-- -----------------------------------------------------

-- Table `wydenflix`.`plano`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `plano` (

`id` INT NOT NULL,

`tipo` VARCHAR(45) NULL UNIQUE,

`preco` DECIMAL(5,2) NOT NULL,

PRIMARY KEY (`id`)

)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`assinatura`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`assinatura` (

`id` INT NOT NULL,

`data\_inicio` DATE NOT NULL,

`data\_final` DATE NOT NULL,

`id\_plano` INT NOT NULL,

PRIMARY KEY (`id`),

CONSTRAINT `fk\_assinatura\_plano1`

FOREIGN KEY (`id\_plano`)

REFERENCES `wydenflix`.`plano` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`conta`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`conta` (

`id` INT NOT NULL,

`email` VARCHAR(255) NOT NULL UNIQUE,

`senha` VARCHAR(20) NOT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`usuario`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`usuario` (

`nome` VARCHAR(150) NOT NULL,

`email` VARCHAR(255) NOT NULL UNIQUE,

`id` INT NOT NULL,

`id\_assinatura` INT NOT NULL,

`conta\_id` INT NOT NULL,

PRIMARY KEY (`id`, `conta\_id`),

CONSTRAINT `fk\_tb\_usuario\_Assinatura1`

FOREIGN KEY (`id\_assinatura`)

REFERENCES `wydenflix`.`assinatura` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_usuario\_conta1`

FOREIGN KEY (`email`)

REFERENCES `wydenflix`.`conta` (`email`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`perfil`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`perfil` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nome` VARCHAR(45) NOT NULL,

`id\_usuario` INT NOT NULL,

PRIMARY KEY (`id`, `id\_usuario`),

CONSTRAINT `fk\_tb\_perfil\_tb\_usuario1`

FOREIGN KEY (`id\_usuario`)

REFERENCES `wydenflix`.`usuario` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`dispositivos\_cadastrados`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`dispositivos\_cadastrados` (

`id` INT NOT NULL AUTO\_INCREMENT,

`modelo` VARCHAR(80) NOT NULL,

`data\_registro` DATE NOT NULL,

`id\_usuario` INT NOT NULL,

PRIMARY KEY (`id`),

CONSTRAINT `fk\_Dispositivos Cadastrados\_tb\_usuario1`

FOREIGN KEY (`id\_usuario`)

REFERENCES `wydenflix`.`usuario` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`metodo\_pagamento`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`metodo\_pagamento` (

`id` INT NOT NULL AUTO\_INCREMENT,

`tipo` INT NOT NULL UNIQUE,

PRIMARY KEY (`id`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`categoria`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`categoria` (

`id` INT NOT NULL AUTO\_INCREMENT,

`tipo` VARCHAR(80) NOT NULL UNIQUE,

PRIMARY KEY (`id`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`produtora`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`produtora` (

`id` INT NOT NULL,

`nome` VARCHAR(45) NULL,

`data\_criacao` DATE NOT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`diretores`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`diretores` (

`id` INT NOT NULL,

`nome` VARCHAR(45) NOT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`classificacao`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`classificacao` (

`id` INT NOT NULL,

`idade\_permitida` INT NOT NULL UNIQUE,

PRIMARY KEY (`id`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`filmes`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`filmes` (

`id` INT NOT NULL,

`titulo` VARCHAR(255) NOT NULL,

`data\_lancamento` DATE NOT NULL,

`id\_categoria` INT NOT NULL,

`id\_produtora` INT NOT NULL,

`id\_diretor` INT NOT NULL,

`id\_classificacao` INT NOT NULL,

PRIMARY KEY (`id`, `id\_diretor`),

CONSTRAINT `fk\_filmes\_Categoria1`

FOREIGN KEY (`id\_categoria`)

REFERENCES `wydenflix`.`categoria` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_filmes\_produtora1`

FOREIGN KEY (`id\_produtora`)

REFERENCES `wydenflix`.`produtora` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_filmes\_diretores1`

FOREIGN KEY (`id\_diretor`)

REFERENCES `wydenflix`.`diretores` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_filmes\_classificacao1`

FOREIGN KEY (`id\_classificacao`)

REFERENCES `wydenflix`.`classificacao` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`series`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`series` (

`id` INT NOT NULL,

`titulo` VARCHAR(255) NOT NULL,

`ano\_lancamento` DATE NOT NULL,

`id\_categoria` INT NOT NULL,

`id\_produtora` INT NOT NULL,

`id\_diretor` INT NOT NULL,

`id\_classificacao` INT NOT NULL,

PRIMARY KEY (`id`, `id\_diretor`),

CONSTRAINT `fk\_Series\_Categoria1`

FOREIGN KEY (`id\_categoria`)

REFERENCES `wydenflix`.`categoria` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_series\_produtora1`

FOREIGN KEY (`id\_produtora`)

REFERENCES `wydenflix`.`produtora` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_series\_diretores1`

FOREIGN KEY (`id\_diretor`)

REFERENCES `wydenflix`.`diretores` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_series\_classificacao1`

FOREIGN KEY (`id\_classificacao`)

REFERENCES `wydenflix`.`classificacao` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`biblioteca`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`biblioteca` (

`Tipo` ENUM("SERIE", "FILME") NOT NULL,

`id` INT NOT NULL AUTO\_INCREMENT,

`id\_filme` INT NULL UNIQUE,

`id\_serie` INT NULL UNIQUE,

PRIMARY KEY (`id`, `Tipo`),

CONSTRAINT `fk\_Biblioteca\_Filmes1`

FOREIGN KEY (`id\_filme`)

REFERENCES `wydenflix`.`filmes` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Biblioteca\_Series1`

FOREIGN KEY (`id\_serie`)

REFERENCES `wydenflix`.`series` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`assistindo`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`assistindo` (

