Министерство образования и науки ДНР

Государственное профессиональное образовательное учреждение

«Донецкий политехнический колледж»

Цикловая комиссия программирования и информатики

**КУРСОВОЙ** **ПРОЕКТ**

по дисциплине Технология раработки и защиты баз данных

(название дисциплины)

на тему: Разработка программного обеспечения института селекции растений

Студента III курса ПКС-15-1 группы

Направления подготовки 09.00.00 «Информатика и

вычислительная техника»

Программирование в компьютерных системах

Ермоленко Евгений Сергеевич\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(фамилия и инициалы)

Руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

преподаватель Прихоженко В.Б.\_\_\_\_\_\_

(должность, ученое звание, научная степень, фамилия и инициалы)

Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

г. Донецк 2018

# ЗАДАНИЕ

Разработать прикладное программное обеспечение института селекции растений.

Данный институт занимается сборов, выведением и продажей различных сортов семян. В его ассортименте можно найти семена практически всех возможных видов растений: от помидоров до редких цветов. Только что выведенные сорта заносятся в отдельный список для дальнейшего тестирования. Каждый сорт семян имеет свои характеристики, такие как урожайность, морозоустойчивость, адаптация к местным условиям, сроки созревания и т.п. Покупатель может выбрать сорт, отвечающий тем иди иным характеристикам. Компания занимается как оптовыми, так и розничными продажами. Оптовые покупатели заносятся в базу главным образом для того, чтобы информировать их о поступлении новых или отсутствовавших в определенный момент в продаже сортов.

# РЕФЕРАТ

Пояснительная записка к курсовому проекту:

40 c., 27 рис., 10 табл., 1 приложение, 6 источников

Цель курсового проекта – создание базы данных для автоматизации работы института селекции растений, который занимается выведением и продажей сортов семян.

Необходимо разработать базу данных для хранения сведений о сотрудниках, покупателях, партиях и сортах готовых для продажи; организовать формирование отчетов о продажах партий и информации о покупателях, сотрудниках, сортов.

Система разрабатывалась в СУБД Microsoft Access 2016.

В результате выполнения курсового проекта разработана база данных, позволяющая сохранять покупателей и данные о их покупках, если покупатель купил больше 3 партий, то он считается оптовым и это значение устанавливается в базе данных.

Ключевые слова:

БАЗА ДАННЫХ, ПАРТИЯ, ИНСТИТУТ, СОРТ, СЕЛЕКЦИЯ, ПРОДАЖА **СОДЕРЖАНИЕ**

[ЗАДАНИЕ 2](#_Toc516412111)

[РЕФЕРАТ 4](#_Toc516412112)

[ВВЕДЕНИЕ 8](#_Toc516412113)

[1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 10](#_Toc516412114)

[1.1 Обозначение границ моделируемой системы 10](#_Toc516412115)

[1.2 Определение цели и точки зрения модели 14](#_Toc516412116)

[1.3 Построение контекстной диаграммы в нотации IDEF0 14](#_Toc516412117)

[1.4 Построение диаграммы декомпозиции в нотации IDEF0 15](#_Toc516412118)

[2 АНАЛИЗ ТЕХНИЧЕСКОГО ЗАДАНИЯ 17](#_Toc516412119)

[3 ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ 20](#_Toc516412120)

[3.1 Концептуальное проектирование 20](#_Toc516412121)

[3.2 Логическое проектирование 20](#_Toc516412122)

[3.3 Обоснование выбора СУБД 22](#_Toc516412123)

[3.4 Физическое проектирование 23](#_Toc516412124)

[3.5 Разработка приложения 29](#_Toc516412125)

[3.5.1 Разработка таблиц 29](#_Toc516412126)

[3.5.2 Разработка запросов 36](#_Toc516412127)

[3.5.3 Разработка форм 37](#_Toc516412128)

[3.5.4 Разработка отчетов 44](#_Toc516412129)

[4 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ 49](#_Toc516412130)

[ЗАКЛЮЧЕНИЕ 53](#_Toc516412131)

[ПЕРЕЧЕНЬ ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 54](#_Toc516412132)

**ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ, СОКРАЩЕНИЙ И ТЕРМИНОВ**

БД – База данных

СУБД – система управления базами данных

т.е. – то есть

т.н. – так называемые

т.к. – так-как

# ВВЕДЕНИЕ

Селекция растений успешно практикуется человеком на протяжении тысяч лет, с самого начала человеческой цивилизации. Используется по всему миру как отдельными людьми: садоводами, фермерами, так и профессиональными селекционерами в организациях, университетах и исследовательских центрах.

По мнению международных агентств развития, важное значение имеет выведение новых сортов [сельскохозяйственных культур](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D0%BB%D1%8C%D1%81%D0%BA%D0%BE%D1%85%D0%BE%D0%B7%D1%8F%D0%B9%D1%81%D1%82%D0%B2%D0%B5%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BA%D1%83%D0%BB%D1%8C%D1%82%D1%83%D1%80%D0%B0), имеющих высокую урожайность, устойчивых к болезням, засухам и адаптированных к региональным условиям выращивания.

Для разработки была использована СУБД Microsoft Access 2016.

Microsoft Access – [реляционная система управления базами данных (СУБД)](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94). Входит в состав пакета Microsoft Office. Имеет широкий спектр функций, включая связанные запросы, связь с внешними таблицами и [базами данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85). Благодаря встроенному языку [VBA](https://ru.wikipedia.org/wiki/VBA), в самом Access можно писать приложения, работающие с базами данных.

Целью разработки данного программного обеспечения является контроль выведения новых сортов с улучшенными характеристиками. Такие как морозоустойчивость, адаптация к окружающим условиям и срок созревания сорта. Также программное обеспечение должно реализовывать покупку сортов.

Достижение указанной цели осуществлялось путем решения следующих основных задач:

* выполнить анализ предметной области;
* провести анализ предметной области;
* выполнить проектирование базы данных;
* произвести концептуальное проектирование;
* произвести логическое проектирование;
* произвести физическое проектирование;
* выполнить разработку приложения;
* составить руководство пользователя.

# АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

## Обозначение границ моделируемой системы

Для моделируемой системы должны быть определены следующие данные:

* список объектов, входящих в систему;
* список функций, выполняемых системой.

Посредством данной информации, есть возможность в достаточной степени определить границы данной моделируемой системы.

Порядок выполнения задания

* определим моделируемую систему;
* далее, составим список всех основных объектов, которые могут лежат в границах моделируемой системы.

Список объектов:

* Название сорта;
* описание сорта;
* год вывода сорта;
* до какой даты годен к посадке (дата);
* адаптирован ли сорт к морозам (да/нет);
* адаптирован ли сорт к погодным условиям (да/нет);
* способ посадки;
* срок созревания;
* название нового сорта;
* дата вывода нового сорта;
* название сорта от какого сорта был выведен новый сорт;
* описание нового сорта;
* адаптирован ли сорт к морозам (да/нет);
* адаптирован ли новый сорт к погодным условиям (да/нет);
* способ посадки нового сорта;
* срок созревания нового сорта;
* номер партии;
* одобрена ли инспекцией (да/нет);
* вид упаковки;
* количество семян в упаковке;
* дата расфасовки;
* стоимость партии;
* название фирмы покупателя;
* дата покупки сорта;
* адрес фирмы;
* контактный телефон фирмы;
* электронная почта;
* оптовый покупатель (да/нет);
* оплата наличными (да/нет);
* имя продавца;
* фамилия продавца;
* отчество продавца;
* телефон продавца;
* электронная почта продавца;
* адрес проживания;
* заработная плата продавца;
* стаж работы;
* имя селекционера;
* фамилия селекционера;
* отчество селекционера;
* телефон селекционера;
* электронная почта селекцонера;
* адрес проживания;
* заработная плата селекционера;
* стаж работы селекционера;
* имя директора;
* фамилия директора;
* отчество директора;
* телефон директора;
* адрес проживания;
* электронная почта директора;
* заработная плата директора;
* имя проректора;
* фамилия проректора;
* отчество проректора;
* телефон проректора;
* адрес проживания;
* заработная плата проректора;
* имя системного администратора;
* фамилия системного администратора;
* отчество системного администратора;
* телефон проректора;
* адрес проживания;
* заработная плата системного администратора.

Из оставшихся элементов, необходимо разделить их по критериям:

Сорт:

* название сорта
* описание сорта;
* год вывода сорта;
* до какой даты годен к посадке (дата);
* адаптирован ли сорт к морозам (да/нет);
* адаптирован ли сорт к погодным условиям (да/нет);
* способ посадки;
* срок созревания.

Новый сорт:

* название нового сорта;
* дата вывода нового сорта;
* название сорта от какого сорта был выведен новый сорт;
* описание нового сорта;
* адаптирован ли сорт к морозам (да/нет);
* адаптирован ли новый сорт к погодным условиям (да/нет);
* способ посадки нового сорта;
* срок созревания нового сорта.

Партия:

* номер партии;
* одобрена ли инспекцией (да/нет);
* вид упаковки;
* количество семян в упаковке;
* дата расфасовки;
* стоимость партии.

Покупатель:

* название фирмы покупателя;
* дата покупки сорта;
* адрес фирмы;
* контактный телефон фирмы;
* оптовый покупатель (да/нет);
* оплата наличными (да/нет).

Продавец:

* имя продавца;
* фамилия продавца;
* телефон продавца;
* адрес проживания;
* заработная плата продавца;
* стаж работы.

Селекционер:

* имя селекционера;
* фамилия селекционера;
* телефон селекционера;
* адрес проживания;
* заработная плата селекционера;
* стаж работы селекционера.

В результате проведенного анализа были сформированы списки объектов и функций для моделируемой систем.

## Определение цели и точки зрения модели

Данное программное обеспечения разработано для использования несколькими группами пользователей, такие как:

* Администрация института;
* Системный администратор;
* Селекционер;
* Продавец;
* Покупатель.

## Построение контекстной диаграммы в нотации IDEF0

IDEF0 предоставляет построение иерархической системы диаграмм - единичных описаний фрагментов системы (т.н. сущностей). Для начала, проводится описание системы в целом и ее взаимодействия с окружающим миром (контекстная диаграмма), затем проводится функциональная декомпозиция - система разбивается на подсистемы (сущности) и каждая подсистема (сущность) опи­сывается отдельно.

Результат построения диаграммы декомпозиции IDEF0 показан на рисунке 1.1.

