SQLCONNECTION + RELATION MAPPING

DOMAIN DEFINITION LANGUAGE

Agenda

- Map relationships
  - 1-1
  - 1-N
  - N-N
  - Inheritance
- SQL connection

Primary Keys

- Primary key inline

```
CREATE TABLE Movies (
  title varchar(50) PRIMARY KEY,
  year varchar(4), studio varchar(50),
  genre varchar(40),
)
```

- Primary key as constraint

```
CREATE TABLE Movies (
  title varchar(50), year varchar(4),
  studio varchar(50), genre varchar(40),
  CONSTRAINT pk_title PRIMARY KEY (title)
)
```

- Primary key spanning multiple columns

```
CREATE TABLE Movies (
  title varchar(50), year varchar(4),
  studio varchar(50), genre varchar(40),
  CONSTRAINT pk_title_year
    PRIMARY KEY (title, year)
)
```

Map 1-1 relations ships

- Interest defines where key is located

```
CREATE TABLE Car (
    reg VARCHAR(20) PRIMARY KEY,
    color VARCHAR(7),
)

CREATE TABLE ParkingSpot (
    id INT PRIMARY KEY,
    address VARCHAR(80),
    car_reg VARCHAR(20) UNIQUE
      FOREIGN KEY REFERENCES Car(reg),
)
```

Map 1-N relations ships

- Set the foreign key constraint in N entity

```
CREATE TABLE Car (
    reg VARCHAR(20) PRIMARY KEY,
    color VARCHAR(7),
)

CREATE TABLE Wheels(
    id INT PRIMARY KEY,
    placement VARCHAR(2),
    car_reg VARCHAR(20)
      FOREIGN KEY REFERENCES Car(reg)
)
```

Map N-N relations ships (1/2)

- In CarDriver driver_cpr/car_reg pair
- Optional define pair as unique

```
CREATE TABLE Car (
    reg VARCHAR(20) PRIMARY KEY,
    color VARCHAR(7),
)

CREATE TABLE Drivers(
    cpr INT PRIMARY KEY,
    Age int,
    Name VARCHAR(50)
)

CREATE TABLE CarDriver(
    driver_cpr INT
       FOREIGN KEY REFERENCES Drivers(cpr),
    car_reg VARCHAR(20)
       FOREIGN KEY REFERENCES Car(reg)
)
```

Map N-N relations ships (2/2)

```
-- Or if the driver/car couple should be unique
CREATE TABLE CarDriver(
    driver_cpr INT
       FOREIGN KEY REFERENCES Drivers(cpr),
    car_reg VARCHAR(20)
        FOREIGN KEY REFERENCES Car(reg),
    CONSTRAINT uc_cpr_reg
        UNIQUE (driver_cpr, car_reg)
)

-- To create relations insert into CarDriver
INSERT INTO CAR VALUES
  ('12345', '123145'), ('123145', '21sdaf')
INSERT INTO Drivers VALUES
  (123456, 12, 'Alice'), (43535, 18, 'Bob')
INSERT INTO CarDriver VALUES
  (123456,'12345'), (43535, '123145'),
  (123456,'123145')
```

Map inheritance

- Either in single table
- or as a 1-1 relation with required existence

```
CREATE TABLE Products (
  id INT PRIMARY KEY,
  price INT, soup_type VARCHAR(10),
  expiration DATE, bakedOn VARCHAR(120),
  slices int, type VARCHAR(5)
)

INSERT INTO Products
 (id, price, soup_type, expiration, type)
VALUES (1, 10, 'mushroom', getDate(), 'soup'),
       (2, 15, 'shrimp', getDate(), 'soup')

INSERT INTO Products
  (id, price, bakedOn, slices, type)
VALUES (3, 9, 'full', 30, 'bread'),
       (4, 12, 'white', 25, 'bread')
```

How to install

- .Net Core 3.0+ (I'll be using 3.1)

  1. $ mkdir SqlConnectionExample; cd SqlConnectionExample
  2. $ dotnet new console
  3. $ dotnet add package Microsoft.Data.SqlClient --version 1.1.0
  4. Or via package manager PM> Install-Package Microsoft.Data.SqlClient -Version 1.1.0
  5. $ dotnet run

SqlConnection

```
private static void CreateCommand(
    string queryString,
    string connectionString)
{
    using (SqlConnection connection
      = new SqlConnection(connectionString))
    {
        SqlCommand command = new SqlCommand(
             queryString, connection);
        command.Connection.Open();
        command.ExecuteNonQuery();
    }
}
```

SqlCommand

```
private static void ReadOrderData(
                    string connectionString) {
  string queryString =
      "SELECT OrderID, CustomerID
         FROM dbo.Orders;";
  using (SqlConnection connection =
       new SqlConnection(
             connectionString)) {
   SqlCommand command = new SqlCommand(
       queryString, connection);
   connection.Open();
   using(SqlDataReader reader =
                 command.ExecuteReader()) {
     while (reader.Read()) {
        Console.WriteLine(String.Format("{0}, {1}",
         reader[0], reader[1]));
} } } }
```

Exercises


YEAH, IF I COULD CONNECT TO THE DATABASE
THAT'D BE GREAT
makeameme.org


SOLIDUM PETIT IN PROFUNDIS
UNIVERSITAS ARHUSIENSIS

References

Frontpage meme: https://rtask.thinkr.fr/the-ten-commandments-for-a-well-formatted-database/

Exercise gif: https://giphy.com/gifs/13HgwGsXF0aiGY/media