`id` INT NOT NULL AUTO\_INCREMENT,

`id\_perfil` INT NOT NULL,

`data\_que\_comecou\_assistir` DATE NOT NULL,

`id\_biblioteca` INT NOT NULL,

PRIMARY KEY (`id`, `id\_biblioteca`, `id\_perfil`),

CONSTRAINT `fk\_Assistindo\_tb\_perfil1`

FOREIGN KEY (`id\_perfil`)

REFERENCES `wydenflix`.`perfil` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Assistindo\_Biblioteca1`

FOREIGN KEY (`id\_biblioteca`)

REFERENCES `wydenflix`.`biblioteca` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`Episodios`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`Episodios` (

`id` INT NOT NULL,

`titulo` VARCHAR(255) NOT NULL,

`temporada` INT NOT NULL,

`duracao` TIME NOT NULL,

`data\_lancamento` DATE NOT NULL,

`id\_serie` INT NOT NULL,

PRIMARY KEY (`id`, `id\_serie`, `temporada`),

CONSTRAINT `fk\_Episodios\_Series1`

FOREIGN KEY (`id\_serie`)

REFERENCES `wydenflix`.`series` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`ator`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`ator` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nome` VARCHAR(100) NOT NULL,

`data\_nascimento` DATE NOT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`elenco`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`elenco` (

`id` INT NOT NULL,

`id\_ator` INT NOT NULL,

`id\_biblioteca` INT NOT NULL,

PRIMARY KEY (`id`, `id\_ator`, `id\_biblioteca`),

CONSTRAINT `fk\_Elenco\_Ator1`

FOREIGN KEY (`id\_ator`)

REFERENCES `wydenflix`.`ator` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Elenco\_Biblioteca1`

FOREIGN KEY (`id\_biblioteca`)

REFERENCES `wydenflix`.`biblioteca` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`historico\_de\_pagamento`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`historico\_de\_pagamento` (

`id` INT NOT NULL AUTO\_INCREMENT,

`id\_metodo\_pagamento` INT NOT NULL,

`data\_pagamento` DATE NOT NULL,

`id\_usuario` INT NOT NULL,

`id\_assinatura` INT NOT NULL,

PRIMARY KEY (`id`),

CONSTRAINT `fk\_Historico de Pagamento\_Metodo de Pagamento1`

FOREIGN KEY (`id\_metodo\_pagamento`)

REFERENCES `wydenflix`.`metodo\_pagamento` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Historico de Pagamento\_Assinatura1`

FOREIGN KEY (`id\_assinatura`)

REFERENCES `wydenflix`.`assinatura` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`Legendas`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`Legendas` (

`id` INT NOT NULL AUTO\_INCREMENT,

`arquivo` BLOB NOT NULL UNIQUE,

`id\_episodio` INT NOT NULL,

PRIMARY KEY (`id`, `id\_episodio`),

CONSTRAINT `fk\_Legendas\_Episodios1`

FOREIGN KEY (`id\_episodio`)

REFERENCES `wydenflix`.`Episodios` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`assistido`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`assistido` (

`id` INT NOT NULL AUTO\_INCREMENT,

`id\_perfil` INT NOT NULL,

`data\_assistido` DATE NOT NULL,

`id\_biblioteca` INT NOT NULL,

PRIMARY KEY (`id`, `id\_perfil`, `id\_biblioteca`),

CONSTRAINT `fk\_Assistido\_tb\_perfil1`

FOREIGN KEY (`id\_perfil`)

REFERENCES `wydenflix`.`perfil` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Assistido\_Biblioteca1`

FOREIGN KEY (`id\_biblioteca`)

REFERENCES `wydenflix`.`biblioteca` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `wydenflix`.`favoritos`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `wydenflix`.`favoritos` (

`id` INT NOT NULL AUTO\_INCREMENT,

`id\_perfil` INT NOT NULL,

`quando\_foi` DATE NOT NULL,

`id\_biblioteca` INT NOT NULL,

PRIMARY KEY (`id`, `id\_perfil`, `id\_biblioteca`),

CONSTRAINT `fk\_Favoritos\_tb\_perfil1`

FOREIGN KEY (`id\_perfil`)

REFERENCES `wydenflix`.`perfil` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Favoritos\_Biblioteca1`

FOREIGN KEY (`id\_biblioteca`)

REFERENCES `wydenflix`.`biblioteca` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

ALTER TABLE metodo\_pagamento

MODIFY COLUMN tipo VARCHAR(100) NOT NULL;

ALTER TABLE elenco

MODIFY COLUMN id INT AUTO\_INCREMENT;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

**5**

INSERT INTO wydenflix.conta (id, email, senha) VALUES

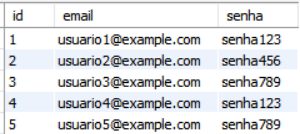
(1, 'usuario1@example.com', 'senha123'),

(2, 'usuario2@example.com', 'senha456'),

(3, 'usuario3@example.com', 'senha789'),

(4, 'usuario4@example.com', 'senha123'),

(5, 'usuario5@example.com', 'senha789');



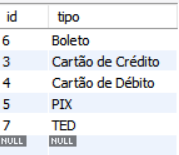
INSERT INTO wydenflix.metodo\_pagamento (tipo) VALUES ('Cartão de Crédito');

INSERT INTO wydenflix.metodo\_pagamento (tipo) VALUES ('Cartão de Débito');

INSERT INTO wydenflix.metodo\_pagamento (tipo) VALUES ('PIX');

INSERT INTO wydenflix.metodo\_pagamento (tipo) VALUES ('Boleto');

INSERT INTO wydenflix.metodo\_pagamento (tipo) VALUES ('TED');



INSERT INTO wydenflix.plano (id, tipo, preco) VALUES

(1, 'Plano Básico', 9.99),

(2, 'Plano Padrão', 14.99),

(3, 'Plano Premium', 19.99),

(4, 'Plano Promocional', 17.99),

(5, 'Plano Gratuito', 0);



INSERT INTO wydenflix.assinatura (id, data\_inicio, data\_final, id\_plano) VALUES

