студент Давыдов Н.В. группы ФН12-63Б

Отчет. Семинар 5. Создание базы данных employee

```
--DROP DATABASE IF EXISTS employee;
CREATE DATABASE employee;
```

Statistics 1

Name	Value
Updated Rows	0
Query	--DROP DATABASE IF EXISTS employee; CREATE DATABASE employee
Start time	Thu Mar 28 11:37:15 MSK 2024
Finish time	Thu Mar 28 11:37:15 MSK 2024

```
-- table department
-- serial -> auto increment
DROP TABLE IF EXISTS department;
CREATE TABLE department (
  departmentID serial NOT NULL PRIMARY KEY,
  name varchar(30)
);
```

Name	Value
Updated Rows	0
Query	CREATE TABLE department (departmentID serial NOT NULL PRIMARY KEY, -- serial -> auto increment name varchar(30))
Start time	Thu Mar 28 11:49:35 MSK 2024
Finish time	Thu Mar 28 11:49:35 MSK 2024

```
INSERT INTO department (departmentID, name)
VALUES
  (1, 'Dep_analit'),
  (2, 'Dep_prog'),
  (3, 'Dep_admin');
```

Name	Value
Updated Rows	3
Query	INSERT INTO department (departmentID, name) VALUES (1, 'Dep_analit'), (2, 'Dep_prog'), (3, 'Dep_admin')
Start time	Thu Mar 28 11:51:45 MSK 2024
Finish time	Thu Mar 28 11:51:45 MSK 2024

```sql
-- table employee
DROP TABLE IF EXISTS employee;
CREATE TABLE employee (
  employeeID serial NOT NULL PRIMARY KEY,
  name varchar(80),
  job varchar(30),
  departmentID int NOT NULL,
  CONSTRAINT "DEP"
    FOREIGN KEY (departmentID)
    REFERENCES department(departmentID)
);
```

```
Updated Rows 0
Query          CREATE TABLE employee (
                 employeeID serial NOT NULL PRIMARY KEY,
                 name varchar(80),
                 job varchar(30),
                 departmentID int NOT NULL,
                 CONSTRAINT "DEP"
                   FOREIGN KEY (departmentID)
                   REFERENCES department(departmentID)
               )
Start time     Thu Mar 28 11:52:56 MSK 2024
Finish time    Thu Mar 28 11:52:56 MSK 2024
```

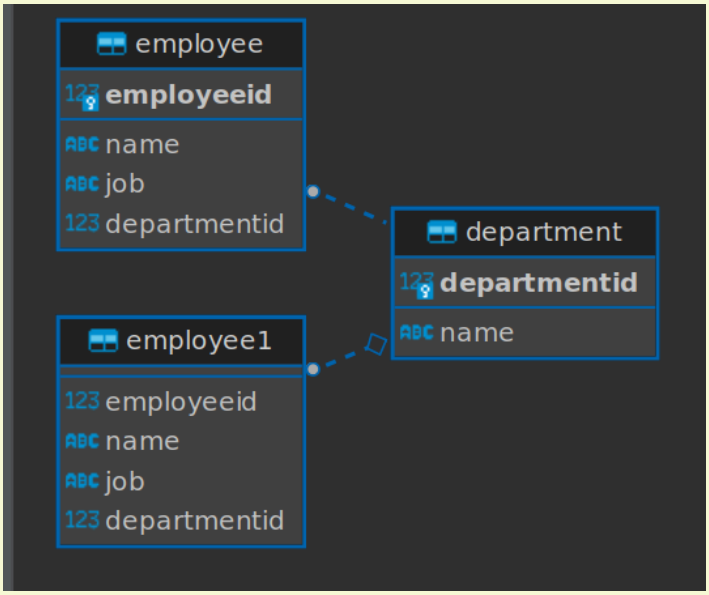
```sql
INSERT INTO employee (employeeID, name, job,
departmentID)
VALUES
  (100, 'Smit N', 'Programmer', 2),
  (101, 'Stone J.', 'manager', 3),
  (102, 'Asser M.', 'analitic', 1),
  (103, 'Wood N.', 'Programmer', 2),
  (104, 'Thomson L.', 'Programmer', 2);
```

```
Updated Rows 5
Query          INSERT INTO employee (employeeID, name, job,
               VALUES
                 (100, 'Smit N', 'Programmer', 2),
                 (101, 'Stone J.', 'manager', 3),
                 (102, 'Asser M.', 'analitic', 1),
                 (103, 'Wood N.', 'Programmer', 2),
                 (104, 'Thomson L.', 'Programmer', 2)
Start time     Thu Mar 28 11:53:55 MSK 2024
Finish time    Thu Mar 28 11:53:55 MSK 2024
```

```sql
-- table employee1
DROP TABLE IF EXISTS employee1;
 CREATE TABLE employee1 AS (
    SELECT * FROM employee
 );

ALTER TABLE employee1
    ADD CONSTRAINT "DEP"
    FOREIGN KEY (departmentID)
    REFERENCES department(departmentID);
```

```
Updated Rows 5
Query          -- table employee1
               DROP TABLE IF EXISTS employee1;
                CREATE TABLE employee1 AS (
                  SELECT * FROM employee
               );
               ALTER TABLE employee1
                  ADD CONSTRAINT "DEP"
                  FOREIGN KEY (departmentID)
                  REFERENCES department(departmentID)
Start time     Thu Mar 28 11:57:27 MSK 2024
Finish time    Thu Mar 28 11:57:27 MSK 2024
```



