Федеральное агентство связи

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Поволжский государственный университет телекоммуникаций и информатики»

Кафедра «Информационные системы и технологии»

Работа принята на проверку Работа допущена к защите

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2019 года «\_\_\_» \_\_\_\_\_\_\_\_\_2019года

Захарова О.И.                   Захарова О.И.

*(ФИО,подпись руководителя) (ФИО,подпись руководителя)*

**КУРСОВАЯ РАБОТА**

**по дисциплине**

**«Базы Данных»**

**на тему:**

«Разработка базы данных предметной области»

ВЫПОЛНИЛА

студентка 3 курса факультета ИСТ

группы ПИвЭ-71

Бородина О.Ю.

*(инициалы и фамилия) (подпись)*

Работа защищена с оценкой

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2019 года

Захарова О.И.

*(ФИО,подпись руководителя*

Содержание

[1.Постановка задачи. 2](#_Toc24484342)

[2.Инфологическая модель. 4](#_Toc24484343)

[3.Нормализация отношений. 9](#_Toc24484347)

[4.Разработка таблиц и схемы базы данных. 14](#_Toc24484348)

[5.Запросы системы. 22](#_Toc24484349)

[6.Выводы. 30](#_Toc24484350)

[7.Список используемых источников 31](#_Toc24484351)

# 1.Постановка задачи.

В качестве предметной области (ПО) рассматривается пункт обмена валют. В пункте обмена валюты ежедневно производится купля-продажа валюты. При совершении сделки покупается или продается валюта. Должна учитываться возможность ведения электронного реестра валютно-обменных операций, формирования необходимых печатных документов и бланков строгой отчетности и ведение их реестра. Каждую из перечисленных операций кассир обменного пункта обязан зафиксировать документально и оформить справку о совершении клиентом валютно-обменной операции, с выдачей копии справки клиенту.

Необходимо спроектировать базу данных Пункт обмена валюты, которая должна обеспечивать хранение, поиск, анализ данных о сделках, совершенных в пункте обмена, и др.

В БД должна храниться информация:

• Клиенты (номер клиента, Ф.И.О. клиента, номер паспорта);

• Сделки (код проданной валюты, код купленной валюты, номер кассира, номер клиента, дата сделки, время сделки, сумма проданной валюты, сумма купленной валюты;

• Валюта (код проданной валюты, код купленной валюты, название валюты, курс продажи, курс покупки);

• Кассир (номер кассира, Ф.И.О. кассира).

При проектировании БД необходимо учитывать следующее:

• клиент может совершать несколько сделок. Сделка совершается одним клиентом;

• любая валюта покупается (продается) при нескольких сделках. Сделка связана с одной валютой.

• кассир обслуживает одну сделку. Сделка совершается одним кассиром.

Кроме того следует учесть:

• каждая сделка обязательно совершается клиентом. Клиент необязательно совершает сделку (его может не устраивать курс валюты);

• каждая сделка обязательно совершается при продаже (покупке) валюты. При совершении сделки обязательно продается или покупается валюта;

• каждая сделка обязательно обслуживается кассиром. Кассир не обязательно обслуживает сделку (например, он только что принят на работу).

Реализация настоящей БД должна быть произведена средствами СУБД

MySQL.

Необходимо также разработать не менее 15 запросов следующих типов:

1. С использованием внутреннего соединения по одному полю.
2. С использованием косвенно связанных таблиц.
3. С использованием таблиц, связанных более чем по одному полю.
4. С использованием внешнего соединения таблиц.
5. С использованием рекурсивного соединения.
6. С использованием соединений по отношению.
7. С использованием функций агрегирования.
8. Перекрестный запрос.
9. С параметрами.
10. С вычисляемым полем.
11. Многотабличный запрос.
12. Независимый и связанный подзапрос с множеством значений.
13. Запрос для расчета с накоплением.
14. Создание представления.
15. Запрос на модификацию данных и слияние. В том числе должны быть сложные запросы с подзапросами.
16. Запрос с использованием транзакции.
17. Создание триггера.

# 2.Инфологическая модель.

**Семантическое моделирование**

Из описания предметной области следуют:

**1. Сущности:**

* Сделки
* Клиент
* Сотрудники
* Должности
* Валюта
* Тип сделки
* Пункт обмена
* Банк
* Вид банка
* Реестр
* Бланк отчетности

**2. Атрибуты:**

* Код сделки
* Код клиента
* Код валюты
* Код сотрудника
* Курс продажи
* Курс покупки
* Код пункта обмена
* Дата сделки
* Код тип сделки
* Сумма продажи
* Сумма покупки

являются атрибутами сущности «Сделки»

* Код клиента
* Фамилия
* Имя
* Отчество
* Паспорт

являются атрибутами сущности «Клиент»

* Код сотрудника
* Фамилия
* Имя
* Отчество
* Код должности
* Код пункта обмена

являются атрибутами сущности «Сотрудники»

* Код должности
* Название
* Оклад

являются атрибутами сущности «Должность»

* Код валюты
* Название валюты

являются атрибутами сущности «Валюта»

* Код типа сделки
* Тип

являются атрибутами сущности «Тип сделки»

* Код пункта обмена
* №отделения
* Адрес
* Код банка

Являются атрибутами сущности «Пункт обмена»

* Код банка
* Название
* Юр. адрес
* Телефон
* Код вид банка

являются атрибутами сущности «Банк»

* Код вида
* Вид

являются атрибутами сущности «Вид банка»

* Код банка
* Справка

являются атрибутами сущности «Бланк отчётности»

* Код реестра
* Код сделки
* Код бланка
* Дата

являются атрибутами сущности «Реестр»

**3. Ключи сущностей:**

* Код\_сделки является первичным ключом сущности «Сделки»;
* Код\_клиента является первичным ключом сущности «Клиенты»;
* Код\_типа сделки является первичным ключом сущности «Тип сделки»;
* Код\_валюты является первичным ключом сущности «Валюта»;
* Код\_должности является первичным ключом сущности «Должность»;
* Код\_сотрудника является первичным ключом сущности «Сотрудники»;
* Код\_пункта обмена является первичным ключом сущности «Пункт обмена»;
* Код\_вида банка является первичным ключом сущности «Вид банка»;
* Код\_банка является первичным ключом сущности «Банк»;
* Код\_бланка отчетности является первичным ключом сущности «Бланк отчетности
* Код\_реестра является первичным ключом сущности «Реестр»

**4. Связь сущностей:**

Между сущностями «Клиенты» и «Сделки»:

Слева направо: «Клиенты должны проводить сделки»

Справа налево: «Хотя бы одна сделка должна принадлежать клиенту»;

Тип связи: один -ко-многим.

Между сущностями «Сделки» и «Тип сделки»:

Слева направо: «У каждой сделки должен быть тип сделки»

Справа налево: «Тип сделки должен принадлежать сделке»;

Тип связи: один -ко-многим.

Между сущностями «Валюта» и «Сделки»:

Слева направо: «Валюта должна участвовать в сделке»

Справа налево: «У сделки должна быть валюта, хотя бы одна»;

Тип связи: один -ко-многим.

Между сущностями «Сделки» и «Сотрудники»:

Слева направо: «Сделка должна проводиться сотрудником»

Справа налево: «У сотрудника может быть множество сделок»;

Тип связи: один -ко-многим.

Между сущностями «Сотрудники» и «Должность»:

Слева направо: «Сотрудники должны иметь должность»

Справа налево: «Должность должна иметь множество сотрудников»;

Тип связи: один -ко-многим.

Между сущностями «Сделки» и «Пункт обмена»:

Слева направо: «Сделки должны проводиться в пункте обмена»;

Справа налево: «Пункт обмена может участвовать во множестве сделок»;

Тип связи: один -ко-многим.

Между сущностями «Сотрудники» и «Пункт обмена»:

Слева направо: «Сотрудники должны работать в пункте обмена»;

Справа налево: «У пункта обмена может быть множество сотрудников»;

Тип связи: один-ко-многим.

Между сущностями «Банк» и «Пункт обмена»:

Слева направо: «Банк может иметь множество пунктов обмена»;

Справа налево: «Пункт обмена должен принадлежать банку»;

Тип связи: один-ко-многим.

Между сущностями «Вид банка» и «Банк»:

Слева направо: «Вид банка может принадлежать множествам банков»;

Справа налево: «Банк должен иметь вид банка»;

Тип связи: один-ко-многим.

Между сущностями «Сделки» и «Реестр»:

Слева направо: «У сделки множество реестров»;

Справа налево: «Множество реестров могут принадлежать одной сделке»;

Тип связи: один-ко-многим.

Между сущностями «Бланк отчётности» и «Реестр»:

Слева направо: «Бланк отчетности указан в множествах реестрах»;

Справа налево: «Множество реестров могут иметь несколько бланков отчётности»;

Тип связи: один-ко-многим.

Концептуальная модель

Диаграмма построенна с помощью программы «Untitled Diagram.drawio».

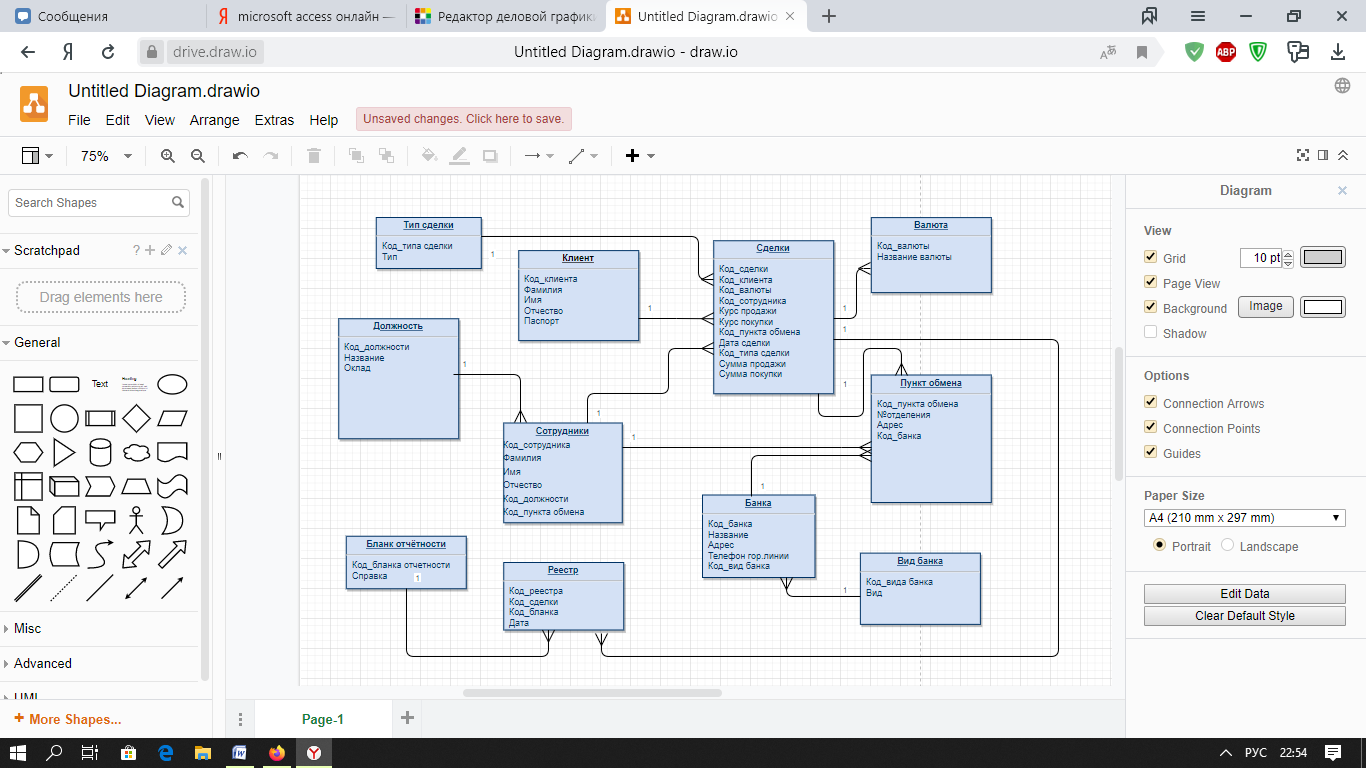


Рис. 1 – ER – диаграмма предметной области

# 3.Нормализация отношений.

Нормализация - процесс, в результате которого можно избавиться от дефектов проектирования базы данных.

**Приведение к первой нормальной форме (1NF):**

Табл.1 **-** Клиент

|  |  |  |  |
| --- | --- | --- | --- |
| Фамилия | Имя | Отчество | Паспорт |
| Иванов | Иван | Иванович | 325664 |
| … | … | … | … |

Табл.2 **-** Сделки

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Валюта | Сотрудник | Курс продажи | Курс покупки | Дата сделки | Тип сделки |
| Евро | Борисов | 54 | 55.5 | 2019-12-02 | Покупка |
| … | … | … | … | … | … |

Табл.3 **-** Пункт обмена

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № отделения | Адрес | Банк | Сотрудники | Должность сотрудника |
| 404 | Ул.  Кирова д.56 | ВТБ | Борисов | Кассир |
| … | … | … | … | … |

Табл.4 **-** Банк

|  |  |  |
| --- | --- | --- |
| Название | Адрес | Телефон гор. линии |
| ВТБ | Ул.  Победы д.76 | 8800555 |
| … | … | … |

**Приведение ко второй нормальной форме (2NF):**

БД имеет не атомарные атрибуты (сотрудники и должности сотрудников в сущности пункт обмена). Наша БД находится в 1НФ, но не во 2-ой НФ, т.к у нас есть зависимость неключевого поля тип сделки, т.к зависят атрибуты сумма продажи и покупка продажи. Избавимся от зависимости и добавим первичные коды:

Табл.5 **-** Клиент

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Фамилия | Имя | Отчество | Паспорт | Код\_клиента |
| Иванов | Иван | Иванович | 325664 | 1 |
| … | … | … | … | … |

Табл.6 **-** Сделки

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Валюта | Сотрудник | Курс продажи | Курс покупки | Дата сделки | Тип сделки | Код\_ сделки |
| Евро | Борисов | 54 | 55.5 | 2019-12-02 | Покупка | 01 |
| … | … | … | … | … | … | … |

Табл.7 **-** Пункт обмена

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № отделения | Адрес | Банк | Сотрудники | Должность сотрудника | Код\_пункта обмена |
| 404 | Ул.  Кирова д.56 | ВТБ | Борисов | Кассир | 11 |
| … | … | … | … | … | … |

Табл.8 **-** Банк

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Адрес | Телефон гор. линии | Код\_банка |
| ВТБ | Ул.  Победы д.76 | 8800555 | 25 |
| … | … | … | … |

Табл.9 **-** Тип сделки

|  |  |
| --- | --- |
| Тип | Код\_типа сделки |
| Покупка | 42 |
| … | … |

Табл.10 **-** Сотрудники

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Фамилия | Имя | Отчество | Должность | Код\_сотрудника |
| Федотов | Степан | Петрович | Кассир | 15 |
| … | … | … | … | … |

**Приведение к третьей нормальной форме (3NF):**

Приведем нашу таблицу к 3-ей НФ, вынесем все неключевые поля, содержимое которых может относиться к нескольким записям таблицы в отдельные таблицы, так же создадим таблицы для реестра и бланка отчётности, в которых хранится все совершенные сделки клиентов.

Табл.11 **-** Клиент

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Фамилия | Имя | Отчество | Паспорт | Код\_клиента |
| Иванов | Иван | Иванович | 325664 | 1 |
| … | … | … | … | … |

Табл.12 **-** Сделки

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Код\_валюты | Код\_сотрудника | Курс продажи | Курс покупки | Дата сделки | Код\_тип сделки | Код\_ сделки |
| 02 | Борисов | 54 | 55.5 | 2019-12-02 | Покупка | 1 |
| … | … | … | … | … | … | … |

Продолжение табл.12

|  |  |  |
| --- | --- | --- |
| Код\_клиента | Сумма продажи | Сумма покупки |
| 16 | 100 500 | 45 700 |
| … | … | … |

Табл.13 **-** Пункт обмена

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № отделения | Адрес | Банк | Сотрудники | Должность сотрудника | Код\_пункта обмена | Код\_Банка |
| 404 | Ул.  Кирова д.56 | ВТБ | Борисов | Кассир | 11 | 24 |
| … | … | … | … | … | … | … |

Табл.14 **-** Банк

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название | Адрес | Телефон гор. линии | Код\_банка | Код\_вида банка |
| ВТБ | Ул.  Победы д.76 | 8800555 | 25 | 7 |
| … | … | … | … | … |

Табл.14 **-** Тип сделки

|  |  |
| --- | --- |
| Тип | Код\_типа сделки |
| Покупка | 42 |
| … | … |

Табл.15 **-** Сотрудники

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Фамилия | Имя | Отчество | Должность | Код\_сотрудника | Код\_должности | Код\_пункта обмена |
| Федотов | Степан | Петрович | Кассир | 15 | 1 | 3 |
| … | … | … | … | … | … | … |

Табл.16 **-** Вид банка

|  |  |
| --- | --- |
| Код\_вида банка | Вид |
| 4 | Центральный |
| … | … |

Табл.17 **-** Должность

|  |  |  |
| --- | --- | --- |
| Название | Оклад | Код\_должности |
| Кассир | 15 000 | 33 |
| … | … | … |

Табл.18 **-** Валюта

|  |  |
| --- | --- |
| Код\_валюты | Название |
| 03 | Доллар |
| … | … |

Табл.19 **-** Бланк отчётности

|  |  |
| --- | --- |
| Код\_бланка отчетности | Справка |
| 1 | О продаже |
| … | … |

Табл.20 **-** Реестр

|  |  |  |  |
| --- | --- | --- | --- |
| Код\_реестра | Код\_сделки | Код\_бланка | Дата |
| 4 | 2 | 1 | 2019-11-10 |
| … | … | … | … |

# 4.Разработка таблиц и схемы базы данных.

Для создания данной системы управления данными понадобятся следующие таблицы:

* Сотрудники;
* Клиенты;
* Пункт обмена;
* Банк;
* Вид банка;
* Валюта;
* Должность;
* Сделки;
* Тип сделки;
* Бланк отчётности;
* Реестр.

В СУБД MySQL создание таблицы происходит с помощью команды CREATE TABLE.

Создание таблицы Клиент:

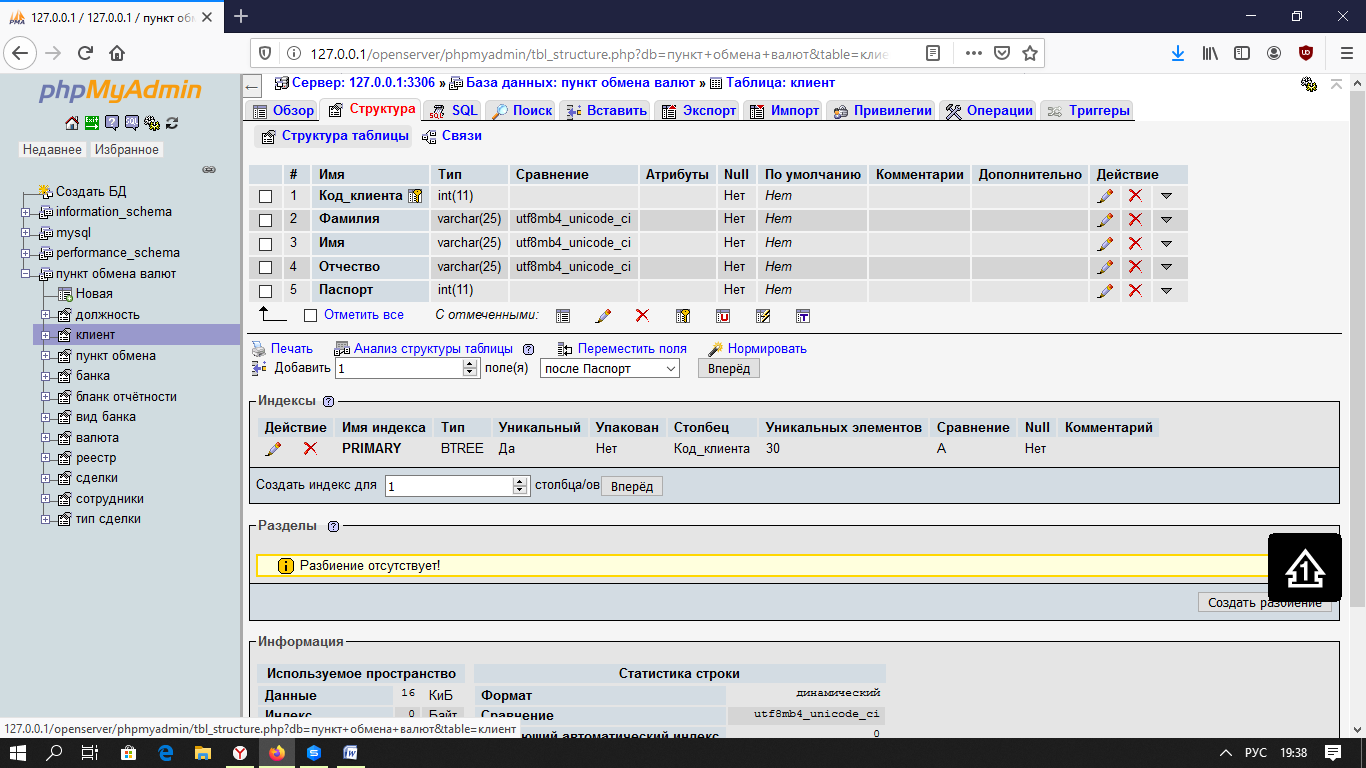
CREATE TABLE `Пункт обмена валют`.`Клиент` (`Код\_клиента` INT NOT NULL, `Фамилия` varchar(25), `Имя` varchar(25) NOT NULL, `Отчество` varchar(25) NOT NULL,`Паспорт` int(11) NOT NULL) ENGINE = MyISAM;

Добавление первичного ключа таблицы Клиент:

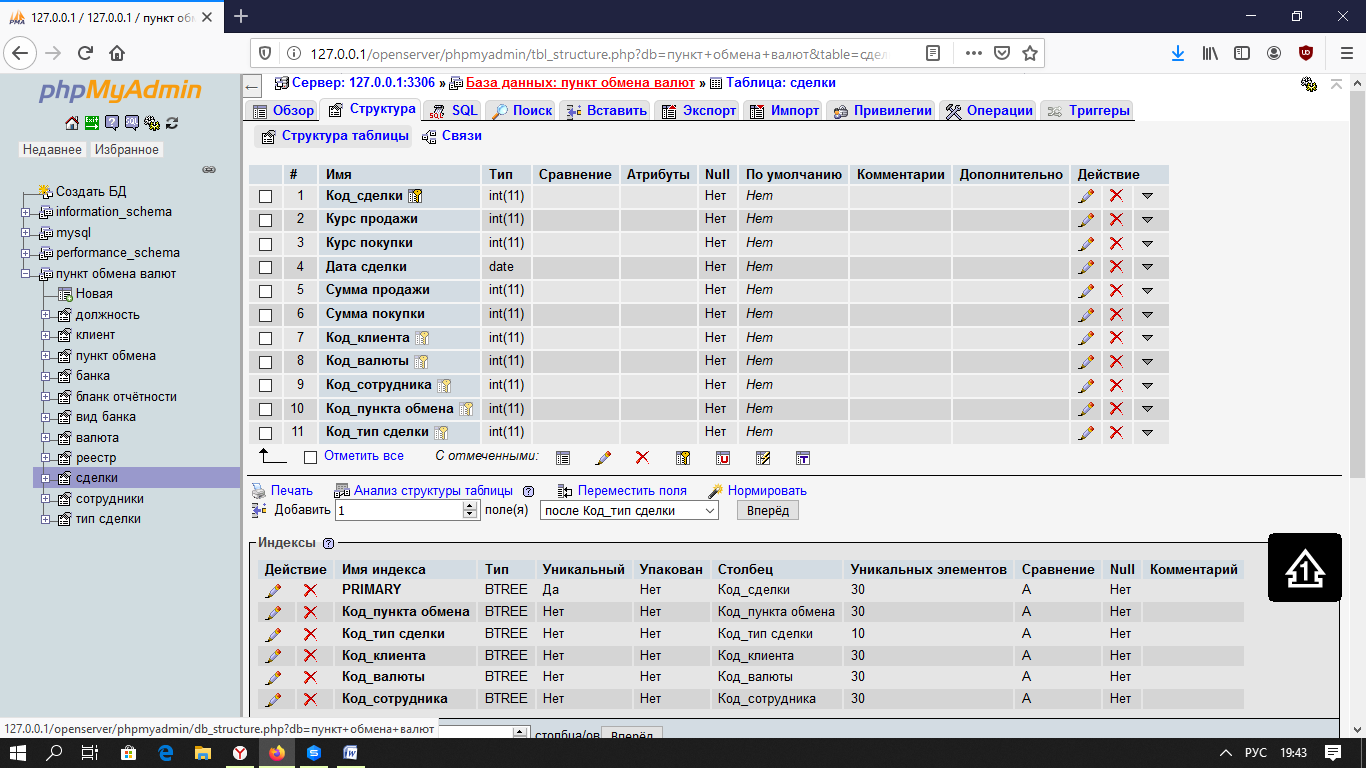
ALTER TABLE `Клиент` ADD PRIMARY KEY(`Код\_клиента`)

Далее все таблицы и первичные ключи делаются по данному примеру.