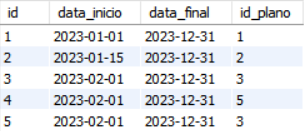
(1, '2023-01-01', '2023-12-31', 1),

(2, '2023-01-15', '2023-12-31', 2),

(3, '2023-02-01', '2023-12-31', 3),

(4, '2023-02-01', '2023-12-31', 5),

(5, '2023-02-01', '2023-12-31', 3);



INSERT INTO wydenflix.usuario (nome, email, id, id\_assinatura, conta\_id) VALUES

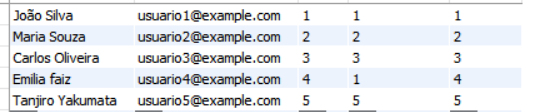
('João Silva', 'usuario1@example.com', 1, 1, 1),

('Maria Souza', 'usuario2@example.com', 2, 2, 2),

('Carlos Oliveira', 'usuario3@example.com', 3, 3, 3),

('Emilia faiz', 'usuario4@example.com', 4, 1, 4),

('Tanjiro Yakumata', 'usuario5@example.com', 5, 5, 5);



INSERT INTO `wydenflix`.`historico\_de\_pagamento` (`id\_metodo\_pagamento`, `data\_pagamento`, `id\_usuario`, `id\_assinatura`) VALUES

(6, '2023-10-15', 1, 1),

(7, '2023-11-20', 2, 2),

(7, '2023-12-25', 3, 3),

(6, '2024-01-10', 1, 1),

(6, '2024-02-12', 2, 2),

(3, '2024-03-18', 3, 3),

(3, '2024-04-05', 1, 1),

(3, '2024-05-22', 2, 2),

(4, '2024-06-30', 3, 3),

(4, '2024-07-15', 1, 1),

(4, '2024-08-20', 2, 2),

(5, '2024-09-25', 3, 3),

(5, '2024-10-10', 1, 1),

(5, '2024-11-12', 2, 2),

(7, '2024-12-18', 3, 3),

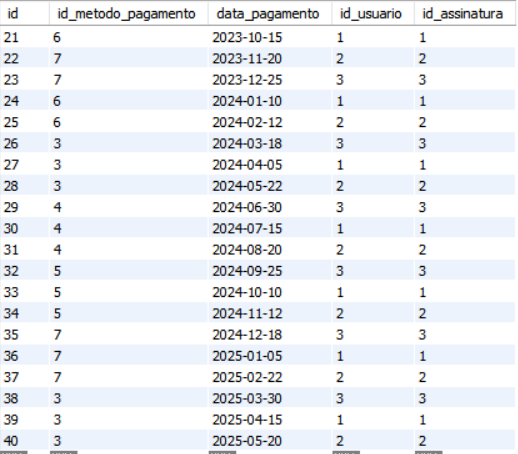
(7, '2025-01-05', 1, 1),

(7, '2025-02-22', 2, 2),

(3, '2025-03-30', 3, 3),

(3, '2025-04-15', 1, 1),

(3, '2025-05-20', 2, 2);



INSERT INTO wydenflix.dispositivos\_cadastrados (id, modelo, data\_registro, id\_usuario) VALUES

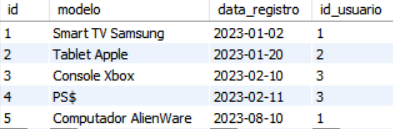
(1, 'Smart TV Samsung', '2023-01-02', 1),

(2, 'Tablet Apple', '2023-01-20', 2),

(3, 'Console Xbox', '2023-02-10', 3),

(4, 'PS$', '2023-02-11', 3),

(5, 'Computador AlienWare', '2023-08-10', 1);



INSERT INTO wydenflix.perfil (nome, id\_usuario) VALUES

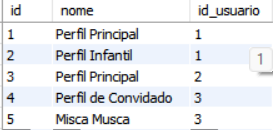
('Perfil Principal', 1),

('Perfil Infantil', 1),

('Perfil Principal', 2),

('Perfil de Convidado', 3),

('Misca Musca', 3);



INSERT INTO wydenflix.diretores (id, nome) VALUES (1, 'Christopher Nolan');

INSERT INTO wydenflix.diretores (id, nome) VALUES (2, 'Quentin Tarantino');

INSERT INTO wydenflix.diretores (id, nome) VALUES (3, 'Steven Spielberg');

INSERT INTO wydenflix.diretores (id, nome) VALUES (4, 'Denzel Washigton');

INSERT INTO wydenflix.diretores (id, nome) VALUES (5, 'Morgan Freeman');



INSERT INTO wydenflix.classificacao (id, idade\_permitida) VALUES

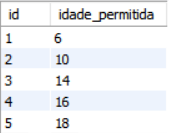
(1, 6),

(2, 10),

(3, 14),

(4, 16),

(5, 18);



INSERT INTO wydenflix.categoria (tipo) VALUES ('Ação');

INSERT INTO wydenflix.categoria (tipo) VALUES ('Comédia');

INSERT INTO wydenflix.categoria (tipo) VALUES ('Drama');

INSERT INTO wydenflix.categoria (tipo) VALUES ('Suspense');

INSERT INTO wydenflix.categoria (tipo) VALUES ('Romance');



INSERT INTO wydenflix.produtora (id, nome, data\_criacao) VALUES (1, 'Warner Bros', '1923-04-04');

INSERT INTO wydenflix.produtora (id, nome, data\_criacao) VALUES (2, 'Paramount Pictures', '1912-05-08');

INSERT INTO wydenflix.produtora (id, nome, data\_criacao) VALUES (3, 'The Walt Disney Company', '1923-10-16');

INSERT INTO wydenflix.produtora (id, nome, data\_criacao) VALUES (4, '21 Century FOX', '1919-12-16');

INSERT INTO wydenflix.produtora (id, nome, data\_criacao) VALUES (5, 'Netflix Production', '2005-08-27');



INSERT INTO wydenflix.series (id, titulo, ano\_lancamento, id\_categoria, id\_produtora, id\_diretor, id\_classificacao) VALUES