Проверка контроля целостности

1. Попытаться вставить в таблицы employee и employee1 следующий кортеж:

('Wirt C','Programmer',5);

```
INSERT INTO employee (name, job, departmentID)
VALUES
  ('Wirt C', 'Programmer', 5);
```

2. Вставить в таблицу department кортеж

(10,'Test DELETE');

```
INSERT INTO department
VALUES
  (10, 'Test DELETE');
```

SQL Error [23503]: ERROR: insert or update on table "employee" violates foreign key constraint "DEP"
Detail: Key (departmentid)=(5) is not present in table "department".

<< Details

org.jkiss.dbeaver.model.sql.DBSQLException: SQL Error [23503]: ERROR: insert or update on table "employee
Detail: Key (departmentid)=(5) is not present in table "department".
at org.jkiss.dbeaver.model.impl.jdbc.exec.JDBCStatementImpl.executeStatement(JDBCStatementImpl.ja
at org.jkiss.dbeaver.ui.editors.sql.execute.SQLQueryJob.executeStatement(SQLQueryJob.java:582)

Updated Rows	1
Query	INSERT INTO department VALUES (10, 'Test DELETE')
Start time	Thu Mar 28 12:06:13 MSK 2024
Finish time	Thu Mar 28 12:06:13 MSK 2024

3. Вставить в таблицу employee кортеж

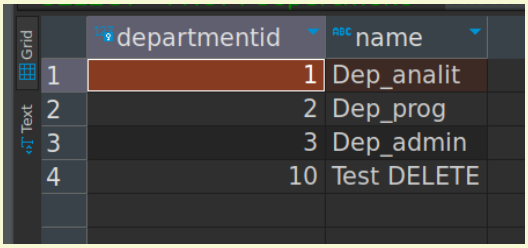
('Wirt C','Programmer',10);

```
INSERT INTO employee (name, job, departmentID)
VALUES
  ('Wirt C', 'Programmer', 10);
```

Updated Rows	1
Query	INSERT INTO employee (name, job, departmentID) VALUES ('Wirt C', 'Programmer', 10)
Start time	Thu Mar 28 12:07:46 MSK 2024
Finish time	Thu Mar 28 12:07:46 MSK 2024

