

**ОТЧЕТ**

по учебной практике

**УП.02.01. «Разработка и администрирование баз данных»**

по специальности 09.02.03 «Программирование в компьютерных системах»

Выполнил студент гр. П2-18

Михайлов Данила Алексеевич

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (подпись)

Проверил преподаватель

Тулупов А.В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(оценка)

Королев, 2021

**Оглавление**

[1. Проектирование базы данных 2](#_Toc87873620)

[1.1 Построение ER-модели базы данных 2](#_Toc87873621)

[2. Разработка приложения 2](#_Toc87873622)

[2.1 Разработка таблиц и связей 2](#_Toc87873623)

[2.2 Разработка хранимых процедур 4](#_Toc87873624)

[2.3 Разработка клиентского приложения 4](#_Toc87873625)

[Заключение 8](#_Toc87873626)

[Список литературы и интернет ресурсов 9](#_Toc87873627)

# Проектирование базы данных

## Построение ER-модели базы данных

В результате изучения предметной области и существующих форм данных была построена ER-модель базы данных для приложения.

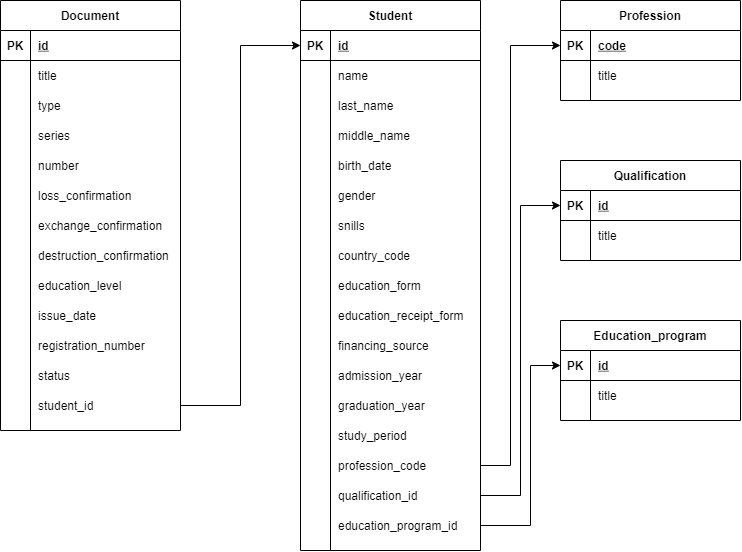


Рисунок 1. ER-модель базы данных

# Разработка приложения

## Разработка таблиц и связей

В программе IBExpert[1], предназначенная для разработки и администрирования баз данных Firebird[2], были созданы таблицы и связи, которые были спроектированы в ER-модели.

