Министерство образования и науки Российской Федерации

Севастопольский государственный университет

Институт информационных технологий и управления в технических системах

Кафедра ИС

ОТЧЕТ

по лабораторной работе №3

СОЗДАНИЕ СХЕМЫ БД. ССЫЛОЧНАЯ ЦЕЛОСТНОСТЬ

Выполнил:

ст. гр. ИС/б-21-о

Куркчи А. Э.

Проверил:

Тимофеева Т. И.

Севастополь

2016

1. ЦЕЛЬ РАБОТЫ

Научиться анализировать предметную область с целью создания схемы БД, учитывая ссылочную целостность набора.

2. ВАРИАНТ ЗАДАНИЯ

Вариант № 7

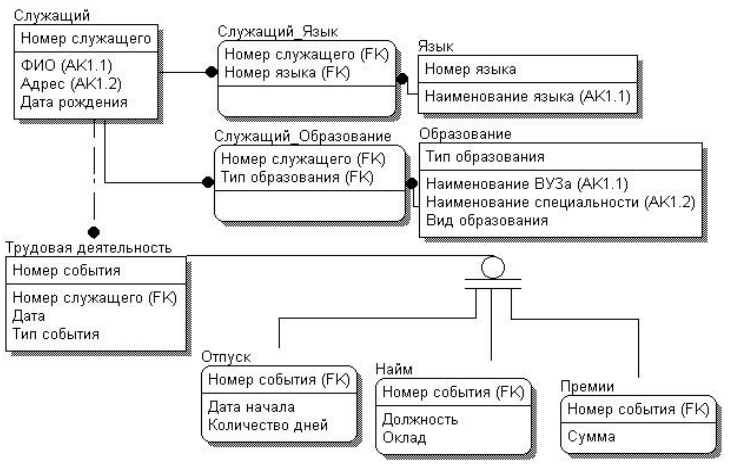


Рисунок 1 — Вариант задания

3. ХОД РАБОТЫ

Ниже представлен набор команд на языке SQL, реализующий создание недостающих таблиц и изменение уже существующей, а также заполнение таблиц тестовыми данными и запросы для проверки.

--CONNECT 'D:\university\semestr.04\DM\Lab.03\db.fdb' USER 'SYSDBA' PASSWORD 'masterkey';

-- from labs before

--CREATE TABLE employee (

-- id INTEGER NOT NULL,

-- name VARCHAR(32) NOT NULL,

-- address VARCHAR(32),

-- b\_date DATE

--);

--INSERT INTO employee VALUES (1, 'Andrew', 'Sevastopol', '12.09.1978');

--INSERT INTO employee VALUES (2, 'John', 'Yalta', NULL);

--INSERT INTO employee VALUES (3, 'Bob', 'Kerch', NULL);

--INSERT INTO employee VALUES (4, 'Mary', 'Sevastopol', '22.09.1996');

--INSERT INTO employee VALUES (5, 'Jerry', 'Simferopol', '09.05.1945');

--INSERT INTO employee VALUES (6, 'Arif', 'Simferopol', '06.10.1997');

--INSERT INTO employee VALUES (7, 'Jane', 'Alushta', NULL);

--INSERT INTO employee VALUES (8, 'Kate', NULL, NULL);

--INSERT INTO employee VALUES (9, 'Iris', NULL, NULL);

--INSERT INTO employee VALUES (10, 'Vladimir', NULL, NULL);

-- changes

**ALTER** **TABLE** employee **ADD** **PRIMARY** **KEY** **(**id**);**

**ALTER** **TABLE** employee **ADD** **UNIQUE** **(**name**,** address**);**

-- create

--- basic tables

**CREATE** **TABLE** lang **(**

id **INTEGER** **NOT** **NULL** **PRIMARY** **KEY,**

name **VARCHAR(**32**)** **NOT** **NULL,**

**UNIQUE** **(**name**)**

**);**

**CREATE** **TABLE** education **(**

id **INTEGER** **NOT** **NULL** **PRIMARY** **KEY,**

name **VARCHAR(**32**)** **NOT** **NULL,**

specialty **VARCHAR(**32**)** **NOT** **NULL,**

description **VARCHAR(**64**),**

**UNIQUE** **(**name**,** specialty**)**

**);**

--- one fk tables

**CREATE** **TABLE** activity **(**

id **INTEGER** **NOT** **NULL** **PRIMARY** **KEY,**

employee\_id **INTEGER** **NOT** **NULL,**

a\_date **DATE,**

a\_type **VARCHAR(**16**)** **NOT** **NULL,**

**FOREIGN** **KEY** **(**employee\_id**)** **REFERENCES** employee **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

