

Projeto Escola - Atividade

Tabela: usuario

Coluna	Tipo	Restrição	Descrição
id_usuario	int	PK	Identificador único do usuário
nome	varchar(100)	NULL	Nome do usuário
data_nascimento	date	NULL	Data de nascimento do usuário
idade	int	NULL	Idade do usuário
tipo_usuario	ENUM ('ADMIN', 'USUARIO', 'ALUNO', 'DOCENTE')	NULL	Tipo de usuário
email	varchar(100)	NULL	Email do usuário
telefone	varchar(15)	NULL	Telefone do usuário
data_cadastro	datetime	NULL	Data de cadastro do usuário
senha	varchar(255)	NULL	Senha do usuário
cpf	varchar(14)	NULL	CPF do usuário

Tabela: curso

Coluna	Tipo	Restrição	Descrição
id_curso	int	PK	Identificador único do curso
nome_curso	varchar(100)	NULL	Nome do curso
descricao	text	NULL	Descrição do curso
carga_horaria	int	NULL	Carga horária do curso

Tabela: matricula

Coluna	Tipo	Restrição	Descrição
id_matricula	int	PK	Identificador único da matrícula
data_matricula	datetime	current_timestamp()	Data da matrícula, insere automaticamente.
id_usuario	int	FK	Referência para id_usuario na tabela usuarios
id_curso	int	FK	Referência para id_curso na tabela cursos
status	boolean	NULL	Status da matrícula
data_inicio	date	NULL	Data de início do curso
data_fim	date	NULL	Data de término do curso

Tabela: nota

Coluna	Tipo	Restrição	Descrição
id_nota	int	PK	Identificador único da nota
id_matricula	int	FK	Referência para id_matricula na tabela matriculas
nota	decimal	NULL	Nota obtida
data_atividade	date	NULL	Data da avaliação
status	boolean	NULL	Status da nota (0 reprovado, 1 aprovado)
observacoes	text	NULL	Observações sobre a avaliação
id_usuario	int	FK	Referência para id_usuario na tabela usuarios
id_curso	int	FK	Referência para id_curso na tabela cursos

```
CREATE TABLE IF NOT EXISTS `usuario` (  
  `id_usuario` int NOT NULL,  
  `nome` varchar(255) DEFAULT '100',  
  `data_nascimento` date,  
  `idade` int,
```

```

    `tipo_usuario` enum DEFAULT '50',
    `email` varchar(255) DEFAULT '100',
    `telefone` varchar(255) DEFAULT '15',
    `data_cadastro` datetime,
    `senha` varchar(255) DEFAULT '255',
    `cpf` varchar(255) DEFAULT '14',
    PRIMARY KEY (`id_usuario`)
);

CREATE TABLE IF NOT EXISTS `curso` (
    `id_curso` int NOT NULL,
    `nome_curso` varchar(255) DEFAULT '100',
    `descricao` text,
    `carga_horaria` int,
    PRIMARY KEY (`id_curso`)
);

CREATE TABLE IF NOT EXISTS `matricula` (
    `data_matricula` datetime,
    `id_matricula` int NOT NULL,
    `id_usuario` int,
    `id_curso` int,
    `status` EMUN("Ativa", "Cancelada", "Pendente"),
    `data_inicio` date,
    `data_fim` date,
    PRIMARY KEY (`id_matricula`)
);

CREATE TABLE IF NOT EXISTS `nota` (
    `status` boolean,
    `id_matricula` int,
    `id_nota` int NOT NULL,
    `nota` decimal(10,0),
    `data_avaliacao` date,
    `observacoes` text,
    `id_usuario` int,
    `id_curso` int,
    PRIMARY KEY (`id_nota`)
);

ALTER TABLE `matricula` ADD CONSTRAINT `matriculas_fk2` FOREIGN KEY (`id_usuario`)
REFERENCES `usuario`(`id_usuario`);
ALTER TABLE `matricula` ADD CONSTRAINT `matriculas_fk3` FOREIGN KEY (`id_curso`)
REFERENCES `curso`(`id_curso`);
ALTER TABLE `notas` ADD CONSTRAINT `notas_fk1` FOREIGN KEY (`id_matricula`) REFERENCES
`matricula`(`id_matricula`);
ALTER TABLE `notas` ADD CONSTRAINT `notas_fk6` FOREIGN KEY (`id_usuario`) REFERENCES
`usuario`(`id_usuario`);
ALTER TABLE `notas` ADD CONSTRAINT `notas_fk7` FOREIGN KEY (`id_curso`) REFERENCES
`cursos`(`id_curso`);