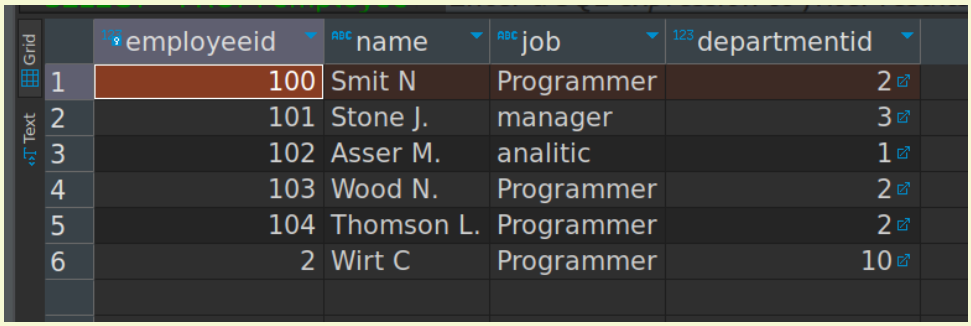
4. Посмотреть содержимое таблиц department , employee

```
SELECT * FROM department;
```



	departmentid	name
1	1	Dep_analit
2	2	Dep_prog
3	3	Dep_admin
4	10	Test DELETE

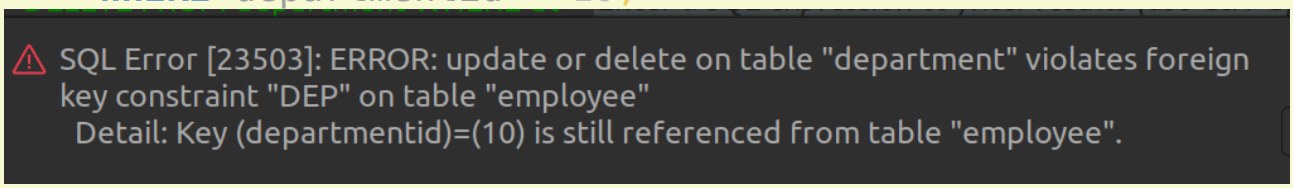
```
SELECT * FROM employee;
```



	employeeid	name	job	departmentid
1	100	Smit N	Programmer	2
2	101	Stone J.	manager	3
3	102	Asser M.	analitic	1
4	103	Wood N.	Programmer	2
5	104	Thomson L.	Programmer	2
6	2	Wirt C	Programmer	10

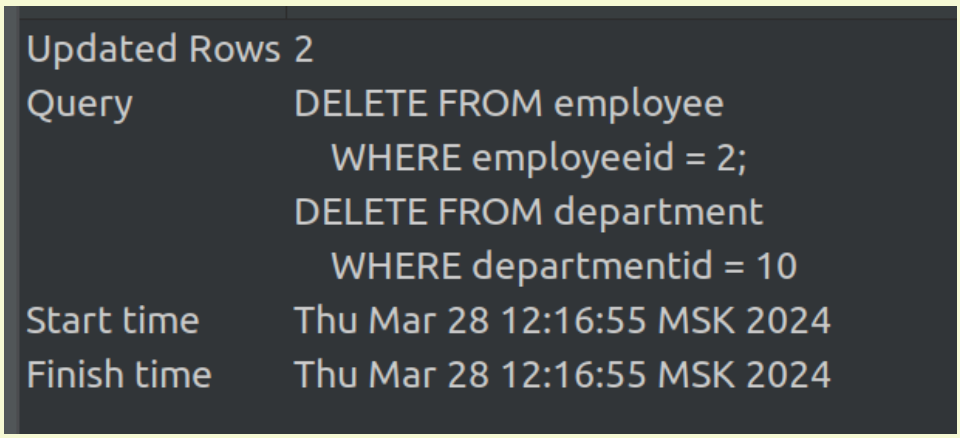
5. Удалить из таблицы department кортеж
(10,'Test DELETE');

```
DELETE FROM department
    WHERE departmentid = 10;
```