**);**

**CREATE** **TABLE** vacation **(**

id **INTEGER** **NOT** **NULL,**

start\_date **DATE** **NOT** **NULL,**

days\_count **INTEGER** **NOT** **NULL,**

**FOREIGN** **KEY** **(**id**)** **REFERENCES** activity **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

**);**

**CREATE** **TABLE** recruitment **(**

id **INTEGER** **NOT** **NULL,**

post **VARCHAR(**32**)** **NOT** **NULL,**

salary **DECIMAL** **NOT** **NULL,**

**FOREIGN** **KEY** **(**id**)** **REFERENCES** activity **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

**);**

**CREATE** **TABLE** premium **(**

id **INTEGER** **NOT** **NULL,**

amount **DECIMAL** **NOT** **NULL,**

**FOREIGN** **KEY** **(**id**)** **REFERENCES** activity **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

**);**

--- pivot tables

**CREATE** **TABLE** employee\_lang **(**

employee\_id **INTEGER** **NOT** **NULL,**

lang\_id **INTEGER** **NOT** **NULL,**

**FOREIGN** **KEY** **(**employee\_id**)** **REFERENCES** employee **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,**

**FOREIGN** **KEY** **(**lang\_id**)** **REFERENCES** lang **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

**);**

**CREATE** **TABLE** employee\_education **(**

employee\_id **INTEGER** **NOT** **NULL,**

education\_id **INTEGER** **NOT** **NULL,**

**FOREIGN** **KEY** **(**employee\_id**)** **REFERENCES** employee **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,**

**FOREIGN** **KEY** **(**education\_id**)** **REFERENCES** education **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

**);**

-- fill

--- basic

---- lang

**INSERT** **INTO** lang **VALUES** **(**1**,** 'Russian'**);**

**INSERT** **INTO** lang **VALUES** **(**2**,** 'English'**);**

**INSERT** **INTO** lang **VALUES** **(**3**,** 'SQL'**);**

**INSERT** **INTO** lang **VALUES** **(**4**,** 'Java'**);**

**INSERT** **INTO** lang **VALUES** **(**5**,** 'PHP'**);**

**INSERT** **INTO** lang **VALUES** **(**6**,** 'C++'**);**

---- education

**INSERT** **INTO** education **VALUES** **(**1**,** 'SevSU'**,** 'IS'**,** 'Information systems'**);**

**INSERT** **INTO** education **VALUES** **(**2**,** 'CFUV'**,** 'BI'**,** 'Buisnes informatics'**);**

--- one fk

---- activity

**INSERT** **INTO** activity **VALUES** **(**1**,** 4**,** '01.09.2014'**,** 'recruitment'**);**

**INSERT** **INTO** activity **VALUES** **(**2**,** 6**,** '01.09.2014'**,** 'recruitment'**);**

**INSERT** **INTO** activity **VALUES** **(**3**,** 4**,** '14.02.2016'**,** 'premium'**);**

**INSERT** **INTO** activity **VALUES** **(**4**,** 6**,** '14.02.2016'**,** 'vacation'**);**

---- vacation

**INSERT** **INTO** vacation **VALUES** **(**4**,** '14.02.2016'**,** 30**);**

---- recruitment

**INSERT** **INTO** recruitment **VALUES** **(**1**,** 'Senior programmer'**,** 200000**);**

**INSERT** **INTO** recruitment **VALUES** **(**2**,** 'TeamLead'**,** 300000**);**

---- premium

**INSERT** **INTO** premium **VALUES** **(**3**,** 150000**);**

--- pivot

---- employee\_lang

**INSERT** **INTO** employee\_lang **VALUES** **(**1**,** 1**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**2**,** 1**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**2**,** 2**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**3**,** 1**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**4**,** 1**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**4**,** 2**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**4**,** 3**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**4**,** 6**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**5**,** 2**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**6**,** 1**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**6**,** 2**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**6**,** 3**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**6**,** 4**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**6**,** 5**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**6**,** 6**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**7**,** 1**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**8**,** 1**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**9**,** 2**);**

**INSERT** **INTO** employee\_lang **VALUES** **(**10**,**1**);**

---- employee\_education

**INSERT** **INTO** employee\_education **VALUES** **(**1**,** 1**);**

**INSERT** **INTO** employee\_education **VALUES** **(**2**,** 2**);**

**INSERT** **INTO** employee\_education **VALUES** **(**3**,** 1**);**

**INSERT** **INTO** employee\_education **VALUES** **(**4**,** 1**);**

**INSERT** **INTO** employee\_education **VALUES** **(**5**,** 1**);**

**INSERT** **INTO** employee\_education **VALUES** **(**6**,** 1**);**

**INSERT** **INTO** employee\_education **VALUES** **(**7**,** 1**);**

**INSERT** **INTO** employee\_education **VALUES** **(**8**,** 2**);**

**INSERT** **INTO** employee\_education **VALUES** **(**9**,** 2**);**

**INSERT** **INTO** employee\_education **VALUES** **(**10**,**2**);**

-- querys

--- insert wrong data to pivot table

**INSERT** **INTO** employee\_lang **VALUES** **(**11**,** 1**);**

--- insert wrong data to one fk table

**INSERT** **INTO** premium **VALUES** **(**5**,** 100500**);**

--- select current langs for employee 6

**SELECT** el**.**lang\_id**,** l**.**name **FROM** employee\_lang el **JOIN** lang l **ON** l**.**id **=** el**.**lang\_id **WHERE** el**.**employee\_id **=** 6**;**