(2, 'Stranger Things', '2016-07-15', 1, 1, 2, 2),

(3, 'The Crown', '2016-11-04', 1, 2, 3, 1),

(4, 'Black Mirror', '2011-12-04', 2, 1, 4, 3),

(5, 'Narcos', '2015-08-28', 1, 3, 5, 4);

INSERT INTO wydenflix.series (id, titulo, ano\_lancamento, id\_categoria, id\_produtora, id\_diretor, id\_classificacao) VALUES

(6, 'The Witcher', '2019-12-20', 1, 1, 4, 3),

(7, 'Money Heist', '2017-05-02', 2, 2, 4, 4),

(8, 'The Mandalorian', '2019-11-12', 1, 3, 2, 2),

(9, 'Breaking Bad', '2008-01-20', 1, 1, 2, 5);

INSERT INTO wydenflix.series (id, titulo, ano\_lancamento, id\_categoria, id\_produtora, id\_diretor, id\_classificacao) VALUES

(10, 'The Office', '2005-03-24', 2, 2, 4, 2),

(11, 'Stranger Things 2', '2017-10-27', 1, 1, 2, 2),

(12, 'Westworld', '2016-10-02', 1, 3, 1, 4),

(13, 'Friends', '1994-09-22', 2, 2, 5, 2);



INSERT INTO wydenflix.Episodios (id, titulo, temporada, duracao, data\_lancamento, id\_serie) VALUES

(1, 'Chapter One: The Vanishing of Will Byers', 1, '00:50:00', '2016-07-15', 2),

(2, 'Chapter Two: The Weirdo on Maple Street', 1, '00:45:00', '2016-07-15', 2),

(3, 'Chapter One: Wolferton Splash', 1, '00:55:00', '2016-11-04', 3),

(4, 'Chapter Two: Hyde Park Corner', 1, '01:00:00', '2016-11-04', 3),

(5, 'The National Anthem', 1, '00:44:00', '2011-12-04', 4);

INSERT INTO wydenflix.Episodios (id, titulo, temporada, duracao, data\_lancamento, id\_serie) VALUES

(6, 'Fifteen Million Merits', 1, '00:49:00', '2011-12-04', 4),

(7, 'Nosedive', 3, '00:58:00', '2016-10-21', 4),

(8, 'USS Callister', 4, '00:51:00', '2017-12-29', 4),

(9, 'The Original', 1, '00:58:00', '2016-10-02', 12);

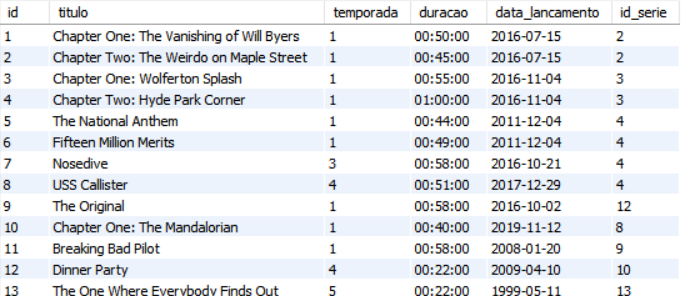
INSERT INTO wydenflix.Episodios (id, titulo, temporada, duracao, data\_lancamento, id\_serie) VALUES

(10, 'Chapter One: The Mandalorian', 1, '00:40:00', '2019-11-12', 8),

(11, 'Breaking Bad Pilot', 1, '00:58:00', '2008-01-20', 9),

(12, 'Dinner Party', 4, '00:22:00', '2009-04-10', 10),

(13, 'The One Where Everybody Finds Out', 5, '00:22:00', '1999-05-11', 13);



INSERT INTO wydenflix.Legendas (arquivo, id\_episodio) VALUES

('Legenda\_Episodio1.srt', 1),

('Legenda\_Episodio2.srt', 2),

('Legenda\_Episodio3.srt', 3),

('Legenda\_Episodio4.srt', 4),

('Legenda\_Episodio5.srt', 5);

INSERT INTO wydenflix.Legendas (arquivo, id\_episodio) VALUES

('Legenda\_Episodio6.srt', 6),

('Legenda\_Episodio7.srt', 7),

('Legenda\_Episodio8.srt', 8),

('Legenda\_Episodio9.srt', 9),

('Legenda\_Episodio10.srt', 10);

INSERT INTO wydenflix.Legendas (arquivo, id\_episodio) VALUES

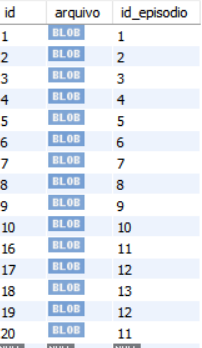
('Legenda\_Episodio11.srt', 11),

('Legenda\_Episodio12.srt', 12),

('Legenda\_Episodio13.srt', 13),

('Legenda\_Episodio14.srt', 12),

('Legenda\_Episodio15.srt', 11);



INSERT INTO filmes(id, titulo, data\_lancamento, id\_categoria, id\_produtora, id\_diretor, id\_classificacao) VALUES (1, "A Cor Púrpura", "1985-12-18",3, 1,3,3);

INSERT INTO filmes(id, titulo, data\_lancamento, id\_categoria, id\_produtora, id\_diretor, id\_classificacao) VALUES (2, "A.I. Artificial Intelligence", "1985-07-29",3, 1,3,3);

INSERT INTO filmes(id, titulo, data\_lancamento, id\_categoria, id\_produtora, id\_diretor, id\_classificacao) VALUES (3, "Kill Bill: Volume 1", "2003-10-10",1, 1,2,5);

INSERT INTO `wydenflix`.`filmes` (`id`, `titulo`, `data\_lancamento`, `id\_categoria`, `id\_produtora`, `id\_diretor`, `id\_classificacao`)

VALUES (4, 'Schindler\'s List', '1993-12-15', 2, 2, 2, 3);