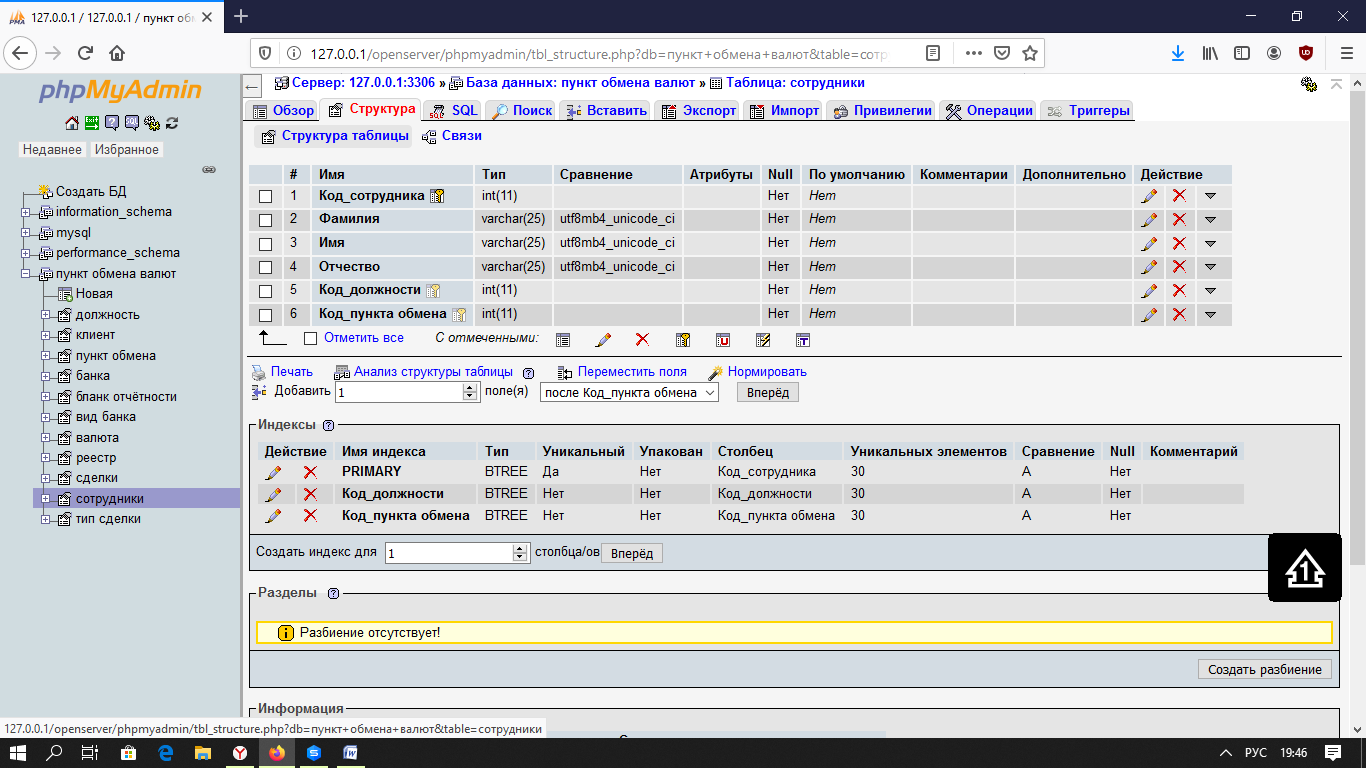
Структура таблицыКлиент**:**



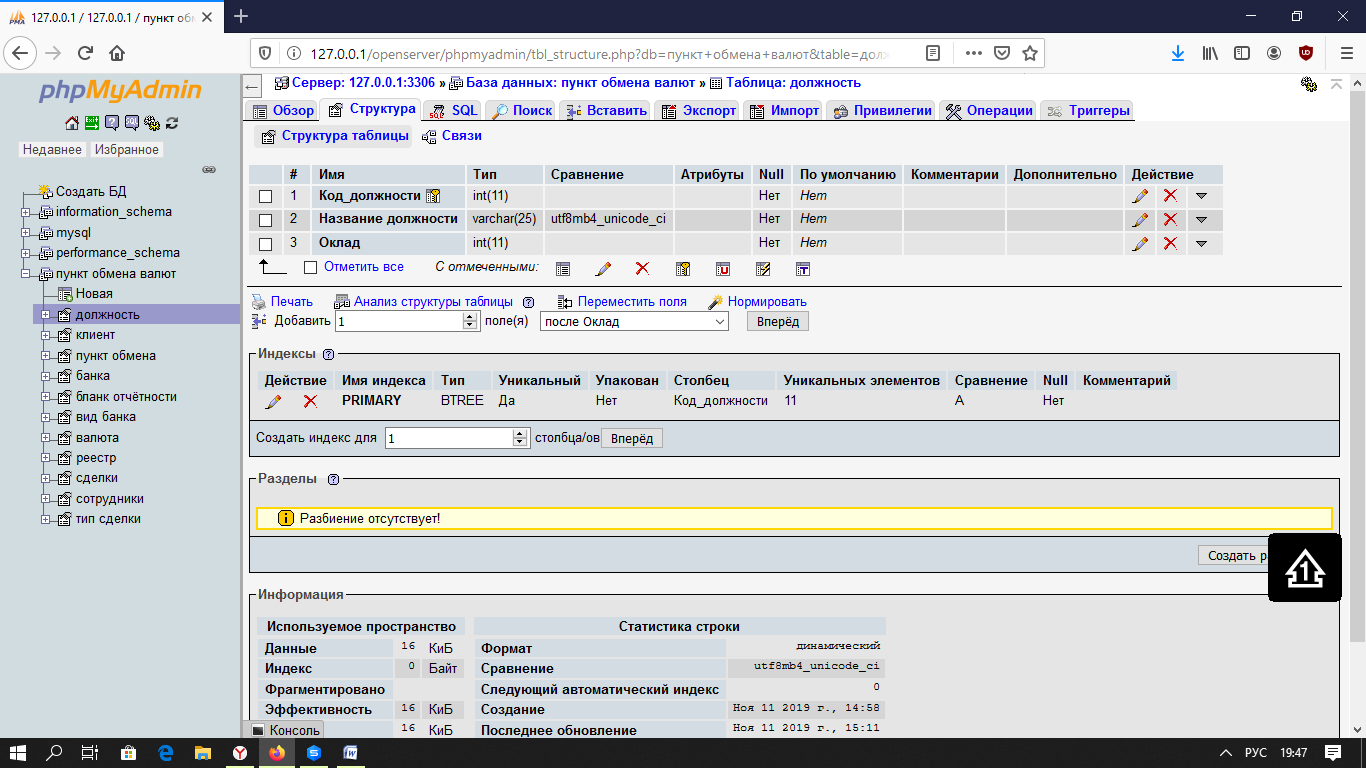
Структура таблицы Сделки**:**

****

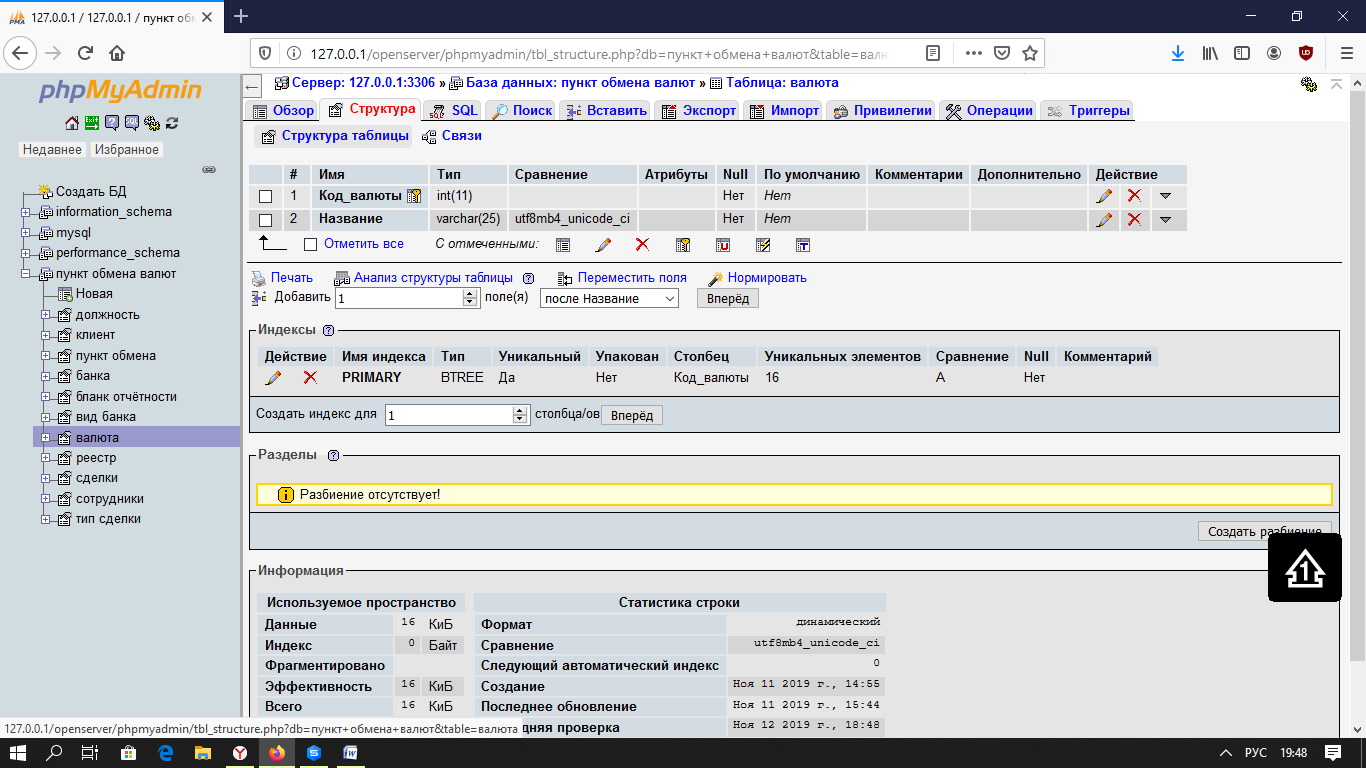
Структура таблицы Сотрудники**:**

****

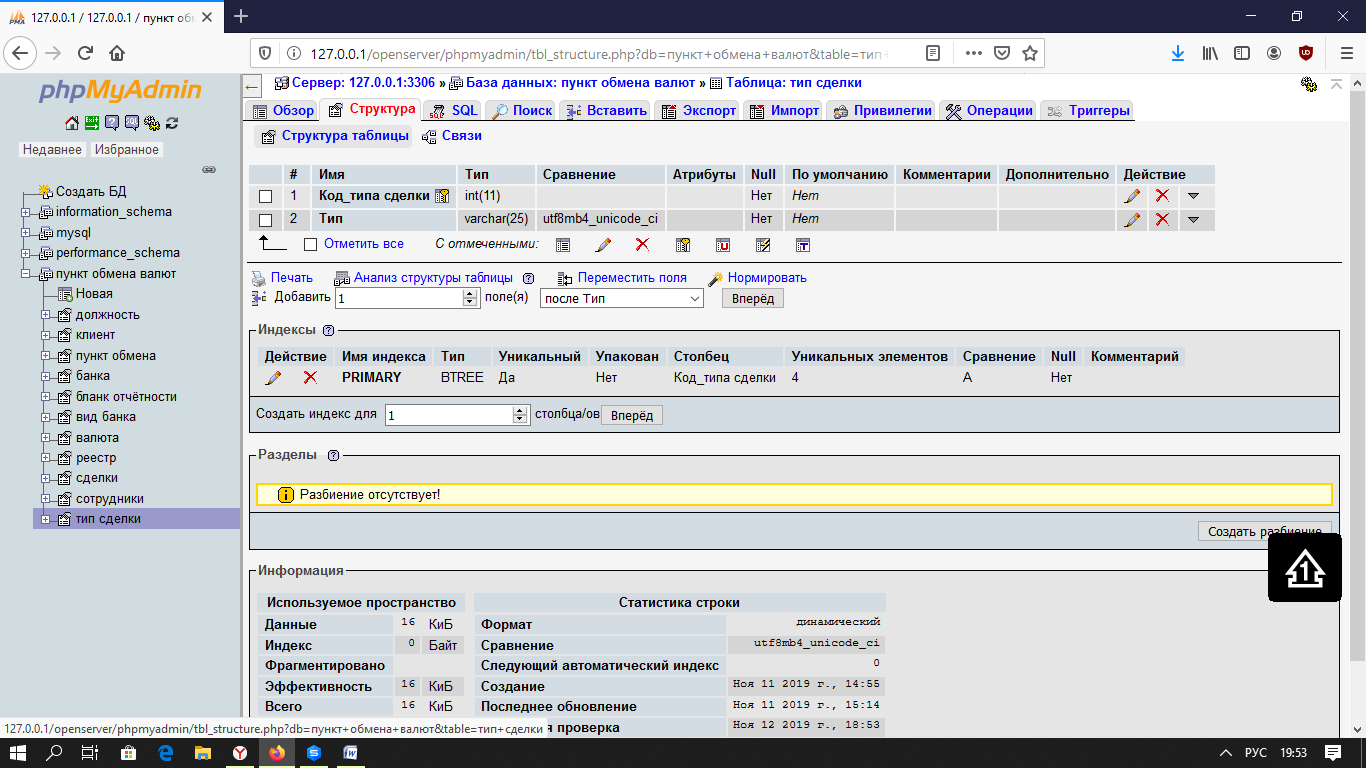
Структура таблицы Должность**:**

****

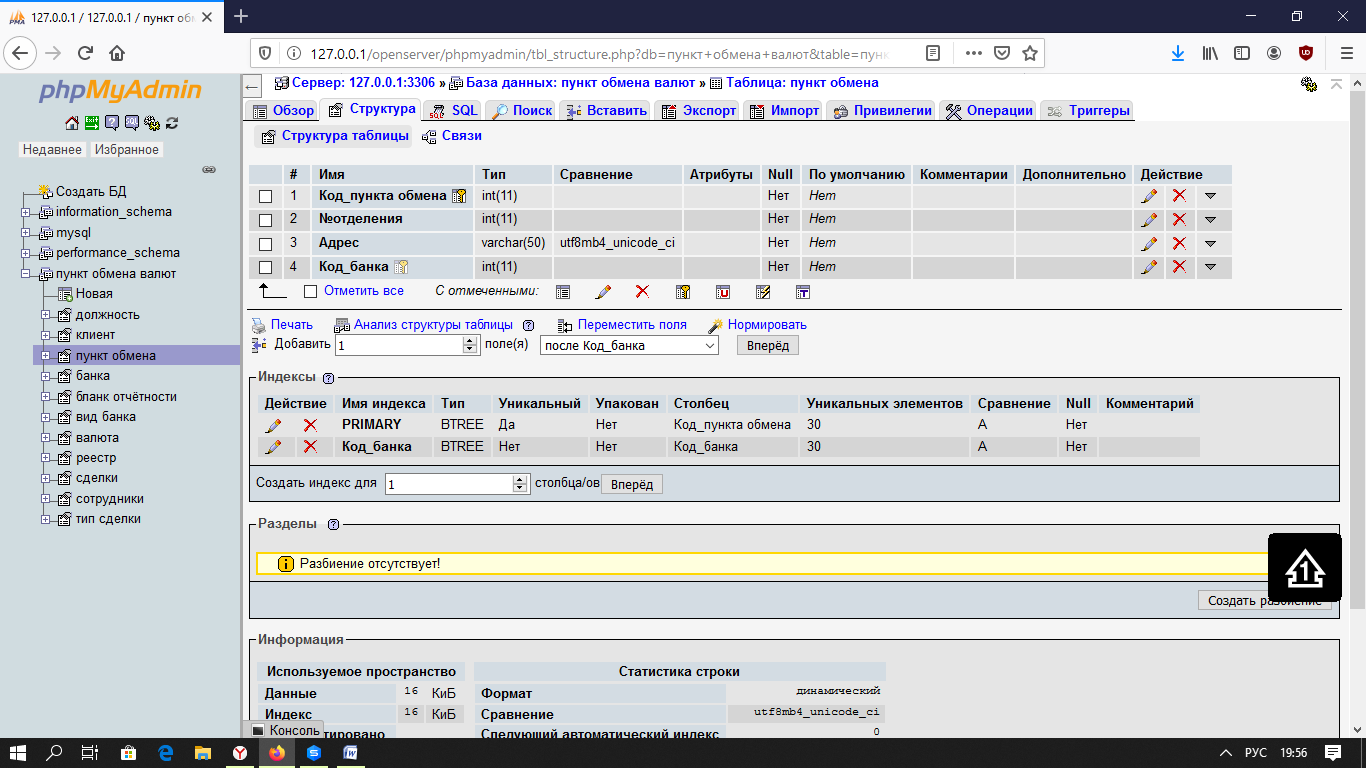
Структура таблицы Валюта**:**

****

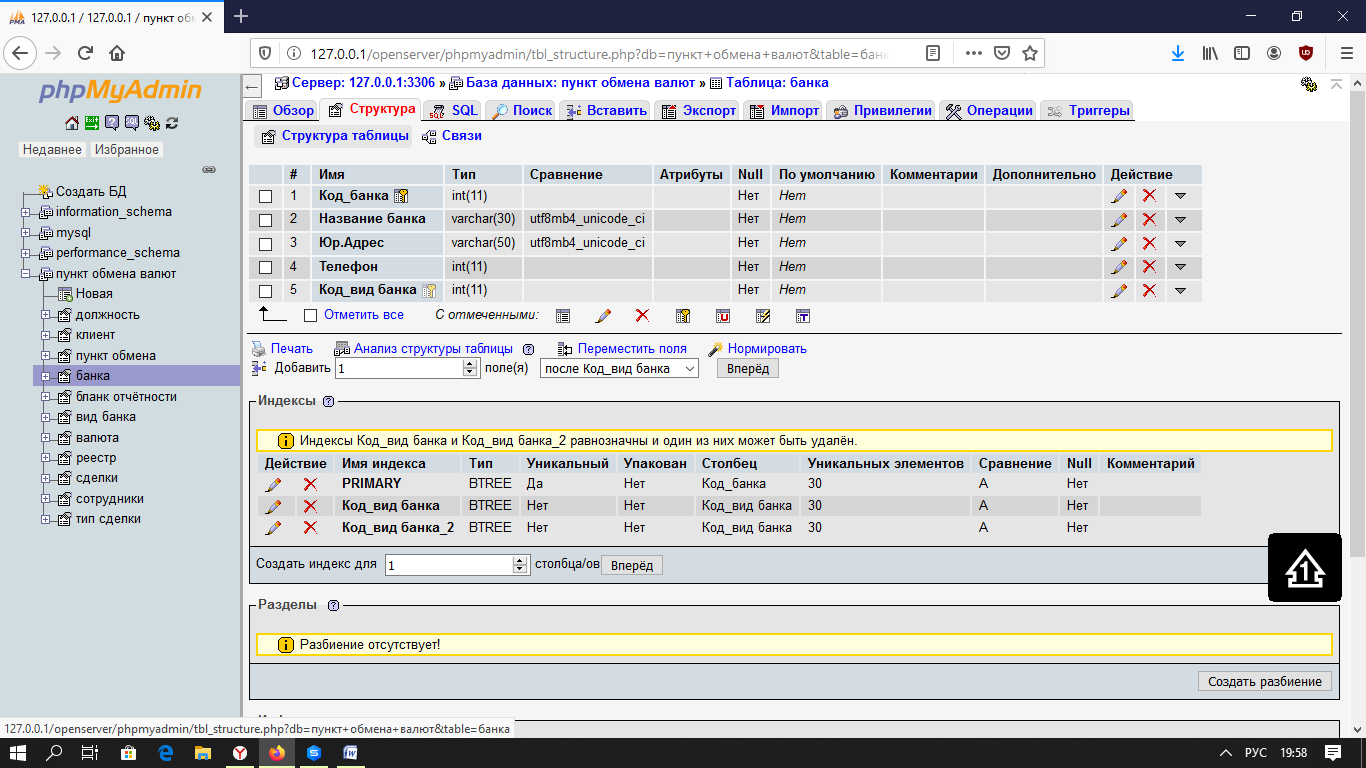
Структура таблицы Тип сделки**:**

****

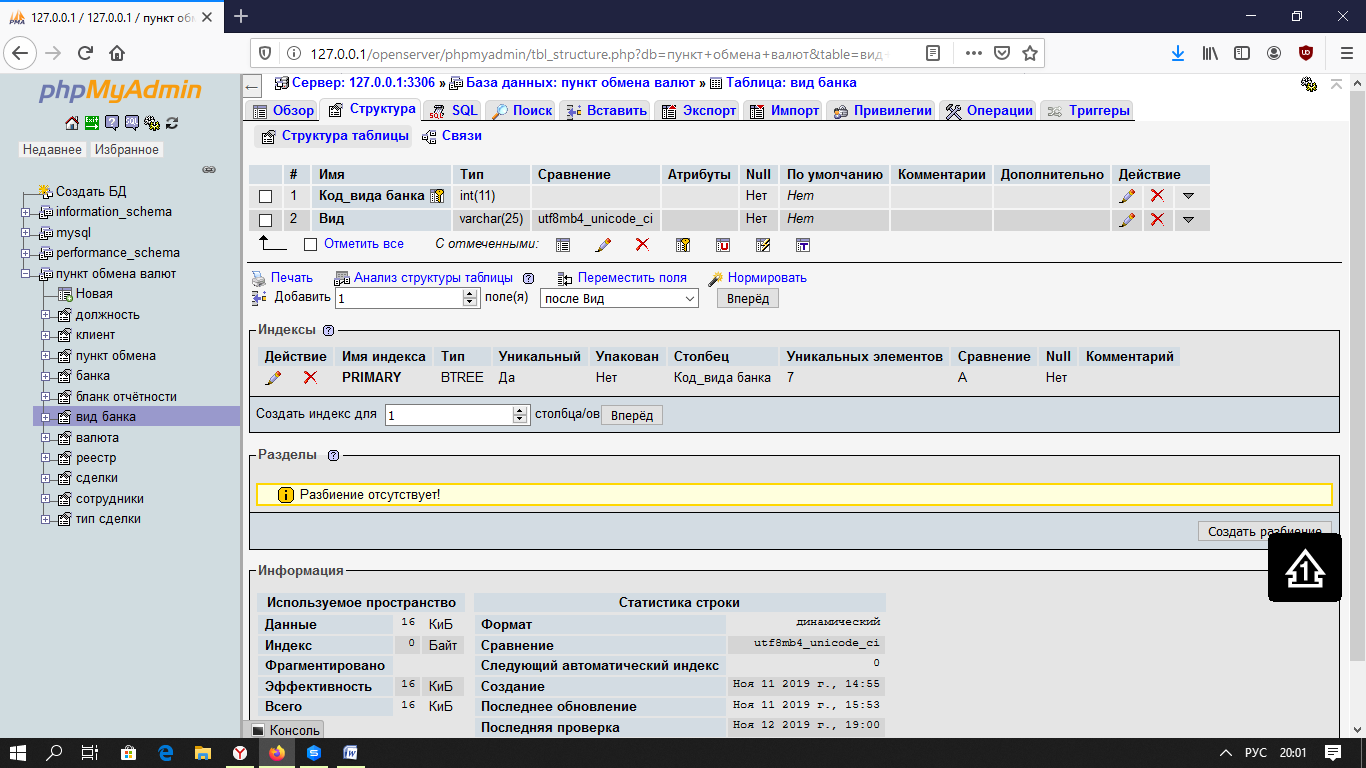
Структура таблицы Пункт обмена**:**

****

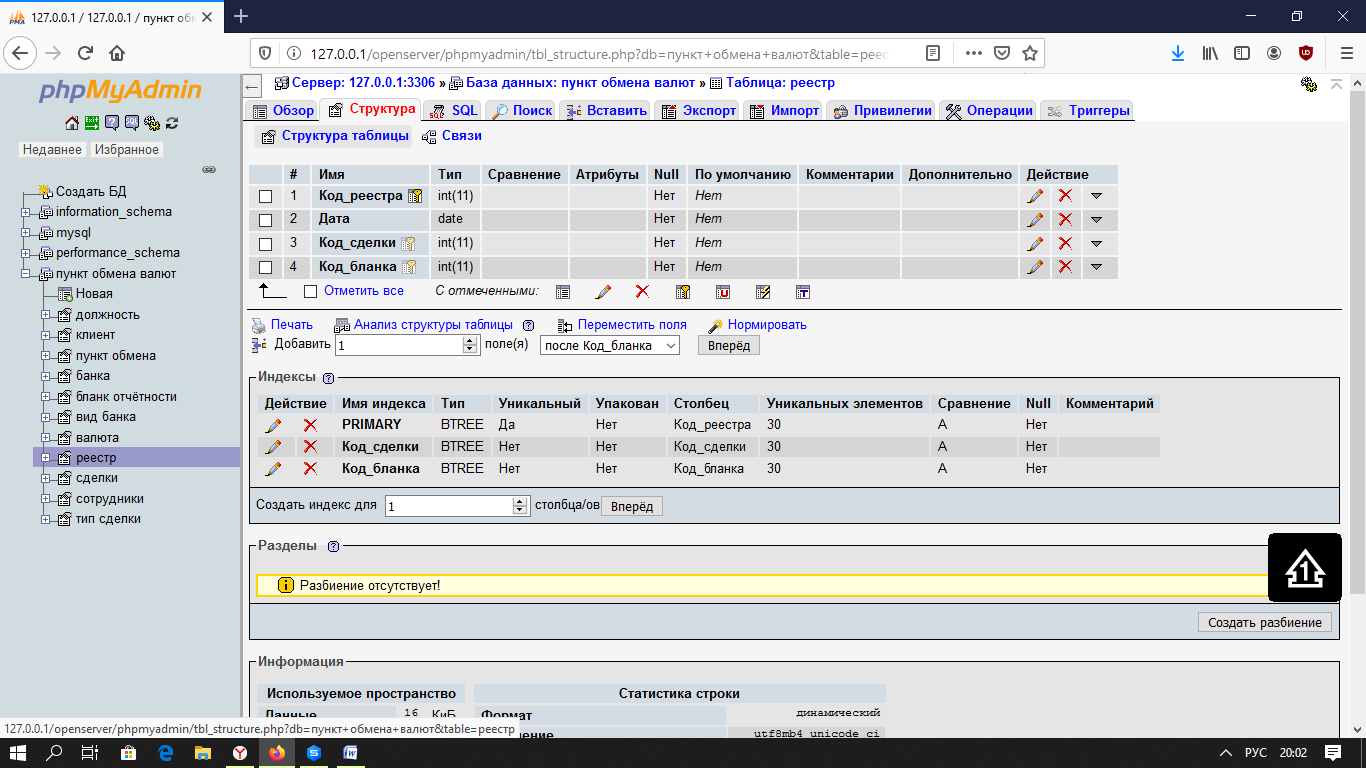
Структура таблицы Банк**:**

****

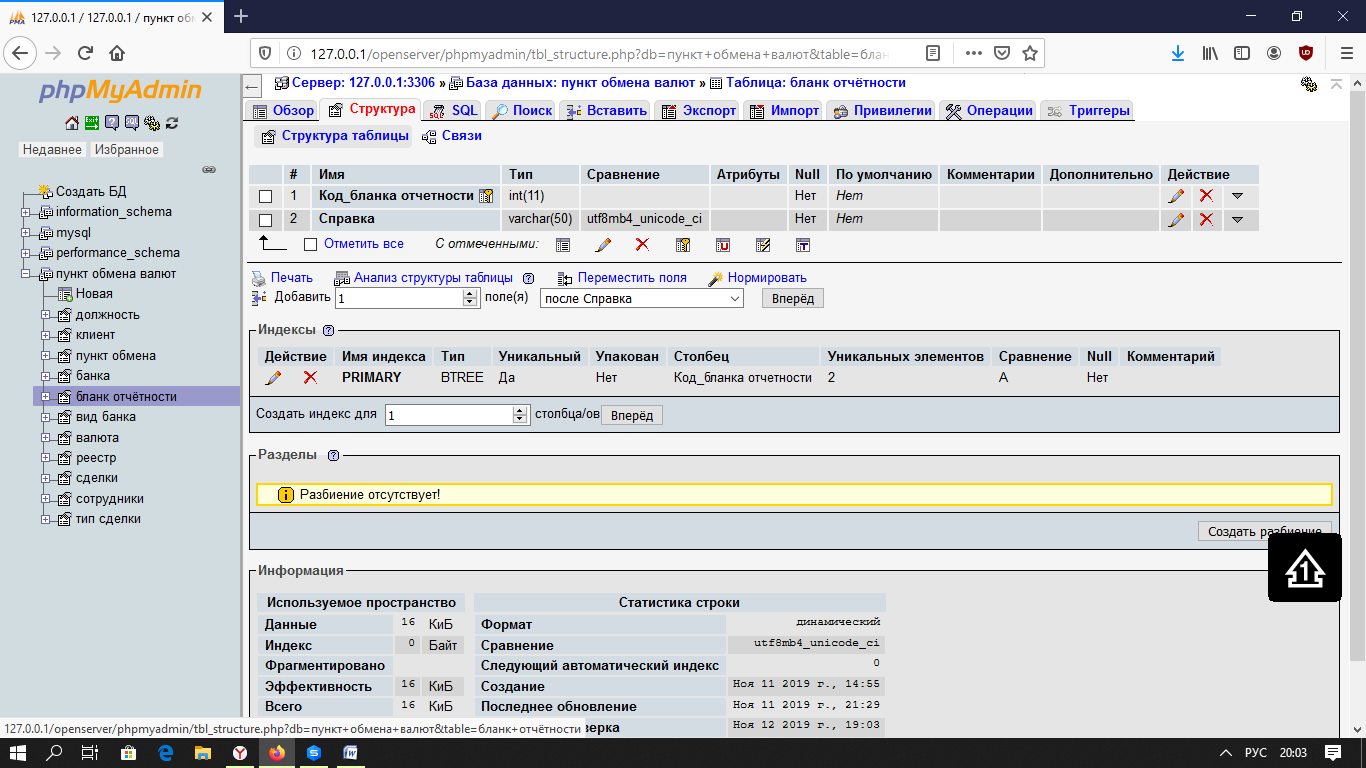
Структура таблицы Вид банка**:**

****

Структура таблицы Реестр**:**

