МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное образовательное учреждение высшего образования

«ТЮМЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ НАУК

КАФЕДРА ИНФОРМАЦИОННЫХ СИСТЕМ

ОТЧЕТ ОБ ИНДИВИДУАЛЬНОЙ РАБОТЕ

ДИСЦИПЛИНЕ «БАЗЫ ДАННЫХ»

Тема

«Автоматизация учета документооборота и исполнения поручений»

Выполнил:

студент группы ИСиТ-189-1.

Бугаков Д.Ю.

Руководитель:

канд. физ.-мат. наук,

доцент Моор П. К.

Тюмень – 2019

ОГЛАВЛЕНИЕ

[ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 3](#_Toc39090760)

[ФУНКЦИИ ПРИЛОЖЕНИЯ 4](#_Toc39090761)

[ПРИВЕДЕНИЕ К НОРМАЛЬНЫМ ФОРМАМ 4](#_Toc39090762)

[ПЛОСКАЯ ТАБЛИЦА 5](#_Toc39090763)

[ПЕРВАЯ НОРМАЛЬНАЯ ФОРМА 5](#_Toc39090764)

[ВТОРАЯ НОРМАЛЬНАЯ ФОРМА 7](#_Toc39090765)

[ТРЕТЬЯ НОРМАЛЬНАЯ ФОРМА 7](#_Toc39090766)

[ER-МОДЕЛИРОВАНИЕ 10](#_Toc39090767)

[РАЗРАБОТАННЫЕ ФОРМЫ 13](#_Toc39090768)

[ФОРМА С ПОДЧИНЕННЫМИ ТАБЛИЦАМИ 16](#_Toc39090769)

[ФОРМА С ИТОГАМИ 18](#_Toc39090770)

[ФОРМА С ПРОЦЕДУРАМИ 19](#_Toc39090771)

ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

Программа “Учета документооборота и исполнения поручений” предназначена для учета документов. Для организации учета документов и поручений на каждый документ заводится карточка.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Код документа | Назва ние документа | Код автора документа | Автор документа | Код типа | Тип документа | Дата документа |
| 1 | Согласие на обработку персональных данных | 1 | Иванов | 2 | Приказ | 21.09.2019 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № | Поручение | Код исполнителя | Исполнитель | Код подразделения. | Название подр. | Срок исполнения  план | Дата исполнения |
| 1 | Подготовить  Данные | 2 | Сидоров | 2 | Отдел управления | 22.09.2019 | 23.09.2019 |
| 2 | Собрать подписи | 3 | Петров | 2 | Отдел управления | 25.09.2019 | 25.09.2019 |

Документ, к которому прилагаются поручения содержит в себе: код документа, название, код автора, ФИО автора документа, код типа документа, тип документа и дату документа.

Автор документа может быть, как автором, так и исполнителем поручений. Автор документа может быть автором нескольких документов. У документа может быть только один автор.

Тип документа определяется кодом. Документ может иметь лишь один тип, но один и тот же тип может быть у нескольких документов.

В карточку учета заносятся строки поручений по этому документу.

На один документ может быть выдано несколько поручений. Строка поручений относится только к одному документу

Каждое поручение содержит в себе: последовательный номер поручения, описание поручения, код исполнителя, ФИО исполнителя, код подразделения и его название, срок исполнения, фактическую дату исполнения.

Поручение отдается одному исполнителю. На одного исполнителя может быть отдано несколько поручений в одном или в других документах. Исполнитель может являться исполнителем и в других документах и исполнениях.

В строке может быть указано одно поручение

Номер поручения в документе всегда уникален.

ФУНКЦИИ ПРИЛОЖЕНИЯ

Программа “Учет документооборота и исполнения поручений” выполняет следующие функции:

1. предоставляет возможность хранить реальный документ в базе данных;
2. предоставляет возможность вводить, изменять и удалять поручения для документов;
3. делает выводы исходя из некоторой результирующей информации о документах;
4. классифицирует документы.