```

Projeto Games online

Dicionário de dados

Tabela: jogadores

Coluna	Tipo	Restrição	Descrição
id_jogador	int	PK	Identificador único do jogador
nome	varchar(100)	NULL	Nome do jogador
data_nascimento	date	NULL	Data de nascimento do jogador
email	varchar(100)	NULL	Email do jogador
senha	varchar(255)	NULL	Senha do jogador
data_cadastro	datetime	NULL	Data de cadastro do jogador

Tabela: jogos

Coluna	Tipo	Restrição	Descrição
id_jogo	int	PK	Identificador único do jogo
nome_jogo	varchar(100)	NULL	Nome do jogo
descricao	text	NULL	Descrição do jogo
data_lancamento	date	NULL	Data de lançamento do jogo
desenvolvedor	varchar(100)	NULL	Desenvolvedor do jogo

Tabela: historico

Coluna	Tipo	Restrição	Descrição
id_historico	int	PK	Identificador único do histórico
id_jogador	int	FK	Referência para id_jogador na tabela jogadores
id_jogo	int	FK	Referência para id_jogo na tabela jogos
data_jogo	datetime	NULL	Data e hora do jogo
duracao	int	NULL	Duração do jogo em minutos
pontuacao	int	NULL	Pontuação obtida no jogo

Tabela: favoritos

Coluna	Tipo	Restrição	Descrição
id_favorito	int	PK	Identificador único do favorito
id_jogador	int	FK	Referência para id_jogador na tabela jogadores
id_jogo	int	FK	Referência para id_jogo na tabela jogos
data_adicao	datetime	NULL	Data de adição aos favoritos

Modelo Entidade-Relacionamento (MER)

O Modelo Entidade-Relacionamento (MER) detalha as entidades, atributos e relacionamentos:

- **jogadores:**
 - PK: id_jogador
 - Atributos: nome, data_nascimento, email, senha, data_cadastro
- **jogos:**
 - PK: id_jogo
 - Atributos: nome_jogo, descricao, data_lancamento, desenvolvedor
- **historico:**
 - PK: id_historico
 - FK: id_jogador -> jogadores.id_jogador
 - FK: id_jogo -> jogos.id_jogo
 - Atributos: data_jogo, duracao, pontuacao

- **favoritos:**
 - PK: id_favorito
 - FK: id_jogador -> jogadores.id_jogador
 - FK: id_jogo -> jogos.id_jogo
 - Atributos: data_adicao

Diagrama de Entidade-Relacionamento (DER)