****

Структура таблицы Бланк отчетности**:**

****

С помощью команд INSERT INTO и VALUES происходит вставка строк внутрь той или иной таблицы.

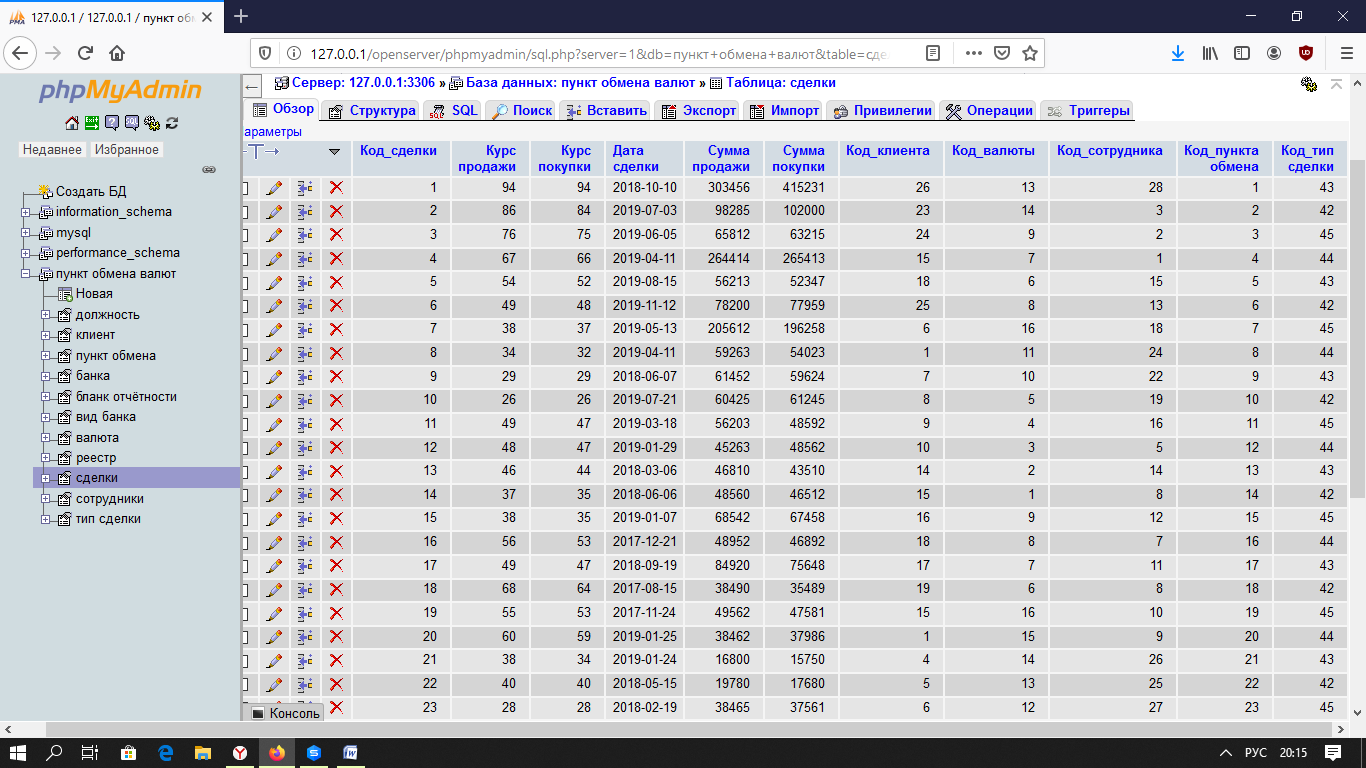
Вставка строки для таблицы «Валюта» выполняется следующим образом:

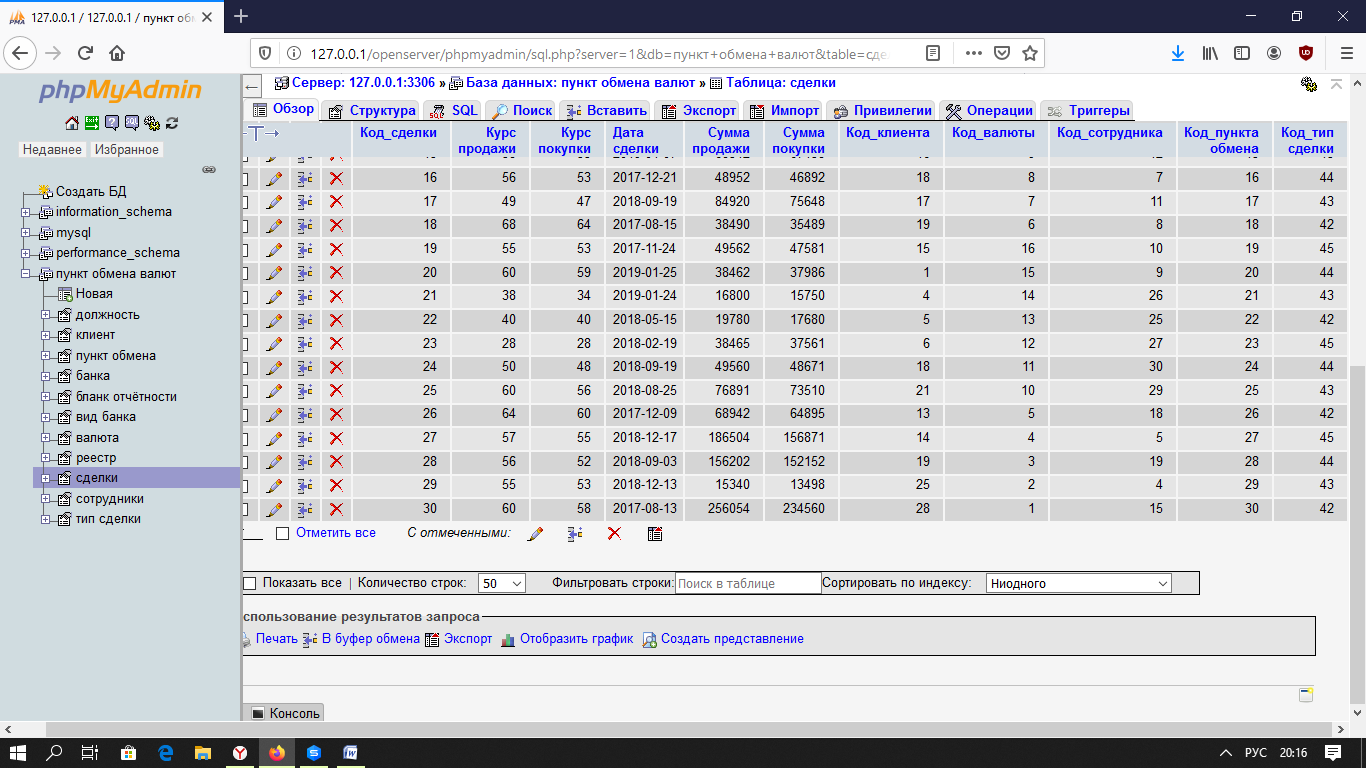


INSERT INTO `Пункт обмена валют`.`Валюта` (`id\_валюта`, `Название валюты`) VALUES ('16', 'Гвинейский доллар');

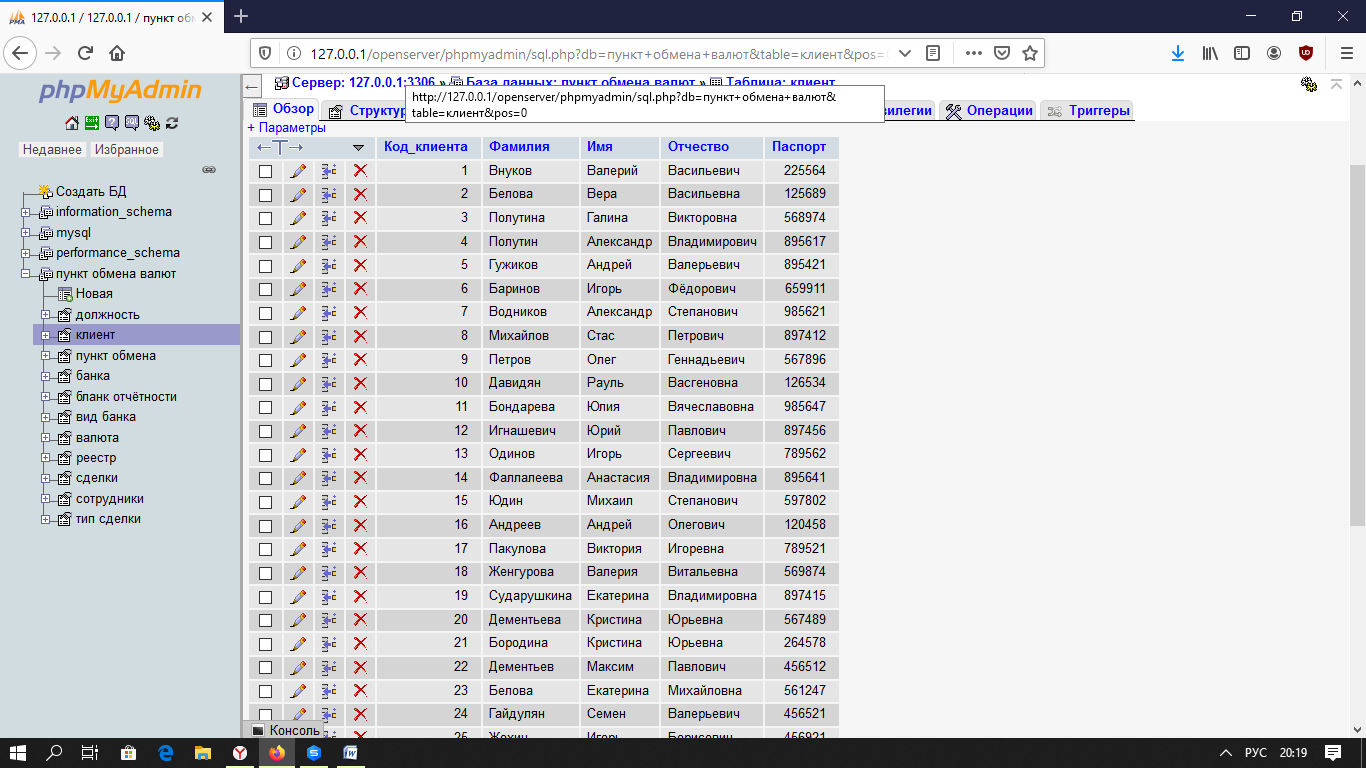
Для вставки последующих значений в каждую таблицу команды INSERT INTOиVALUES выполняются аналогично.

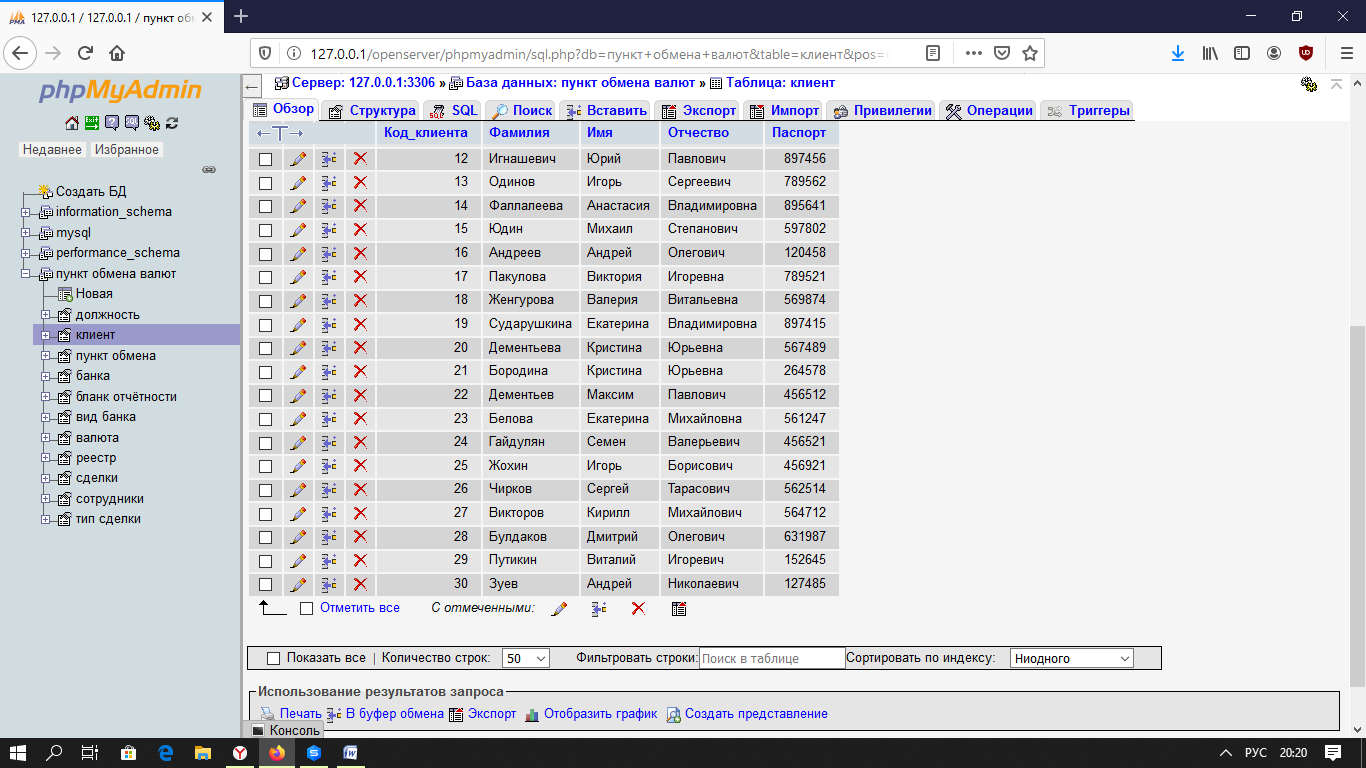
Данные в таблице Сделки:



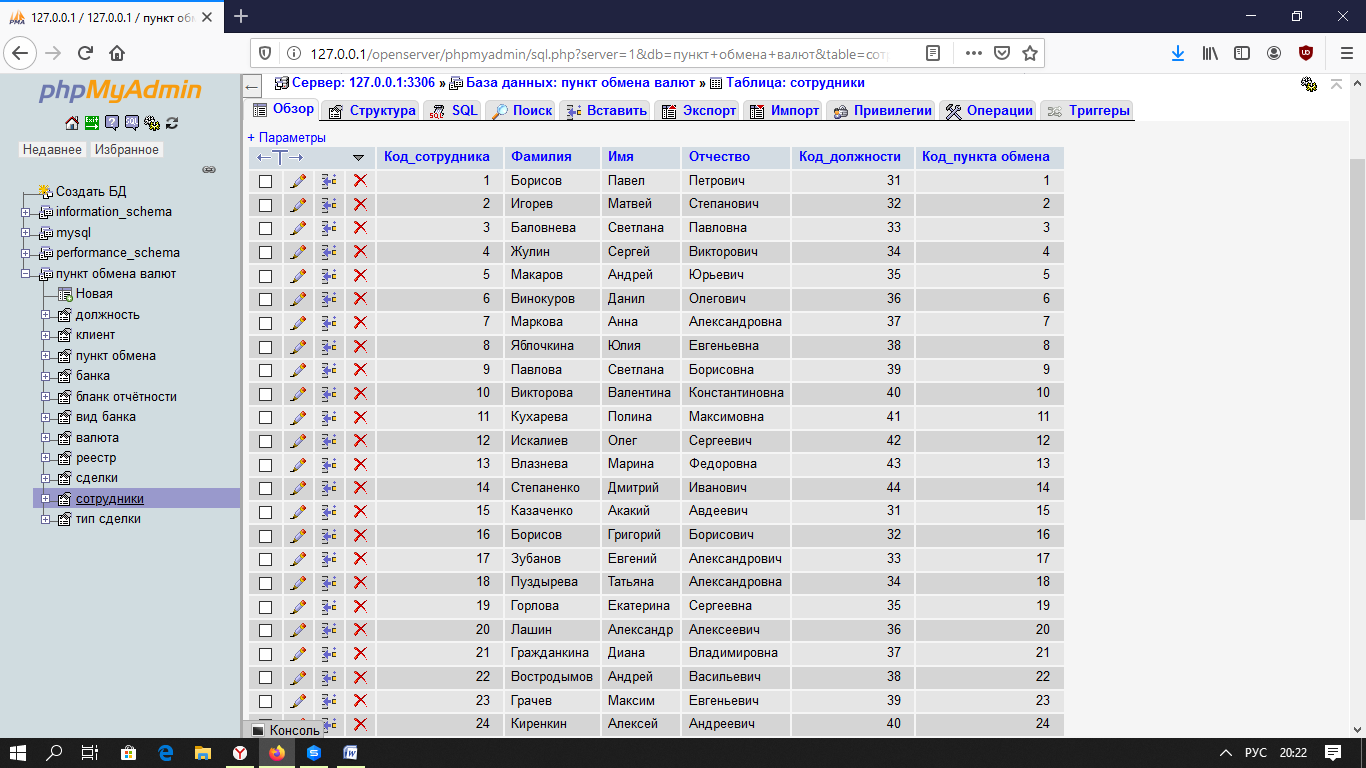


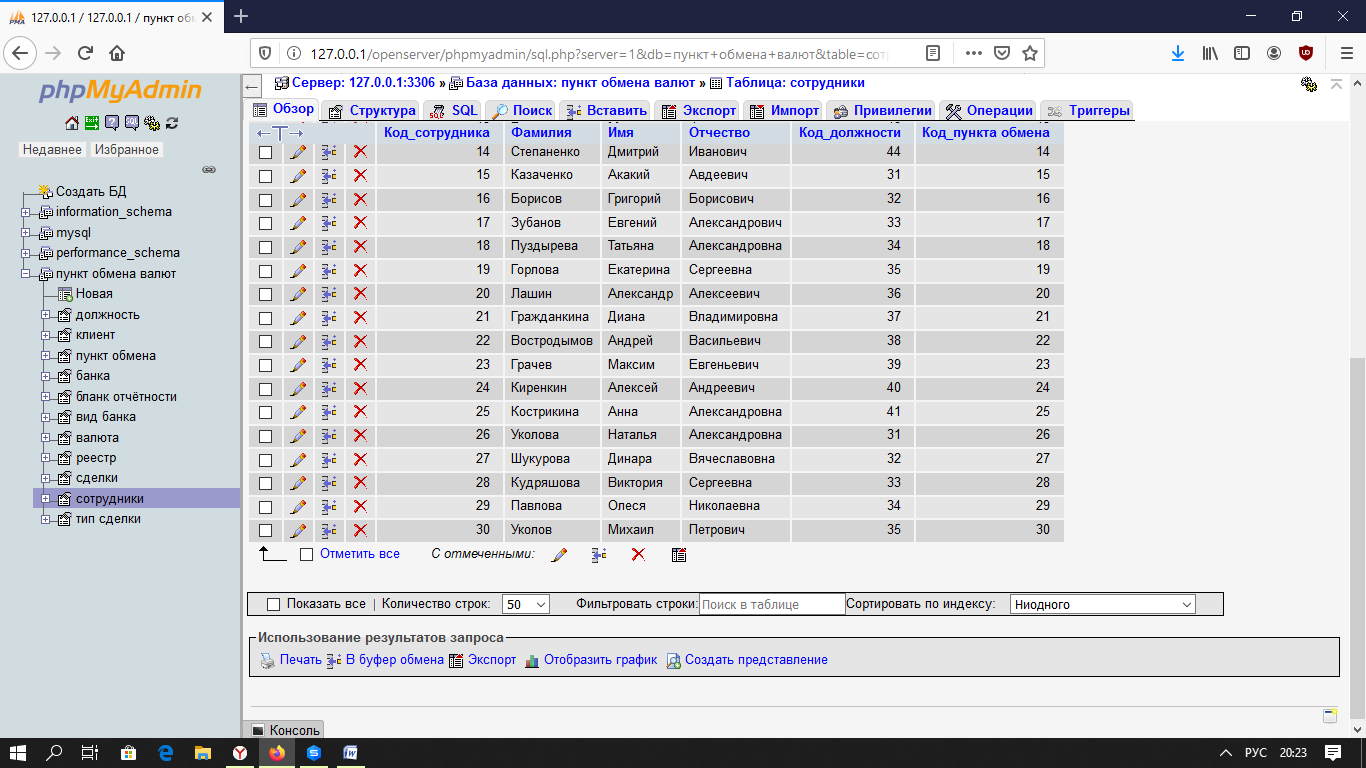
Данные в таблице Клиент:



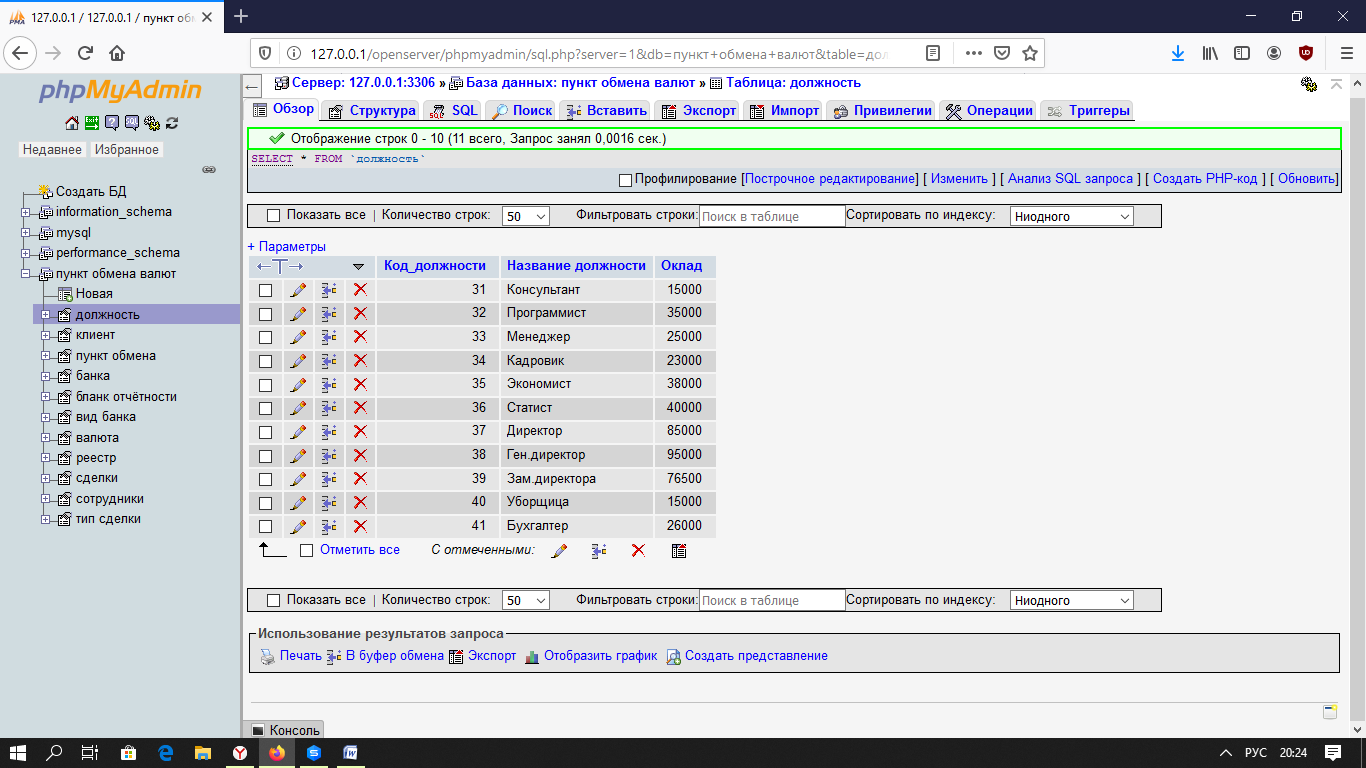


Данные в таблице Сотрудник:

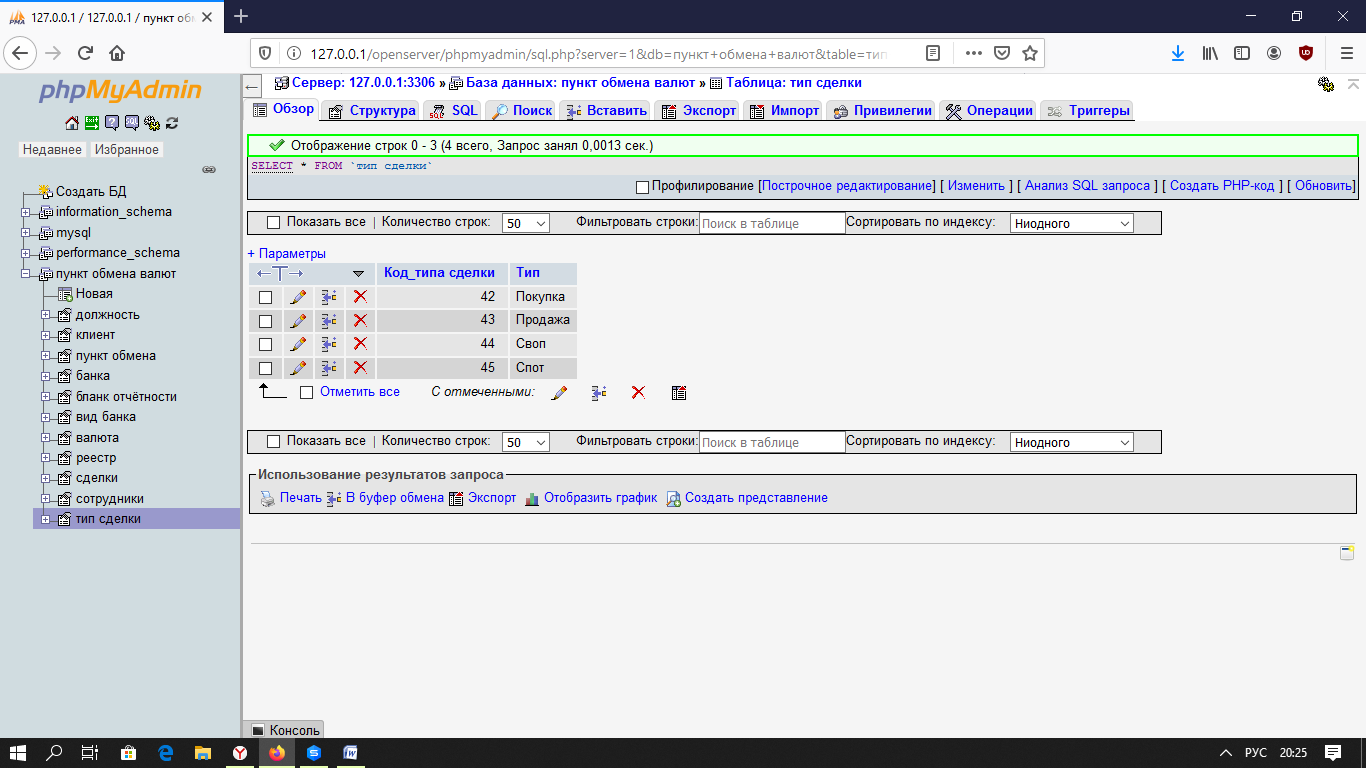
****

****

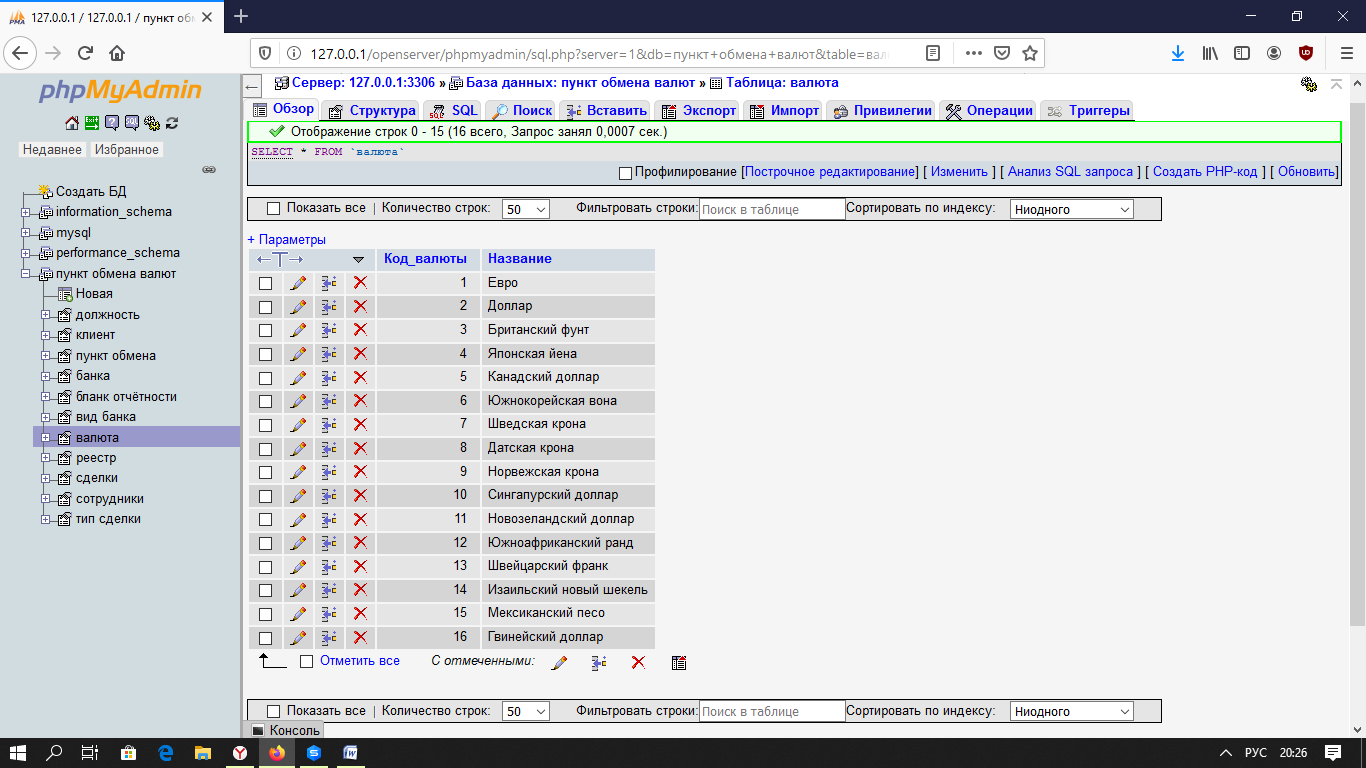
Данные в таблице Должность:



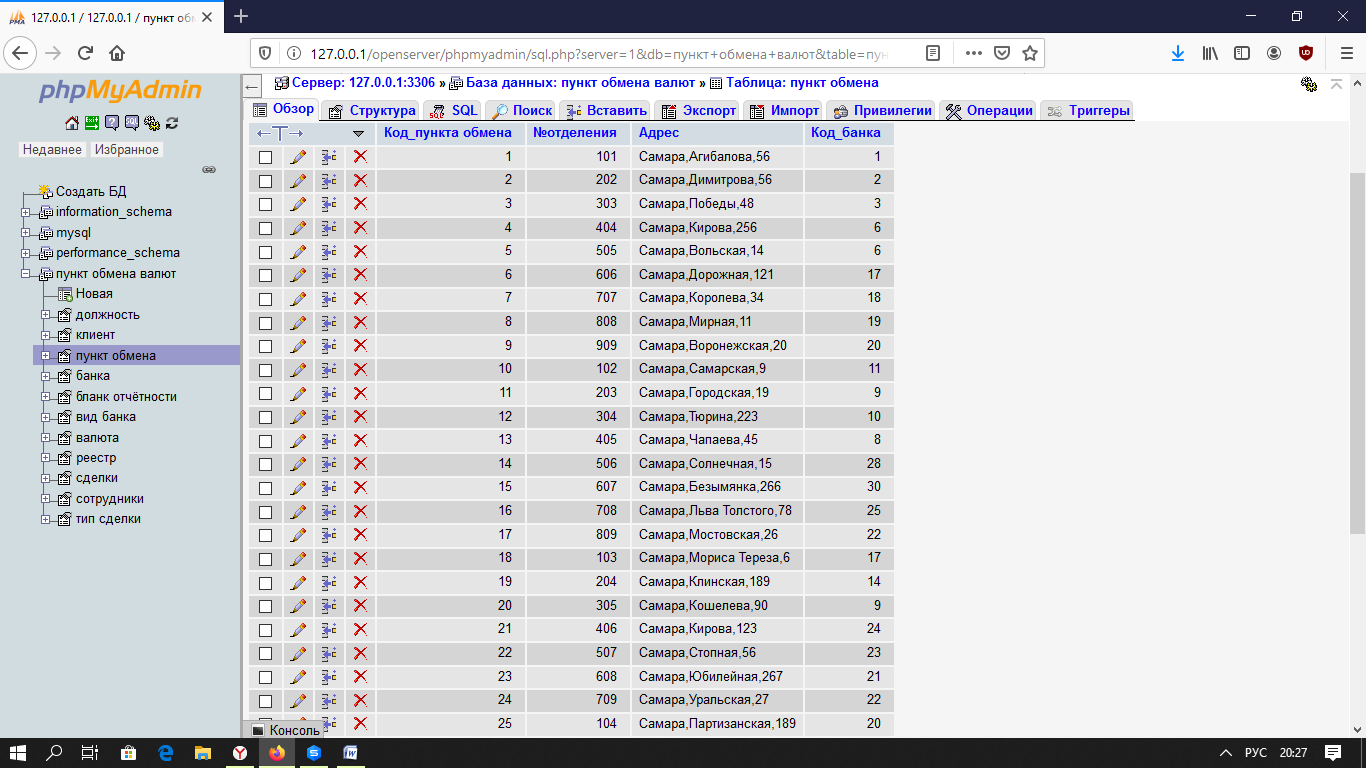
Данные в таблице Тип сделки:

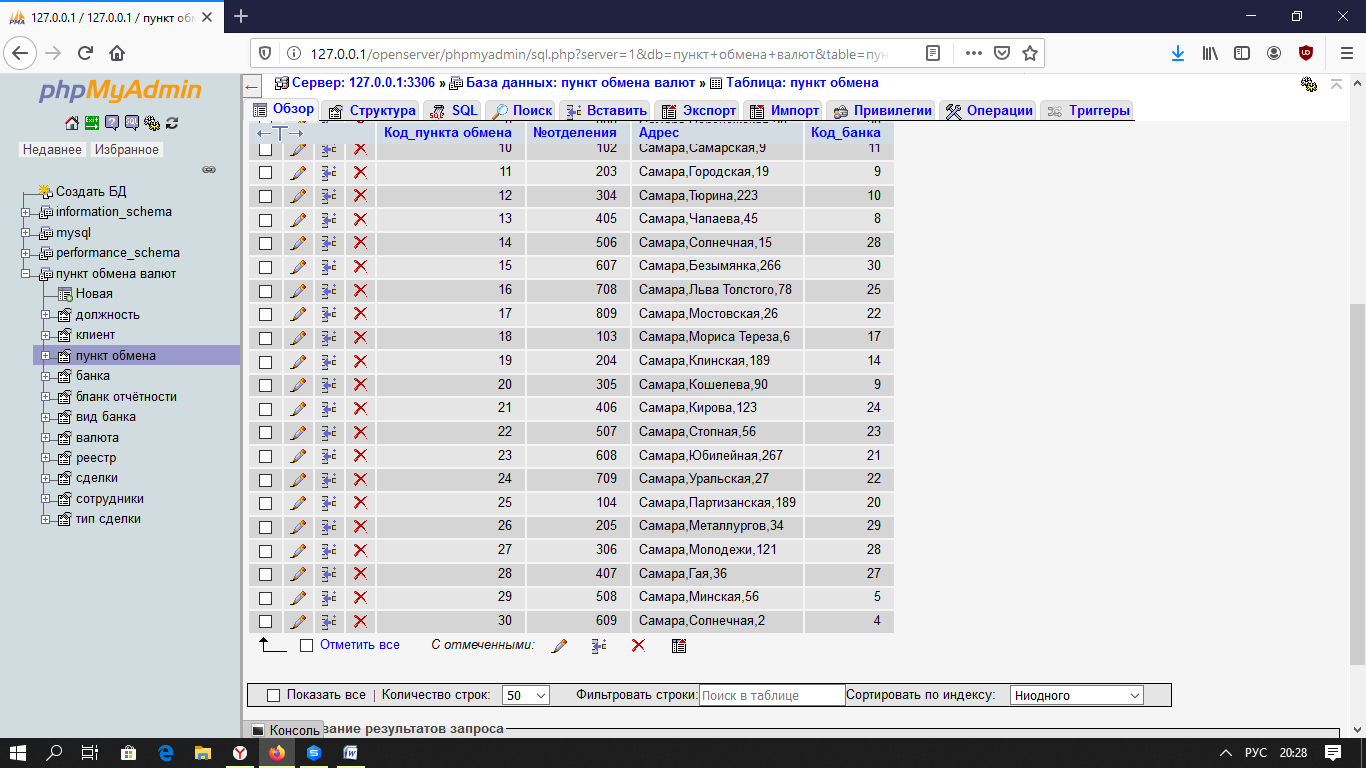
****

Данные в таблице Валюта:

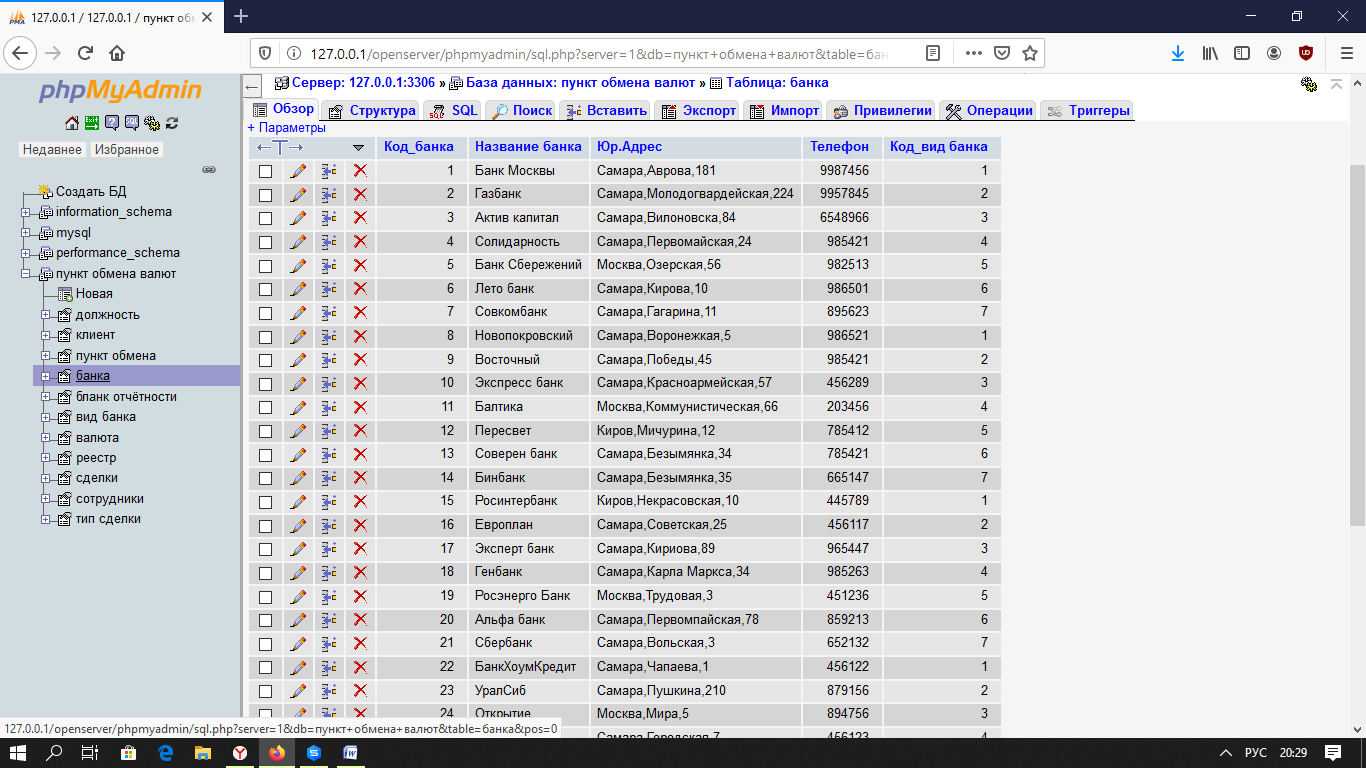
****

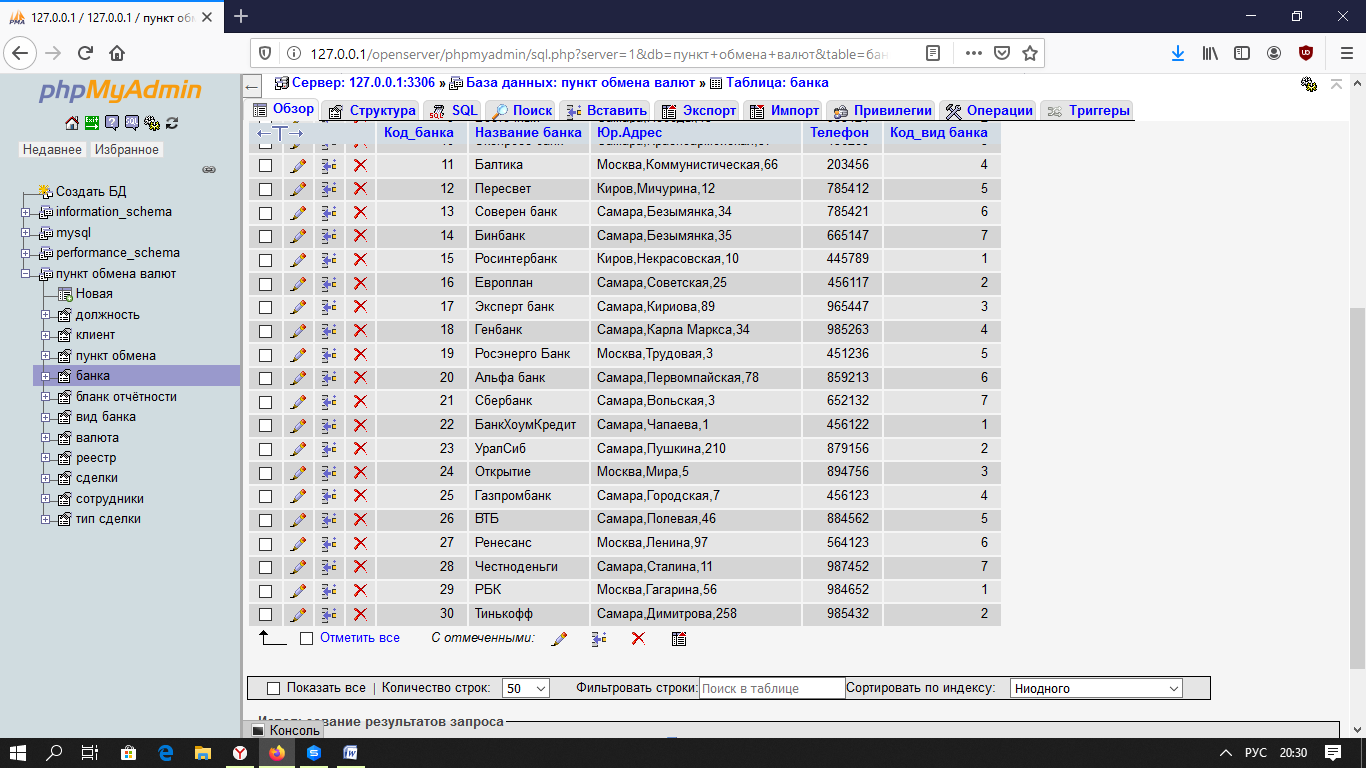
Данные в таблице Пункт обмена:



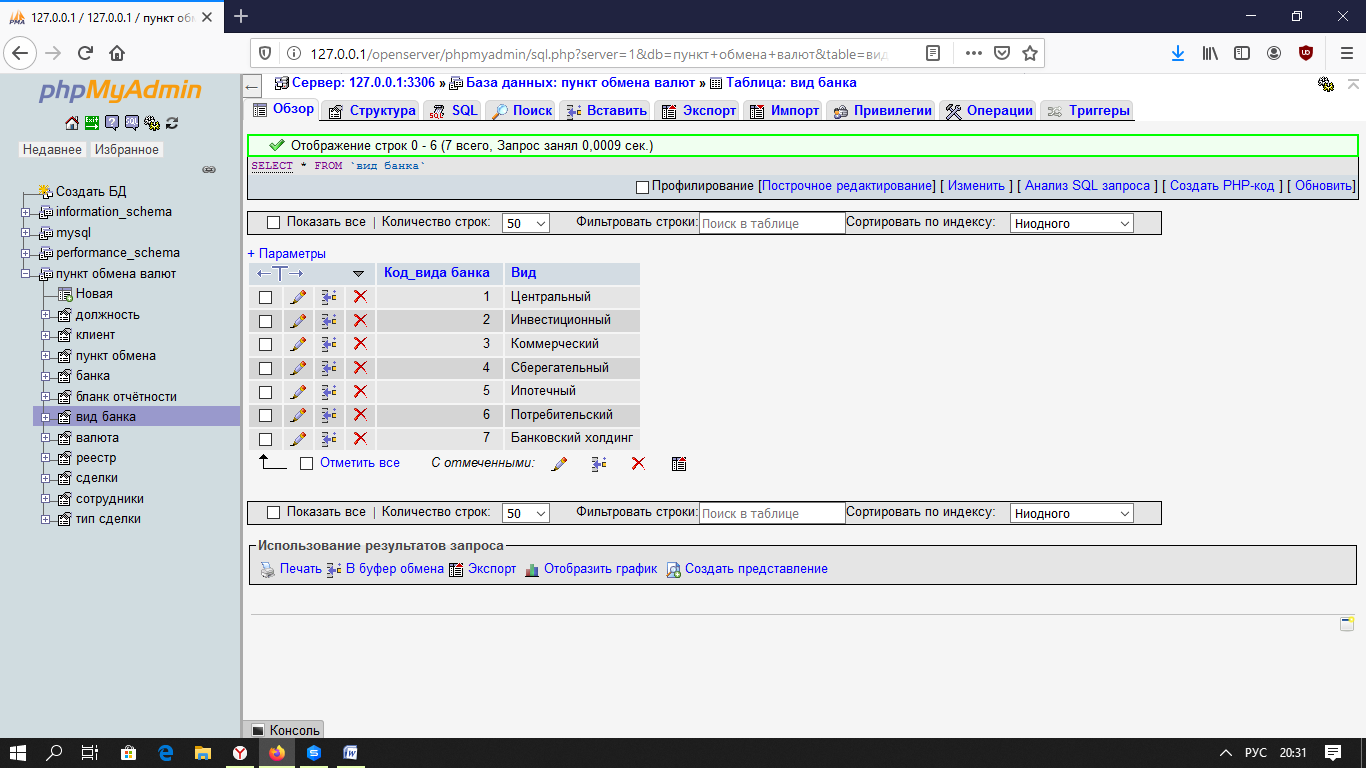


Данные в таблице Банк:

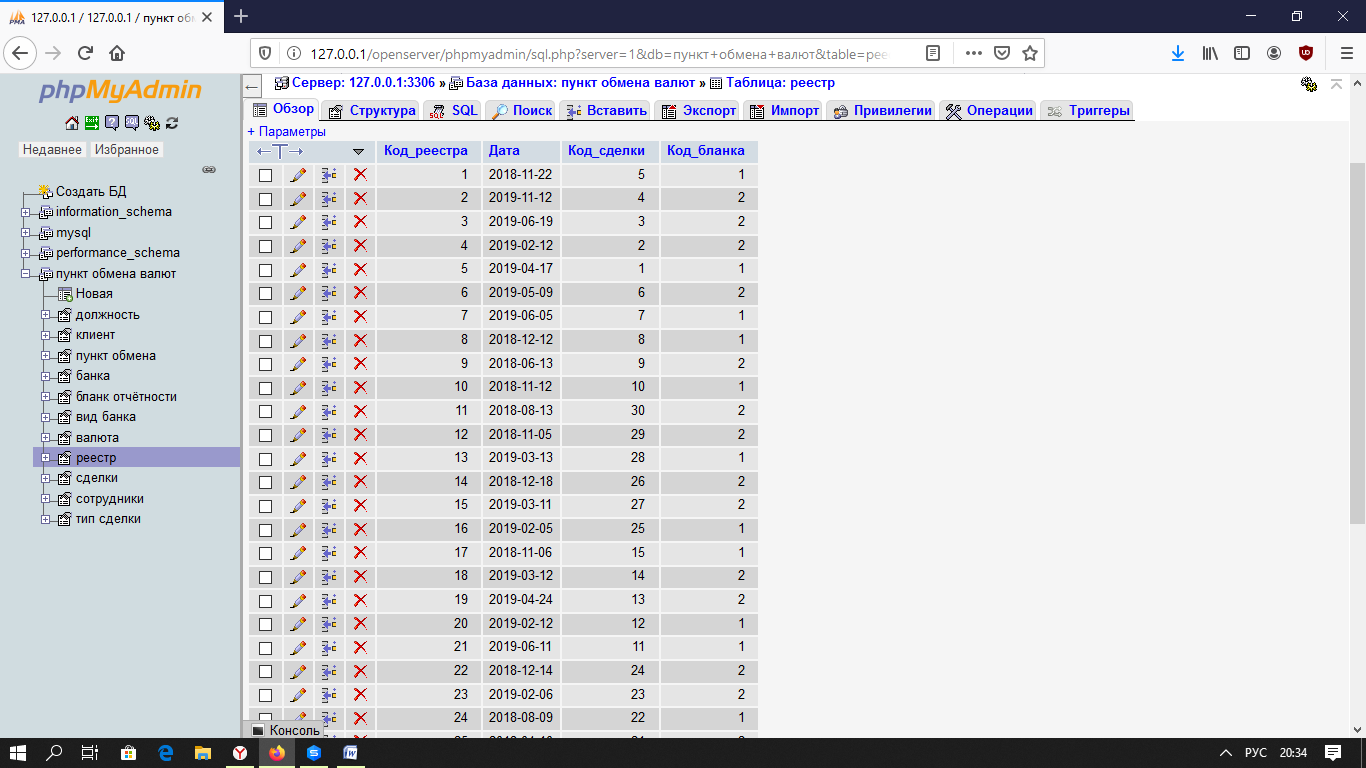


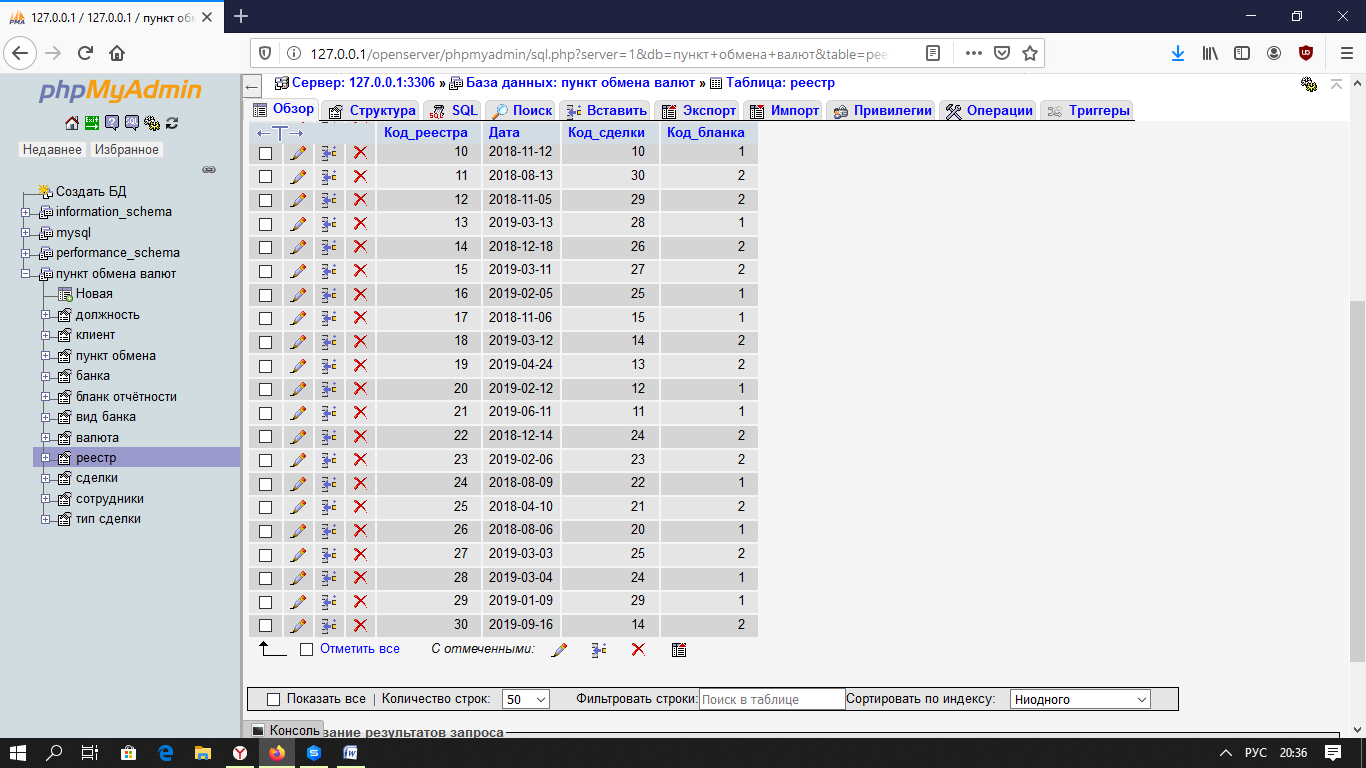


Данные в таблице Вид банка:

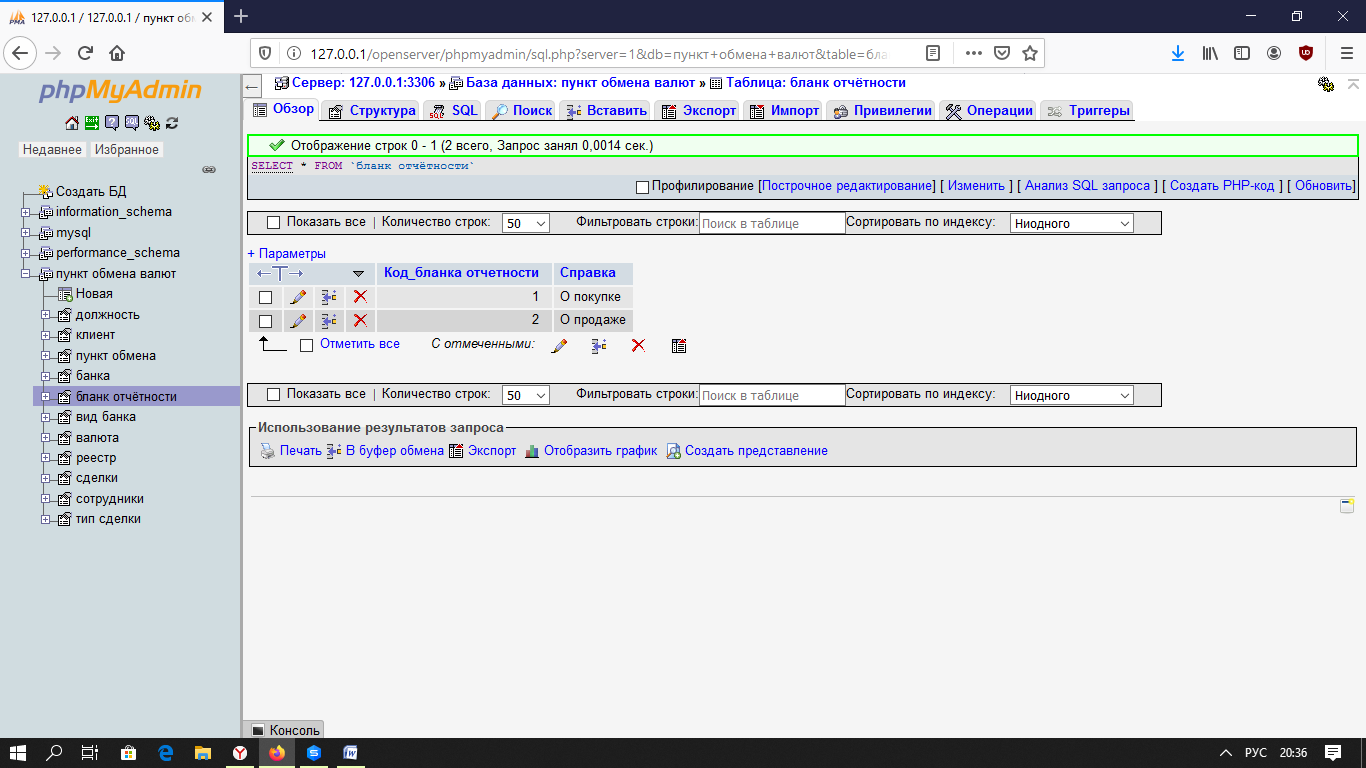


Данные в таблице Реестр:





Данные в таблице Бланк отчетности:



# 5.Запросы системы.

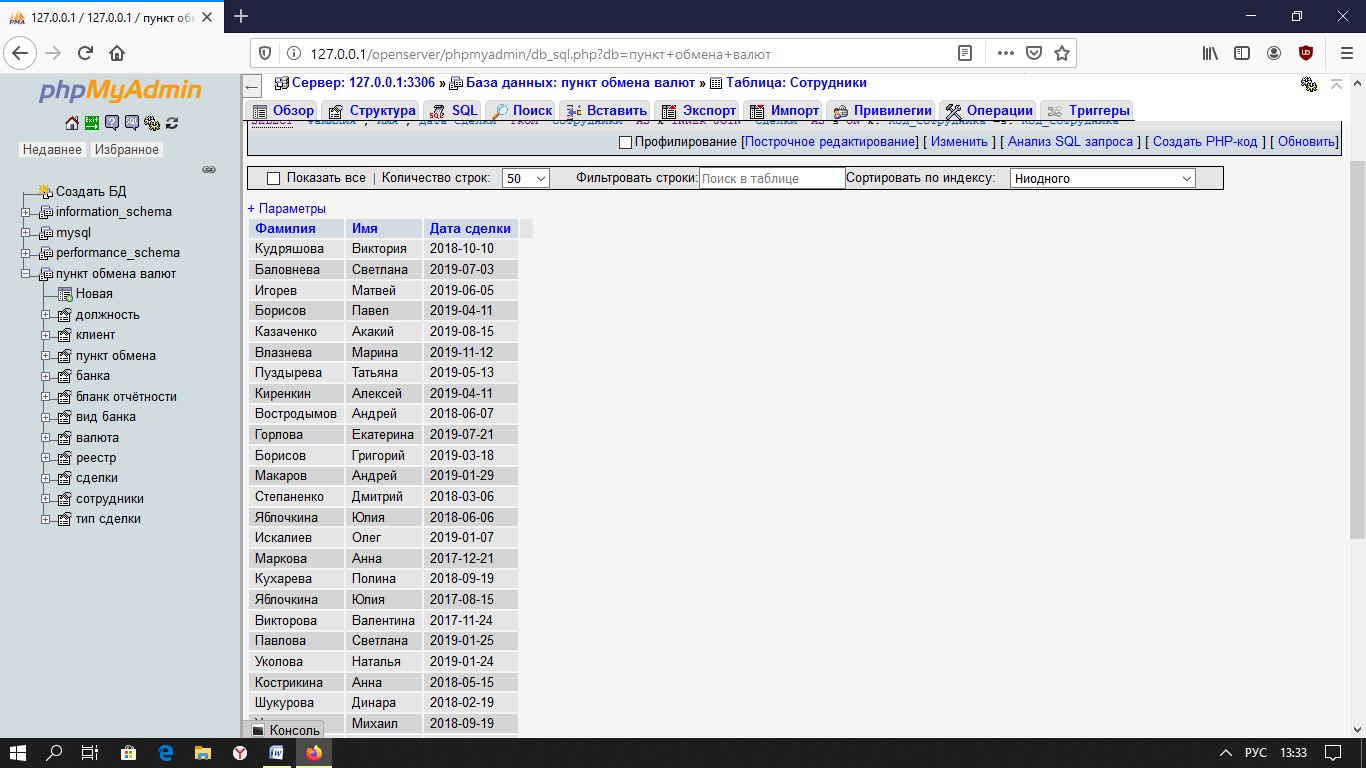
Для данной системы было создано 17 запросов:

**Запрос 1.** Внутреннее соединение по одному полю:

Выведем сотрудника и дату сделки которую провел сотрудник.

SELECT `Фамилия`,`Имя`,`Дата сделки` FROM `Сотрудник` AS k

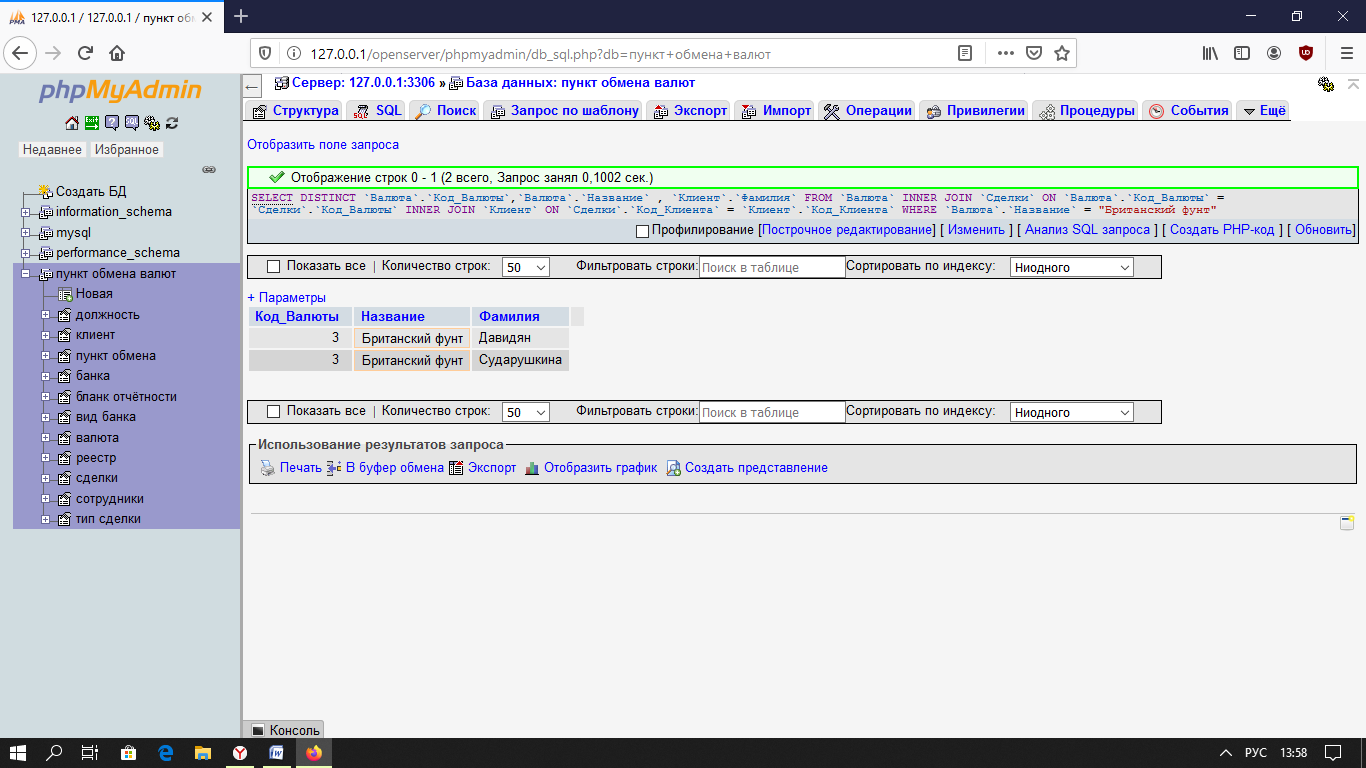
INNER JOIN `Сделки` AS s ON k.`Код\_Сотрудника`=s.`Код\_Сотрудника`



**Запрос 2.** С использованием косвенно связанных таблиц.

Выведем клиентов, совершивших сделку с валютой = британский фунт.