Приведение к нормальным формам

ПЛОСКАЯ ТАБЛИЦА

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| К\_док | Назв\_Д | К\_авт | авт | К\_тип | тип | Д | № | Поруч | К\_исп | исп | К\_под | под | ДП | ДИ |
| 1 | Согласие на обработку персональных данных | 1 | Иванов | 2 | Уведомление | 21.09.2019 | 1 | Подготовить  Данные | 2 | Сидоров | 2 | Отдел управления | 22.09.2019 | 23.09.2019 |
| 2 | Собрать подписи | 3 | Петров | 2 | Отдел управления | 25.09.2019 | 25.09.2019 |

Каждая повторяющаяся группа образует новую группу выделяется кортеж

ПЕРВАЯ НОРМАЛЬНАЯ ФОРМА

Отношение находится в 1НФ, если все его атрибуты являются простыми. Не должно быть повторений строк в таблице.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| К\_док  KD | Назв\_Д  ND | К\_авт  KA | Авт  A | К\_тип  KT | Тип  T | Д  D | №  N | Поруч  PO | К\_исп  KI | Исп  I | К\_под  KP | Под  P | ДП  DP | ДИ  DI |
| 1 | Согласие на обработку персональных данных | 1 | Иванов | 2 | Уведомление | 21.09.2019 | 1 | Подготовить  Данные | 2 | Сидоров | 2 | Отдел управления | 22.09.2019 | 23.09.2019 |
| 1 | Согласие на обработку персональных данных | 1 | Иванов | 2 | Уведомление | 21.09.2019 | 2 | Собрать подписи | 3 | Петров | 2 | Отдел управления | 22.09.2019 | 25.09.2019 |
| 2 | Приказ об утверждении структуры IT-отдела | 2 | Петров | 2 | Приказ | 28.09.2019 | 1 | Разработать схему отдела | 4 | Алексеев | 3 | Отдел разработки | 5.10.2019 | 7.10.2019 |
| 2 | Приказ об утверждении структуры IT-отдела | 2 | Петров | 2 | Приказ | 28.09.2019 | 2 | Установить иерархию | 5 | Павлов | 3 | Отдел разработки | 5.10.2019 | 8.10.2019 |

Потенциальные ключи: (KD, N);

Первичный ключ: KD, N

ВТОРАЯ НОРМАЛЬНАЯ ФОРМА

Для начала необходимо выписать общее отношение, из которого будет в дальнейшем выявляться зависимости:

All (**KD**, ND, KA, A, KT, T, D, **N**, PO, KI, I, KP, P, DP, DI)

Затем для нахождения второй нормальной формы мы должны выявить частичные зависимости:

KD->ND, KA, A, KT, T, D

В данном случае получается частичная зависимость названия документа, кода автора, автора, кода типа, типа и даты от кода документа.

Выписываем отношение после выявления зависимости, в котором первичным ключом будет являться код документа.

Dock (**KD**, ND, KA, A, KT, T, D)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| К\_док  **KD** | Назв\_д  ND | К\_авт  KA | Авт  A | К\_тип  KT | Тип  T | Дата  D |
| 1 | Согласие на обработку персональных данных | 1 | Иванов | 2 | Уведомление | 21.09.2019 |

Выписываем изначальное отношение убрав из него зависимые атрибуты.

O (**KD, N**, PO, KI, I, KP, P, DP, DI)

В итоге получим:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| К\_док  **KD** | №  **N** | Поруч  PO | К\_исп  KI | Исп  I | К\_под  KP | Под  P | ДП  DP | ДИ  DI |
| 1 | 1 | Подготовить  Данные | 2 | Сидоров | 2 | Отдел управления | 22.09.2019 | 23.09.2019 |
| 1 | 2 | Собрать подписи | 3 | Петров | 2 | Отдел управления | 22.09.2019 | 25.09.2019 |

ТРЕТЬЯ НОРМАЛЬНАЯ ФОРМА

Для нахождения третьей нормальной формы мы должны выявить транзитивные зависимости.

Выявление транзитивных зависимостей в отношении Dock:

Dock (**KD**, ND, KA, A, KT, T, D)

В данном отношении будет транзитивная зависимость, а именно код автора будет зависим от кода документа, а в свою очередь автор зависит от кода автора:

KD->KA KA->A

Запишем в отдельное отношение:

O2(**KA**, A) \* Автор

|  |  |
| --- | --- |
| К\_авт  **KA** | Авт  A |
| 1 | Иванов |

Следующей транзитивной зависимостью будет являться зависимость кода типа документа от кода документа, где от кода типа документа зависит тип документа.