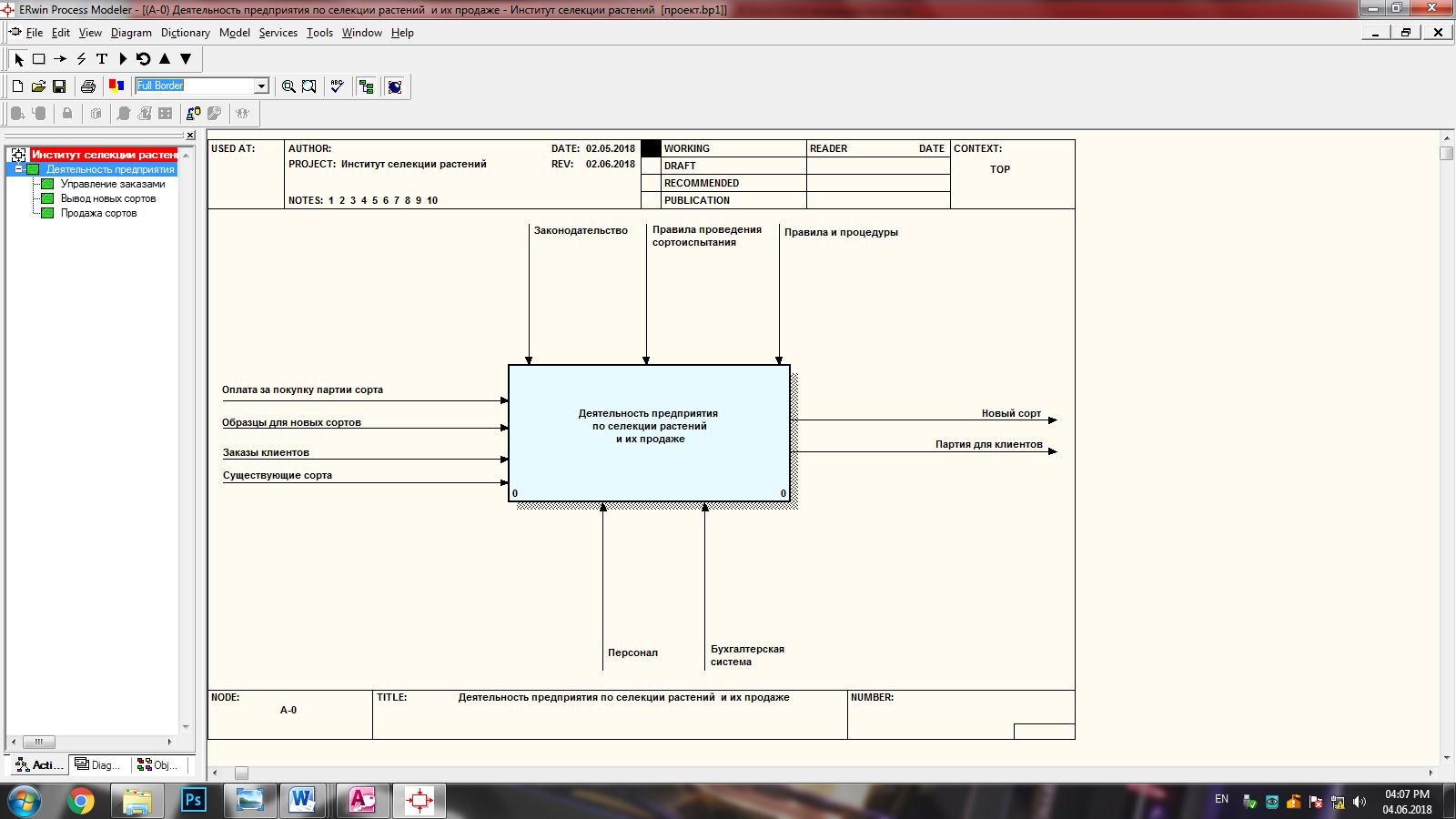


Рисунок 1.1 – Контекстная диаграмма IDEF0

## Построение диаграммы декомпозиции в нотации IDEF0

Диаграмма декомпозиции контекстной диаграммы представлена на рисунке 1.2.

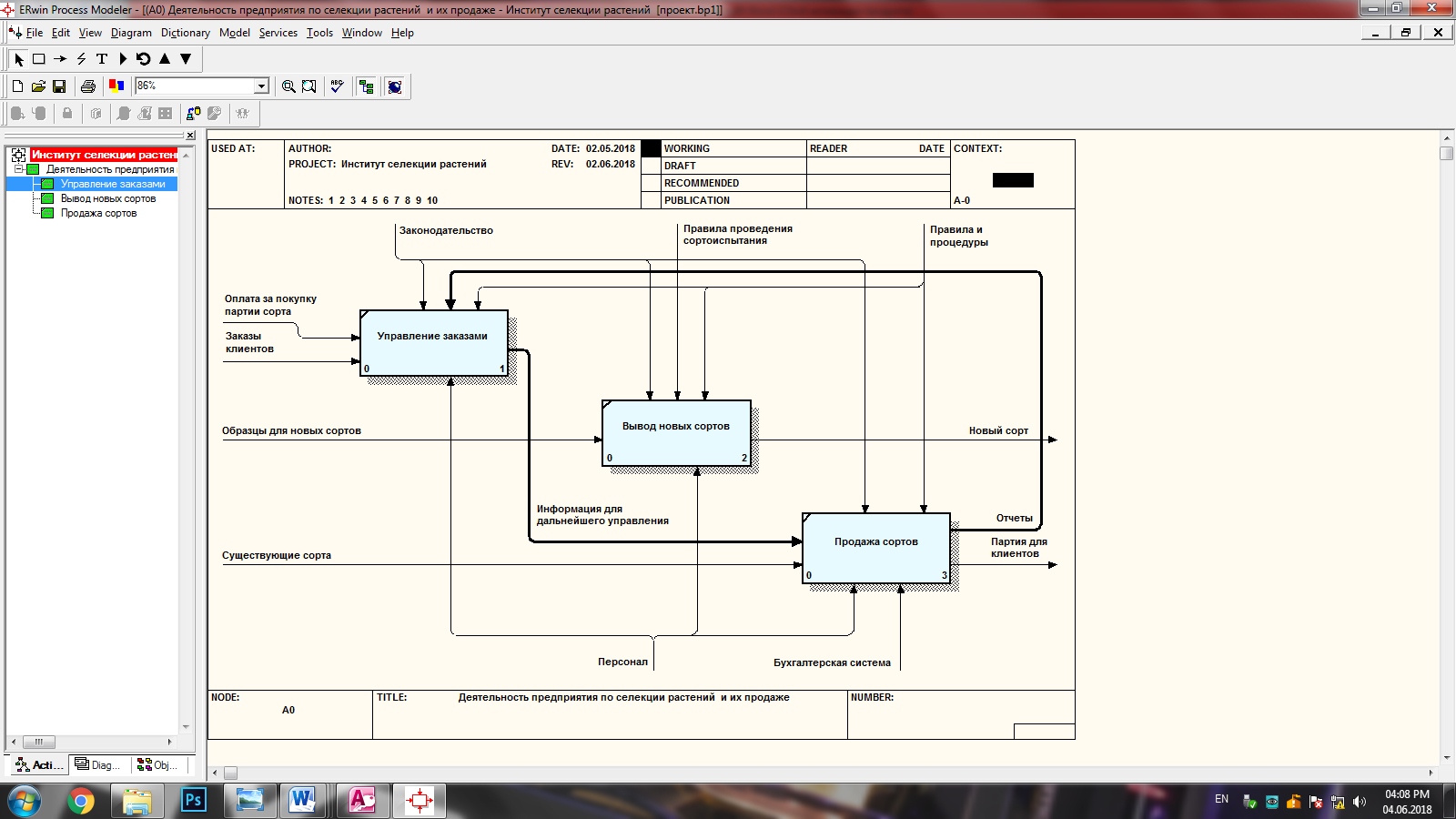


Рисунок 1.2 ­­­­­­­­­– Декомпозиция главного функционального блока

В результате исследования предметной области было разработано техническое задание (Приложение А).