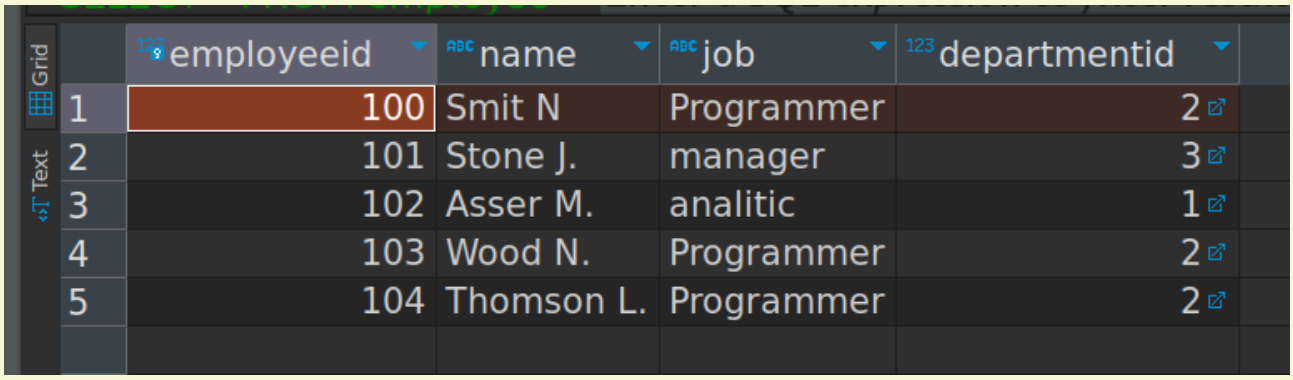
SQL Error [23503]: ERROR: update or delete on table "department" violates foreign key constraint "DEP" on table "employee"
Detail: Key (departmentid)=(10) is still referenced from table "employee".

```
DELETE FROM employee
    WHERE employeeid = 2;
DELETE FROM department
    WHERE departmentid = 10;
```

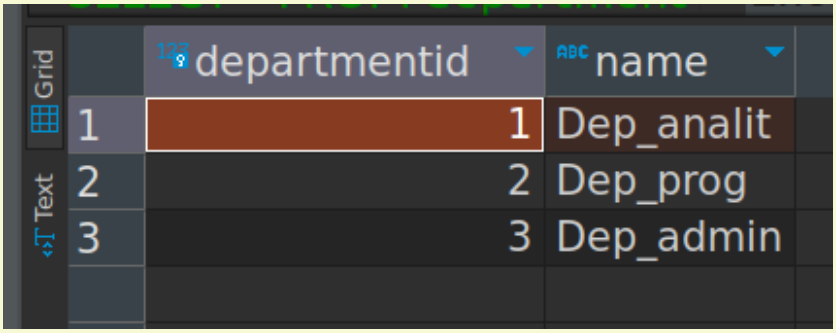


Updated Rows 2
Query DELETE FROM employee
WHERE employeeid = 2;
DELETE FROM department
WHERE departmentid = 10
Start time Thu Mar 28 12:16:55 MSK 2024
Finish time Thu Mar 28 12:16:55 MSK 2024

6. Проверить содержимое таблиц department , employee.



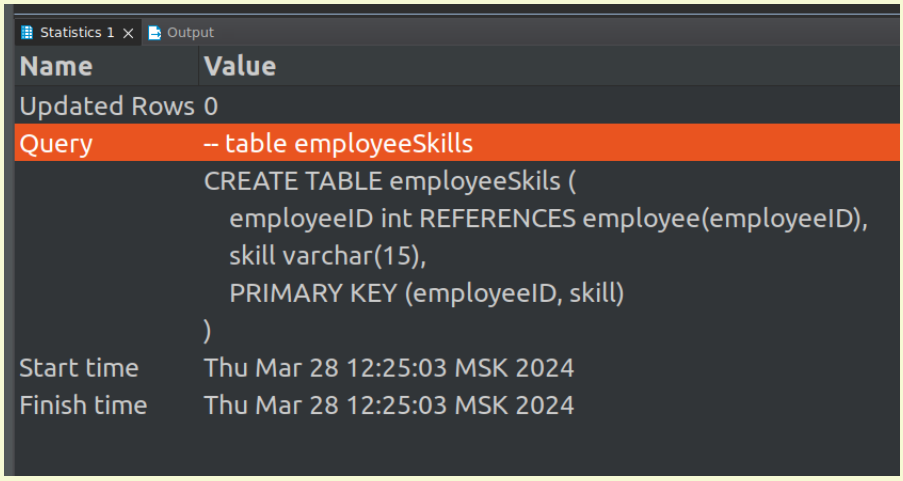
	employeeid	name	job	departmentid
1	100	Smit N	Programmer	2
2	101	Stone J.	manager	3
3	102	Asser M.	analitic	1
4	103	Wood N.	Programmer	2
5	104	Thomson L.	Programmer	2