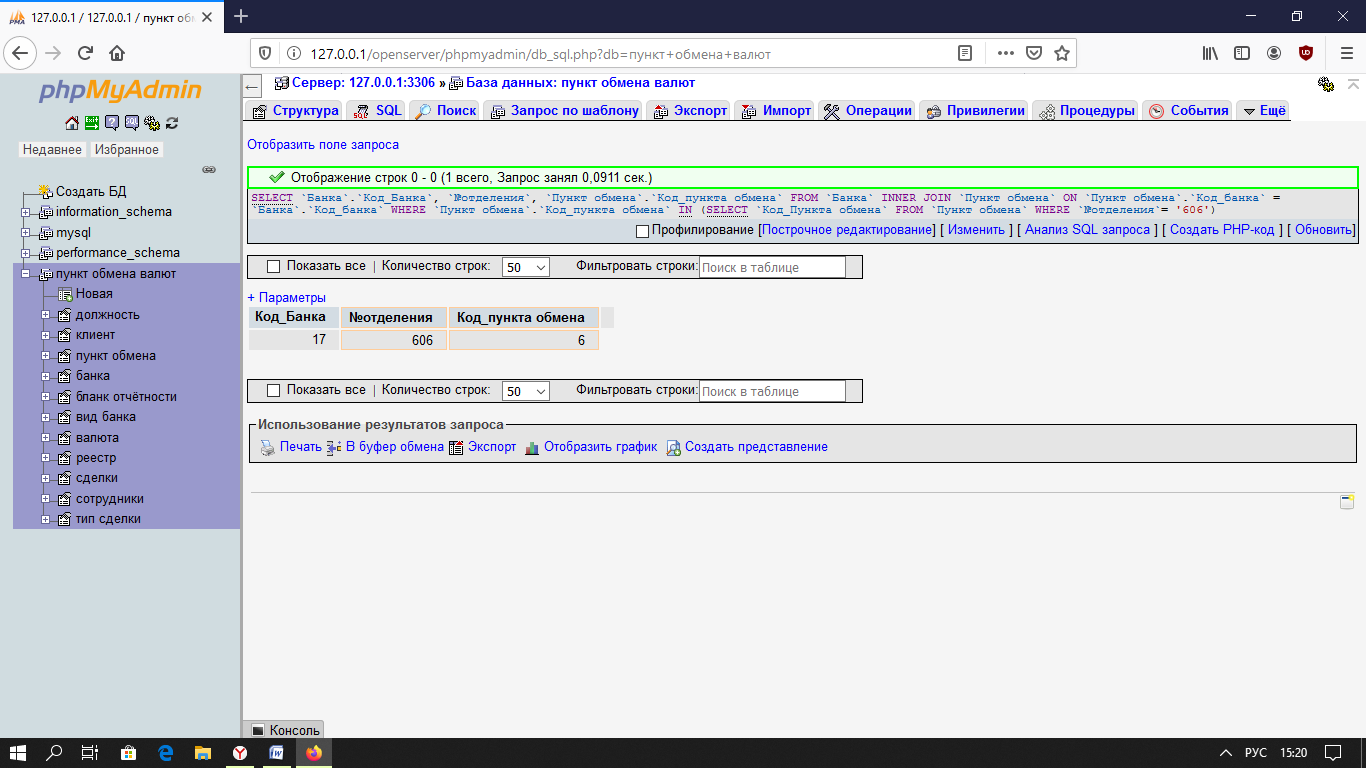
[SELECT](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) DISTINCT `Валюта`.`Код\_Валюты`,`Валюта`.`Название` , `Клиент`.`Фамилия` FROM `Валюта` INNER JOIN `Сделки` ON `Валюта`.`Код\_Валюты` = `Сделки`.`Код\_Валюты` INNER JOIN `Клиент` ON `Сделки`.`Код\_Клиента` = `Клиент`.`Код\_Клиента` WHERE `Валюта`.`Название` = "Британский фунт"



**Запрос 3.** С использованием таблиц, связанных более чем по одному полю.

Выводили код банка где номер отделения = 606.

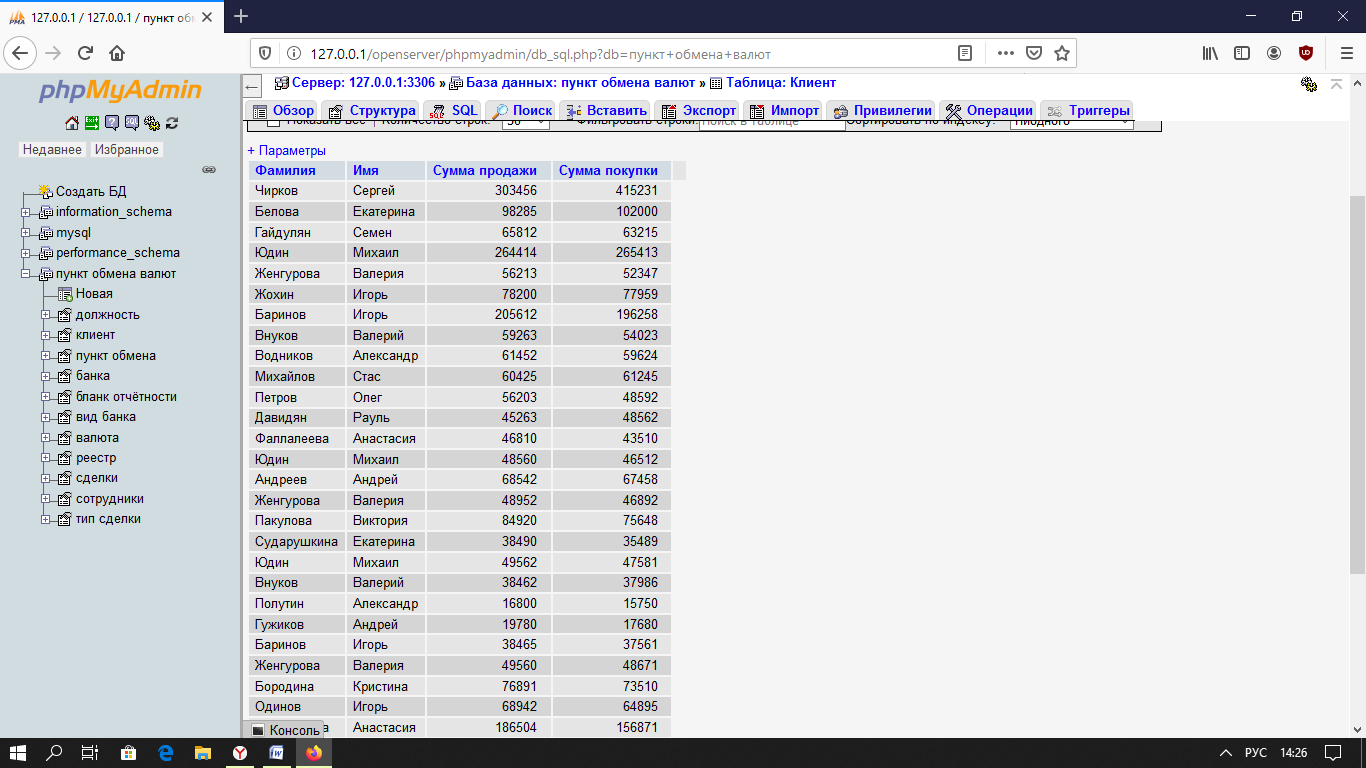
[SELECT](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) `Банка`.`Код\_Банка`, `№отделения`, `Пункт обмена`.`Код\_пункта обмена` FROM `Банка` INNER JOIN `Пункт обмена` ON `Пункт обмена`.`Код\_банка` = `Банка`.`Код\_банка` WHERE `Пункт обмена`.`Код\_пункта обмена` [IN](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/comparison-operators.html#function_in) ([SELECT](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) `Код\_Пункта обмена` FROM `Пункт обмена` WHERE `№отделения`= '606')

****

**Запрос 4.**С использованием внешнего соединения таблиц.

Выведем клиентов, посмотрим сколько вышла сумма покупки, сумма продажи, а так же убедимся не отказался ли клиент от сделки.

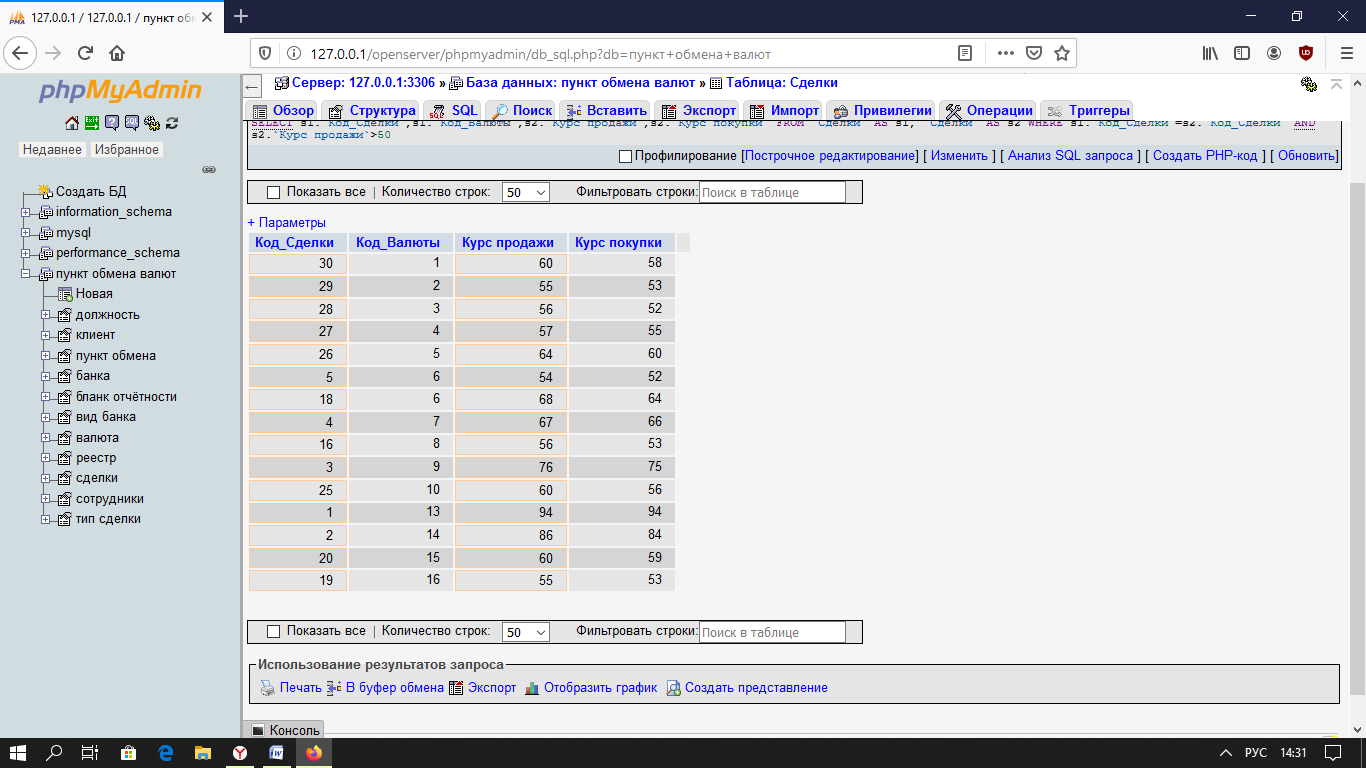
[SELECT](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) `Фамилия`,`Имя`,`Сумма продажи`,`Сумма покупки` FROM `Клиент` AS k [RIGHT](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/string-functions.html#function_right) OUTER JOIN `Сделки` AS s ON k.`Код\_Клиента`=s.`Код\_клиента`



**Запрос 5.** С использованием рекурсивного соединения.

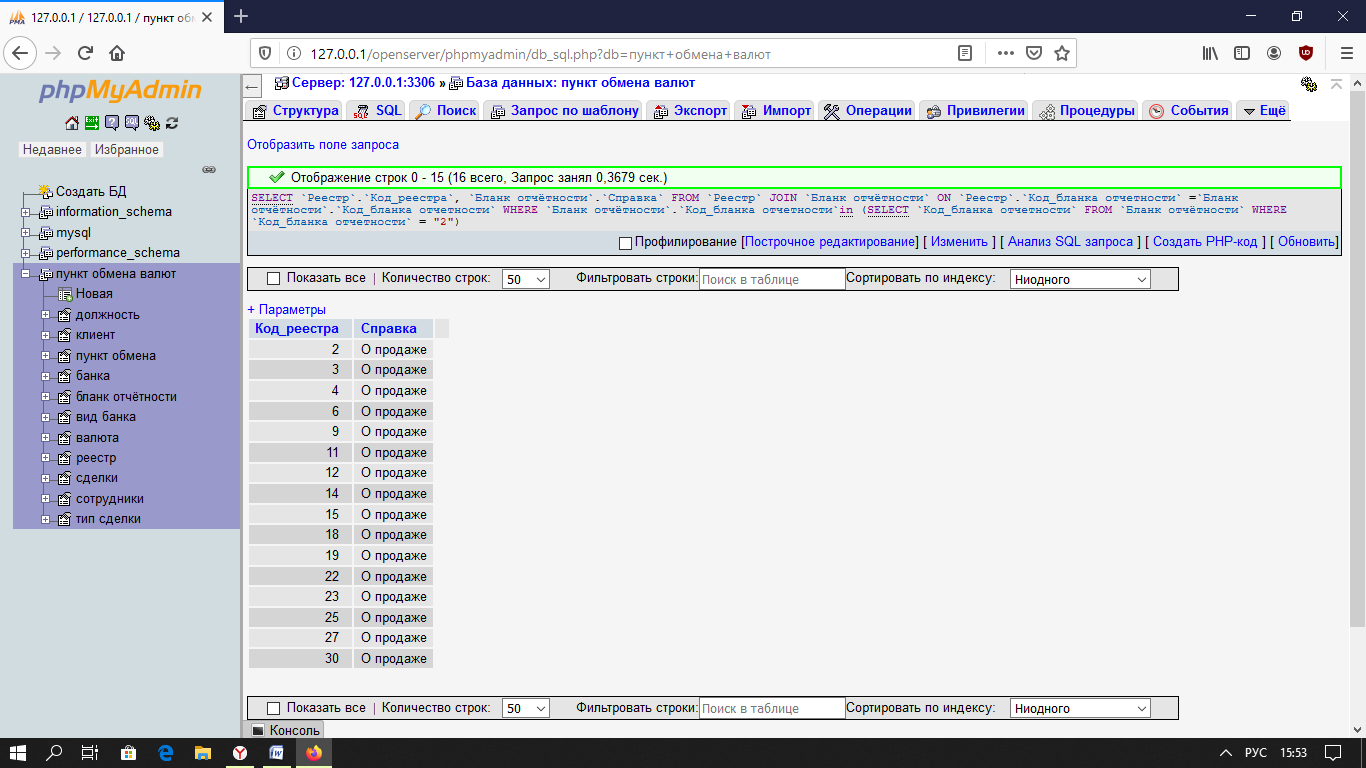
Выведем сделки, где курс продаж был более 50.

[SELECT](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) s1.`Код\_Сделки`,s1.`Код\_Валюты`,s2.`Курс продажи`,s2.`Курс покупки` FROM `Сделки` AS s1, `Сделки` AS s2 WHERE s1.`Код\_Сделки`=s2.`Код\_Сделки` [AND](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_and) s2.`Курс продажи`>50



**Запрос 6.** С использованием соединения по отношению

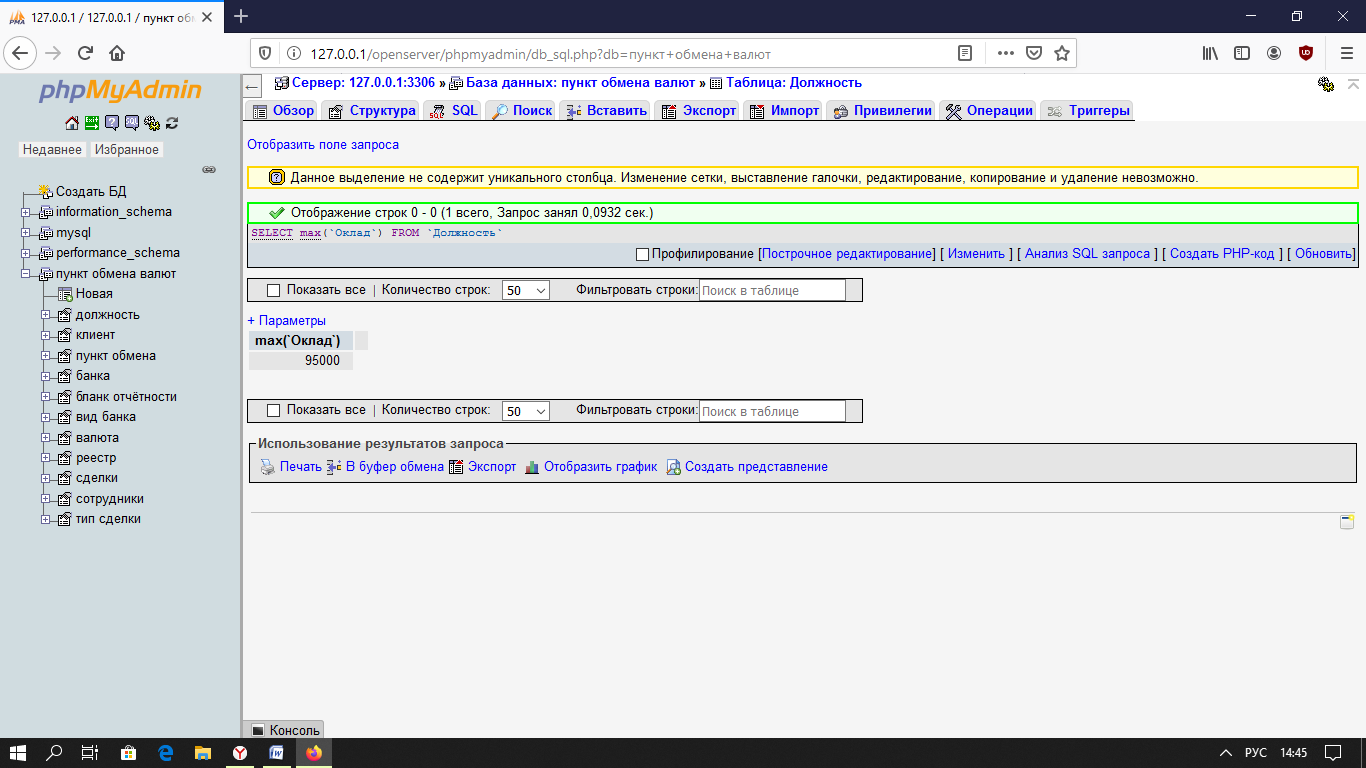
[SELECT](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) `Реестр`.`Код\_реестра`, `Бланк отчётности`.`Справка` FROM `Реестр` JOIN `Бланк отчётности` ON `Реестр`.`Код\_бланка отчетности` =`Бланк отчётности`.`Код\_бланка отчетности` WHERE `Бланк отчётности`.`Код\_бланка отчетности`[in](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/comparison-operators.html#function_in) ([SELECT](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) `Код\_бланка отчетности` FROM `Бланк отчётности` WHERE `Код\_бланка отчетности` = "2")



**Запрос 7.** С использованием функции агрегирования.

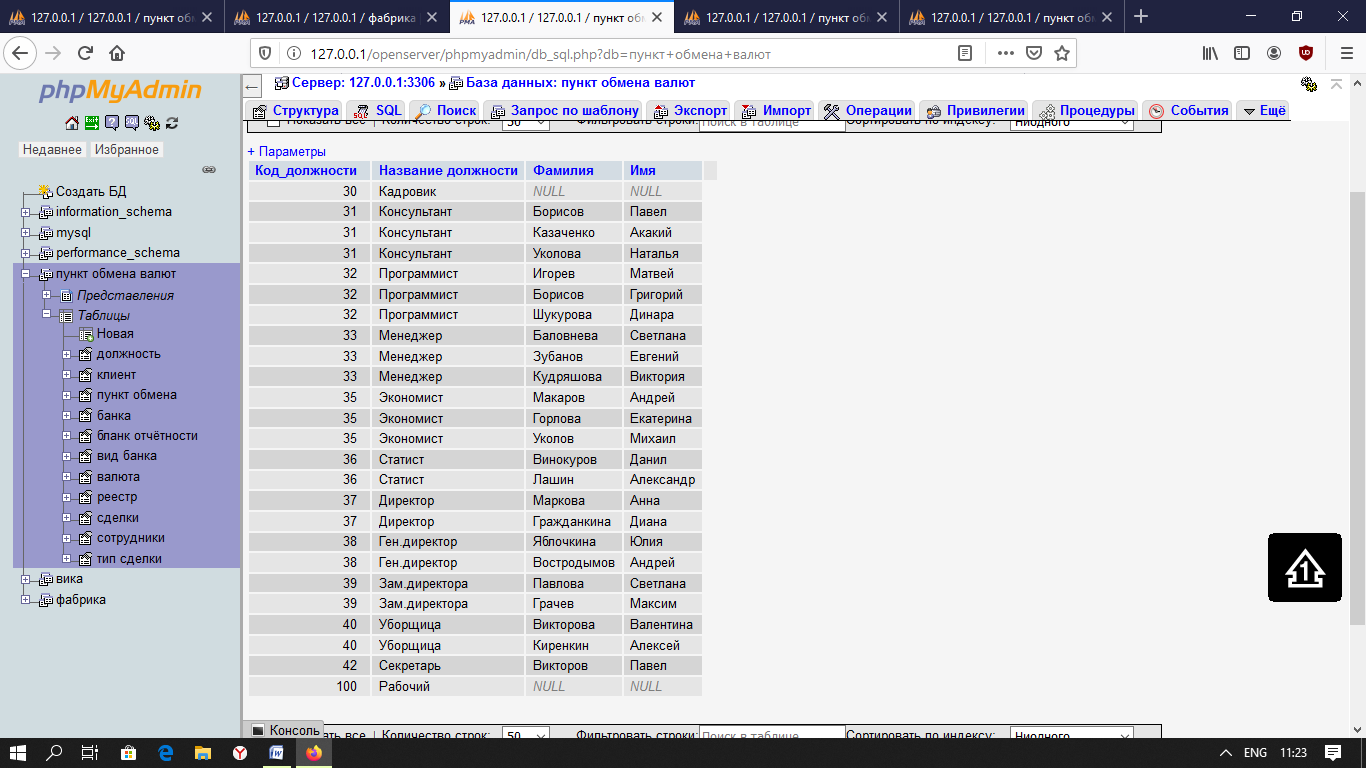
Узнаем максимальный оклад у должности.

[SELECT](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) [max](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html#function_max)(`Оклад`) FROM `Должность`



**Запрос 8.** Перекрестный запрос

SELECT`Должность`.`Код\_должности`, `Должность`.`Название должности`, `Сотрудники`.`Фамилия`,`Сотрудники`.`Имя` FROM `Должность` LEFT JOIN `сотрудники` ON `Сотрудники`.`Код\_должности` = `Должность`.`Код\_должности`



**Запрос 9.** С параметрами.

Выведем валюту, у которой Код = 1, 4, 10.