# АНАЛИЗ ТЕХНИЧЕСКОГО ЗАДАНИЯ

Анализ описания предметной области института селекции растений позволяет определить требования к введению и хранению информации в базе данных:

* о хранении данных о сорте (номер сорта, название, год вывода, описание, технология посадки, срок созревания, адаптация к внешним условиям, морозоустойчивость, фото);
* о хранении данных о новом сорте (номер сорта, название, дата вывода, описание, технология посадки, срок созревания, адаптация к внешним условиям, морозоустойчивость, фото);
* о хранении данных о партии (номер партии, тип упаковки, кол-во семян, вес, дата упаковки, проверка инспекцией, дата окончания срока годности, стоимость);
* о хранении данных о покупателе (номер покупателя, название компании, адрес, телефон, электронная почта, оптовый покупатель);
* о хранении данных о продавце (номер продавца, имя, фамилия, отчество, фото, дата рождения, телефон, электронная почта, адрес, стаж, заработная плата);
* о хранении данных о селекционере (номер селекционера, имя, фамилия, отчество, фото, образование, дата рождения, адрес, телефон, электронная почта, заработная плата);
* о хранении данных заказа (номер заказа, номер покупателя, дата заказа, сумма заказа, оплата наличными);

Таким образом, результатом анализа технического задания должен быть конкретный перечень объектов, которые необходимо спроектировать при выполнении курсового проекта:

* установить связь между таблицами «сорт» и «новый сорт» в виде 1:M;
* установить связь между таблицами «новый сорт» и «селекционер» в виде 1:M;
* установить связь между таблицами «сорт» и «партия» в виде 1:M;
* установить связь между таблицами «партия» и «отправка» в виде 1:M;
* установить связь между таблицами «отправка» и «покупатель» в виде M:1;
* установить связь между таблицами «покупатель» и «продавец» в виде 1:M;
* установить связь между таблицами «покупатель» и «заказ» в виде 1:M.
* создать запрос на обновление пароля для авторизации в базе данных;
* создать запрос на удаление сорта;
* создать перекрестный запрос, который будет выводить селекционера и его выведенный сорт;
* создать запрос на выборку, который выводит оптовых покупателей;
* создать параметрический запрос, который выводит партии на заданную дату;
* создать формы для добавления нового покупателя;
* создать формы для добавления нового сотрудника;
* создать форму для добавления нового сорта;
* создать форму для оформления заказа;
* создать форму в которой указывается количество выбранных партий для заказа и сумму покупки;
* создать форму для изменения данных о определенном сотруднике;
* создать форму для изменения пароля для группы сотрудников;
* создать отчет о покупателях и их купленных партиях;
* создать отчет о партиях и их покупателях;
* создать отчет для вывода совершенной покупки;
* создать отчет для вывода сотрудников института.

# ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ

## Концептуальное проектирование

В результате разработки сформировал данную концептуальную схему (рис.3.1).