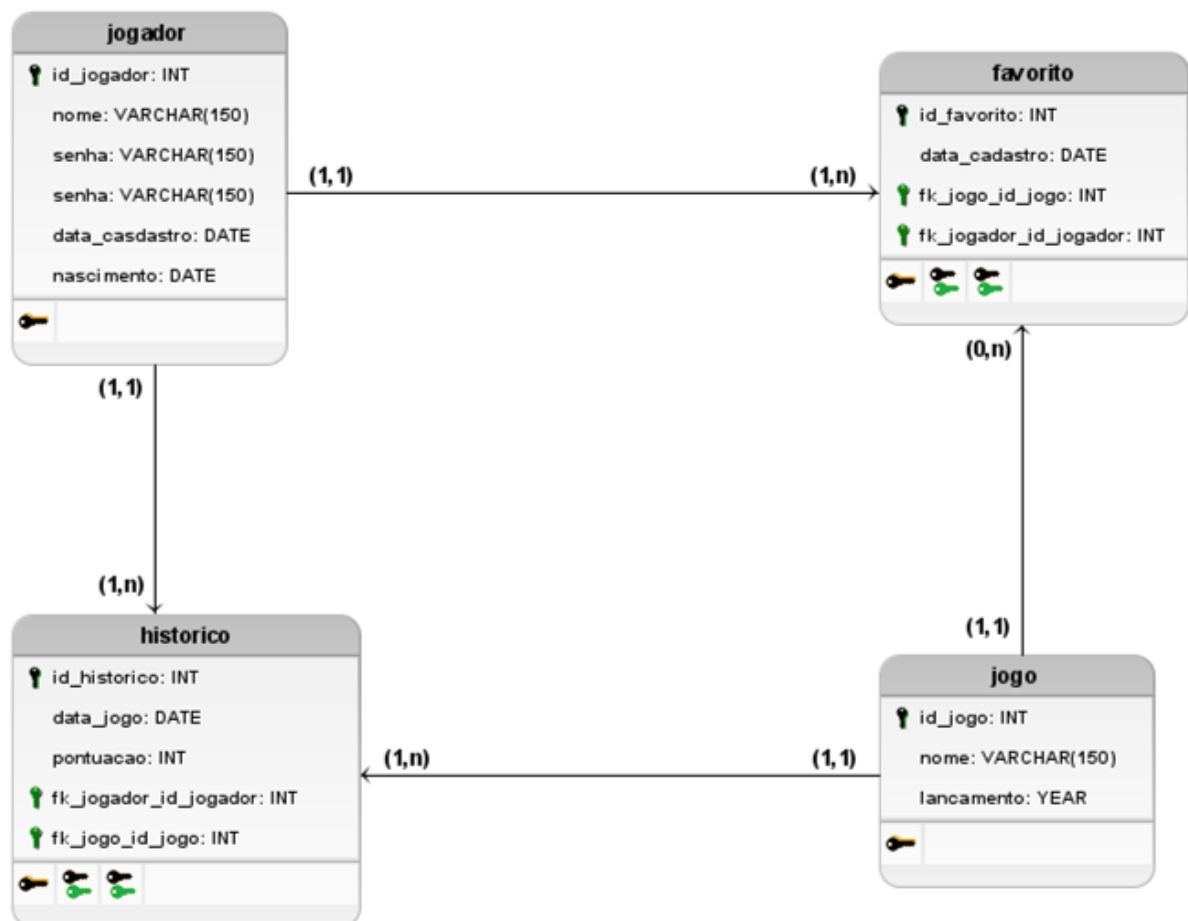
O Diagrama Entidade-Relacionamento (ER) é uma representação gráfica do **MER**.

Descrição textual do ER:

- **jogadores** (id_jogador, nome, data_nascimento, email, senha, data_cadastro)
- **jogos** (id_jogo, nome_jogo, descricao, data_lancamento, desenvolvedor)
- **historico** (id_historico, id_jogador, id_jogo, data_jogo, duracao, pontuacao)
- **favoritos** (id_favorito, id_jogador, id_jogo, data_adicao)

Relacionamentos:

- **jogadores** (1) <-> (N) **historico**
- **jogos** (1) <-> (N) **historico**
- **jogadores** (1) <-> (N) **favoritos**
- **jogos** (1) <-> (N) **favoritos**



Modelo físico - Games

```

CREATE TABLE jogador (
  id_jogador INT PRIMARY KEY,
  nome VARCHAR(100) NULL,
  data_nascimento DATE NULL,
  email VARCHAR(100) NULL,
  senha VARCHAR(255) NULL,

```

```

        data_cadastro DATETIME NULL
    );

CREATE TABLE jogo (
    id_jogo INT PRIMARY KEY,
    nome_jogo VARCHAR(100) NULL,
    descricao TEXT NULL,
    data_lancamento DATE NULL,
    desenvolvedor VARCHAR(100) NULL
);

CREATE TABLE historico (
    id_historico INT PRIMARY KEY,
    id_jogador INT,
    id_jogo INT,
    data_jogo DATETIME NULL,
    duracao INT NULL,
    pontuacao INT NULL,
    FOREIGN KEY (id_jogador) REFERENCES jogadores(id_jogador),
    FOREIGN KEY (id_jogo) REFERENCES jogos(id_jogo)
);

CREATE TABLE favorito (
    id_favorito INT PRIMARY KEY,
    id_jogador INT,
    id_jogo INT,
    data_adicao DATETIME NULL,
    FOREIGN KEY (id_jogador) REFERENCES jogadores(id_jogador),
    FOREIGN KEY (id_jogo) REFERENCES jogos(id_jogo)
);