KD->KT KT->T

Выпишем в отдельное отношение:

O3(**KT**, T) \* Тип документа

|  |  |
| --- | --- |
| К\_тип  **KT** | Тип  T |
| 2 | Уведомление |

Таким образом после выявления всех транзитивных зависимостей и их исключения получится следующее отношение:

Dock (**KD**, ND, KA, KT, D) \* Документ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| К\_док  KD | Назв\_Д  ND | К\_авт  KA | К\_тип  KT | Д  D |
| 1 | Согласие на обработку персональных данных | 1 | 2 | 21.09.2019 |

Затем необходимо выделить транзитивные зависимости в общем отношении:

O (**KD, N**, PO, KI, I, KP, P, DP, DI)

В данном случае транзитивной зависимостью будет являться зависимость кода исполнителя от кода документа и номера документа, а свою очередь зависимость фамилии исполнителя, кода подразделения в котором он работает и название подразделения от кода исполнителя.

KD, N -> KIKI->I, KP, P

Выписываем отношение:

I (**KI**, I, KP, P)

В получившемся отношении можно выявить еще одну транзитивную зависимость, а именно зависимость кода подразделения от кода исполнителя, где от кода подразделения зависит название подразделения:

KI->KP KP->P

Выписываем в отдельное отношение:

O6(**KP**, P) \* Подразделение

|  |  |
| --- | --- |
| К\_под  **KP** | Под  P |
| 2 | Отдел управления |

Далее выписываем итоговое отношение по исполнителям

I (**KI**, I, KP) \* Исполнитель

|  |  |  |
| --- | --- | --- |
| К\_исп  **KI** | Исп  I | К\_под  KP |
| 2 | Сидоров | 2 |
| 3 | Петров | 2 |

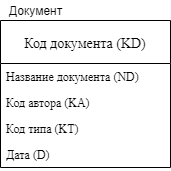
Итоговым отношением является:

Gen (**KD**, **N**, PO, KI, DP, DI) \* Поручение

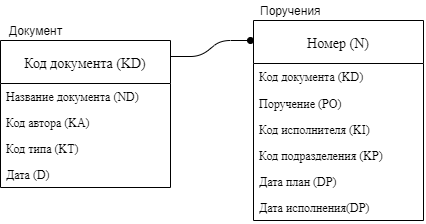
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| К\_док  KD | №  N | Поруч  PO | К\_исп  KI | ДП  DP | ДИ  DI |
| 1 | 1 | Подготовить  Данные | 2 | 22.09.2019 | 23.09.2019 |
| 2 | 1 | Разработать схему отдела | 4 | 5.10.2019 | 7.10.2019 |

ER-МОДЕЛИРОВАНИЕ

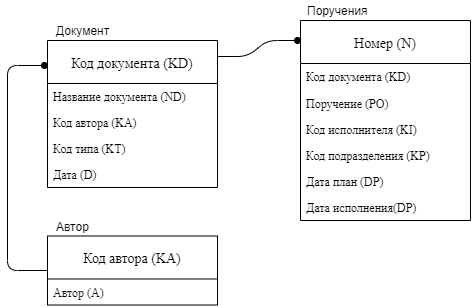
Для построения ER- модели мы должны выделить все сущности и определить связь между ними. Первая сущность- **документ**.



Документ содержит множество поручений, а следовательно, вторая сущность — это поручения. Также множество поручений может быть у одного документа. Документ и поручения имеют связь один ко многим (Один документ-много поручений).

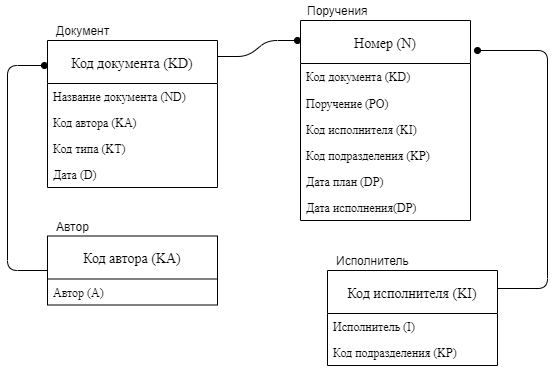


Каждый документ имеет лишь одного автора, но один и тот же автор может быть автором многих документов (Один ко многим).