SET @val1=1;

SET @val2=4;

SET @val3=10;

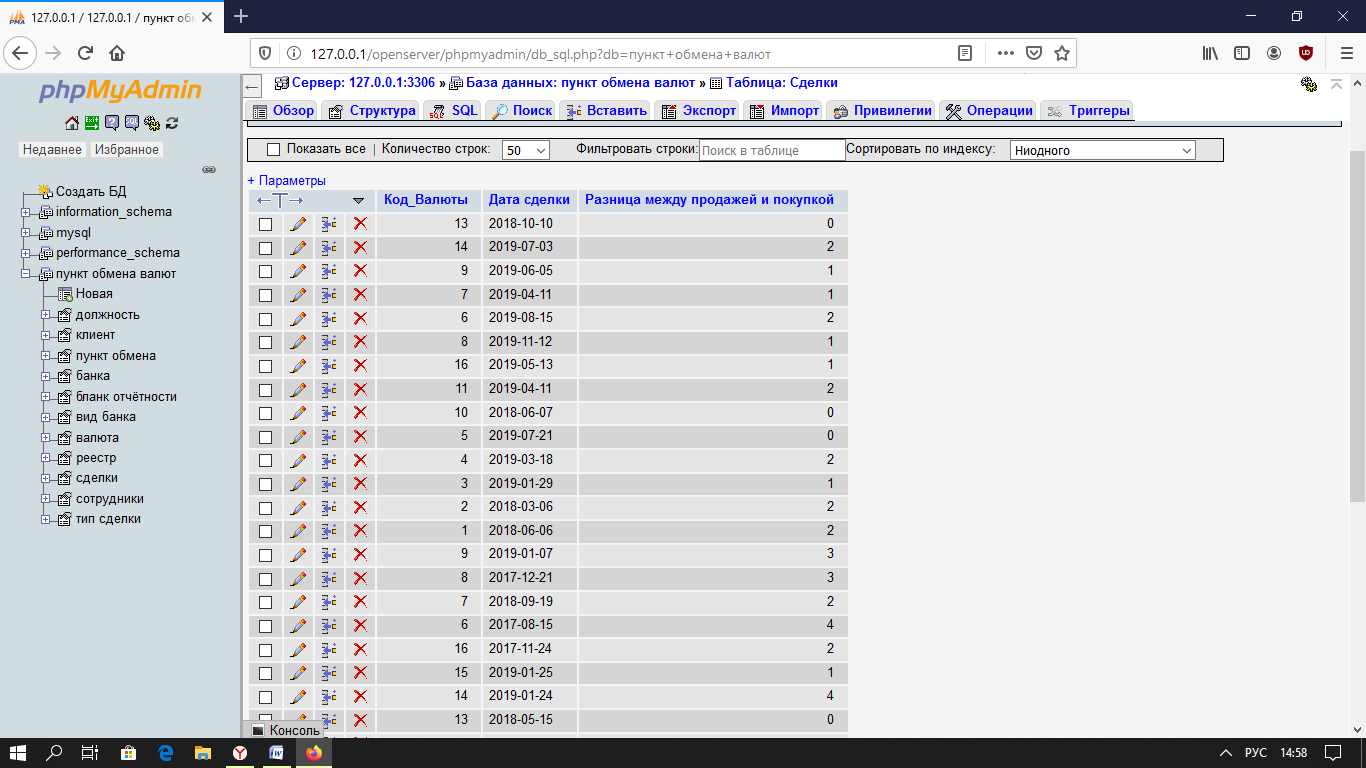
[SELECT](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) `Название` FROM `Валюта` WHERE `Код\_Валюты`=@val1 [OR](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_or) `Код\_Валюты`=@val2 [OR](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/logical-operators.html#operator_or) `Код\_Валюты`=@val3

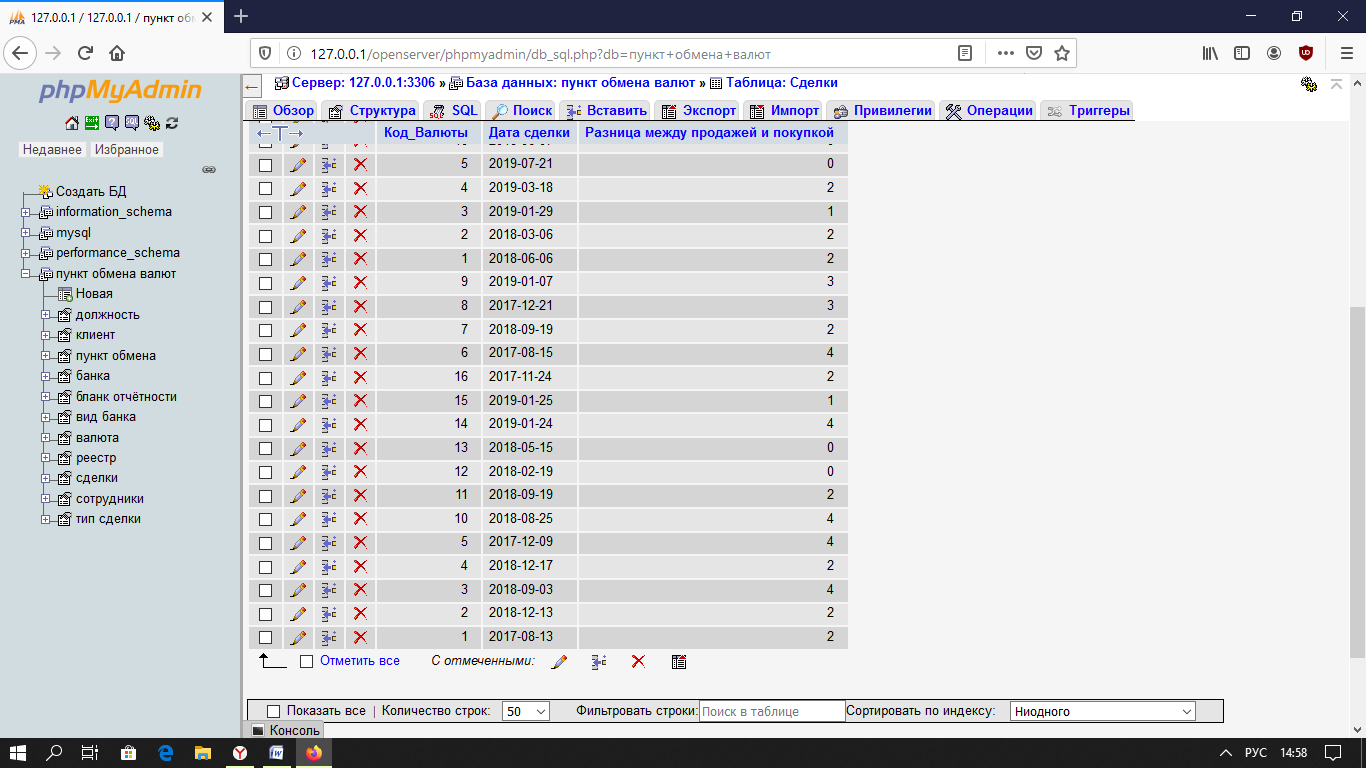


**Запрос 10.** С вычисляемым полем.

Выведем разницу между продажей и покупкой на опредеденную дату, для каждой валюты.

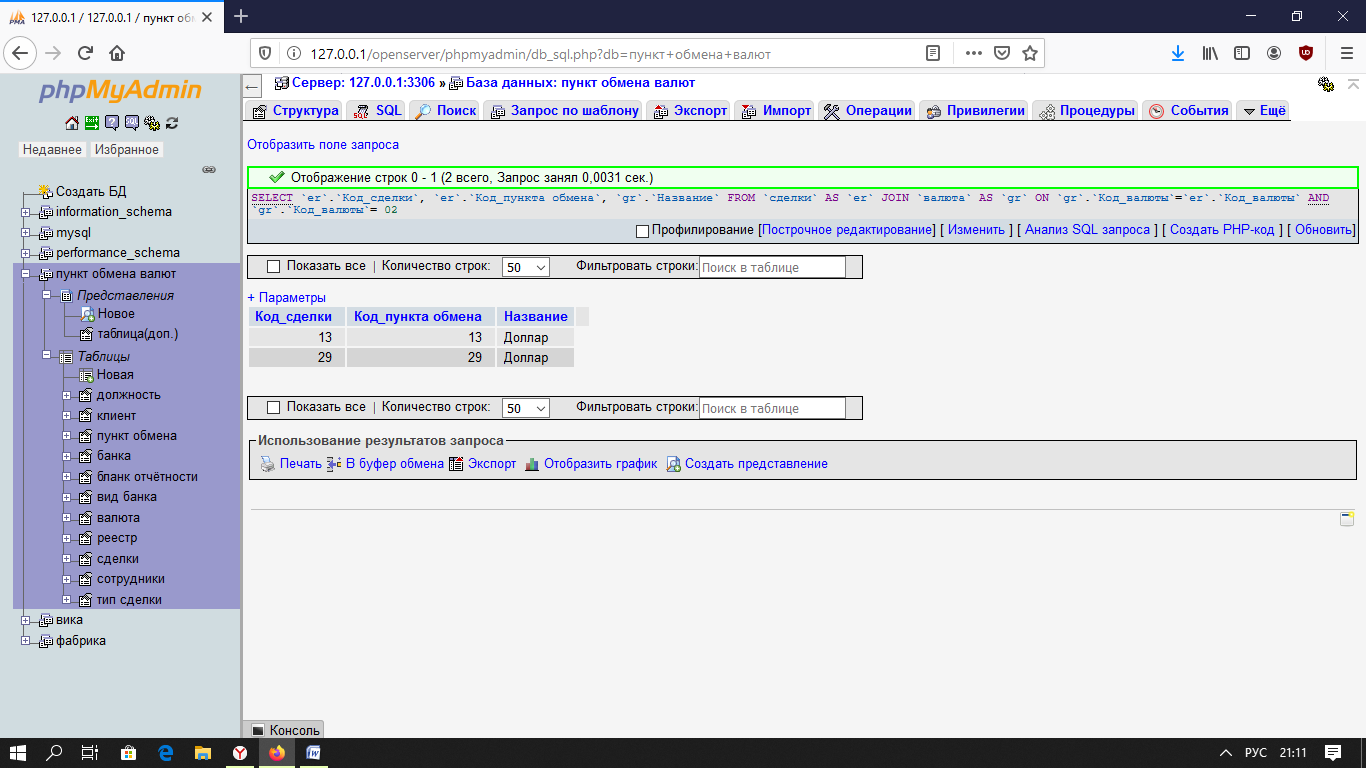
[SELECT](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) `Код\_Валюты`,`Дата сделки`,`Курс продажи`-`Курс покупки` AS `Разница между продажей и покупкой` FROM `Сделки`





**Запрос 11.** Многотабличный запрос:

SELECT `er`.`Код\_сделки`, `er`.`Код\_пункта обмена`, `gr`.`Название` FROM `сделки` AS `er`   
JOIN `валюта` AS `gr` ON `gr`.`Код\_валюты`=`er`.`Код\_валюты` AND `gr`.`Код\_валюты`= 02



**Запрос12.**Независимый и связанный подзапрос с множеством значений

Выведем на какую сумму покупки были совершены сделки с валютой «датская крона»:

SELECT `Код\_сделки`, `Код\_клиента`,`Код\_валюты`,`Код\_сотрудника`,`Курс покупки`,`Сумма покупки`

FROM `сделки`

WHERE EXISTS

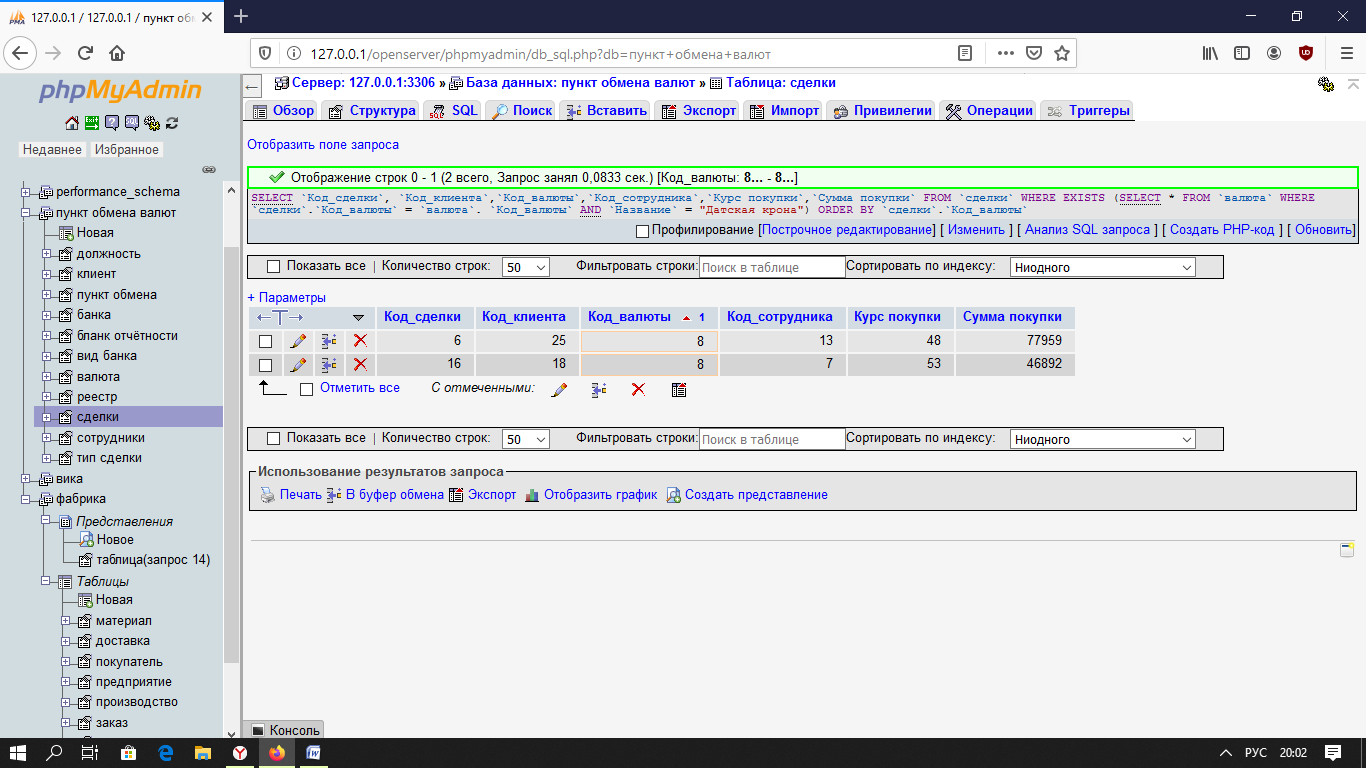
(SELECT \*

FROM `валюта`

WHERE `сделки`.`Код\_валюты` =

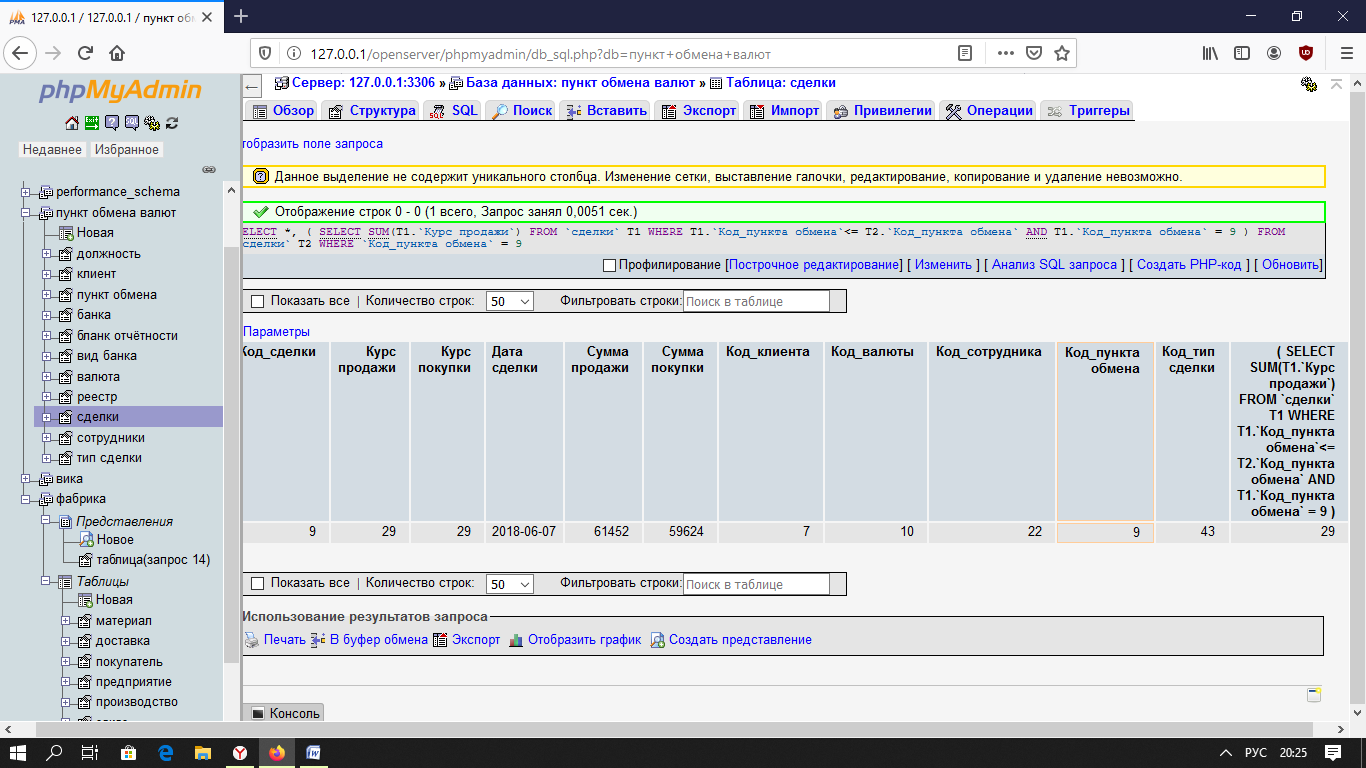
`валюта`. `Код\_валюты` AND `Название` = "Датская крона")

ORDER BY `сделки`.`Код\_валюты`



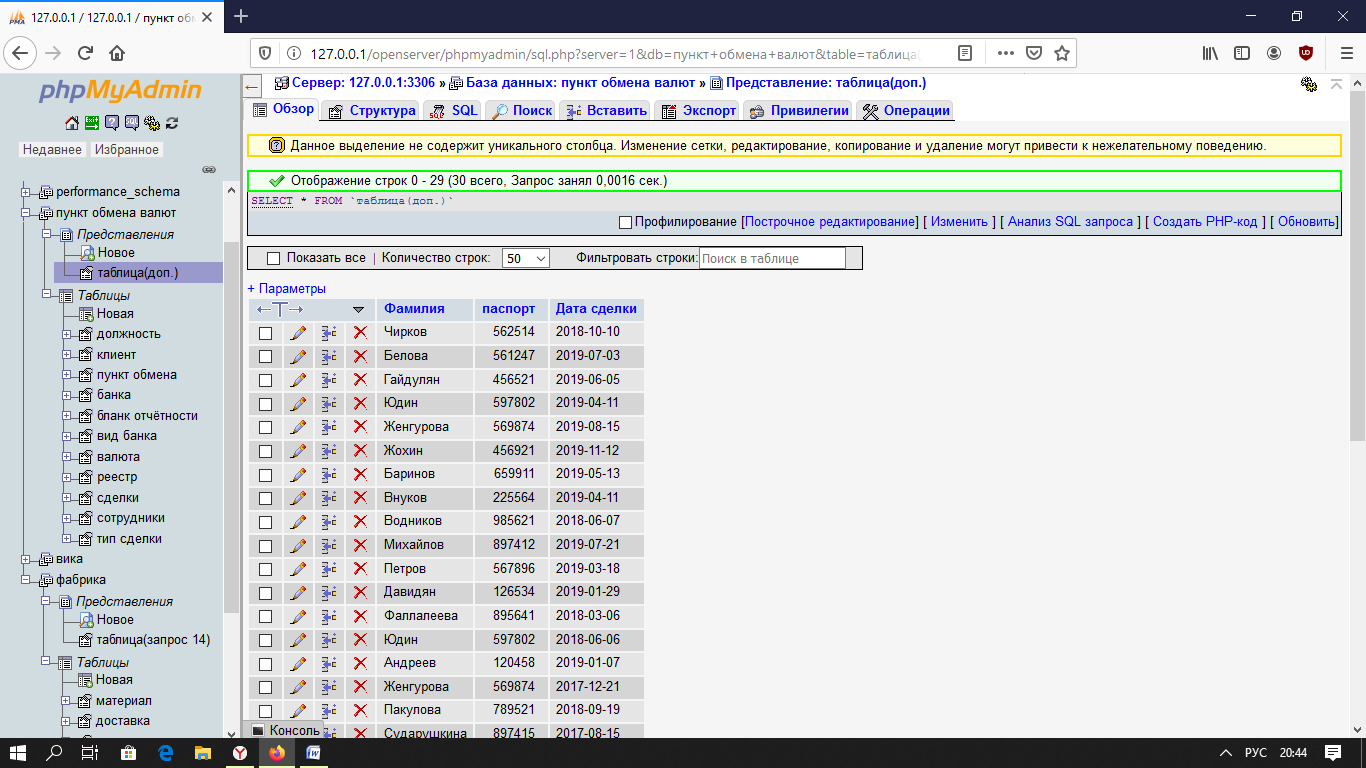
**Запрос 13.** Запрос для расчета с накоплением

SELECT \*, ( SELECT SUM(T1.`Курс продажи`) FROM `сделки` T1 WHERE T1.`Код\_пункта обмена`<= T2.`Код\_пункта обмена` AND T1.`Код\_пункта обмена` = 9 ) FROM `сделки` T2 WHERE `Код\_пункта обмена` = 9



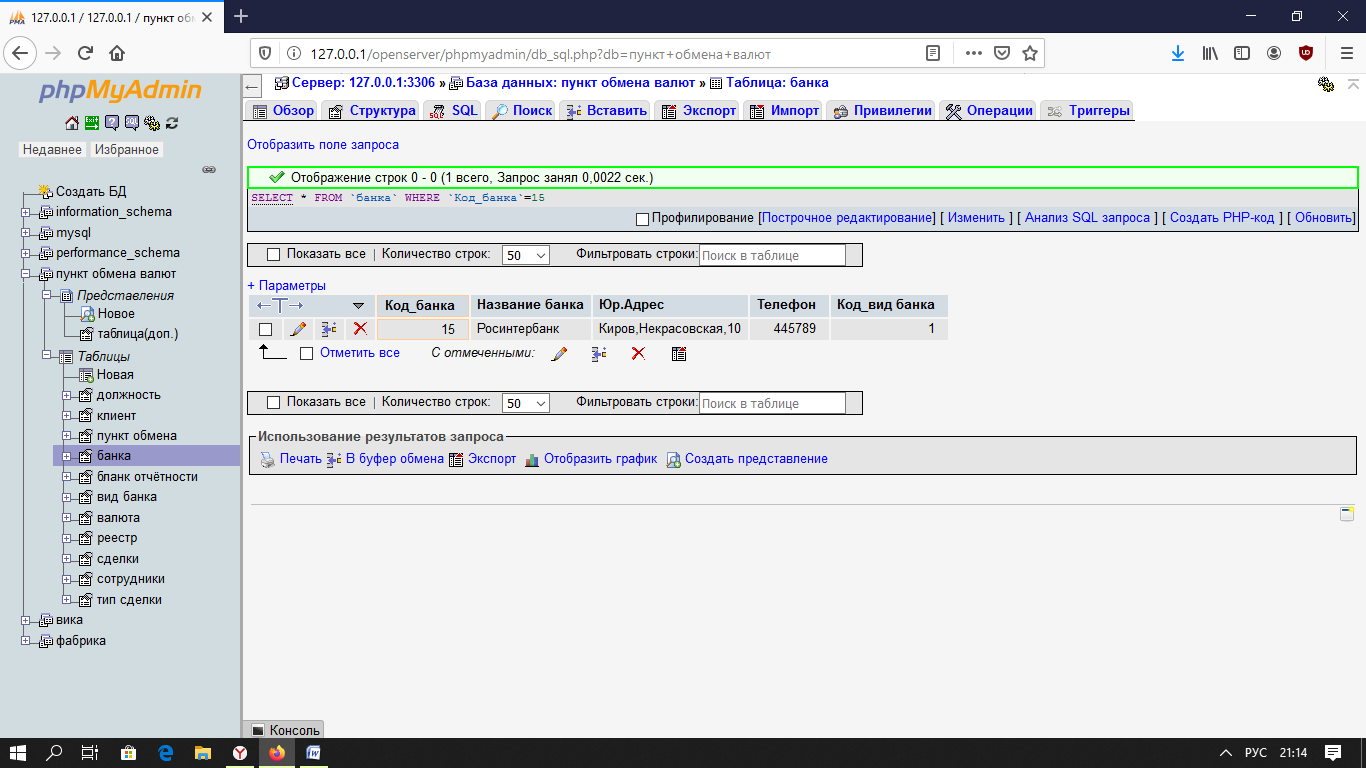
**Запрос 14.** Создание представления:

CREATE VIEW `Таблица(доп.)`AS   
SELECT `клиент`.`Фамилия`, `клиент`.`паспорт`, `сделки`.`Дата сделки`   
FROM`клиент`   
INNER JOIN `сделки` ON `сделки`.`Код\_клиента` = `клиент`.`Код\_клиента`

