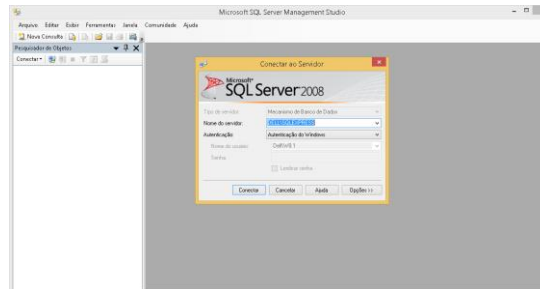


PARA
SER,
UTP

SQL Service Management Studio

utp.edu.br

- Ferramenta para manipular o banco de dados



PARA SER, UTP.



Universidade
Tuiuti do
Paraná

PARA
SER,
UTP

Login

utp.edu.br

- Um login é uma conta utilizada para acessar a **instância** do banco de dados.
- Create login <username> with password = '<password>'
- Create login meuLogin with WITH PASSWORD = '1234'

SELECT * FROM syslogins

EXEC sp_helplogins

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

PARA
SER,
UTP

Usuários

utp.edu.br

- Um usuário é uma conta utilizada para acessar o banco de dados.
- `Create user <username> for login <loginname>`
- `create user TestUser for login TestLogin`

Usuario corrente
`SELECT SYSTEM_USER`

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

PARA
SER,
UTP

Permissões

utp.edu.br

- Permissões são regras governando o nível de acesso. Permissões podem ser concedidas (Grant) ou revogadas (Revoke).
- `Grant select on TestTable to TestUser`
- `Revoke select on TestTable from TestUser`

`GRANT SELECT ON dbo.PILOTS TO teste` PARA SER, UTP.

`REVOKE SELECT ON dbo.PILOTS FROM teste`



Universidade
Tuiuti do
Paraná

PARA
SER,
UTP.

Permissões

utp.edu.br

- GRANT ALL ON dbo.PILOTS TO teste
- Permissões de função escalar: EXECUTE, REFERENCES.
- Permissões de função com valor de tabela: DELETE, INSERT, REFERENCES, SELECT, UPDATE.
- Permissões de procedimento armazenado: EXECUTE.
- Permissões de tabela: DELETE, INSERT, REFERENCES, SELECT, UPDATE.
- Permissões de exibição: DELETE, INSERT, REFERENCES, SELECT, UPDATE.

PARA SER, UTP.

Universidade
Tuiuti do
ParanáPARA
SER,
UTP.

Esquema

utp.edu.br

- Esquema é um namespace distinto que existe independentemente do usuário do banco de dados que o criou.
- É um contêiner de objetos que pode ser de propriedade de qualquer usuário e sua propriedade é transferível.

PARA SER, UTP.

Universidade
Tuiuti do
Paraná

PARA
SER,
UTP.

utp.edu.br

DDL

Data definition language

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

PARA
SER,
UTP.

Criar um banco de dados

utp.edu.br

Criar banco de dados

```
CREATE DATABASE Aviacao
```

Selecionar o banco de dados

```
USE Aviacao
```

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

PARA
SER,
UTP

Normalização

utp.edu.br

- **1ª Forma Normal**

- A tabela não deve conter grupos repetidos e nem atributos com mais de um valor.
- De:
 - PESSOAS = { ID + NOME + ENDERECO + TELEFONES }
- Para:
 - PESSOAS = { ID + NOME + ENDERECO }
 - TELEFONES = { PESSOA_ID + TELEFONE }

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

PARA
SER,
UTP

Normalização

utp.edu.br

- **2ª Forma Normal**

- Atributos não chaves da tabela devem depender unicamente da chave primária (não podendo depender apenas de parte dela).
- De:
 - ALUNOS_CURSOS = { ID_ALUNO + ID_CURSO + NOTA + DESCRICAO_CURSO }
- Para:
 - ALUNOS_CURSOS = { ID_ALUNO + ID_CURSO + NOTA }
 - CURSOS = { ID_CURSO + DESCRICAO }

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

PARA
SER,
UTP

Normalização

utp.edu.br

• 3ª Forma Normal

- Atributos não chave de uma tabela devem ser mutuamente independentes e dependentes unicamente e exclusivamente da chave primária.
- De:
 - FUNCIONARIOS = { ID + NOME + ID_CARGO + DESCRICAO_CARGO }
- Para:
 - FUNCIONARIOS = { ID + NOME + ID_CARGO }
 - CARGOS = { ID_CARGO + DESCRICAO }

PARA SER, UTP.



Universidade
Tuiuti do
Paraná

PARA
SER,
UTP

Schema

utp.edu.br

```
CREATE TABLE Pilots (
  ID int PRIMARY KEY,
  First_name nvarchar(100) NOT
  NULL,
  Last_name nvarchar(100) NOT
  NULL,
  Title nvarchar(100) NOT NULL,
  Birth_date date NOT NULL
);
```

```
CREATE TABLE Planes (
  ID int PRIMARY KEY,
  Model nvarchar(100) NOT NULL,
  Built_date date NOT NULL,
  Number_of_seats int NOT NULL
);
```

```
CREATE TABLE Flights (
  ID int PRIMARY KEY,
  [From] nvarchar(100) NOT NULL,
  [To] nvarchar(100) NOT NULL,
  Departure_date date NOT NULL,
  Planes_ID int NOT NULL,
  constraint Planes_fk foreign key (Planes_ID) references
  Planes(ID)
);
```



Universidade
Tuiuti do
Paraná

Schema

```
CREATE TABLE Pilots_on_flights (
  Flights_ID int NOT NULL,
  Pilots_ID int NOT NULL,
  constraint Pilots_on_flights_pk primary key (Flights_ID,
  Pilots_ID),
  constraint Flightsfk foreign key (Flights_ID) references
  Flights(ID),
  constraint Pilots_fk foreign key (Pilots_ID) references
  Pilots(ID)
);
```

Schema Browser

- ✦ Flights (TABLE)
- ✦ Pilots (TABLE)
- ✦ Pilots_on_flights (TABLE)
- ✦ Planes (TABLE)

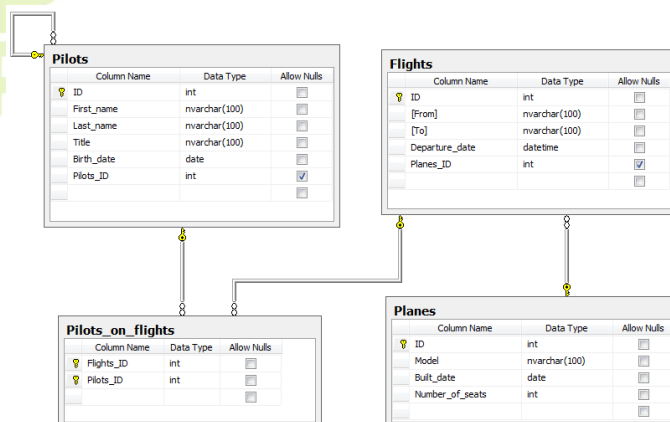
PARA SER, UTP.



Universidade
Tuiuti do
Paraná

utp.edu.br

Esquema banco de dados



PARA SER, UTP.



Universidade
Tuiuti do
Paraná

utp.edu.br