	departmentid	name
1	1	Dep_analit
2	2	Dep_prog
3	3	Dep_admin

Таблицы **employeeSkills, client, assignment**

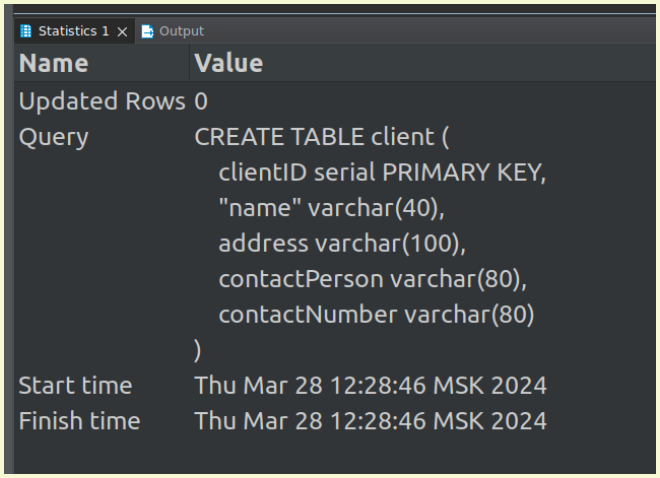
```
CREATE TABLE employeeSkils (
    employeeID int REFERENCES employee(employeeID),
    skill varchar(15),
    PRIMARY KEY (employeeID, skill)
);
```



Statistics 1 | Output

Name	Value
Updated Rows	0
Query	-- table employeeSkills CREATE TABLE employeeSkils (employeeID int REFERENCES employee(employeeID), skill varchar(15), PRIMARY KEY (employeeID, skill))
Start time	Thu Mar 28 12:25:03 MSK 2024
Finish time	Thu Mar 28 12:25:03 MSK 2024

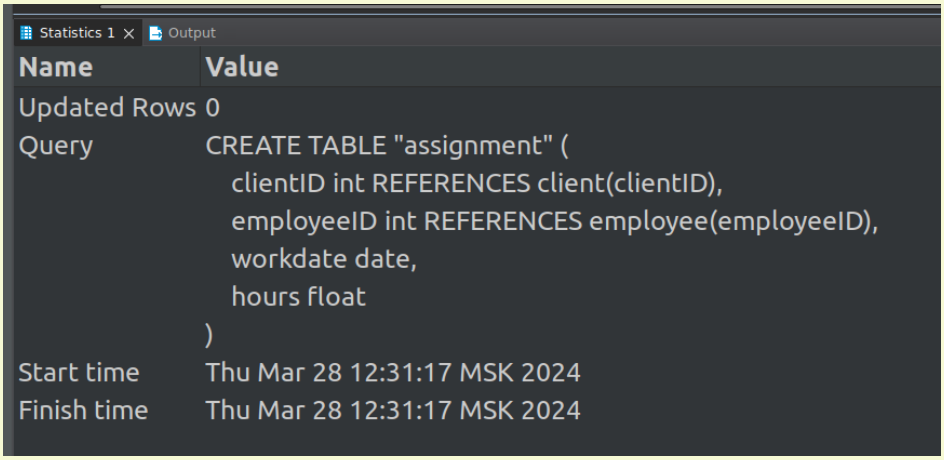
```sql
CREATE TABLE client (
    clientID serial PRIMARY KEY,
    "name" varchar(40),
    address varchar(100),
    contactPerson varchar(80),
    contactNumber varchar(80)
);
```



