# Упражнения: Типове връзки

## One-To-One връзка

Създайте две таблици. Използвайте подходящи типове данни.

|  |  |  |  |
| --- | --- | --- | --- |
| **Persons** | | | |
| **person\_id** | **first\_name** | **Salary** | **passport\_id** |
| 1 | Roberto | 43300.00 | 102 |
| 2 | Tom | 56100.00 | 103 |
| 3 | Yana | 60200.00 | 101 |

|  |  |
| --- | --- |
| **passports** | |
| **passport\_id** | **passport\_number** |
| 101 | N34FG21B |
| 102 | K65LO4R7 |
| 103 | ZE657QP2 |

Вкарайте данните от примера по-горе.

Изменете таблицата **persons**, така че да направите **person\_id** първичен ключ. Създайте външен ключ между **persons** и **passports** чрез **passport\_id** колоната.

use excercise\_db\_2;

CREATE TABLE passports(

passport\_id INT (11) PRIMARY KEY AUTO\_INCREMENT,

passport\_number VARCHAR(8)

# CONSTRAINT pk\_passports\_pasport\_id

# PRIMARY KEY (passport\_id)

);

CREATE TABLE persons(

person\_id INT(11) PRIMARY KEY AUTO\_INCREMENT,

first\_name VARCHAR(50),

salary DECIMAL(8, 2),

passport\_id INT(11) UNIQUE,

constraint fk\_persons\_passports

foreign key (passport\_id)

references passports(passport\_id)

);

## One-To-Many връзка

Създайте две таблици. Използвайте подходящи типове данни.

|  |  |  |
| --- | --- | --- |
| **manufacturers** | | |
| **manufacturer\_id** | **name** | **established\_on** |
| 1 | BMW | 01/03/1916 |
| 2 | Tesla | 01/01/2003 |
| 3 | Lada | 01/05/1966 |

|  |  |  |
| --- | --- | --- |
| **models** | | |
| **model\_id** | **name** | **manufacturer\_id** |
| 101 | X1 | 1 |
| 102 | i6 | 1 |
| 103 | Model S | 2 |
| 104 | Model X | 2 |
| 105 | Model 3 | 2 |
| 106 | Nova | 3 |

Вкарайте данните от примера по-горе. Добавете първични ключове и външни ключове.

CREATE TABLE students(

student\_id INT PRIMARY KEY,

name VARCHAR(50)

);

CREATE TABLE exams(

exam\_id INT PRIMARY KEY,

name VARCHAR(50)

);

CREATE TABLE students\_exams(

student\_id INT,

exam\_id INT,

CONSTRAINT pk\_students\_exams PRIMARY KEY(student\_id, exam\_id),

CONSTRAINT fk\_students\_exams\_students FOREIGN KEY(student\_id) REFERENCES students(student\_id),

CONSTRAINT fk\_students\_exams\_exams FOREIGN KEY(exam\_id) REFERENCES exams(exam\_id)

);

## Many-To-Many връзка

Създайте три таблици. Използвайте подходящи типове данни.

|  |  |
| --- | --- |
| **students** | |
| **student\_id** | **name** |
| 1 | Mila |
| 2 | Toni |
| 3 | Ron |

|  |  |
| --- | --- |
| **Exams** | |
| **exam\_id** | **Name** |
| 101 | Spring MVC |
| 102 | Neo4j |
| 103 | Oracle 11g |

|  |  |
| --- | --- |
| **students\_exams** | |
| **student\_id** | **exam\_id** |
| 1 | 101 |
| 1 | 102 |
| 2 | 101 |
| 3 | 103 |
| 2 | 102 |
| 2 | 103 |

Вкарайте данните от примера по-горе. Добавете първични ключове и външни ключове. Забележете, че таблицата **student\_exams** трябва да има съставен (от повече от една колони) първичен ключ.

create table students (

student\_id INT (11) PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR (50)

);

CREATE TABLE exams (

exam\_id INT (11) PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(50)

);

CREATE TABLE students\_exams(

student\_id INT,

exam\_id INT,

CONSTRAINT pk\_students\_exams

PRIMARY KEY (student\_id, exam\_id),

CONSTRAINT fk\_students\_exams\_exams

FOREIGN KEY (exam\_id)