```

Projeto Cursos Online

Dicionário de Dados

Tabela: usuarios

Coluna	Tipo	Restrição	Descrição
id_usuario	int	PK	Identificador único do usuário
nome	varchar(100)	NULL	Nome do usuário
data_nascimento	date	NULL	Data de nascimento do usuário
idade	int	NULL	Idade do usuário
tipo_usuario	enum('ADMIN', 'ALUNO', 'PROFESSOR')	NULL	Tipo de usuário
email	varchar(100)	NULL	Email do usuário
telefone	varchar(15)	NULL	Telefone do usuário
data_cadastro	datetime	NULL	Data de cadastro do usuário
senha	varchar(255)	NULL	Senha do usuário
cpf	varchar(14)	NULL	CPF do usuário

Tabela: cursos

Coluna	Tipo	Restrição	Descrição
id_curso	int	PK	Identificador único do curso
nome_curso	varchar(100)	NULL	Nome do curso
descricao	text	NULL	Descrição do curso
carga_horaria	int	NULL	Carga horária do curso

Tabela: matriculas

Coluna	Tipo	Restrição	Descrição
id_matricula	int	PK	Identificador único da matrícula
data_matricula	datetime	NULL	Data da matrícula
id_usuario	int	FK	Ref. para id_usuario na tabela usuarios
id_curso	int	FK	Referência para id_curso na tabela cursos
status	enum('At', 'Can', 'Con')	NULL	Status da matrícula
data_inicio	date	NULL	Data de início do curso
data_fim	date	NULL	Data de término do curso

Tabela: notas

Coluna	Tipo	Restr.	Descrição
id_nota	int	PK	Identificador único da nota
status	enum('PEN', 'APRO', 'REP')	NULL	Status da nota
id_matricula	int	FK	Ref. para id_matricula na tabela matriculas
nota	decimal(5,2)	NULL	Nota obtida
data_avaliacao	date	NULL	Data da avaliação
observacoes	text	NULL	Observações sobre a avaliação
id_usuario	int	FK	Ref. para id_usuario na tabela usuarios
id_curso	int	FK	Ref. para id_curso na tabela cursos

Modelo Entidade-Relacionamento (MER)

O Modelo Entidade-Relacionamento (MER) detalha as entidades, atributos e relacionamentos:

- **usuarios:**
 - PK: id_usuario
 - Atributos: nome, data_nascimento, idade, tipo_usuario, email, telefone, data_cadastro, senha, cpf
- **cursos:**
 - PK: id_curso
 - Atributos: nome_curso, descricao, carga_horaria
- **matriculas:**
 - PK: id_matricula
 - FK: id_usuario -> usuarios.id_usuario
 - FK: id_curso -> cursos.id_curso
 - Atributos: data_matricula, status, data_inicio, data_fim
- **notas:**
 - PK: id_nota
 - FK: id_matricula -> matriculas.id_matricula
 - FK: id_usuario -> usuarios.id_usuario
 - Atributos: status, nota, data_avaliacao, observacoes

Diagrama de Entidade-Relacionamento (DER)

O Diagrama de Entidade-Relacionamento (DER) é uma representação gráfica do **MER**:

Descrição textual do ER:

- **usuarios** (id_usuario, nome, data_nascimento, idade, tipo_usuario, email, telefone, data_cadastro, senha, cpf)

- **cursos** (id_curso, nome_curso, descricao, carga_horaria)
- **matriculas** (id_matricula, data_matricula, id_usuario, id_curso, status, data_inicio, data_fim)
- **notas** (id_nota, status, id_matricula, nota, data_avaliacao, observacoes, id_usuario)

Relacionamentos:

- **usuarios** (1) <-> (N) **matriculas**
- **cursos** (1) <-> (N) **matriculas**
- **matriculas** (1) <-> (N) **notas**
- **usuarios** (1) <-> (N) **notas**