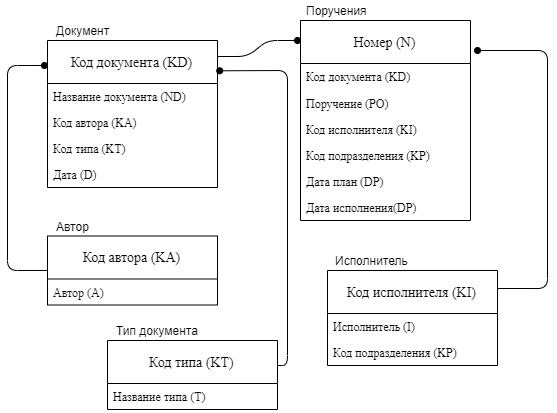


Поручение отдается лишь одному исполнителю. Исполнитель может иметь несколько поручений.

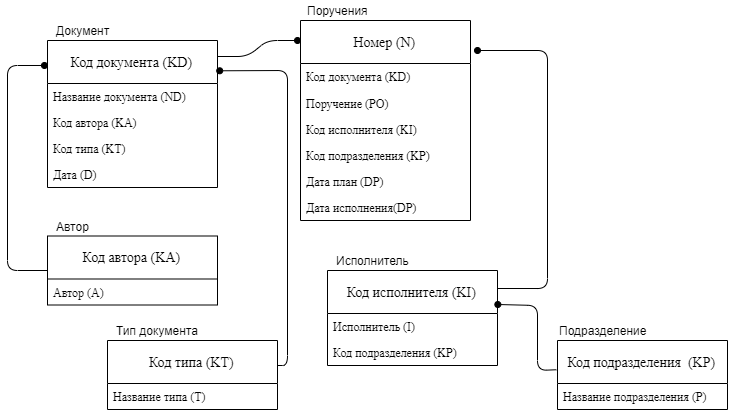
,



Документ может иметь лишь один тип, но один и тот же тип документа может быть у многих документов.



Каждый исполнитель принадлежит к одному подразделению, но каждое подразделение может содержать множество исполнителей.



РАЗРАБОТАННЫЕ ФОРМЫ

Было разработано 4 формы для работы с таблицами базы данных “Учет документооборота”. Первые три формы “Авторы”, “Тип документа”, “Подразделения” (Рис.1,2,3) позволяют добавлять, изменять, удалять выбранные данные в таблицах из баз данных, с которыми они работают.