Statistics 1 | Output

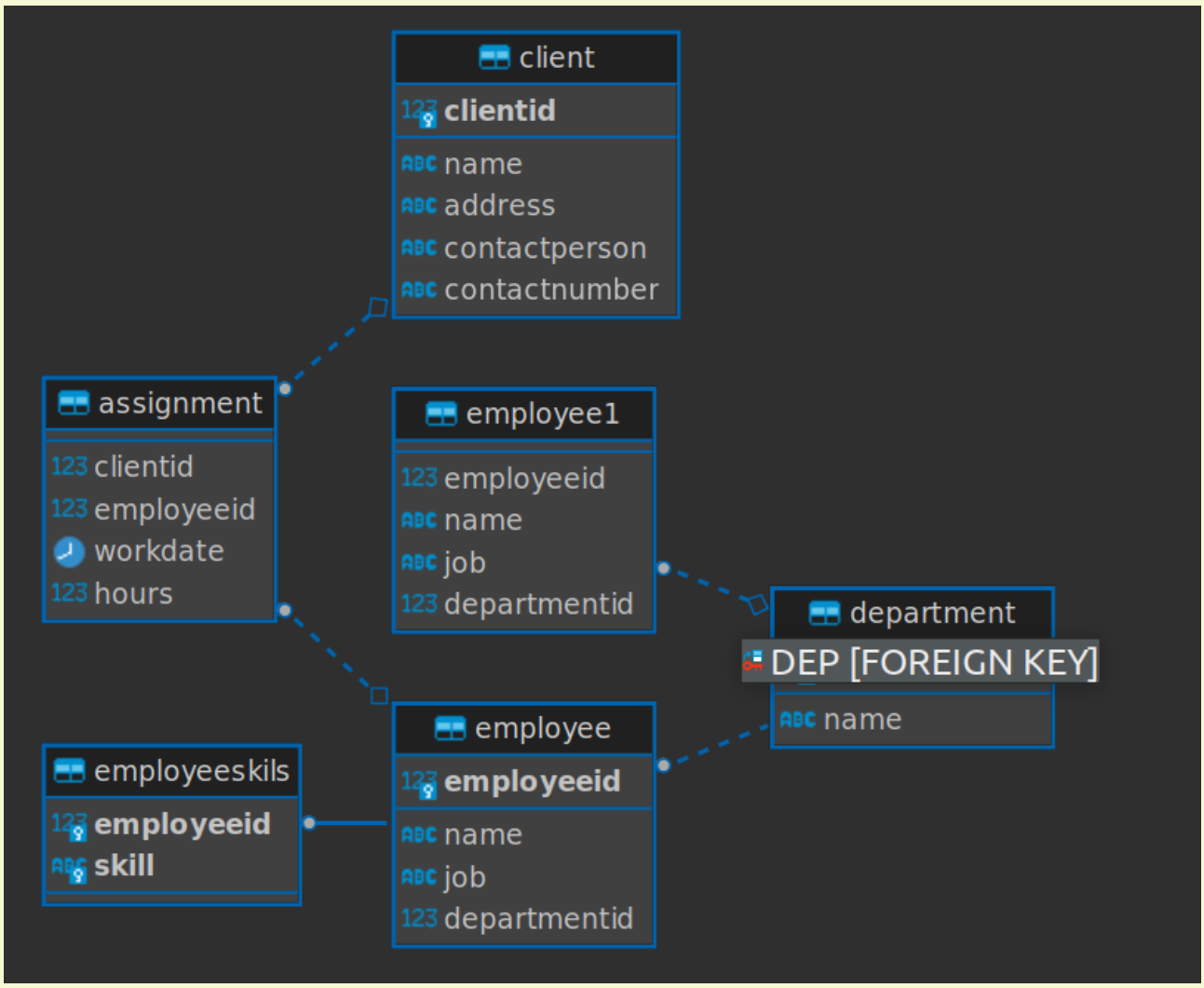
Name	Value
Updated Rows	0
Query	CREATE TABLE client (clientID serial PRIMARY KEY, "name" varchar(40), address varchar(100), contactPerson varchar(80), contactNumber varchar(80))
Start time	Thu Mar 28 12:28:46 MSK 2024
Finish time	Thu Mar 28 12:28:46 MSK 2024

```sql
CREATE TABLE "assignment" (
    clientID int REFERENCES client(clientID),
    employeeID int REFERENCES employee(employeeID),
    workdate date,
    hours float
);
```