-- Filme 5

INSERT INTO `wydenflix`.`filmes` (`id`, `titulo`, `data\_lancamento`, `id\_categoria`, `id\_produtora`, `id\_diretor`, `id\_classificacao`)

VALUES (5, 'Saving Private Ryan', '1998-07-24', 2, 2, 2, 4);

INSERT INTO `wydenflix`.`filmes` (`id`, `titulo`, `data\_lancamento`, `id\_categoria`, `id\_produtora`, `id\_diretor`, `id\_classificacao`)

VALUES (6, 'The Shawshank Redemption', '1994-09-23', 2, 3, 3, 3);

INSERT INTO `wydenflix`.`filmes` (`id`, `titulo`, `data\_lancamento`, `id\_categoria`, `id\_produtora`, `id\_diretor`, `id\_classificacao`)

VALUES (7, 'The Godfather', '1972-03-24', 2, 3, 3, 4);

INSERT INTO `wydenflix`.`filmes` (`id`, `titulo`, `data\_lancamento`, `id\_categoria`, `id\_produtora`, `id\_diretor`, `id\_classificacao`)

VALUES (8, 'The Dark Knight', '2008-07-18', 3, 4, 4, 5);

INSERT INTO `wydenflix`.`filmes` (`id`, `titulo`, `data\_lancamento`, `id\_categoria`, `id\_produtora`, `id\_diretor`, `id\_classificacao`)

VALUES (9, 'Inception', '2010-07-16', 3, 4, 4, 5);

INSERT INTO `wydenflix`.`filmes` (`id`, `titulo`, `data\_lancamento`, `id\_categoria`, `id\_produtora`, `id\_diretor`, `id\_classificacao`)

VALUES (10, 'Pulp Fiction', '1994-05-12', 3, 5, 5, 4);

INSERT INTO `wydenflix`.`filmes` (`id`, `titulo`, `data\_lancamento`, `id\_categoria`, `id\_produtora`, `id\_diretor`, `id\_classificacao`)

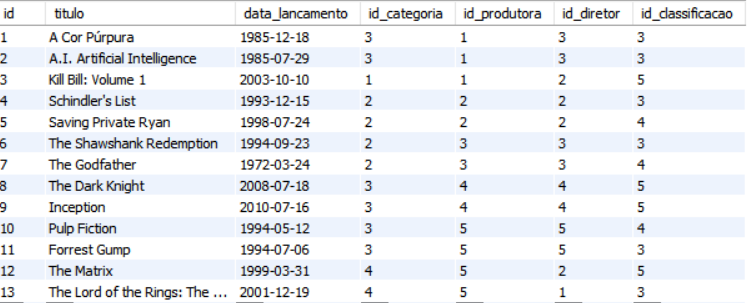
VALUES (11, 'Forrest Gump', '1994-07-06', 3, 5, 5, 3);

INSERT INTO `wydenflix`.`filmes` (`id`, `titulo`, `data\_lancamento`, `id\_categoria`, `id\_produtora`, `id\_diretor`, `id\_classificacao`)

VALUES (12, 'The Matrix', '1999-03-31', 4, 5, 2, 5);

INSERT INTO `wydenflix`.`filmes` (`id`, `titulo`, `data\_lancamento`, `id\_categoria`, `id\_produtora`, `id\_diretor`, `id\_classificacao`)

VALUES (13, 'The Lord of the Rings: The Fellowship of the Ring', '2001-12-19', 4, 5, 1, 3);



INSERT INTO wydenflix.biblioteca (Tipo, id\_filme, id\_serie) VALUES

('FILME', 1, NULL),

('FILME', 2, NULL),

('FILME', 3, NULL),

('FILME', 4, NULL),

('FILME', 5, NULL);

INSERT INTO wydenflix.biblioteca (Tipo, id\_filme, id\_serie) VALUES

('SERIE', NULL, 12),

('SERIE', NULL, 2),

('SERIE', NULL, 3),

('SERIE', NULL, 4),

('SERIE', NULL, 5);

INSERT INTO wydenflix.biblioteca (Tipo, id\_filme, id\_serie) VALUES

('FILME', 6, NULL),

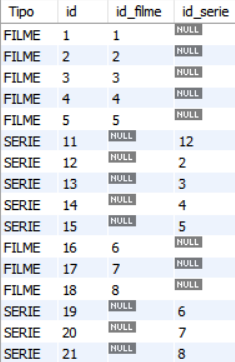
('FILME', 7, NULL),

('FILME', 8, NULL),

('SERIE', NULL, 6),

('SERIE', NULL, 7),

('SERIE', NULL, 8);



INSERT INTO wydenflix.favoritos (id\_perfil, quando\_foi, id\_biblioteca) VALUES

(1, '2023-11-08', 1),

(1, '2023-11-08', 3),

(2, '2023-11-08', 2),

(2, '2023-11-08', 4),

(3, '2023-11-08', 5),

(3, '2023-11-08', 12);

INSERT INTO wydenflix.favoritos (id\_perfil, quando\_foi, id\_biblioteca) VALUES

(1, '2023-11-09', 14),

(2, '2023-11-09', 14),

(3, '2023-11-09',14),

(4, '2023-11-09', 11),

(4, '2023-11-09', 13);

INSERT INTO wydenflix.favoritos (id\_perfil, quando\_foi, id\_biblioteca) VALUES

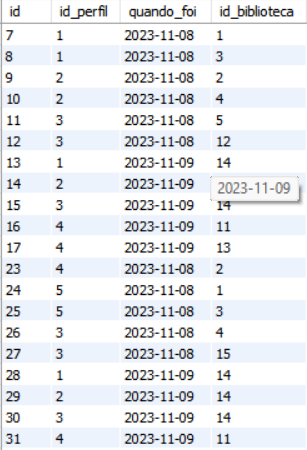
(4, '2023-11-08', 2),

(5, '2023-11-08', 1),

(5, '2023-11-08', 3),

(3, '2023-11-08', 4),

(3, '2023-11-08', 15);



INSERT INTO wydenflix.assistido (id\_perfil, data\_assistido, id\_biblioteca) VALUES