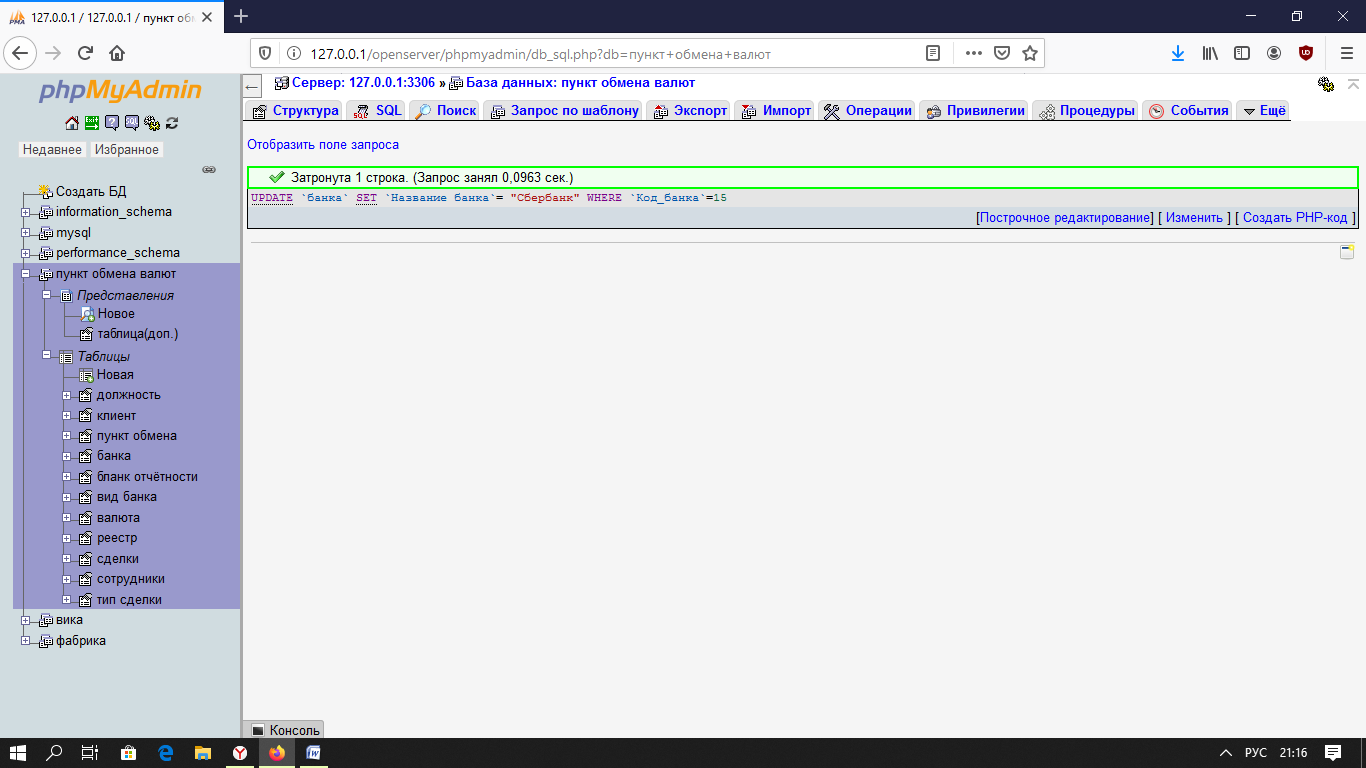


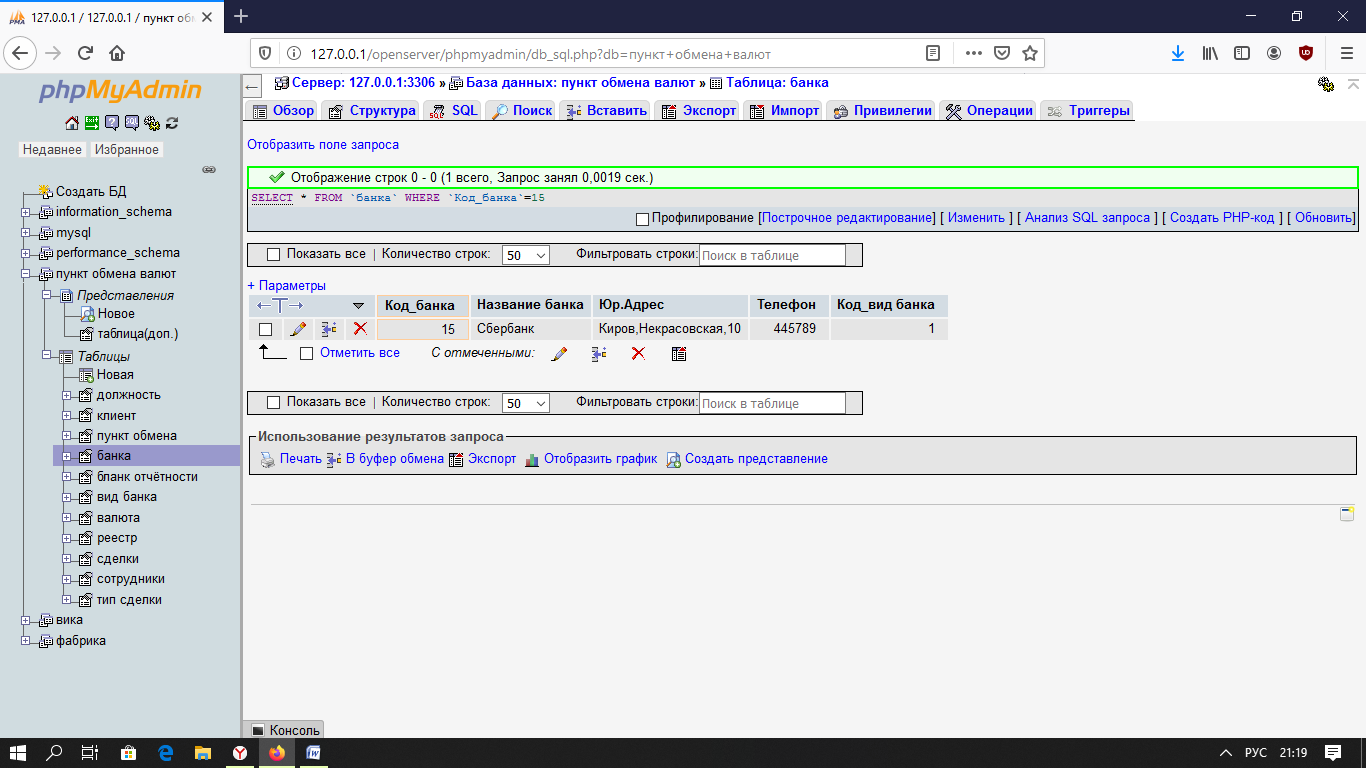
**Запрос 15.** Запрос на модификацию данных и слияние

SELECT \* FROM `банка` WHERE `Код\_банка`=15



UPDATE `банка` SET `Название банка`= "Сбербанк" WHERE `Код\_банка`=15

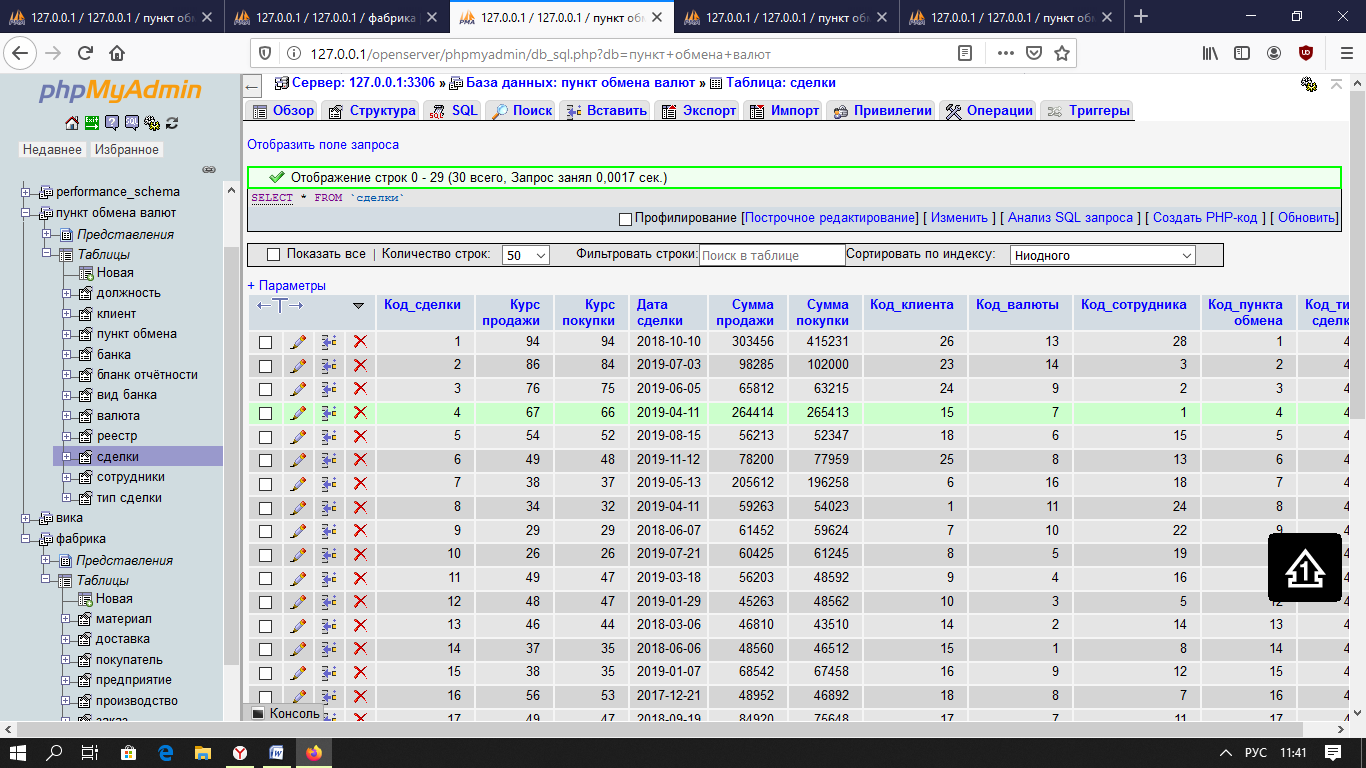




**Запрос 16.** Запрос с использованием транзакции

SELECT \*

FROM `сделки`

****

START TRANSACTION;

UPDATE `клиент` SET `Фамилия` = "Бондарева"

WHERE `Код\_клиента` = "11"



UPDATE `сделки` SET `Сумма покупки ` = `Сумма покупки ` + 100

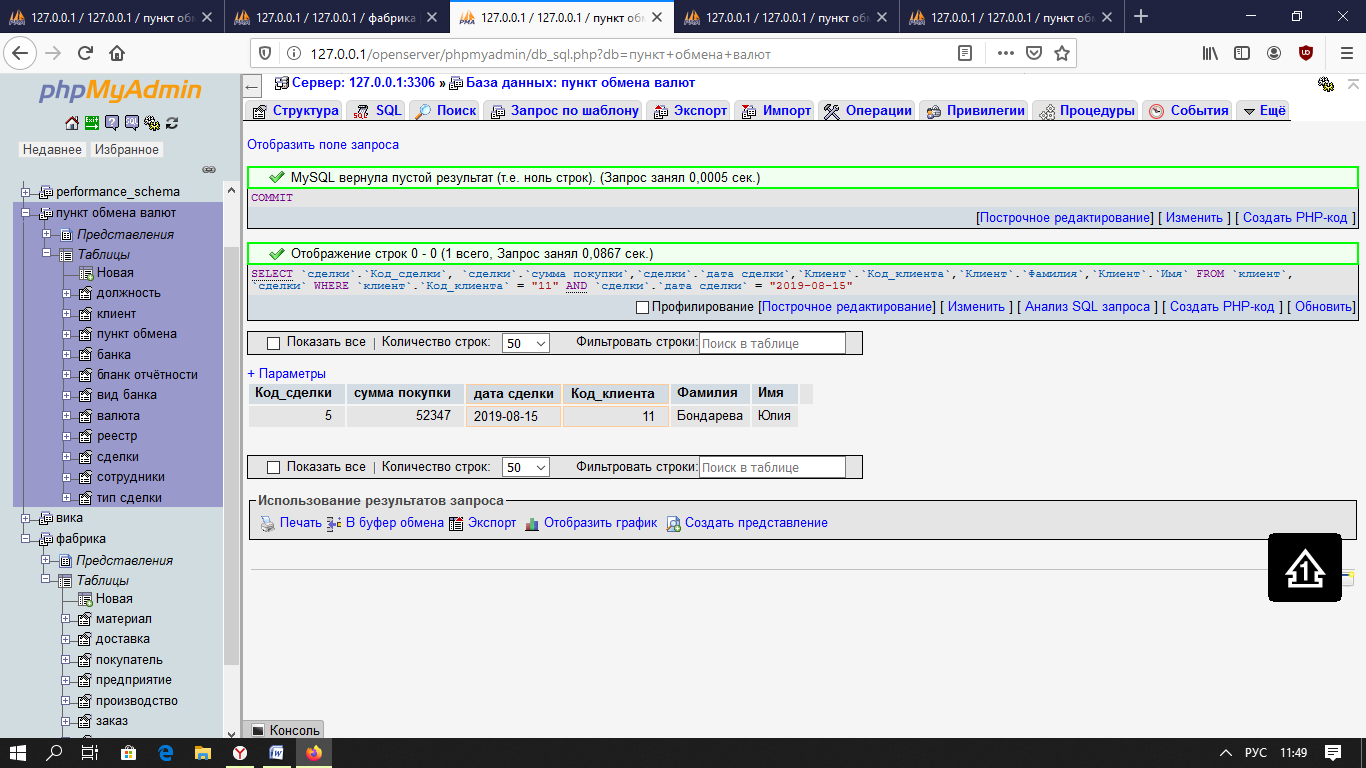
WHERE `дата сделки ` = "2019-08-15";

COMMIT;

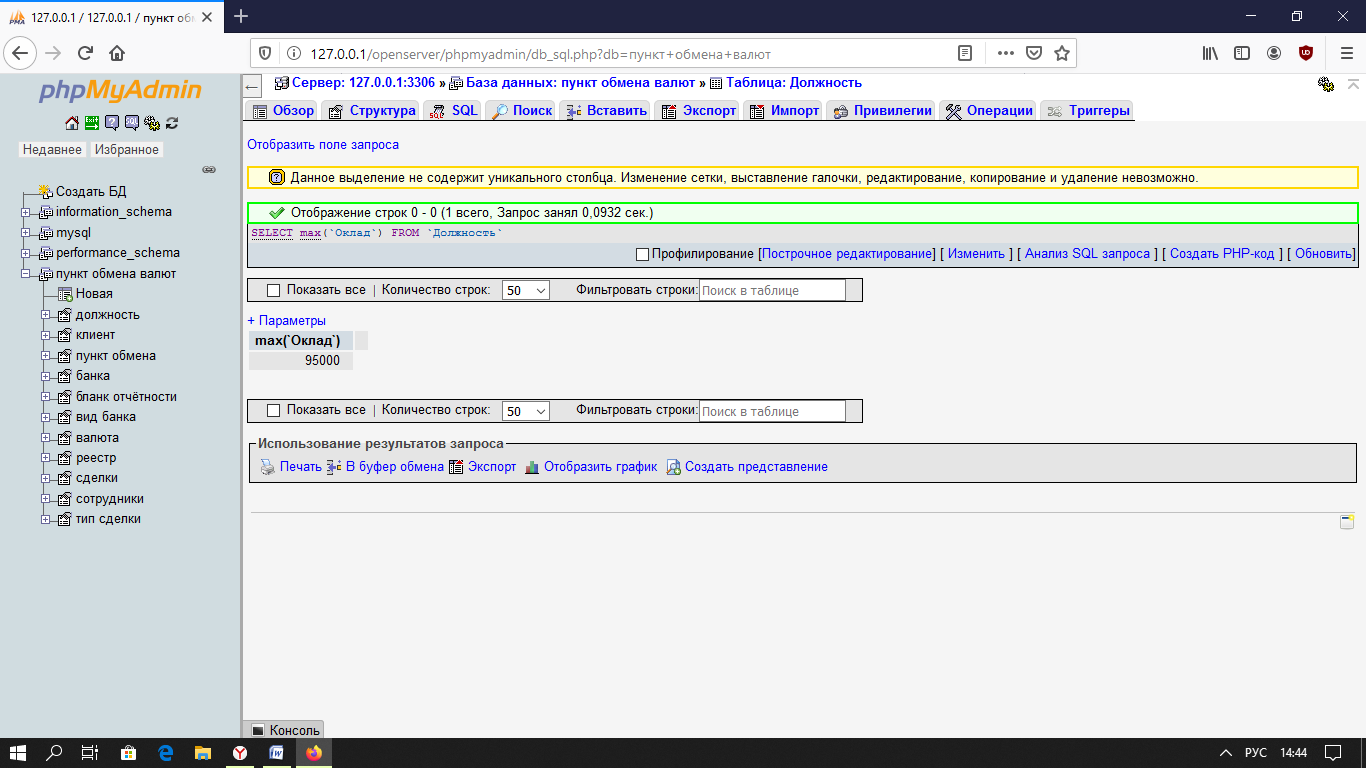
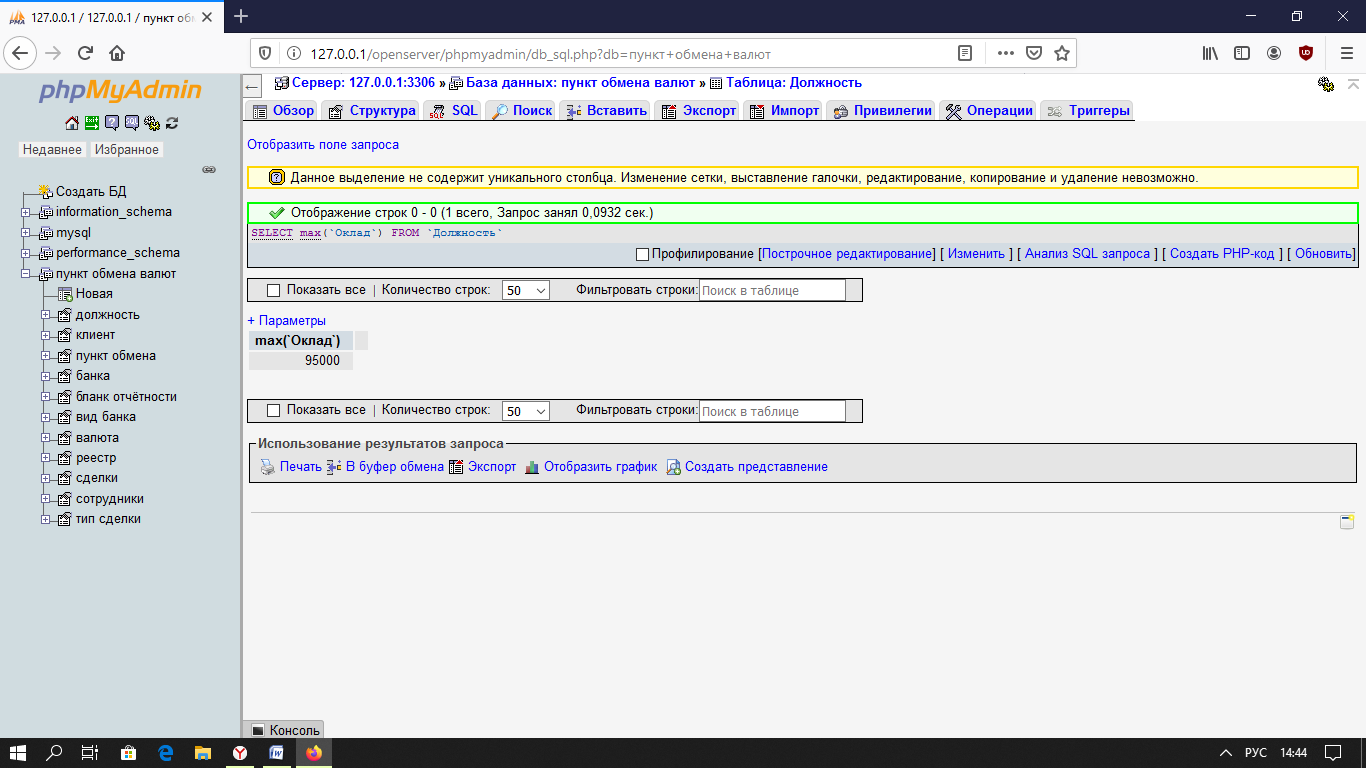
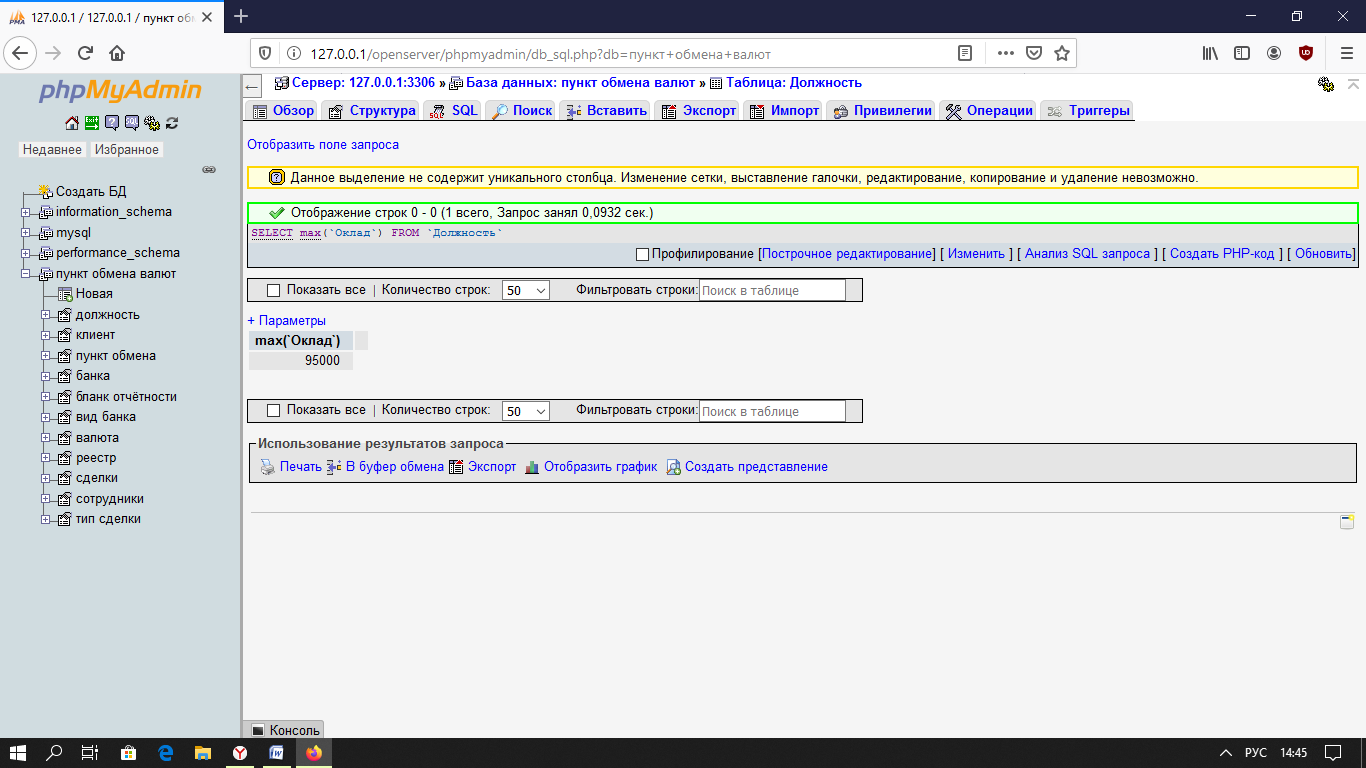
SELECT `сделки`.`Код\_сделки`, `сделки`.`сумма покупки`,`сделки`.`дата сделки`,`Клиент`.`Код\_клиента`,`Клиент`.`Фамилия`,`Клиент`.`Имя`

FROM `клиент`, `сделки`

WHERE `клиент`.`Код\_клиента` = "11" AND `сделки`.`дата сделки` = "2019-08-15"



**Запрос 17.** Создание триггера

Начало формы

# 6.Выводы.

* В данной работе была исследована предметная область «Пункт обмена валюты»;
* Построена инфологическая модель;
* Разработана даталогическая модель базы данных средствами MySQL, включающая 10 таблиц и соответствующие связи между ними;
* Разработаны 10 запросов для отображения данных из таблиц.

7.Список используемых источников.

1. Андон Ф. Язык запросов SQL. Учебный курс [Text] / Андон Ф. - СПб.: Питер, 2006. - 416 с. : ил. - 30 экз. - ISBN 5-469-00394-9 : Б. ц. УДК 004.6004.65(075)
2. Атре Ш. «Microsoft SQL Server 6.5». М.: Комплект документации, 2004 г. -200 с.
3. Голицина ОЛ, Максимов НВ, Попов ИИ. «Базы данных». М.: Диалектика, 2005 г. - 350 с.
4. Диго С. М. Базы данных: проектирование и использование [Text] : учебник для вузов / Диго С. М. - М. : Финансы и статистика, 2005. - 592 с. : ил. - 20 экз. - ISBN 5-279-02571-2
5. Дюбуа П. «Применение MySQL и Perl в Web\_приложениях». М.: Вильямс, 2004 г. - 400 с.
6. Конноли Т.К. «Базы данных проектирование реализация и сопровождение» Берг: Диалектика, 2005 г.-300 с.
7. Малыхина М.П. Базы данных: основы, проектирование, использование, 2-е изд. перераб. и доп. – СПб.: БХВ-Петербург, 2008. 528 с.
8. Нагао М., Катаяма Т., Уэмура С. «Реализация языка SQL в СУБД MySQL». М.: Мир, 2006 г.-350 с.
9. Хомоненко А. Д., Цыганков В. М., Мальцев М. Г. Базы данных: Учебник для высших учебных заведений /Под. ред. проф. А. Д. Хомоненко. 5-е ззд.,доп.- М:Бином-Пресс;СПб.: КОРОНА принт, 2009. - 736 с.
10. Лекции по дисциплине «Базы данных»