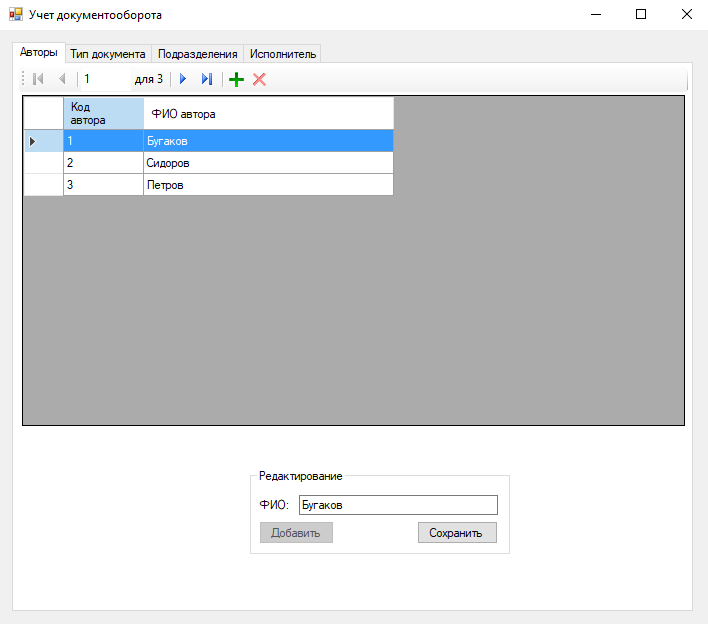


Рисунок – 1

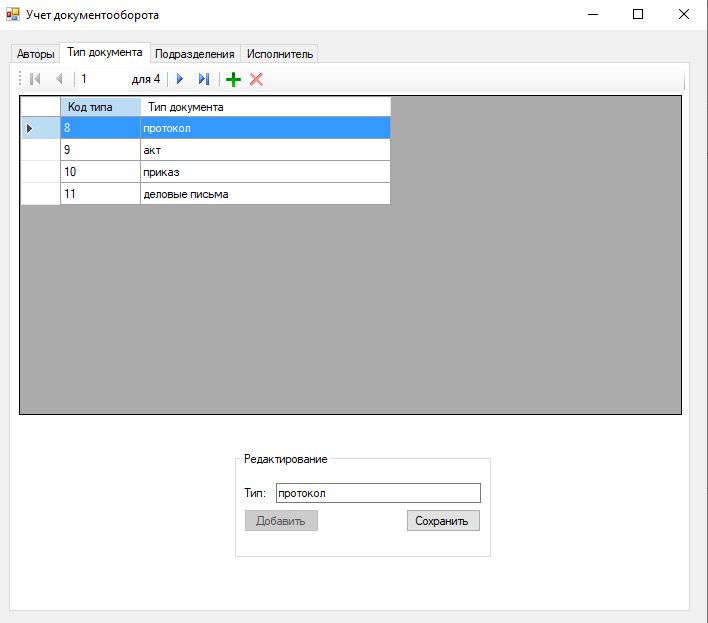


Рисунок – 2

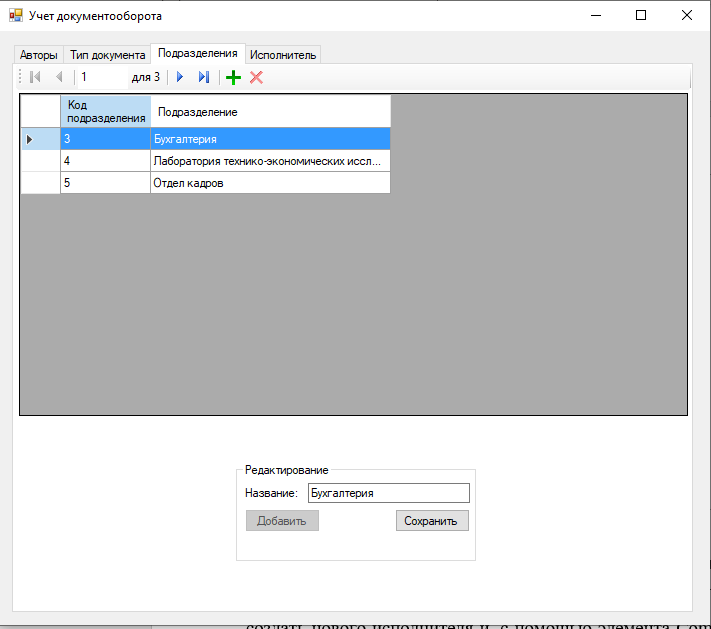


Рисунок – 3

Форма “Исполнителей” позволяет формировать новые данные на основе данных, которые были внесены в предыдущие таблицы. Таким образом можно создать нового исполнителя и ,с помощью элемента ComboBox, назначить ему подразделение, данные о котором были внесены в предыдущую таблицу.

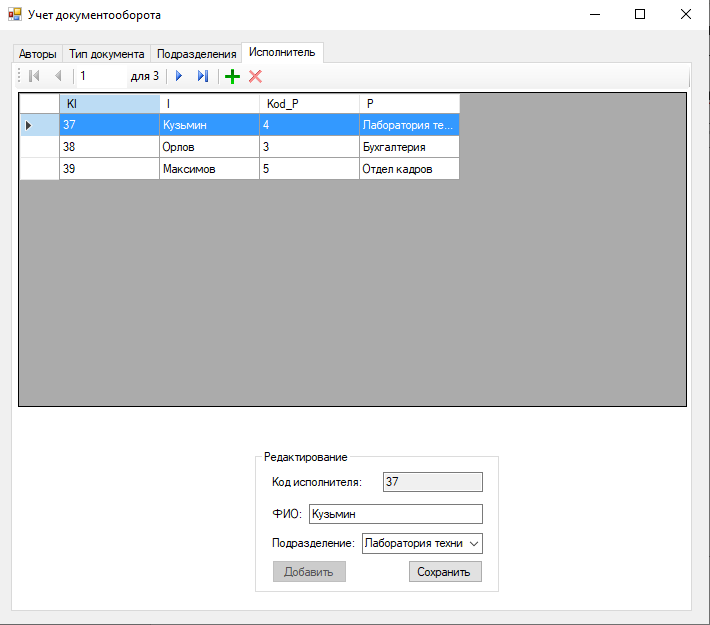


Рисунок – 4

При нажатии на зеленый крестик (Рисунок 5) в bindingNavigator появляется возможность добавления новой записи в любую из таблиц. Отображается надпись (Рисунок 5), предупреждающая о том, что таблица готова к добавлению новой записи. Кнопка “Добавить” становится доступной. Пользователь заполняет значения формы и нажимает кнопку “Добавить”, после этого в базу данных добавляется новая запись. Если пользователь нажимает кнопку “Добавить”, не заполнив форму, появляется сообщение о том, что пользователь не заполнил необходимые поля и эти поля подсвечиваются (Рисунок 6).