(1, '2023-11-08', 1),

(1, '2023-11-08', 3),

(2, '2023-11-08', 2),

(2, '2023-11-08', 4),

(3, '2023-11-08', 5),

(3, '2023-11-08', 1);

INSERT INTO wydenflix.assistido (id\_perfil, data\_assistido, id\_biblioteca) VALUES

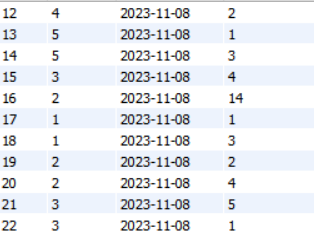
(4, '2023-11-08', 2),

(5, '2023-11-08', 1),

(5, '2023-11-08', 3),

(3, '2023-11-08', 4),

(2, '2023-11-08', 14);



INSERT INTO wydenflix.assistindo (id\_perfil, data\_que\_comecou\_assistir, id\_biblioteca) VALUES

(1, '2023-11-08', 1),

(1, '2023-11-08', 3),

(2, '2023-11-08', 2),

(2, '2023-11-08', 4),

(3, '2023-11-08', 5),

(3, '2023-11-08', 12);

INSERT INTO wydenflix.assistindo (id\_perfil, data\_que\_comecou\_assistir, id\_biblioteca) VALUES

(1, '2023-11-09', 15),

(2, '2023-11-09', 13),

(3, '2023-11-09', 13),

(4, '2023-11-09', 13),

(4, '2023-11-09', 13);

INSERT INTO wydenflix.assistindo (id\_perfil, data\_que\_comecou\_assistir, id\_biblioteca) VALUES

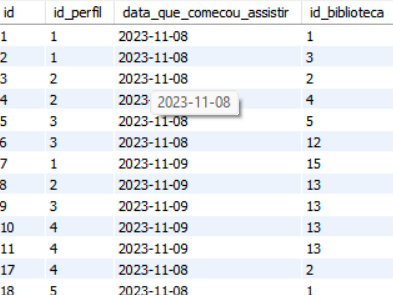
(4, '2023-11-08', 2),

(5, '2023-11-08', 1),

(5, '2023-11-08', 3),

(2, '2023-11-08', 4),

(1, '2023-11-08', 12);



-- Inserir atores fictícios

INSERT INTO wydenflix.ator (nome, data\_nascimento) VALUES

('Tom Hanks', '1956-07-09'),

('Scarlett Johansson', '1984-11-22'),

('Leonardo DiCaprio', '1974-11-11'),

('Jennifer Lawrence', '1990-08-15'),

('Chris Hemsworth', '1983-08-11');

-- Inserir mais atores fictícios

INSERT INTO wydenflix.ator (nome, data\_nascimento) VALUES

('Gal Gadot', '1985-04-30'),

('Robert Downey Jr.', '1965-04-04'),

('Emma Stone', '1988-11-06'),

('Chris Evans', '1981-06-13'),

('Margot Robbie', '1990-07-02');

-- Inserir atores adicionais

INSERT INTO wydenflix.ator (nome, data\_nascimento) VALUES

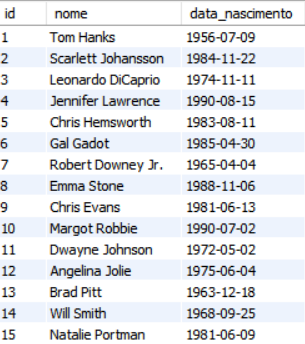
('Dwayne Johnson', '1972-05-02'),

('Angelina Jolie', '1975-06-04'),

('Brad Pitt', '1963-12-18'),

('Will Smith', '1968-09-25'),

('Natalie Portman', '1981-06-09');



INSERT INTO wydenflix.elenco (id\_ator, id\_biblioteca) VALUES

(1, 1),

(2, 2),

(3, 3),

(4, 4),

(5, 5),

(6, 12),

(7, 14),

(8, 13),

(9, 14),

(10, 14),

(11, 12),

(12, 12);

INSERT INTO wydenflix.elenco (id\_ator, id\_biblioteca) VALUES

(13, 1),

(14, 2),

(15, 3),

(12, 4),

(15, 5),

(8, 11),

(7, 12),

(3, 15),

(2, 15),

(2, 15),

(3, 11),

(4, 12);

INSERT INTO wydenflix.elenco (id\_ator, id\_biblioteca) VALUES

(5, 1),

(6, 2),

(7, 3),

(8, 4),

(9, 5),

(4, 15),

(1, 14),

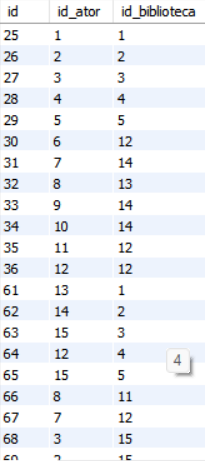
(2, 14),

(3, 14),

(4, 14),

(5, 15),

(6, 12);



**6.1**

| **Usuário** | **Select** | **Insert** | **Update** | **Delete** |
| --- | --- | --- | --- | --- |
| **Adm** | **SIM** | **SIM** | **SIM** | **SIM** |
| **Usuário Pago** | **SIM** | **SIM** | **SIM** | **SIM** |
| **Produtora** | **SIM** | **SIM** | **NÃO** | **NÃO** |
| **Usuário não pagante** | **NÃO** | **NÃO** | **NÃO** | **NÃO** |
| **Consultor** | **SIM** | **NÃO** | **NÃO** | **NÃO** |

**6.2**

CREATE USER 'usuario pago'@'localhost' IDENTIFIED VIA mysql\_native\_password USING '\*\*\*';GRANT SELECT, INSERT, UPDATE, DELETE ON \*.\* TO 'usuario pago'@'localhost' REQUIRE NONE WITH MAX\_QUERIES\_PER\_HOUR 0 MAX\_CONNECTIONS\_PER\_HOUR 0 MAX\_UPDATES\_PER\_HOUR 0 MAX\_USER\_CONNECTIONS 0;GRANT ALL PRIVILEGES ON `wydenflix`.\* TO 'usuario pago'@'localhost';