Statistics 1 | Output

Name	Value
Updated Rows	0
Query	CREATE TABLE "assignment" (clientID int REFERENCES client(clientID), employeeID int REFERENCES employee(employeeID), workdate date, hours float)
Start time	Thu Mar 28 12:31:17 MSK 2024
Finish time	Thu Mar 28 12:31:17 MSK 2024



Заполнение таблиц

```
INSERT INTO employeeskils
VALUES
    (101, 'Basic'),
    (102, 'Python'),
    (103, 'SQL'),
    (104, 'C++'),
    (100, 'Pascal'),
    (104, 'Delphi');
SELECT * FROM employeeskils;
```

	employeeid	skill
1	101	Basic
2	102	Python
3	103	SQL
4	104	C++
5	100	Pascal
6	104	Delphi

```
INSERT INTO client (clientID, "name", address,
contactperson, contactnumber)
VALUES
    (1100, 'ACER', 'M.12.st.', 'Nora', '112233445566'),
    (1101, 'MTS', 'S.P.11.st.', 'Lena', '665544332211'),
    (1102, 'Dog', 'N.N.13 st.', 'Ivan', '123456123456'),
    (1103, 'Cat', 'K.14 st.', 'Petr', '654321654321');
SELECT * FROM client;
```

	clientid	name	address	contactperson	contactnumber
1	1,100	ACER	M.12.st.	Nora	112233445566
2	1,101	MTS	S.P.11.st.	Lena	665544332211
3	1,102	Dog	N.N.13 st.	Ivan	123456123456
4	1,103	Cat	K.14 st.	Petr	654321654321

```
INSERT INTO "assignment" (clientID, employeeID,
workdate, hours)
VALUES
    (1100, 100, '2009-01-10', 120),
    (1101, 101, '2008-11-01', 10),
    (1102, 102, '2009-12-01', 70),
    (1103, 102, '2009-02-01', 100);
SELECT * FROM "assignment";
```

	clientid	employeeid	workdate	hours
1	1,100	100	2009-01-10	120
2	1,101	101	2008-11-01	10
3	1,102	102	2009-12-01	70
4	1,103	102	2009-02-01	100

	clientid	employeeid	workdate	hours
1	1,100	100	2009-01-10	120
2	1,101	101	2008-11-01	10
3	1,102	102	2009-12-01	70
4	1,103	102	2009-02-01	100

А также манипуляции, связанные с изменением таблицы **Employee** из файла "Семинар 6-8-WM.pdf"

Задание 8 (a).
Добавить в таблицу employee столбец AGE (возраст), salary (зарплату), perks (надбавки)

```
ALTER TABLE employee
     ADD COLUMN Age int,
     ADD COLUMN Salary int,
     ADD COLUMN perks int;
```

Задание 8 (b).
Заполнить новые столбцы данными (зарплата: 20000 -50000, надбавки: 1000 – 5000, возраст 20-45 лет).

```
UPDATE employee
     SET
          Salary = ROUND((20+RANDOM()*30)::int)*1000,
          Age = ROUND(20 + RANDOM()*25)::int,
          perks = ROUND(1+ RANDOM()*4)*1000::int;
```

Задание 8 (c).
Добавить в таблицу employee сотрудников, которые являются системными программистами и программистами - администраторами баз данных. Например, ‘syst.programmer’, ‘admin. Programmer’

```
INSERT INTO employee
VALUES
     (105, 'Fedor K.', 'syst. Programmer', 2, 46, 49000, 3000),
     (106, 'Maria T.', 'admin. Programmer', 2, 37, 44000, 2000);
SELECT * FROM employee;
```

	employeeid	name	job	departmentid	age	salary	perks
1	100	Smit N	Programmer	2	40	48,000	2,000
2	101	Stone J.	manager	3	42	23,000	4,000
3	102	Asser M.	analitic	1	44	50,000	5,000
4	103	Wood N.	Programmer	2	40	37,000	4,000
5	104	Thomson L.	Programmer	2	34	29,000	2,000
6	105	Fedor K.	syst. Programmer	2	46	49,000	3,000
7	106	Maria T.	admin. Programmer	2	37	44,000	2,000