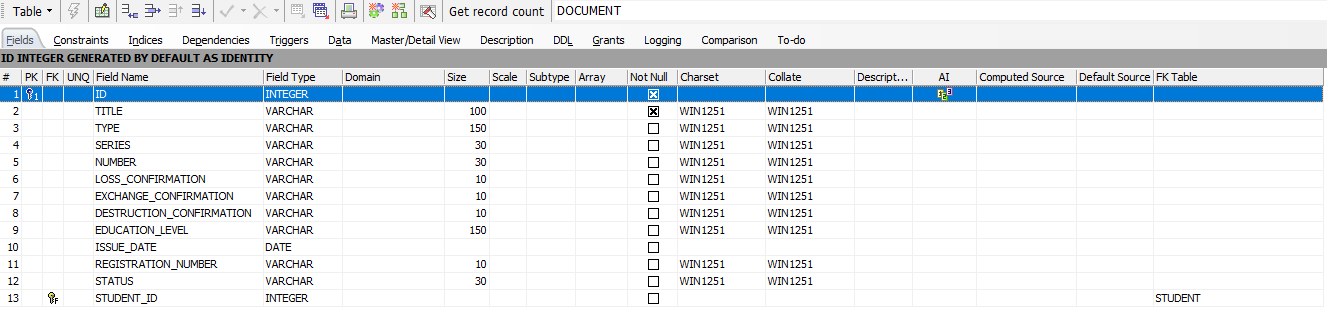


Рисунок 2. Таблица Document

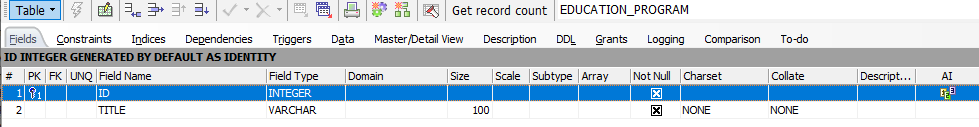


Рисунок 3. Таблица Education\_program

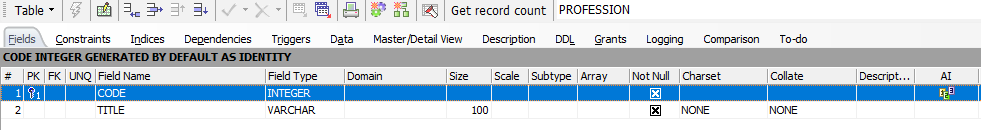


Рисунок 4. Таблица Profession

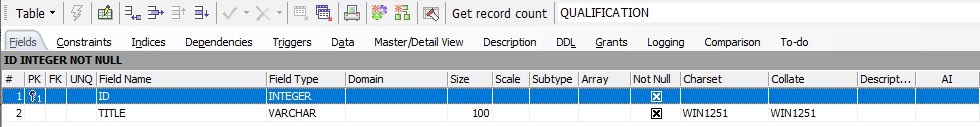


Рисунок 5. Таблица Qualification

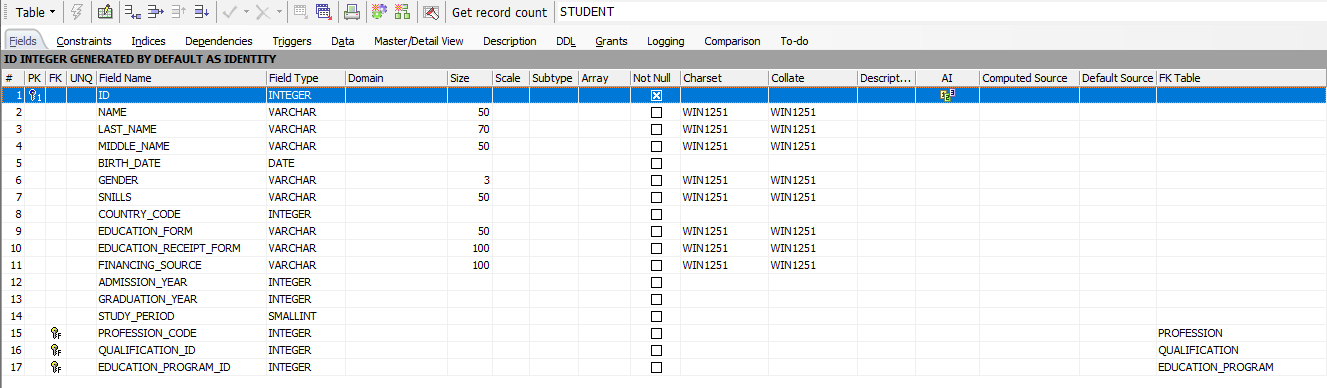


Рисунок 6. Таблица Student

## Разработка хранимых процедур

После создания таблиц была создана хранимая процедура для добавления профессии в таблицу Profession.