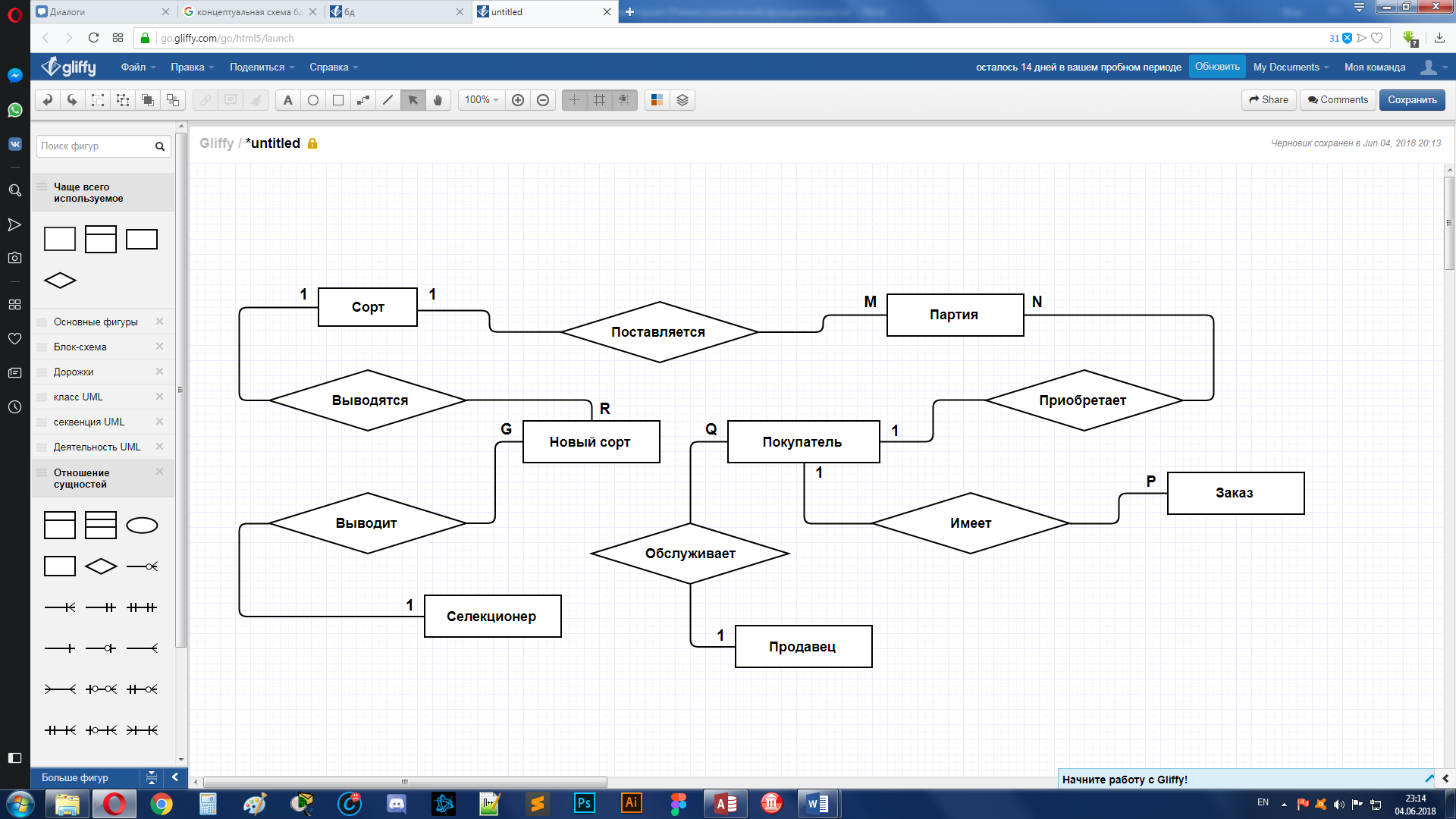


Рисунок 3.1 ­­­­­­­­– Концептуальная схема

## Логическое проектирование

В результате разработки сформировал данную логическую схему (рис.3.2).

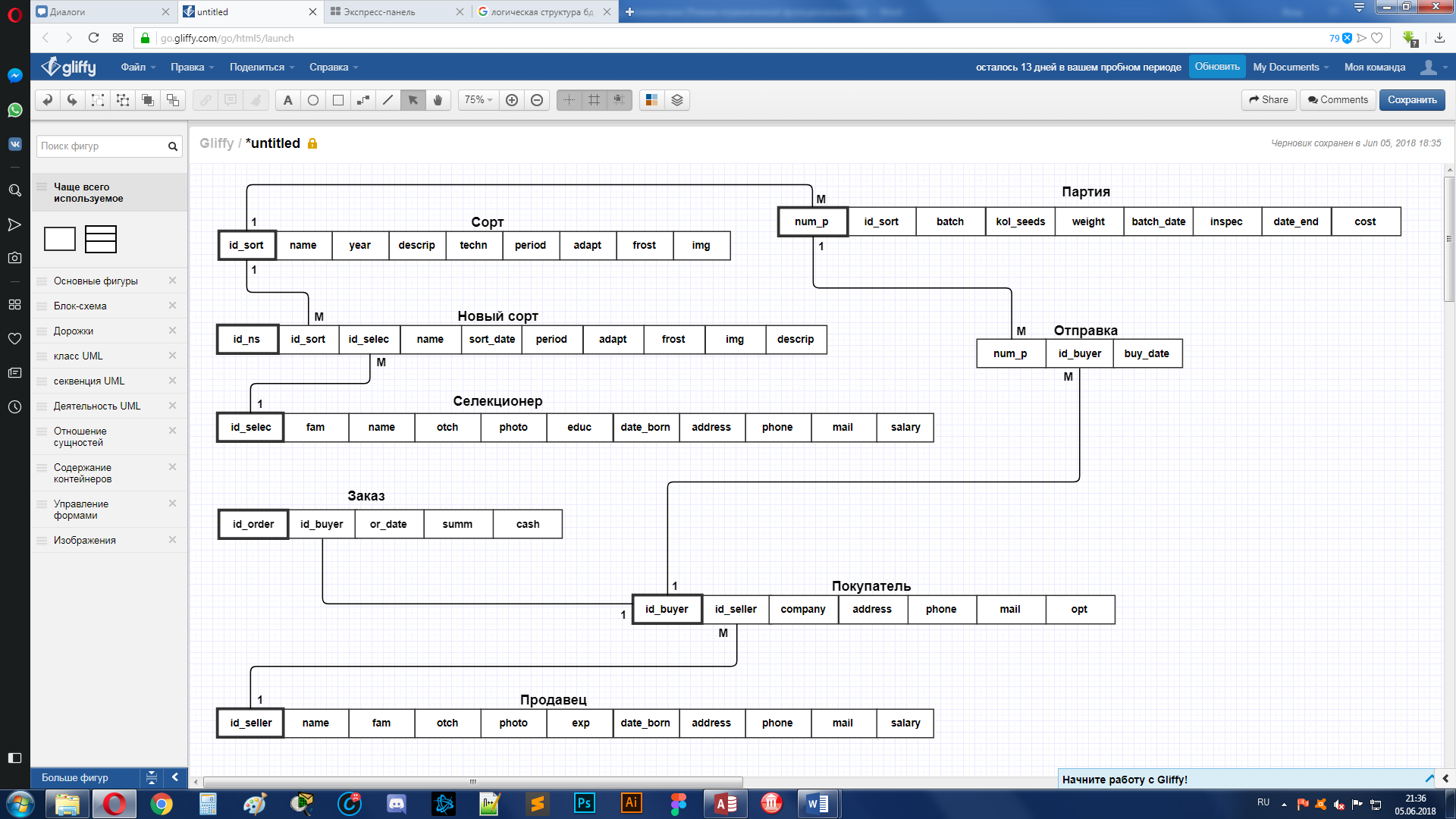


Рисунок 3.2 ­­­­­­­­– Логическая схема

В результате разработки сформировал данную ER-диаграмму (рис.3.3).