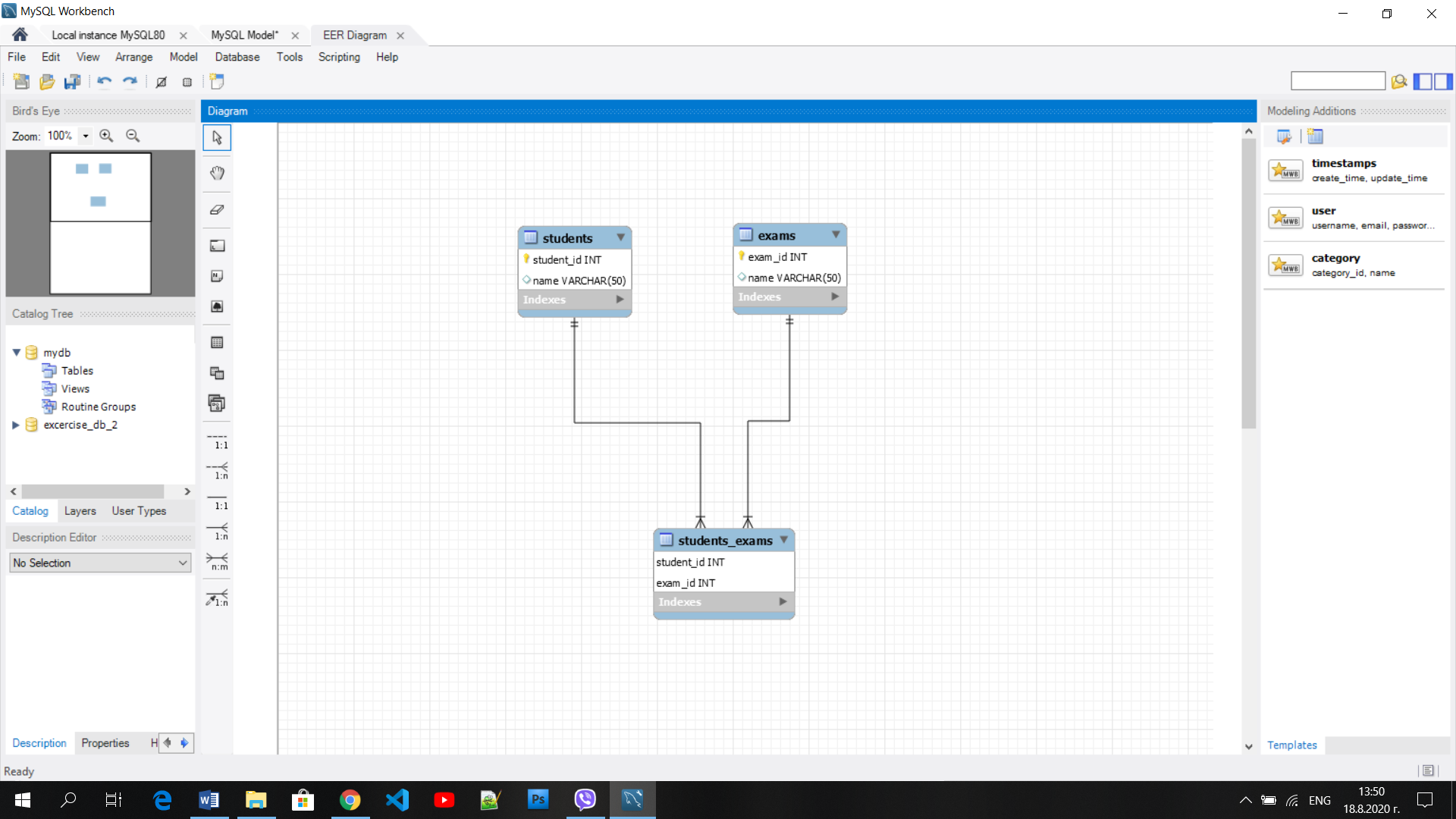
REFERENCES exams(exam\_id),

CONSTRAINT fk\_students\_exams\_students

FOREIGN KEY (student\_id)

REFERENCES students(student\_id)

);



## Самообръщаша се връзка

Създайте една таблица. Използвайте подходящи типове данни.

Това не може при самореферираща се таблица. Затова първо вкарваме в manager\_id да е NULL и след това UPDATE

|  |  |  |
| --- | --- | --- |
| **teachers** | | |
| **teacher\_id** | **name** | **manager\_id** |
| 101 | John |  |
| 102 | Maya | 106 |
| 103 | Silvia | 106 |
| 104 | Ted | 105 |
| 105 | Mark | 101 |
| 106 | Greta | 101 |

Вкарайте данните от примера по-горе. Добавете първични ключове и външни ключове. Външният ключ трябва да е между **manager\_id** и **teacher\_id**.

CREATE TABLE teachers (

teacher\_id INT PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(50),

manager\_id INT,

CONSTRAINT fk\_teachers\_teachers

FOREIGN KEY (manager\_id)

REFERENCES teachers (teacher\_id)

);

INSERT INTO teachers (teacher\_id, name, manager\_id)

VALUES (101, 'John', NULL),

(102, 'Maya', NULL),

(103, 'Silvia', NULL),

(104, 'Ted', NULL),

(105, 'Mark', NULL),

(106, 'Greta', NULL);

Тук се променя за всеки запис и се дава Execute и така се попълват данните

UPDATE teachers

SET manager\_id=101

WHERE teacher\_id IN (105, 106);

SELECT \* FROM teachers – така виждаме табличката с попълнените данни долу:

