
Домашняя работа № 7. Селекции в Больнице

=====Код создания и наполнения БД=====

```
CREATE database Hospital;
```

```
CREATE table Departments (  
id INT IDENTITY(1,1) NOT NULL PRIMARY KEY,  
Name NVARCHAR(100) NOT NULL UNIQUE check(Name <> N'')  
);
```

```
CREATE table Doctors(  
    Id INT NOT NULL IDENTITY(1, 1) PRIMARY KEY,  
    Name NVARCHAR(MAX) NOT NULL CHECK (NAME <> N''),  
    Premium MONEY NOT NULL CHECK (PREMIUM >= 0.0) DEFAULT 0.0,  
    Salary MONEY NOT NULL CHECK (SALARY > 0.0),  
    Surname NVARCHAR(MAX) NOT NULL CHECK (SURNAME <> N'')  
);
```

```
CREATE table DoctorsSpecializations  
(  
    Id INT NOT NULL IDENTITY(1, 1) PRIMARY KEY,  
    DoctorId INT NOT NULL,  
    SpecializationId INT NOT NULL  
);
```

```
CREATE table Donations  
(  
    Id INT NOT NULL IDENTITY(1, 1) PRIMARY KEY,  
    Amount MONEY NOT NULL CHECK (Amount > 0.0),  
    Date DATE NOT NULL CHECK (Date <= getdate()) default  
getdate(),  
    DepartmentId INT NOT NULL,  
    SponsorId INT NOT NULL  
);
```

```
CREATE table Specializations  
(  
    Id INT NOT NULL IDENTITY(1, 1) PRIMARY KEY,  
    Name NVARCHAR(100) NOT NULL UNIQUE CHECK (Name <> N'')  
);
```

```
CREATE table Sponsors
(
    Id INT NOT NULL IDENTITY(1, 1) PRIMARY KEY,
    Name NVARCHAR(100) NOT NULL UNIQUE CHECK (Name <> N'')
);
```

```
CREATE TABLE Vacations
(
    Id INT NOT NULL IDENTITY(1, 1) PRIMARY KEY,
    EndDate DATE NOT NULL,
    StartDate DATE NOT NULL,
    DoctorId INT NOT NULL
);
```

```
CREATE table Wards
(
    Id INT NOT NULL IDENTITY(1, 1) PRIMARY KEY,
    Name NVARCHAR(20) NOT NULL UNIQUE CHECK(Name <> N''),
    DepartmentId INT NOT NULL
);
```

```
ALTER TABLE Vacations
add FOREIGN KEY (DoctorID) REFERENCES Doctors(Id);
```

```
ALTER TABLE DoctorsSpecializations
add FOREIGN KEY (SpecializationId) REFERENCES
Specializations(Id) ON UPDATE CASCADE;
```

```
ALTER TABLE DoctorsSpecializations
add FOREIGN KEY (DoctorId) REFERENCES Doctors(Id) ON UPDATE
CASCADE;
```

```
ALTER TABLE Donations
add FOREIGN KEY (SponsorId) REFERENCES Sponsors(Id);
```

```
ALTER TABLE Donations
add FOREIGN KEY (DepartmentId) REFERENCES Departments(Id);
```

```
ALTER TABLE Wards
add FOREIGN KEY (DepartmentId) REFERENCES Departments(Id);
```

```

INSERT INTO Departments (Name)
VALUES ('Intensive Treatment'),
('Рытьё мозга'),
('Пятка горло рот'),
('Рентген души'),
('Экспериментальные камеры');

INSERT INTO Doctors (Name, Premium, Salary, Surname)
VALUES ('Helen', 500, 10000, 'Williams'),
('Слава', 200, 8000, 'Косторез'),
('Чебурпых', 700, 5000, 'Фон Венециано'),
('Селентий', 200, 15000, 'Уголков'),
('Гвидон', 0, 200, 'Голодный');

INSERT INTO Sponsors (Name)
VALUES ('Umbrella Corporation'),
('Людоеды INC'),
('Театр анатомии'),
('Независимые производители неизвестного мяса'),
('Клуб четырёхпалых левой руки');

INSERT INTO Donations (Amount, Date, DepartmentID, SponsorID)
VALUES (100500, '26-11-2012', 3, 1),
(98000, '12-10-2021', 2, 1),
(25000, '18-08-2019', 3, 2),
(200700, '11-11-2022', 1, 3),
(200, '31-12-2000', 5, 4);

INSERT INTO Specializations (Name)
VALUES ('31 Бровные выделения'),
('35 Ушибы языка'),
('30 Перегрев ногтей '),
('21 Неконтролируемость мыслей'),
('15 Ложная вонь');

INSERT INTO Vacations (EndDate, StartDate, DoctorId)
VALUES ('21-12-2014', '26-11-2014', 1),
('11-09-2014', '11-04-2014', 2),
('05-10-2021', '26-06-2021', 3),
('23-11-2018', '22-08-2018', 1),
('14-02-2021', '26-01-2021', 1);

```

```

INSERT INTO Wards (Name, DepartmentId)
VALUES
('Притихшие', 1),
('Кривые', 2),
('Смрадородящие', 3),
('Шутники', 4),
('Пижамомаратели', 5);

```

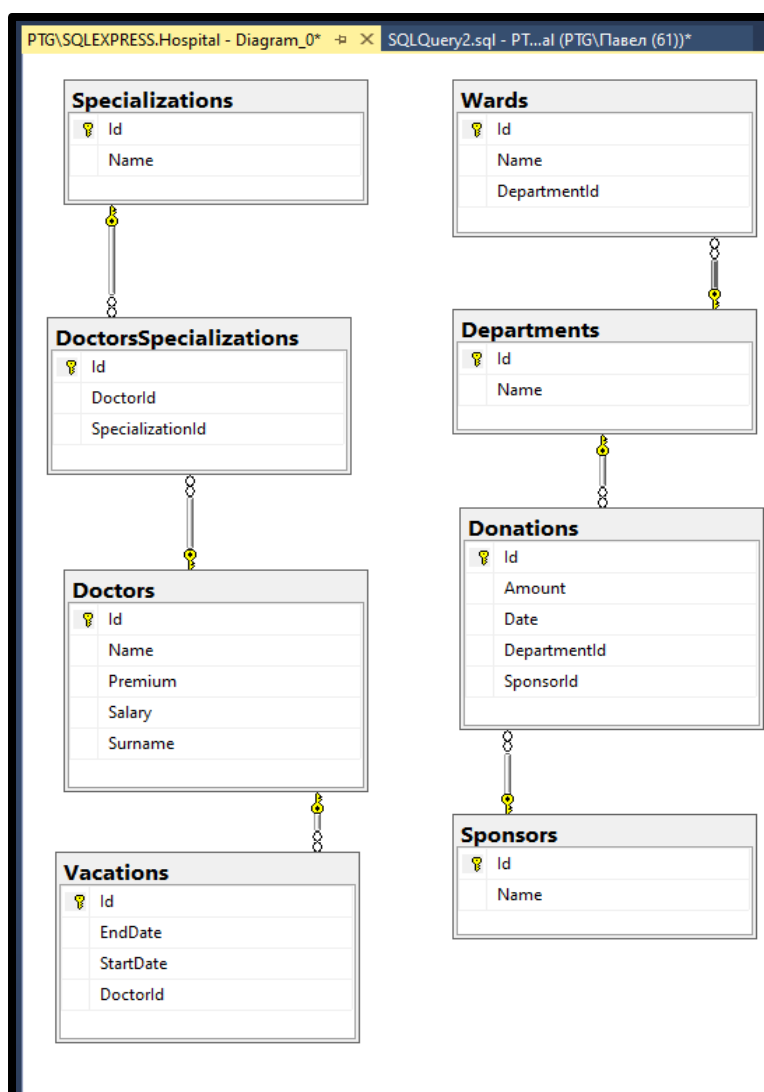
```

INSERT INTO DoctorsSpecializations (DoctorId, SpecializationId)
VALUES
(2, 3), (1, 1), (3, 4), (4, 5), (5, 2);

```

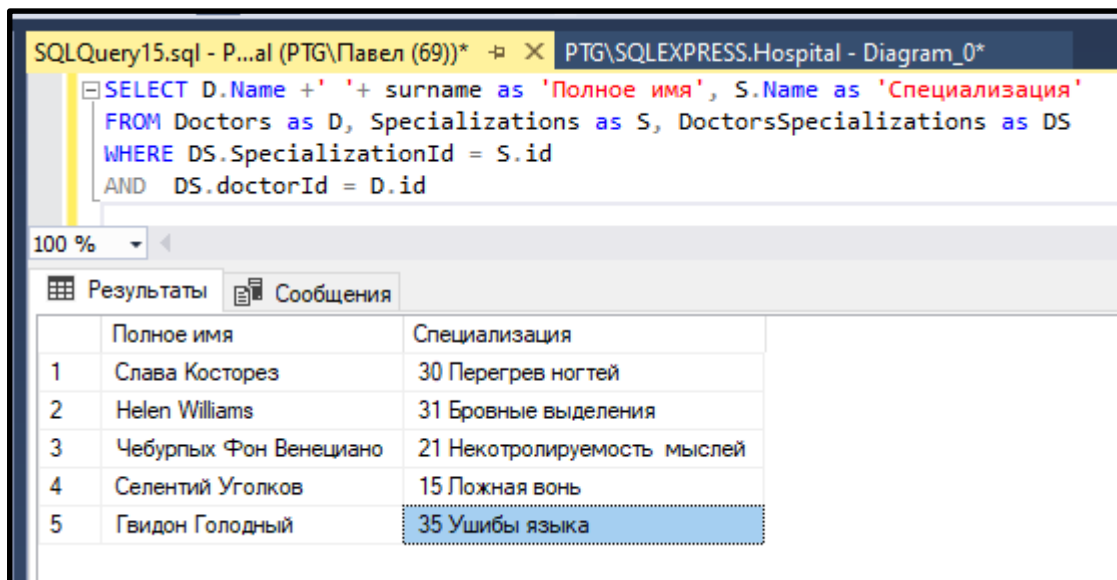
=====Код создания и наполнения БД=====

Диаграмма со связями получилась как в задании:



1. *Вывести полные имена врачей и их специализации.*

```
SELECT D.Name + ' ' + surname as 'Полное имя', S.Name as  
'Специализация'  
FROM Doctors as D, Specializations as S, DoctorsSpecializations  
as DS  
WHERE DS.SpecializationId = S.id  
AND DS.doctorId = D.id
```



The screenshot shows a SQL Server Enterprise Manager window with a query editor and a results pane. The query editor contains the following SQL code:

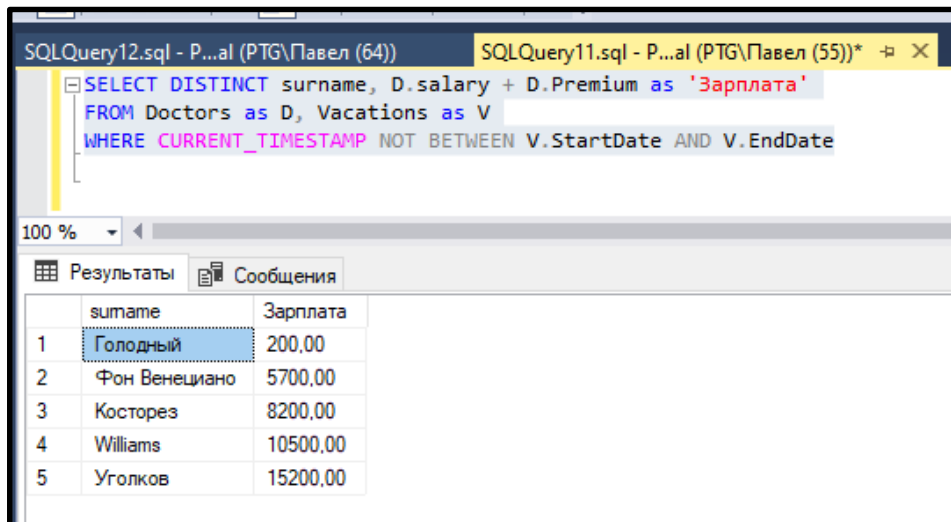
```
SELECT D.Name + ' ' + surname as 'Полное имя', S.Name as 'Специализация'  
FROM Doctors as D, Specializations as S, DoctorsSpecializations as DS  
WHERE DS.SpecializationId = S.id  
AND DS.doctorId = D.id
```

The results pane shows a table with two columns: 'Полное имя' and 'Специализация'. The table contains five rows of data, with the last row highlighted in blue.

| | Полное имя | Специализация |
|---|------------------------|------------------------------|
| 1 | Слава Косторез | 30 Перегрев ногтей |
| 2 | Helen Williams | 31 Бровные выделения |
| 3 | Чебурных Фон Венециано | 21 Неконтролируемость мыслей |
| 4 | Селентий Уголков | 15 Пожная вонь |
| 5 | Гвидон Голодный | 35 Ушибы языка |

2. Вывести фамилии и зарплаты (сумма ставки и надбавки) врачей, которые не находятся в отпуске.

```
SELECT DISTINCT surname, D.salary + D.Premium as 'Зарплата'  
FROM Doctors as D, Vacations as V  
WHERE CURRENT_TIMESTAMP NOT BETWEEN V.StartDate AND V.EndDate
```



The screenshot shows a SQL query execution window with two tabs. The active tab, 'SQLQuery12.sql - P...al (PTG\Павел (64))', displays the following query:

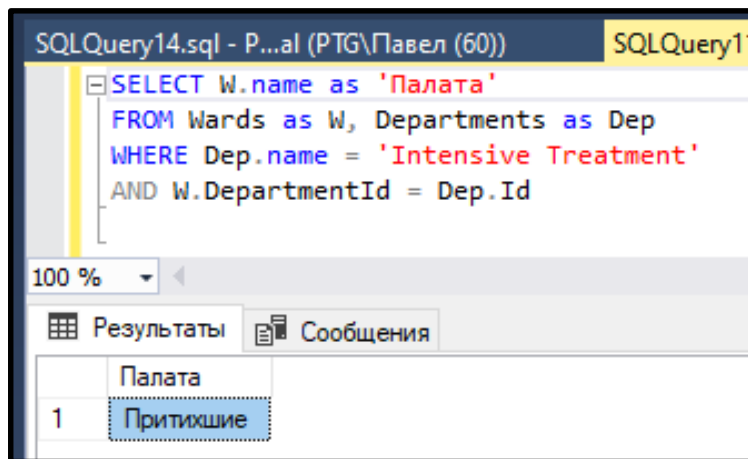
```
SELECT DISTINCT surname, D.salary + D.Premium as 'Зарплата'  
FROM Doctors as D, Vacations as V  
WHERE CURRENT_TIMESTAMP NOT BETWEEN V.StartDate AND V.EndDate
```

Below the query editor, the 'Результаты' (Results) tab is selected, showing a table with two columns: 'surname' and 'Зарплата'. The table contains five rows of data:

| | surname | Зарплата |
|---|---------------|----------|
| 1 | Голодный | 200,00 |
| 2 | Фон Венециано | 5700,00 |
| 3 | Косторез | 8200,00 |
| 4 | Williams | 10500,00 |
| 5 | Уголков | 15200,00 |

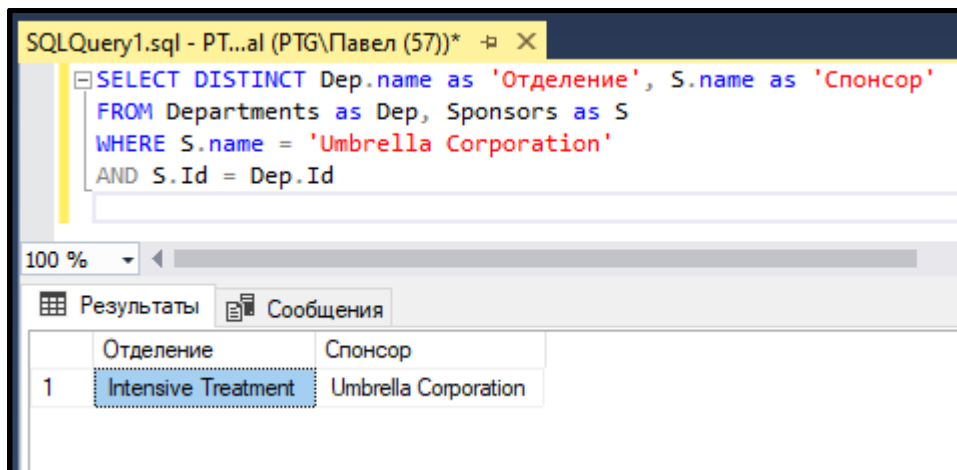
3. Вывести названия палат, которые находятся в отделении "Intensive Treatment".

```
SELECT W.name as 'Палата'  
FROM Wards as W, Departments as Dep  
WHERE Dep.name = 'Intensive Treatment'  
AND W.DepartmentId = Dep.Id
```



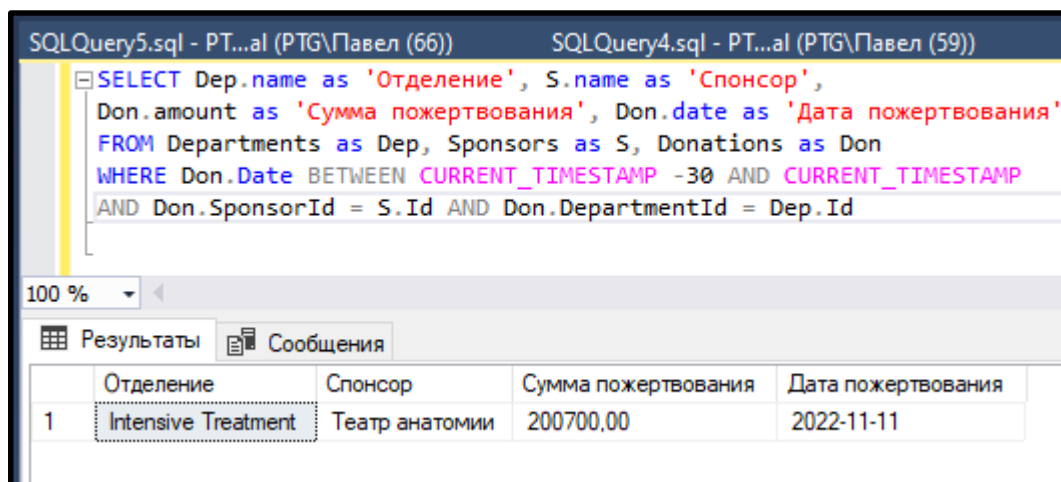
4. Вывести названия отделений без повторений, которые спонсируются компанией "Umbrella Corporation".

```
SELECT DISTINCT Dep.name as 'Отделение', S.name as 'Спонсор'  
FROM Departments as Dep, Sponsors as S  
WHERE S.name = 'Umbrella Corporation'  
AND S.Id = Dep.Id
```



5. Вывести все пожертвования за последний месяц в виде: отделение, спонсор, сумма пожертвования, дата пожертвования.

```
SELECT Dep.name as 'Отделение', S.name as 'Спонсор',  
Don.amount as 'Сумма пожертвования', Don.date as 'Дата  
пожертвования'  
FROM Departments as Dep, Sponsors as S, Donations as Don  
WHERE Don.Date BETWEEN CURRENT_TIMESTAMP -30 AND  
CURRENT_TIMESTAMP  
AND Don.SponsorId = S.Id AND Don.DepartmentId = Dep.Id
```



The screenshot shows a SQL Server Enterprise Manager window with two tabs: 'SQLQuery5.sql - PT...al (PTG\Павел (66))' and 'SQLQuery4.sql - PT...al (PTG\Павел (59))'. The active tab displays the following SQL query:

```
SELECT Dep.name as 'Отделение', S.name as 'Спонсор',  
Don.amount as 'Сумма пожертвования', Don.date as 'Дата пожертвования'  
FROM Departments as Dep, Sponsors as S, Donations as Don  
WHERE Don.Date BETWEEN CURRENT_TIMESTAMP -30 AND CURRENT_TIMESTAMP  
AND Don.SponsorId = S.Id AND Don.DepartmentId = Dep.Id
```

Below the query editor, the 'Results' tab is selected, showing a table with 5 columns: 'Отделение', 'Спонсор', 'Сумма пожертвования', and 'Дата пожертвования'. The table contains one row of data:

| | Отделение | Спонсор | Сумма пожертвования | Дата пожертвования |
|---|---------------------|----------------|---------------------|--------------------|
| 1 | Intensive Treatment | Театр анатомии | 200700,00 | 2022-11-11 |

6. *Вывести фамилии врачей с указанием отделений, в которых они проводят обследования. Необходимо учитывать обследования, проводимые только в будние дни.*

Это задание я не делаю, поскольку у нас нет таблицы, где хранятся данные о датах проведения обследований и в каких отделениях работают конкретные доктора.

Но я сделаю, то что могу придумать как сделать=)

Добавим поле DepWork в таблицу Doctors

```
ALTER TABLE Doctors
add DepWork INT NULL;
ALTER TABLE Doctors
add FOREIGN KEY (DepWork) REFERENCES Departments (Id);
```

И введём в это поле место работы каждого доктора

```
UPDATE Doctors
SET DepWork = 1
WHERE id=2;
```

```
UPDATE Doctors
SET DepWork = 2
WHERE id=1;
```

```
UPDATE Doctors
SET DepWork = 3
WHERE id=3;
```

```
UPDATE Doctors
SET DepWork = 4
WHERE id=5;
```

```
UPDATE Doctors
SET DepWork = 5
WHERE id=4;
```

SQLQuery23.sql - P...al (PTG\Павел (72)) SQLQuery15.sql - P...al (PTG\Павел (72))

```
SELECT *
FROM Doctors
```

100 %

Результаты Сообщения

| | Id | Name | Premium | Salary | Surname | DepWork |
|---|----|----------|---------|----------|---------------|---------|
| 1 | 1 | Helen | 500,00 | 10000,00 | Williams | 2 |
| 2 | 2 | Слава | 200,00 | 8000,00 | Косторез | 1 |
| 3 | 3 | Чебурных | 700,00 | 5000,00 | Фон Венециано | 3 |
| 4 | 4 | Селентий | 200,00 | 15000,00 | Уголков | 5 |
| 5 | 5 | Гвидон | 0,00 | 200,00 | Голодный | 4 |

Тогда запрос без дат:

```
SELECT D.surname as 'Фамилия', Dep.name as 'Отделение'
FROM Departments as Dep, Doctors as D
WHERE D.Depwork = Dep.Id
```

SQLQuery23.sql - P...al (PTG\Павел (72)) SQLQuery15.sql - P...al (PTG\Павел (72))

```
SELECT D.surname as 'Фамилия', Dep.name as 'Отделение'
FROM Departments as Dep, Doctors as D
WHERE D.Depwork = Dep.Id
```

100 %

Результаты Сообщения

| | Фамилия | Отделение |
|---|---------------|--------------------------|
| 1 | Williams | Рытьё мозга |
| 2 | Косторез | Intensive Treatment |
| 3 | Фон Венециано | Пятка горло рот |
| 4 | Уголков | Экспериментальные камеры |
| 5 | Голодный | Рентген души |

7. Вывести названия палат и корпуса отделений, в которых проводит обследования врач "Helen Williams".

Тоже не делаю, поскольку нет таблицы, связывающей корпуса и палаты. Однако добавим поле Corpus в таблицу Wards.

```
ALTER TABLE Wards  
add Corpus INT NULL;
```

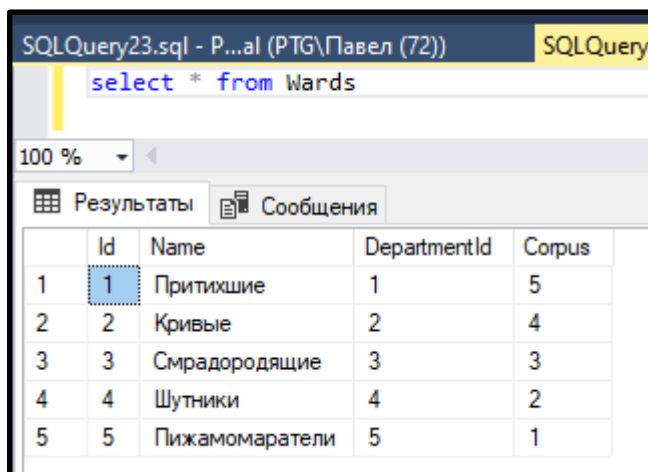
```
UPDATE Wards  
SET Corpus = 1  
WHERE id=5;
```

```
UPDATE Wards  
SET Corpus = 2  
WHERE id=4;
```

```
UPDATE Wards  
SET Corpus = 3  
WHERE id= 3;
```

```
UPDATE Wards  
SET Corpus = 4  
WHERE id= 2;
```

```
UPDATE Wards  
SET Corpus = 5  
WHERE id= 1;
```



SQLQuery23.sql - P...al (PTG\Павел (72)) SQLQuery

```
select * from Wards
```

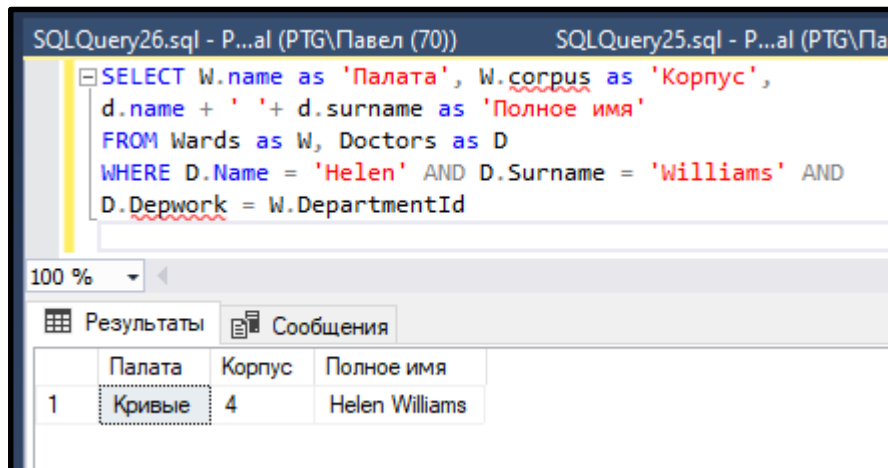
100 %

Результаты Сообщения

| | Id | Name | DepartmentId | Corpus |
|---|----|----------------|--------------|--------|
| 1 | 1 | Притихшие | 1 | 5 |
| 2 | 2 | Кривые | 2 | 4 |
| 3 | 3 | Смрадороящие | 3 | 3 |
| 4 | 4 | Шутники | 4 | 2 |
| 5 | 5 | Пижамомаратели | 5 | 1 |

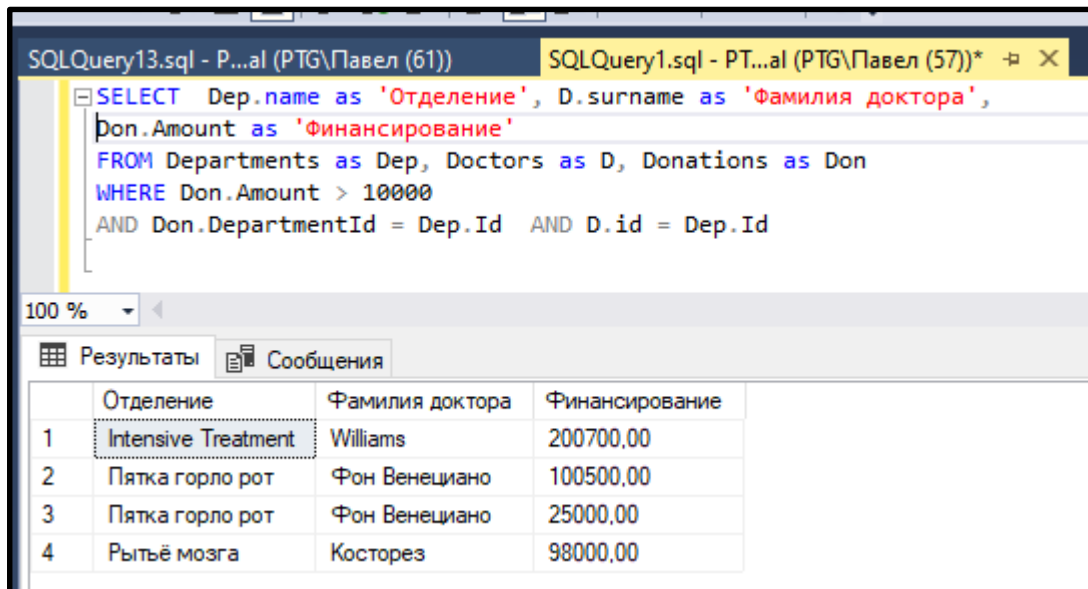
Тогда искомый запрос будет таким:

```
SELECT W.name as 'Палата', W.corpus as 'Корпус',  
d.name + ' ' + d.surname as 'Полное имя'  
FROM Wards as W, Doctors as D  
WHERE D.Name = 'Helen' AND D.Surname = 'Williams' AND  
D.Depwork = W.DepartmentId
```



8. Вывести названия отделений, которые получали пожертвование в размере больше 100000, с указанием их врачей.

```
SELECT Dep.name as 'Отделение', D.surname as 'Фамилия доктора',  
Don.Amount as 'Финансирование'  
FROM Departments as Dep, Doctors as D, Donations as Don  
WHERE Don.Amount > 10000  
AND Don.DepartmentId = Dep.Id AND D.id = Dep.Id
```



The screenshot shows a SQL Server Enterprise Manager window with a query editor and a results grid. The query editor contains the following SQL code:

```
SELECT Dep.name as 'Отделение', D.surname as 'Фамилия доктора',  
Don.Amount as 'Финансирование'  
FROM Departments as Dep, Doctors as D, Donations as Don  
WHERE Don.Amount > 10000  
AND Don.DepartmentId = Dep.Id AND D.id = Dep.Id
```

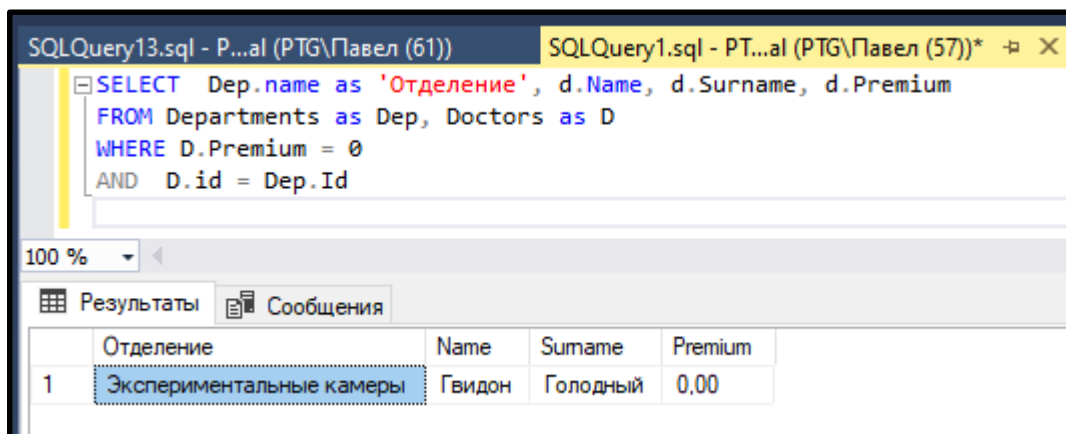
The results grid displays the following data:

| | Отделение | Фамилия доктора | Финансирование |
|---|---------------------|-----------------|----------------|
| 1 | Intensive Treatment | Williams | 200700,00 |
| 2 | Пятка горло рот | Фон Венециано | 100500,00 |
| 3 | Пятка горло рот | Фон Венециано | 25000,00 |
| 4 | Рытьё мозга | Косторез | 98000,00 |

Добавил Финансирование для наглядности

9. Вывести названия отделений, в которых есть врачи не получающие надбавки.

```
SELECT Dep.name as 'Отделение', d.Name, d.Surname, d.Premium  
FROM Departments as Dep, Doctors as D  
WHERE D.Premium = 0  
AND D.id = Dep.Id
```



The screenshot shows a SQL Server Enterprise Manager window with two tabs: 'SQLQuery13.sql - P...al (PTG\Павел (61))' and 'SQLQuery1.sql - PT...al (PTG\Павел (57))*'. The active tab displays the following SQL query:

```
SELECT Dep.name as 'Отделение', d.Name, d.Surname, d.Premium  
FROM Departments as Dep, Doctors as D  
WHERE D.Premium = 0  
AND D.id = Dep.Id
```

Below the query editor, the 'Results' tab is selected, showing a table with the following data:

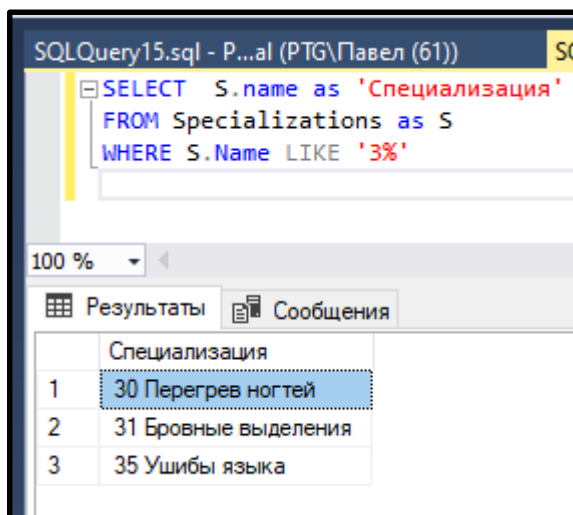
| | Отделение | Name | Surname | Premium |
|---|--------------------------|--------|----------|---------|
| 1 | Экспериментальные камеры | Гвидон | Голодный | 0,00 |

Добавил имя, фамилию и премию для осознания глубины отчаяния.

10. Вывести названия специализаций, которые используются для лечения заболеваний со степенью тяжести выше 3.

Это задание тоже предполагает наличие таблицы с данными которых нет, но я кое-что придумал. Просто в названия специализаций добавил некий цифровой код специальности.

```
SELECT S.name as 'Специализация'
FROM Specializations as S
WHERE S.Name LIKE '3%'
```



Но это не совсем подходит к условию задачи, так что пришлось добавить поле Dangerous - Опасность или Тяжесть.

```
ALTER TABLE Specializations
add Dangerous INT NULL;
```

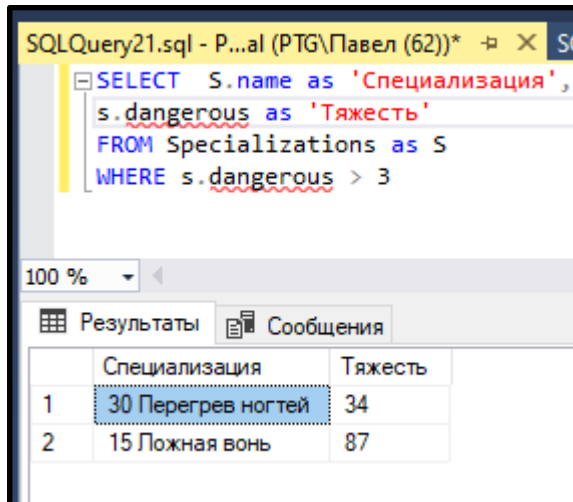
```
UPDATE Specializations
SET Dangerous = 87
WHERE id=5
/// и так далее///
```

The screenshot shows the same SQL query window, but now the 'Результаты' tab displays a table with four columns: 'Id', 'Name', and 'Dangerous'. The results are:

| | Id | Name | Dangerous |
|---|----|------------------------------|-----------|
| 1 | 1 | 31 Бровные выделения | 1 |
| 2 | 2 | 35 Ушибы языка | 0 |
| 3 | 3 | 30 Перегрев ногтей | 34 |
| 4 | 4 | 21 Неконтролируемость мыслей | 2 |
| 5 | 5 | 15 Пожная вонь | 87 |

И запрос теперь можно сделать точным.

```
SELECT S.name as 'Специализация', s.dangerous as 'Опасность'  
FROM Specializations as S  
WHERE s.dangerous > 3
```



The screenshot shows a SQL query window titled 'SQLQuery21.sql - P...al (PTG\Павел (62))*'. The query is as follows:

```
SELECT S.name as 'Специализация',  
       s.dangerous as 'Тяжесть'  
FROM Specializations as S  
WHERE s.dangerous > 3
```

Below the query editor, the 'Results' tab is active, displaying a table with two columns: 'Специализация' and 'Тяжесть'. The table contains two rows of data. The first row is highlighted with a blue selection box.

| | Специализация | Тяжесть |
|---|--------------------|---------|
| 1 | 30 Перегрев ногтей | 34 |
| 2 | 15 Ложная вонь | 87 |

11. Вывести названия отделений и заболеваний, обследования по которым они проводили за последние полгода.

Это задание я не выполняю, поскольку у нас нет таблицы с данными о датах обследований и заболеваниях (возможно речь о специализации?).

А так бы я использовал конструкцию:

```
SELECT dep.name as 'Отделение', s.name as 'Заболевание'
FROM Departments as Dep, Specializations as S, *некая таблица с
датоми работы* as WD
WHERE WD.Date BETWEEN CURRENT_TIMESTAMP - 180 AND
CURRENT_TIMESTAMP ;
```

12. Вывести названия отделений и палат, в которых проводились обследования по заразительным заболеваниям.

И снова у нас нет данных о заразительности заболеваний, которых, впрочем, тоже нет.

Предположим, что заразительность либо есть, либо нет, 1 или 0. И данные о ней содержатся в таблице Wards в поле Contagiousness, поскольку заразных селят с заразными, не иначе.

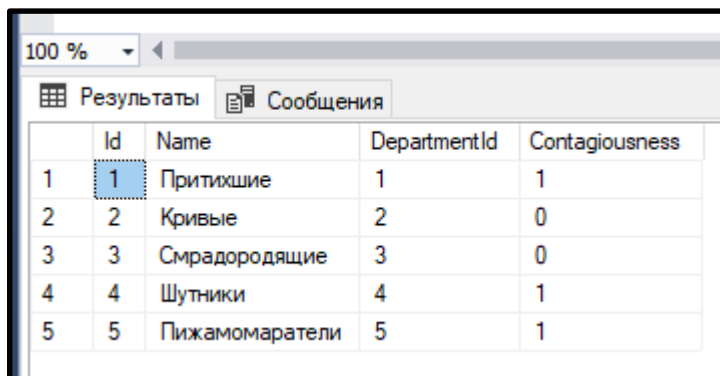
Добавим заразное поле (Contagiousness) в таблицу Палаты

```
ALTER TABLE Wards  
add Contagiousness INT NOT NULL;
```

Введём Заразность в каждую ячейку примерно так:

```
UPDATE Wards  
SET Contagiousness = 1  
WHERE id=1;  
//и тому подобное//
```

Получили Wards в таком виде:



The screenshot shows a database application window with a toolbar at the top containing a zoom dropdown set to 100%, a back arrow, and two buttons labeled 'Результаты' (Results) and 'Сообщения' (Messages). Below the toolbar is a table with the following data:

| | Id | Name | DepartmentId | Contagiousness |
|---|----|----------------|--------------|----------------|
| 1 | 1 | Притихшие | 1 | 1 |
| 2 | 2 | Кривые | 2 | 0 |
| 3 | 3 | Смрадороящие | 3 | 0 |
| 4 | 4 | Шутники | 4 | 1 |
| 5 | 5 | Пижамомаратели | 5 | 1 |

Тогда наш запрос о заразных палатах и отделениях будет таким:

```
SELECT Dep.name as 'Отделение', W.name as 'Палата',  
W.Contagiousness as 'Заразность'  
FROM Departments as Dep, Wards as W  
WHERE W.Contagiousness = 1  
AND W.DepartmentId = Dep.id
```