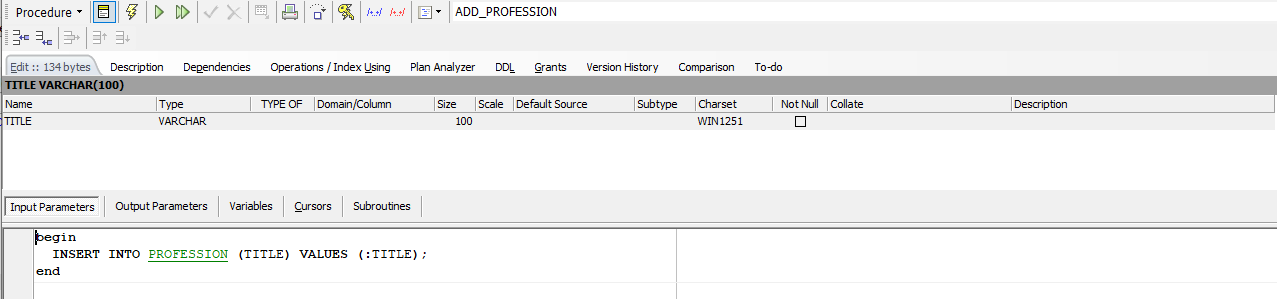


Рисунок 7. Хранимая процедура Add\_profession

## Разработка клиентского приложения

Клиентское приложение было разработано на языке программирования Python[3]. За БД отвечает класс SQL в файле sql.py.

Для того чтобы подключиться к БД, надо лишь вызвать класс SQL и передать ему как аргумент имя БД, к которой надо подключаться.

Класс SQL выполняет следующие функции:

1. Подключение к БД
2. Подключение к таблице
3. Получение ID записи
4. Добавление записи в таблицу
5. Удаление записи из таблицы
6. Обновление записи в таблице
7. Получение ID последней добавленной записи

Листинг 1. Функция подключения к базе данных.

def connect\_table(self, table):

if table is None:

table = self.table

with self.connection:

is\_valid = self.is\_valid(table)

if is\_valid:

self.table = table

return is\_valid

Листинг 2. Функция обновления базы данных.

def update(self, table: str, pk: dict, data: dict):

"""

Обновит выбранные данные (data) из выбранной таблицы (table).

Выборка происходит благодаря id таблицы (pk)

:param: все словари имеют следующий вид => {'имя\_колоны': значение\_колоны}

:return:

"""

try:

with self.connection:

column\_values = list()

command = f"UPDATE `{table}`"

key\_pk = list(pk.keys())[0]

command += f" SET `{key\_pk}` WHERE "

column\_values.append(pk[key\_pk])

count = len(data)

for key, value in data.items():

column\_values.append(value)

command += f"`{key}` = ?"

count -= 1

if count:

command += ", "

return self.cursor.execute(command, tuple(column\_values)).fetchall()

except sqlite3.IntegrityError:

return -1

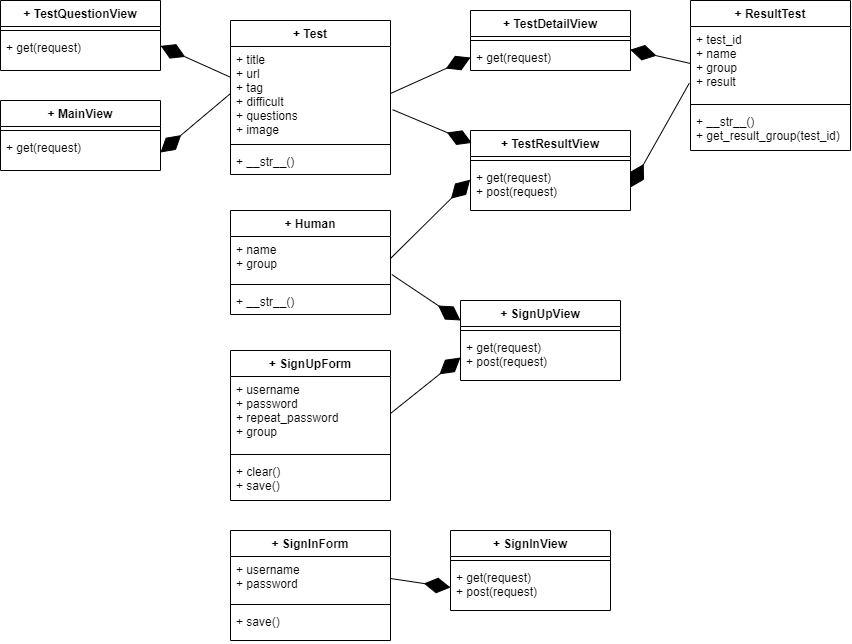


Рисунок 8. Диаграмма классов приложения

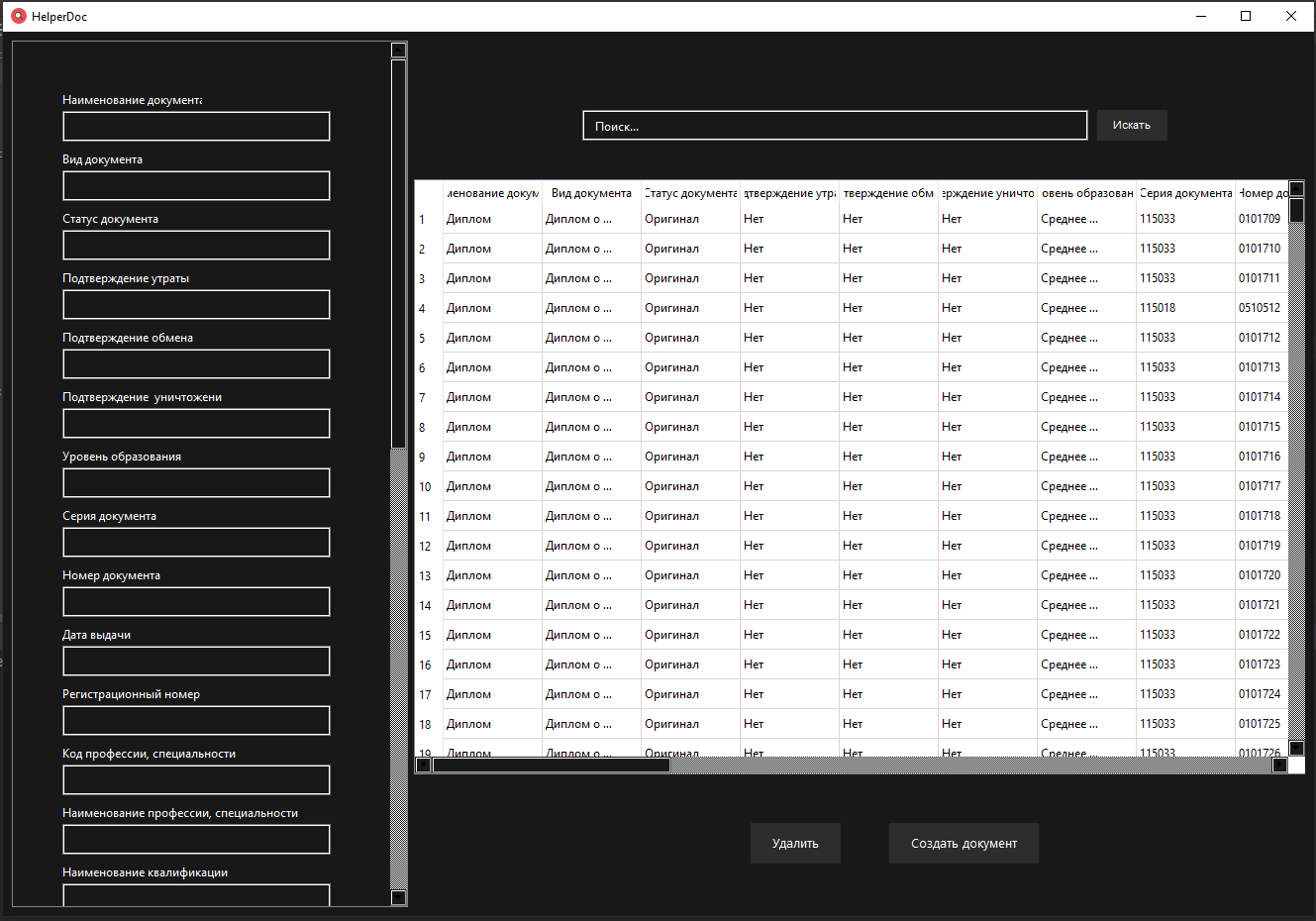


Рисунок 9. Клиентское приложение

# Заключение

В результате учебной практики была спроектирована и создана база данных. Также было создано клиентское приложение для использования данной базы данных.

# Список литературы и интернет ресурсов

1. IBExpert

<https://www.ibexpert.net>

1. Firebird

<https://firebirdsql.org>

1. Python

<https://www.python.org/>