Рисунок – 5

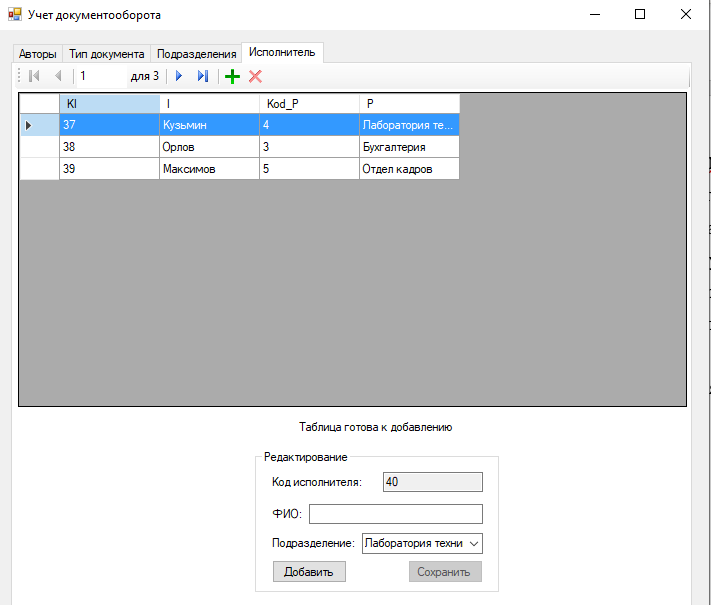


Рисунок – 6

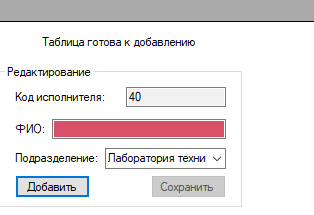
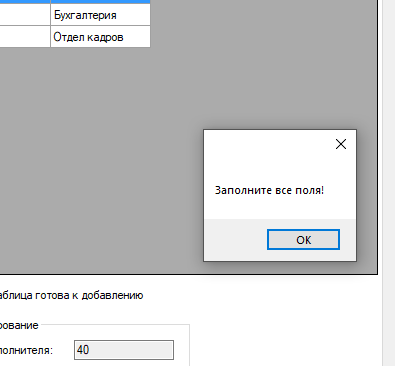


Рисунок – 7

После успешного добавления новой записи кнопка “Добавить” блокируется, пока не будет происходить новое добавление.

При нажатии на красный крестик в bindingNavigator (Рисунок 5) удаляется выбранная запись из БД.

После всех изменений таблицы необходимо нажать кнопку сохранить, чтобы сохранить изменения.

ФОРМА С ПОДЧИНЕННЫМИ ТАБЛИЦАМИ

Данное приложение содержит форму с подчиненными таблицами, где содержатся все авторы, их документы и поручения по этим документам. Всего на форме (Рисунок 8-9) расположено 3 таблицы и предусмотрена возможность редактирования, добавления и удаления поручений по выбранному документу с соответствующим автором.