CREATE USER 'usuario nao pago'@'localhost' IDENTIFIED VIA mysql\_native\_password USING '\*\*\*';GRANT USAGE ON \*.\* TO 'usuario nao pago'@'localhost' REQUIRE NONE WITH MAX\_QUERIES\_PER\_HOUR 0 MAX\_CONNECTIONS\_PER\_HOUR 0 MAX\_UPDATES\_PER\_HOUR 0 MAX\_USER\_CONNECTIONS 0;GRANT ALL PRIVILEGES ON `wydenflix`.\* TO 'usuario nao pago'@'localhost';

REVOKE ALL PRIVILEGES ON `wydenflix`.\* FROM 'usuario nao pago'@'localhost'; GRANT USAGE ON `wydenflix`.\* TO 'usuario nao pago'@'localhost';

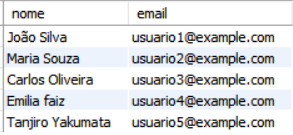
GRANT SELECT, INSERT, UPDATE, DELETE ON \*.\* TO `usuario pago`@`localhost`;

CREATE USER 'produtora'@'localhost' IDENTIFIED VIA mysql\_native\_password USING '\*\*\*';GRANT SELECT, INSERT ON \*.\* TO 'produtora'@'localhost' REQUIRE NONE WITH MAX\_QUERIES\_PER\_HOUR 0 MAX\_CONNECTIONS\_PER\_HOUR 0 MAX\_UPDATES\_PER\_HOUR 0 MAX\_USER\_CONNECTIONS 0;

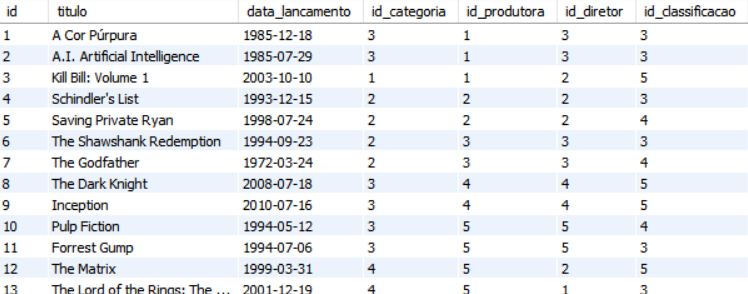
CREATE USER 'consultor'@'localhost' IDENTIFIED VIA mysql\_native\_password USING '\*\*\*';GRANT SELECT ON \*.\* TO 'consultor'@'localhost' REQUIRE NONE WITH MAX\_QUERIES\_PER\_HOUR 0 MAX\_CONNECTIONS\_PER\_HOUR 0 MAX\_UPDATES\_PER\_HOUR 0 MAX\_USER\_CONNECTIONS 0;

**7.1**

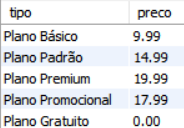
SELECT nome, email FROM usuario;



SELECT \* FROM filmes;



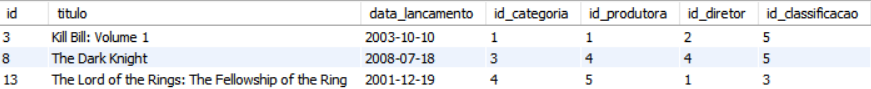
SELECT tipo, preco FROM plano;



**7.2**

SELECT \* FROM filmes

WHERE data\_lancamento BETWEEN "2000-01-01" AND "2010-01-01";



SELECT \* FROM episódios

WHERE temporada = 1 and data\_lancamento >= "2015-01-01";



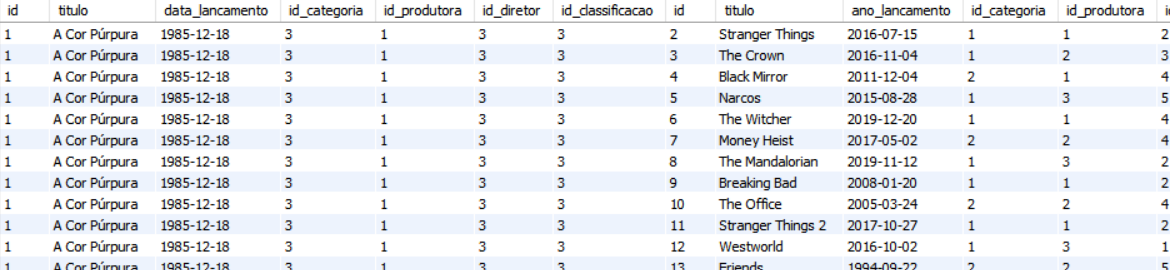
SELECT \* FROM historico\_de\_pagamento

WHERE id >= 10 and id <= 100;

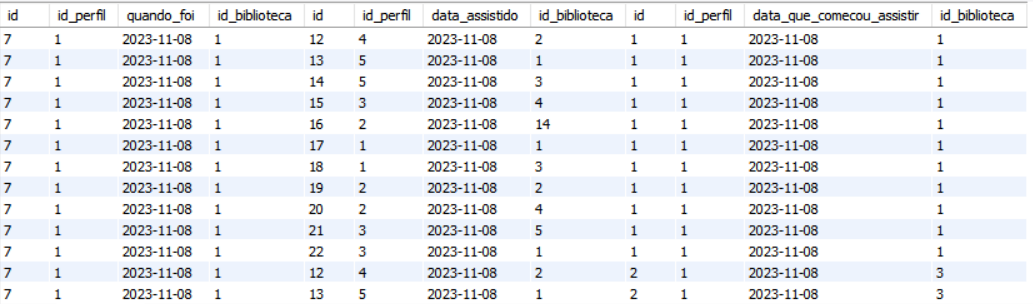


**7.3**

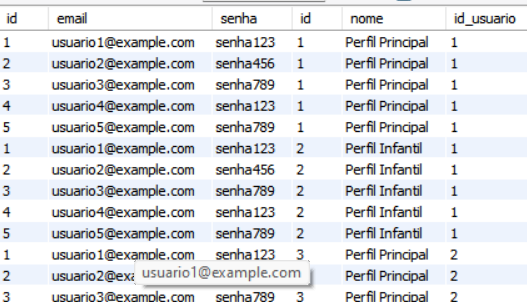
SELECT \* FROM filmes, series;



