

--DDL

CREATE DATABASE RpgBdAds1

GO

USE RpgBdAds1

GO

CREATE TABLE Usuarios

(
 IdUsuario INT PRIMARY KEY IDENTITY,
 Email VARCHAR(100) UNIQUE NOT NULL,
 Apelido VARCHAR(20) UNIQUE NOT NULL,
 Senha VARCHAR(30) NOT NULL
)

GO

CREATE TABLE Classes

(
 IdClasse INT PRIMARY KEY IDENTITY,
 Nome VARCHAR(30) UNIQUE NOT NULL,
 Descricao VARCHAR(255)
)

GO

CREATE TABLE Personagens

(
 IdPersonagem INT PRIMARY KEY IDENTITY,
 Nome VARCHAR(30) UNIQUE NOT NULL,
 IdUsuario INT UNIQUE FOREIGN KEY REFERENCES Usuarios(IdUsuario),
 IdClasse INT FOREIGN KEY REFERENCES Classes(IdClasse)

)

GO

CREATE TABLE Habilidades

(

IdHabilidades INT PRIMARY KEY IDENTITY,

Nome VARCHAR(50) UNIQUE NOT NULL,

Descricao VARCHAR(255)

)

GO

CREATE TABLE ClasseHabilidade

(

IdClasse INT FOREIGN KEY REFERENCES Classes(IdClasse),

IdHabilidades INT FOREIGN KEY REFERENCES Habilidades(IdHabilidades)

)

GO

--DML

INSERT INTO Usuarios (Email, Senha, Apelido) Values ('email@email.com', '1234', 'Rambo')

INSERT INTO Classes VALUES ('Barbaro', 'Descrição do barbaro')

INSERT INTO Habilidades VALUES ('Lança Mortal', 'Descrição da Lança Mortal'), ('Escudo Supremo', 'Descrição do Escudo Supremo')

INSERT INTO Personagens VALUES ('DeuBug', 1, 1)

INSERT INTO ClasseHabilidade VALUES (1, 1), (1, 2)

```
INSERT INTO Usuarios VALUES('email@email2.com', 'SoulKiller', 1234)
```

```
INSERT INTO Classes VALUES('Monge', 'Descrição Classe Monge')
```

```
INSERT INTO Habilidades VALUES('Recuperar Vida', 'Descrição Recuperar Vida')
```

```
INSERT INTO Personagens VALUES('Bitbug', 3, 3)
```

```
INSERT INTO ClasseHabilidade VALUES(2, 2), (2, 3)
```

```
UPDATE Usuarios
```

```
SET Senha = 123456
```

```
WHERE IdUsuario = 1
```

```
--DQL
```

```
SELECT * FROM Usuarios
```

```
SELECT Senha, Email FROM Usuarios
```

```
SELECT * FROM Classes
```

```
SELECT * FROM Habilidades
```

```
SELECT * FROM Personagens
```

```
INNER JOIN Classes
```

```
ON Personagens.IdClasse = Classes.IdClasse
```

```
SELECT Personagens.Nome, Classes.Nome, Classes.Descricao FROM Personagens
```

```
INNER JOIN Classes
```

```
ON Personagens.IdClasse = Classes.IdClasse
```

```
SELECT P.Nome, C.Nome 'Classe', C.Descricao FROM Personagens P
```

```
INNER JOIN Classes C
```

```
ON P.IdClasse = C.IdClasse
```

```
CREATE DATABASE ExemploJoin
GO
```

```
USE ExemploJoin
GO
```

```
CREATE TABLE NomeA(
    Nome VARCHAR(50) NOT NULL
)
GO
```

```
CREATE TABLE NomeB(
    Nome VARCHAR(50) NOT NULL
)
GO
```

```
INSERT INTO NomeA VALUES ('Fernanda'), ('Maria'), ('Luiz'), ('Caique')
INSERT INTO NomeB VALUES ('Luiz'), ('Caique'), ('Odirlei'), ('Jessica')
```

--JOIN (INNER JOIN) - Traz somente onde ha equivalencia.

```
SELECT * FROM NomeA
JOIN NomeB
ON NomeA.Nome = NomeB.Nome
```

---- LEFT JOIN - Trás os elementos da PRIMEIRA tabela citada e sua equivalencias na SEGUNDA tabela.

```
SELECT * FROM NomeA
LEFT JOIN NomeB
ON NomeA.Nome = NomeB.Nome
```

-- RIGHT JOIN - Trás os elementos da SEGUNDA tabela citada e sua equivalencias na PRIMEIRA tabela.

```
SELECT * FROM NomeA
RIGHT JOIN NomeB
```

ON NomeA.Nome = NomeB.Nome

--FULL OUTER JOIN - Tras todos os elementos.

SELECT * FROM NomeA

FULL OUTER JOIN NomeB

ON NomeA.Nome = NomeB.Nome