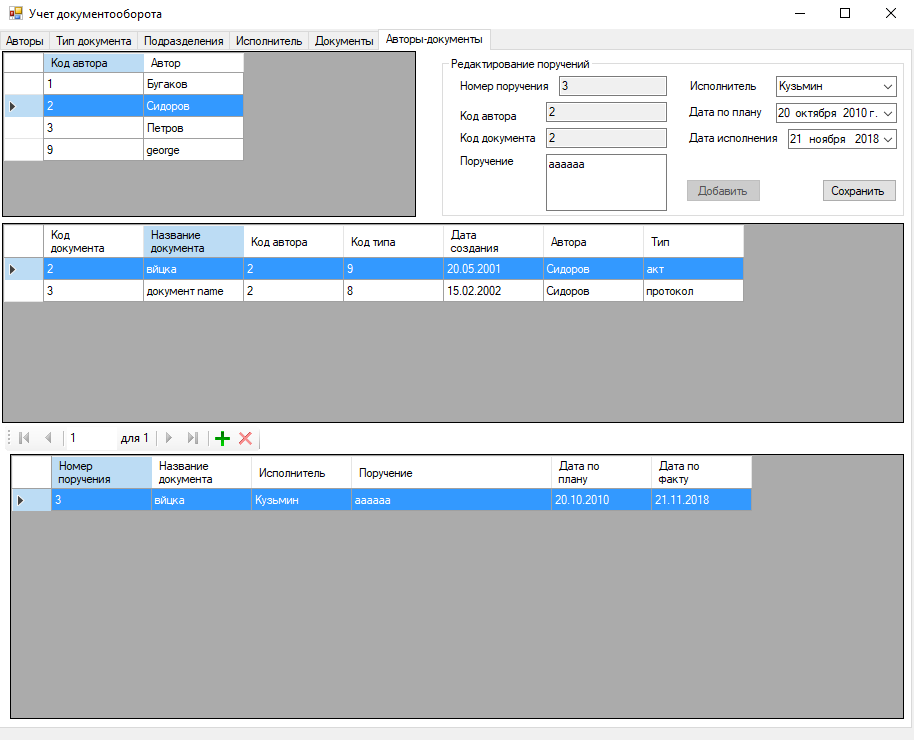
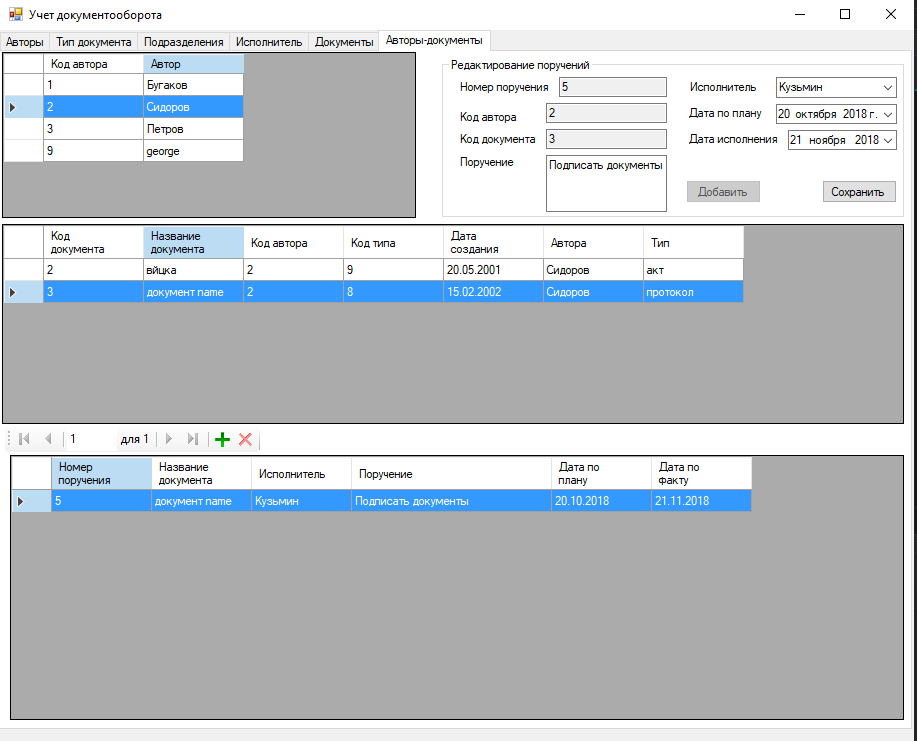
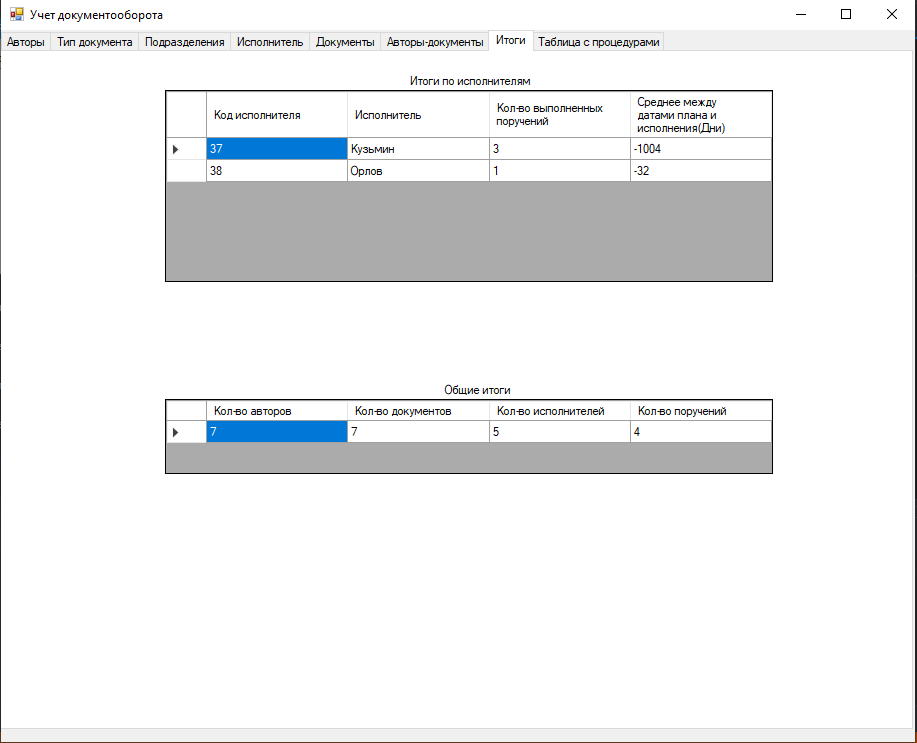