SELECT \* FROM favoritos, assistido, assistindo;



SELECT \* FROM conta, perfil;



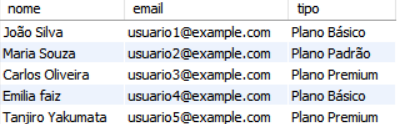
**7.4**

SELECT nome, email, tipo

FROM usuario

INNER JOIN assinatura on assinatura.id = usuario.id\_assinatura

INNER JOIN plano ON plano.id = assinatura.id\_plano;



SELECT titulo, data\_lancamento, diretores.nome AS nome\_diretor, classificacao.idade\_permitida, categoria.tipo, produtora.nome FROM biblioteca

INNER JOIN filmes ON filmes.id = id\_filme

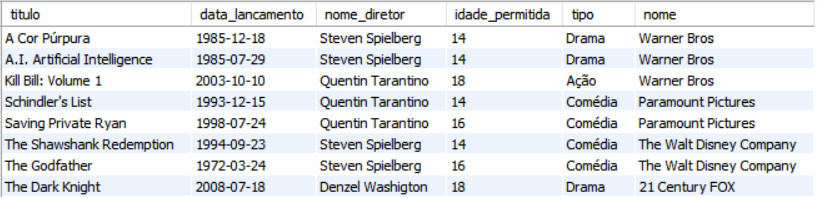
INNER JOIN categoria ON categoria.id = filmes.id\_categoria

INNER JOIN produtora ON produtora.id = filmes.id\_produtora

INNER JOIN classificacao ON classificacao.id = filmes.id\_classificacao

INNER JOIN diretores ON diretores.id = filmes.id\_diretor

WHERE id\_serie IS NULL;



SELECT series.titulo, series.ano\_lancamento, ator.nome FROM biblioteca

INNER JOIN series ON series.id = biblioteca.id\_serie

INNER JOIN elenco ON elenco.id\_biblioteca = biblioteca.id

INNER JOIN ator on ator.id = elenco.id\_ator

WHERE biblioteca.id\_filme IS NULL;



**8. Normalização**

A normalização no modelo de dados do WydenFlix desempenha um papel essencial na estruturação eficiente das informações armazenadas no banco de dados, seguindo princípios fundamentais para garantir a consistência e integridade dos dados.

Ao analisar a arquitetura das tabelas, percebemos uma cuidadosa aplicação dos conceitos de normalização. Cada tabela representa uma entidade única, como Categoria, Diretores, Produtora, Método de Pagamento, Plano, Assinatura, Conta, Usuário, Perfil, Dispositivos Cadastrados, Filmes, Classificação, Séries, Episódios, Ator, Legendas, Biblioteca, Favoritos, Assistido, Assistindo, Elenco e Histórico de Pagamento. A minimização de redundâncias é evidente na estrutura do banco de dados. Por exemplo, as informações relacionadas aos usuários são armazenadas na tabela Usuário, evitando duplicações desnecessárias em outras partes do sistema. Isso não apenas economiza espaço de armazenamento, mas também simplifica a manutenção e atualização dos dados. Os relacionamentos entre as tabelas são estabelecidos através de chaves estrangeiras, criando uma rede de conexões entre as diferentes entidades. Um exemplo notável é a relação entre `Assinatura` e `Plano`, onde a chave estrangeira na tabela Assinatura referencia o id correspondente na tabela `Plano`. Essa abordagem cria uma integração coesa entre as diferentes partes do sistema, permitindo consultas complexas e garantindo a consistência dos dados. Em resumo, a normalização no WydenFlix não é apenas uma prática técnica, mas uma estratégia cuidadosa para garantir que a estrutura de dados seja eficiente, livre de redundâncias desnecessárias e capaz de manter a integridade dos dados em todo o sistema. Essa abordagem sólida contribui para a confiabilidade e desempenho do serviço de streaming, proporcionando uma experiência consistente e confiável aos usuários.

**9. Conclusão**

Após a análise e desenvolvimento detalhado do banco de dados para o serviço de streaming de filmes e séries proposto, é possível concluir que atingimos com sucesso os objetivos definidos no início do processo. Nosso foco estava em representar de forma conceitual, lógica e física a estrutura do banco de dados, proporcionando uma compreensão profunda do funcionamento de um sistema robusto e complexo de streaming. Ao longo do trabalho, exploramos a modelagem de dados, abrangendo entidades como usuários, assinaturas, filmes, séries, planos, entre outras. A divisão em pares facilitou a abordagem de cada aspecto de maneira mais eficiente, garantindo consistência e qualidade em cada etapa do desenvolvimento. A compreensão do fluxo lógico do sistema se revelou fundamental, especialmente ao identificar as entidades participantes e seus atributos. A aplicação dos princípios de normalização permitiu a criação de um banco de dados eficiente, minimizando redundâncias e estabelecendo relações sólidas entre as diferentes partes do sistema. Além disso, a experiência proporcionou insights valiosos sobre técnicas de consulta, otimização e eficiência dos dados. A complexidade do serviço de streaming exigiu uma abordagem estruturada e organizada, refletindo-se na clareza da representação do banco de dados. Em última análise, este projeto não apenas contribuiu para a nossa compreensão prática de design de banco de dados, mas também ofereceu uma visão aprofundada do funcionamento interno de um serviço de streaming. Estamos confiantes de que as habilidades adquiridas neste processo serão aplicáveis em contextos mais amplos, preparando-nos para enfrentar desafios similares no futuro.