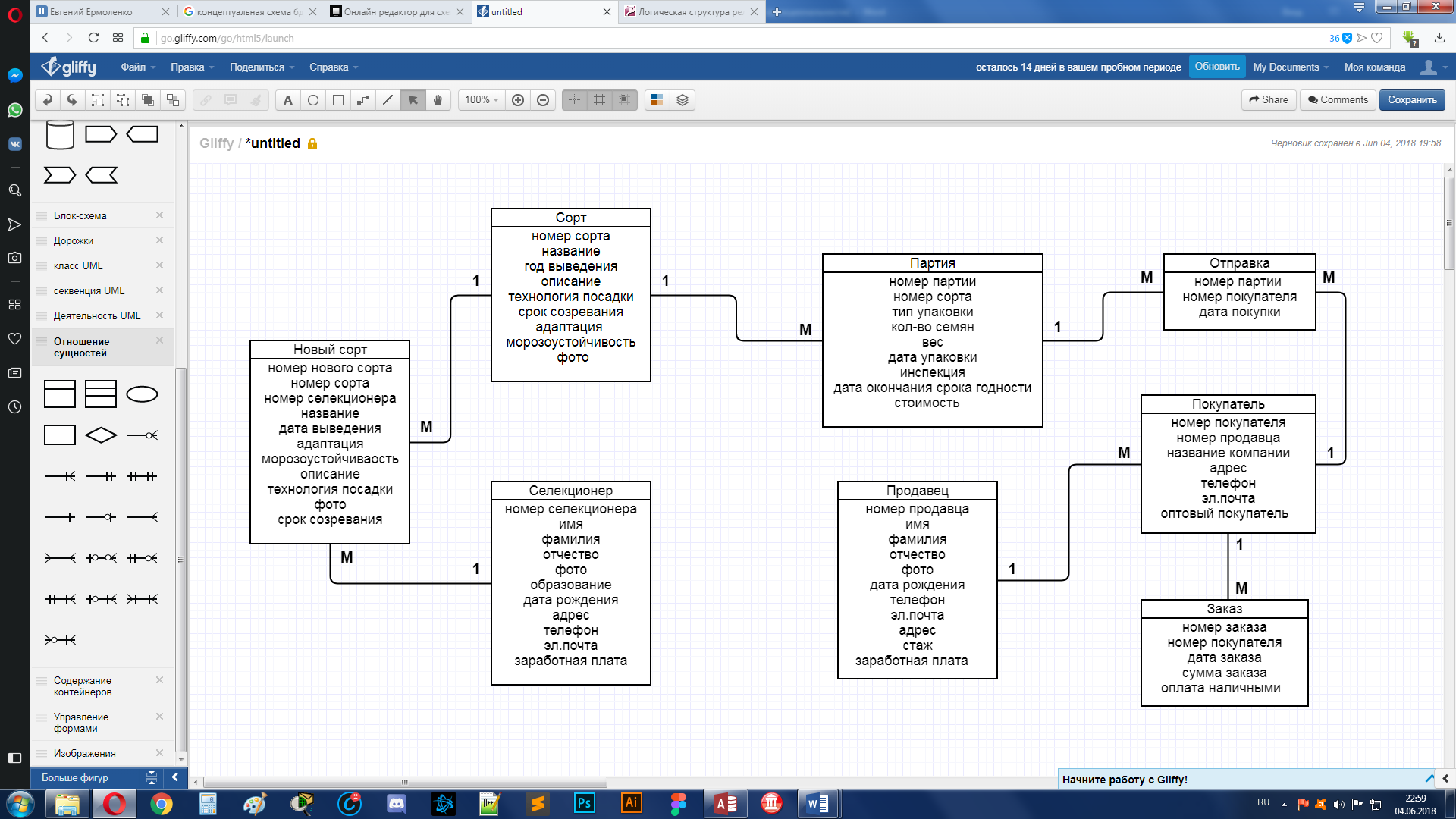


Рисунок 3.3 ­­­­­­­­– ER-диаграмма

## Обоснование выбора СУБД

Access – мощное приложение Windows. При этом производительность СУБД органично сочетаются со всеми удобствами и преимуществами Windows.

Как реляционная СУБД Access обеспечивает доступ ко всем типам данных и позволяет одновременно использовать несколько таблиц базы данных. Можно использовать таблицы, созданные в среде Paradox или dBase. Работая в среде Microsoft Office, пользователь получает в своё распоряжение полностью совместимые с Access текстовые документы(Word), электронные таблицы(Excel), презентации(PowerPoint).С помощью новых расширений для Internet можно напрямую взаимодействовать с данными из World Wide Web и транслировать представление данных на языке HTML, обеспечивая работу с такими приложениями как Internet Explorer и Netscape Navigator.

Access специально спроектирован для создания многопользовательских приложений, где файлы базы данных являются разделяемыми ресурсами в сети. В Access реализована надёжная система защиты от несанкционированного доступа к файлам.

Несмотря на то, что Access является мощной и сложной системой, его использование не сложно для непрофессиональных пользователей.

Элементы базы данных.

Таблицы. В базе данных информация хранится в виде двумерных таблиц. Можно так же импортировать и связывать таблицы из других СУБД или систем управления электронными таблицами. Одновременно могут быть открыты 1024 таблицы.

Запросы*.* При помощи запросов можно произвести выборку данных по какому-нибудь критерию из разных таблиц. В запрос можно включать до 255 полей.

Формы. Формы позволяют отображать данные из таблиц и запросов в более удобном для восприятия виде. С помощью форм можно добавлять и изменять данные, содержащиеся в таблицах. В формы позволяют включать модули.

Отчёты. Отчёты предназначены для печати данных, содержащихся в таблицах и запросах, в красиво оформленном виде. Отчёты так же позволяют включать модули.

Макросы

Модули. Модули содержат VBA-код, используемый для написания процедур обработки событий, таких как например нажатия кнопки в форме или отчёте, для создания функций настройки, для автоматического выполнения операций над объектами базы данных и программного управления операциями, т.е. добавление VBA-кода позволяет создать полную базу данных с настраиваемыми меню, панелями инструментов и другими возможностями. Модули снимают с пользователя приложения необходимость помнить последовательность выбора объектов базы данных для выполнения того или иного действия и повышают эффективность работы.

База данных может содержать до 32768 объектов.