--- delete lang 5

**DELETE** **FROM** lang **WHERE** id **=** 5**;**

--- select changed langs for employee 6

**SELECT** el**.**lang\_id**,** l**.**name **FROM** employee\_lang el **JOIN** lang l **ON** l**.**id **=** el**.**lang\_id **WHERE** el**.**employee\_id **=** 6**;**

--- select activities for users

**SELECT** a**.**id**,** a**.**employee\_id**,** e**.**name**,** a**.**a\_date**,** a**.**a\_type **FROM** activity a **JOIN** employee e **ON** e**.**id **=** a**.**employee\_id**;**

--- update employee id from 6 to 666

**UPDATE** employee **SET** id **=** 666 **WHERE** id **=** 6**;**

--- select changed activities for users

**SELECT** a**.**id**,** a**.**employee\_id**,** e**.**name**,** a**.**a\_date**,** a**.**a\_type **FROM** activity a **JOIN** employee e **ON** e**.**id **=** a**.**employee\_id**;**

4. ПРИМЕРЫ ЗАПРОСОВ

4.1 Попытка вставить в pivot таблицы данных, нарушающих целостность связей.

При помощи команд, представленных на рисунке 2, пытаемся нарушить целостность данных, введя заведомо несуществующие значения id полей и получим ошибку как ответ защиты целостности базы.

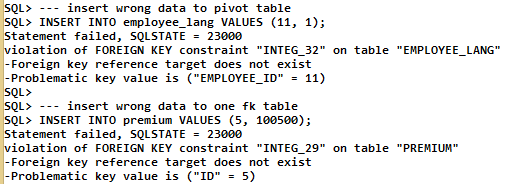


Рисунок 2 — Нарушение целостности через INSERT

4.2 Проверка каскадного удаления:

Выполняя запросы, показанные на рисунке 3, отобразим исходный список языков, которые знает сотрудник 6, затем удалим из базы язык номер 5 и вновь отобразим список языков, известных сотруднику 6. Как видим, язык 5 не отображается в результатах, так как в случае нарушения целостности мы бы наблюдали кортеж (5, <null>).

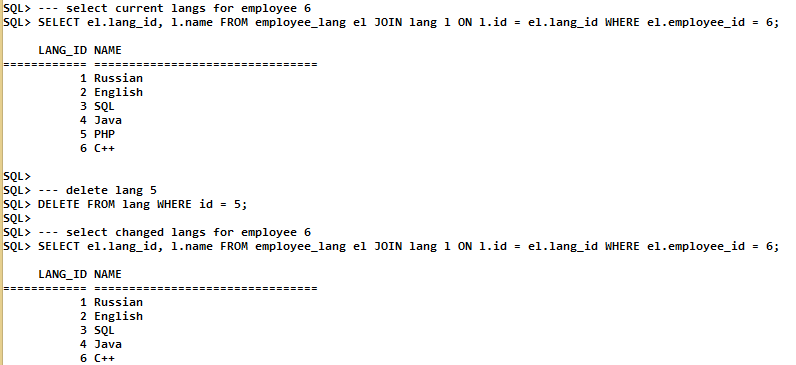


Рисунок 3 — Каскадное удаление

4.3 Проверка каскадной модификации:

Для проверки каскадной модификации таблиц БД выполним запросы, представленные на рисунке 4. Выведем события трудовой деятельности, прикрепив к ним соответствующего им сотрудника, затем изменим id сотрудника 6 на 666 и ещё раз выведем события. В случае нарушения целостности в кортежах 2 и 4 столбцы employee\_id и name были бы равны 6 и <null> соответственно.

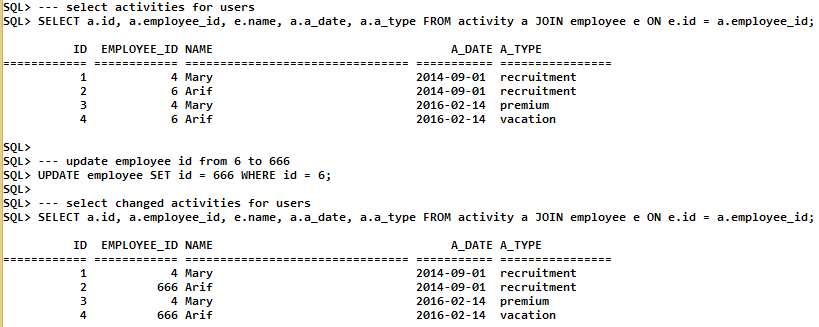


Рисунок 4 — Каскадная модификация

Вывод

В ходе данной лабораторной работы были изучены способы анализа предметной области с целью создания БД, учитывая ссылочную целостность. Создана БД, содержащая первичные, альтернативные и внешние ключи, а также описания ON DELETE и ON UPDATE, которые позволяют поддерживать целостность БД, при работе с ней.