Рисунок – 8



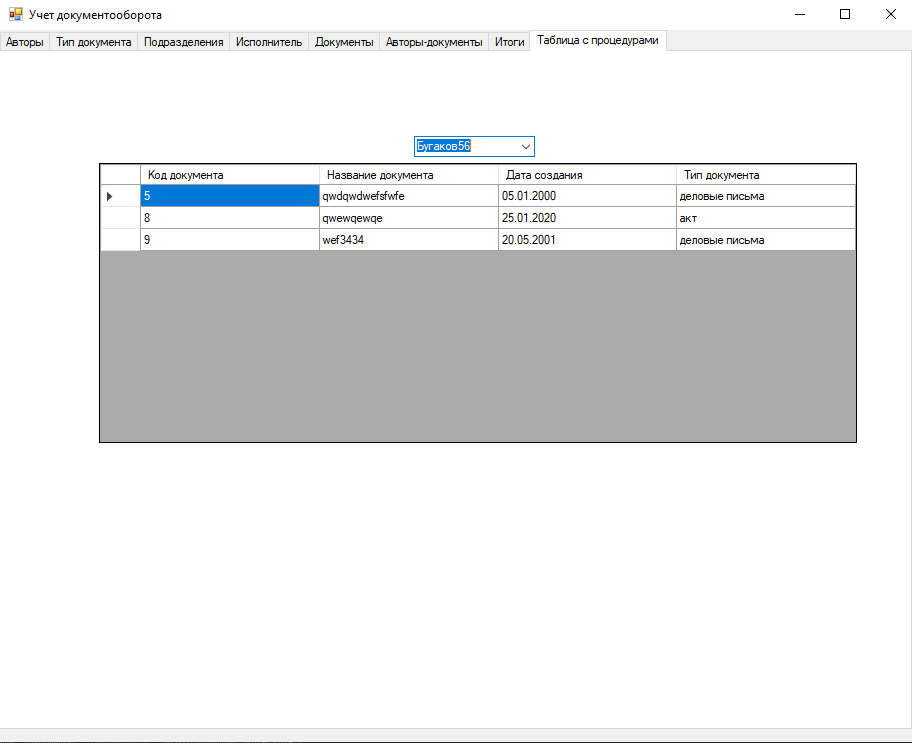
ФОРМА С ИТОГАМИ

В данной форме располагаются данные об исполнителях и кол-ве исполнений, которые они выполнили. Это реализовано черед представления в sql management studio. Так-же присутствует таблица с общими итогами.



ФОРМА С ПРОЦЕДУРАМИ

Данная форма использует процедуры для обработки данных в sql management. В данной форме содержится список с авторами и в datagrid выводится все информация по документам, которые они создали.



При наборе пользователем фамилии автора в combobox появляются фамилии, полученные по вводимым буквам для упрощения задачи.