В состав Access входит множество мастеров, построителей и надстроек, которые позволяют упростить процесс создания объектов базы данных.

## Физическое проектирование

При разработке базы данных было составлено 10 таблиц(табл.3.1-3.10).

Таблица 3.1 – таблица «Сорт»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| id\_sort | Ключевое поля, Счетчик |
| name | Короткий текст |
| year | Числовой |
| descrip | Длинный текст |
| techn | Длинный текст |
| period | Числовой |
| adapt | Логический |
| frost | Логический |
| img | Поле объекта OLE |

Таблица 3.2 – таблица «Новый сорт»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| id\_ns | Ключевое поля, Счетчик |
| name | Короткий текст |
| sort\_date | Дата и время |
| adapt | Логический |
| frost | Логический |
| descrip | Длинный текст |
| techn | Длинный текст |
| img | Поле объекта OLE |
| period | Числовой |

Таблица 3.3 – таблица «Селекционер»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| id\_selec | Ключевое поля, Счетчик |
| fam | Короткий текст |
| name | Короткий текст |
| otch | Короткий текст |
| photo | Поле объекта OLE |
| educ | Короткий текст |
| date\_born | Дата и время |
| address | Короткий текст |
| phone | Короткий текст |
| mail | Короткий текст |
| salary | Числовой |

Таблица 3.4 – таблица «Партия»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| num\_p | Ключевое поля, Счетчик |
| batch | Короткий текст |
| kol\_seeds | Числовой |
| weight | Числовой |
| batch\_date | Дата и время |
| inspection | Логический |
| date\_end | Дата и время |
| cost | Числовой |

Таблица 3.5 – таблица «Отправка»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| num\_p | Числовой |
| id\_buyer | Числовой |
| buy\_date | Дата и время |

Таблица 3.6 – таблица «Покупатель»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| id\_buyer | Ключевое поля, Счетчик |
| company | Короткий текст |
| address | Короткий текст |
| phone | Короткий текст |
| mail | Короткий текст |
| opt | Логический |

Таблица 3.7 – таблица «Заказ»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| id\_order | Ключевое поля, Счетчик |
| or\_date | Дата и время |
| summ | Числовой |
| cash | Логический |

Таблица 3.8 – таблица «Должности»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| id\_user | Ключевое поля, Счетчик |
| dolzh | Короткий текст |
| password | Короткий текст |

Таблица 3.9 – таблица «Продавец»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| id\_seller | Ключевое поля, Счетчик |
| name | Короткий текст |
| fam | Короткий текст |
| otch | Короткий текст |
| date\_born | Дата и время |
| address | Короткий текст |
| phone | Короткий текст |
| mail | Короткий текст |
| exp | Числовой |
| salary | Числовой |

Таблица 3.10 – таблица «Администрация»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| id\_adm | Ключевое поля, Счетчик |
| name | Короткий текст |
| fam | Короткий текст |
| otch | Короткий текст |
| date\_born | Дата и время |
| address | Короткий текст |
| phone | Короткий текст |
| mail | Короткий текст |

После разработки базы данных в СУБД Access 2016 сформировалась схема данных, которая представлена на рисунке 3.4.

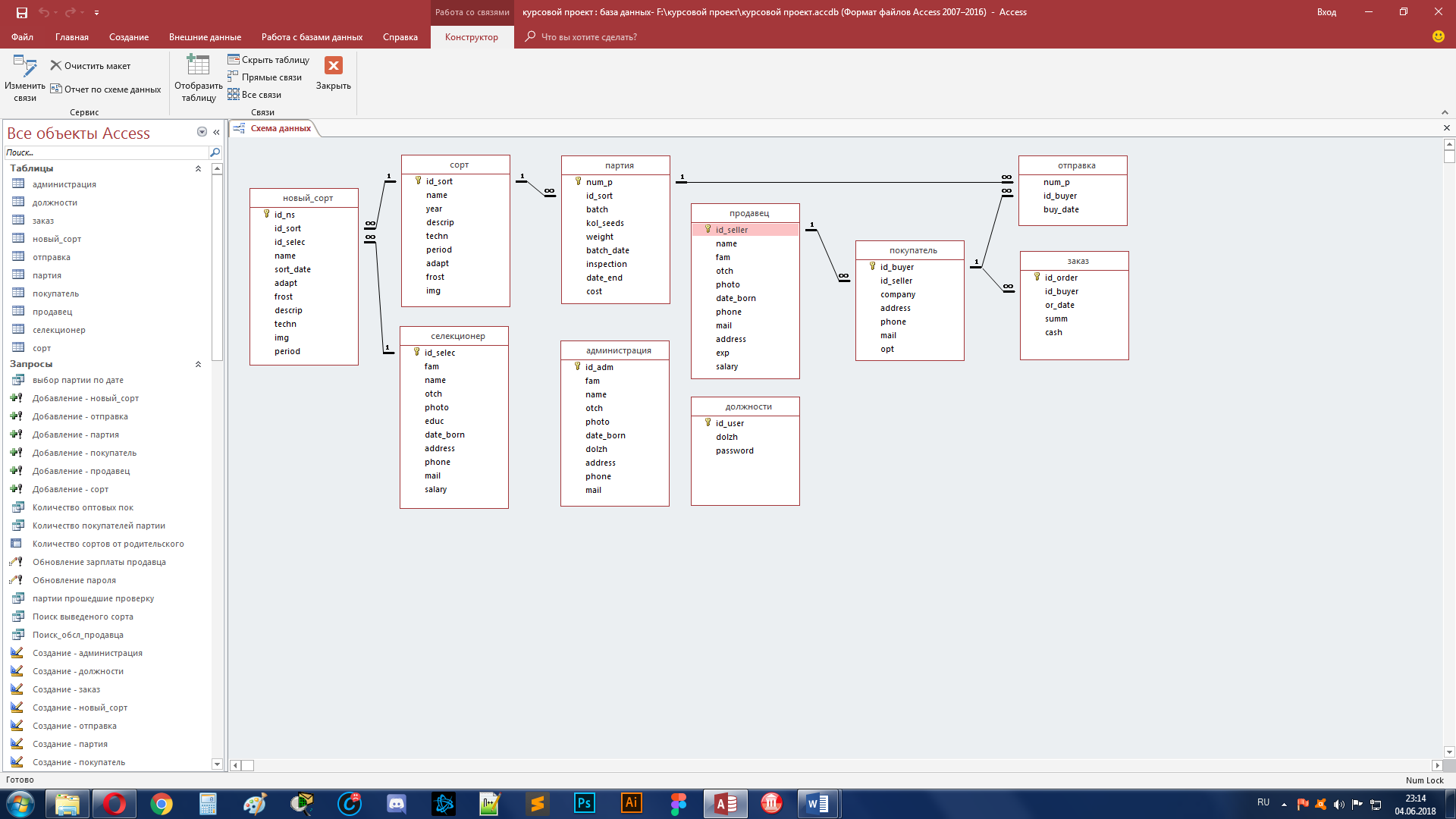


Рисунок 3.4 ­­– Схема данных

## Разработка приложения

### Разработка таблиц

Для создания таблиц в базе данных использовал язык структурированных запросов SQL.

SQL запрос на создание таблицы «сорт».

CREATE TABLE сорт

(

id\_sort COUNTER NOT NULL PRIMARY KEY,

name VARCHAR(40) NOT NULL,

year INT NOT NULL,

adapt BIT NOT NULL,

frost BIT NOT NULL,

descrip LONGTEXT NOT NULL,

techn LONGTEXT NOT NULL,

img IMAGE NULL,

period INT NOT NULL

);

SQL запрос на создание таблицы «новый\_сорт».

CREATE TABLE новый\_сорт

(

id\_ns COUNTER NOT NULL PRIMARY KEY,

id\_sort INT NOT NULL,

id\_selec INT NOT NULL,

name VARCHAR(30) NOT NULL,

sort\_date DATE NOT NULL,

adapt BIT NOT NULL,

frost BIT NOT NULL,

descrip LONGTEXT NOT NULL,

techn LONGTEXT NOT NULL,

img IMAGE NULL,

period INT NOT NULL,

FOREIGN KEY (id\_sort) REFERENCES сорт (id\_sort),

FOREIGN KEY (id\_selec) REFERENCES селекционер (id\_selec)

);

SQL запрос на создание таблицы «селекционер».

CREATE TABLE селекционер

(

id\_selec COUNTER NOT NULL PRIMARY KEY,

fam VARCHAR(40) NOT NULL,

name VARCHAR(30) NOT NULL,

otch VARCHAR(30) NOT NULL,

photo IMAGE,

educ VARCHAR(30) NOT NULL,

date\_born DATE NOT NULL,

address VARCHAR(50) NOT NULL,

phone VARCHAR(20) NOT NULL,

mail CHAR,

salary FLOAT NOT NULL

);

SQL запрос на создание таблицы «партия».

CREATE TABLE партия

(

num\_p COUNTER NOT NULL PRIMARY KEY,

id\_sort INT NOT NULL,

batch VARCHAR(20) NOT NULL,

kol\_seeds INT NOT NULL,

weight FLOAT NOT NULL,

batch\_date DATE NOT NULL,

inspection BIT NOT NULL,

date\_end DATE NOT NULL,

cost FLOAT NOT NULL,

FOREIGN KEY (id\_sort) REFERENCES сорт (id\_sort)

);

SQL запрос на создание таблицы «отправка».

CREATE TABLE отправка

(

num\_p INT NOT NULL,

id\_buyer INT NOT NULL,

buy\_date DATE NOT NULL,

summ FLOAT NOT NULL,

FOREIGN KEY (num\_p) REFERENCES партия(num\_p),

FOREIGN KEY (id\_buyer) REFERENCES покупатель(id\_buyer)

);

SQL запрос на создание таблицы «покупатель».

CREATE TABLE покупатель

(

id\_buyer COUNTER NOT NULL PRIMARY KEY,

id\_seller INT NOT NULL,

company VARCHAR(40) NOT NULL,

address VARCHAR(40),

phone VARCHAR(20) NOT NULL,

mail CHAR,

opt BIT,

FOREIGN KEY (id\_seller) REFERENCES продавец (id\_seller)

);

SQL запрос на создание таблицы «продавец».

CREATE TABLE продавец

(

id\_seller COUNTER NOT NULL PRIMARY KEY,

name VARCHAR(20) NOT NULL,

fam VARCHAR(30) NOT NULL,

otch VARCHAR(30) NOT NULL,

photo IMAGE NULL,

date\_born DATE NOT NULL,

phone VARCHAR(15) NOT NULL,

mail CHAR,

address VARCHAR(40) NOT NULL,

exp INT NOT NULL,

salary INT NOT NULL

);

SQL запрос на создание таблицы «заказ».

CREATE TABLE заказ

(

id\_order COUNTER NOT NULL PRIMARY KEY,

id\_buyer int NOT NULL,

or\_date DATE NOT NULL,

summ FLOAT NOT NULL,

cash BIT NOT NULL,

FOREIGN KEY (id\_buyer) REFERENCES покупатель(id\_buyer)

);

SQL запрос на создание таблицы «администрация».

CREATE TABLE администрация

(

id\_adm COUNTER NOT NULL PRIMARY KEY,

fam VARCHAR(40) NOT NULL,

name VARCHAR(30) NOT NULL,

otch VARCHAR(30) NOT NULL,

photo IMAGE NULL,

date\_born DATE NOT NULL,

dolzh VARCHAR(20) NOT NULL,

address VARCHAR(30) NOT NULL,

phone VARCHAR(20) NOT NULL,

mail CHAR

);

SQL запрос на создание таблицы «должности».

CREATE TABLE должности

(

id\_user COUNTER NOT NULL PRIMARY KEY,

dolzh VARCHAR(30) NOT NULL,

password VARCHAR(9)

);

Результаты выполнений запросов на создание таблиц представлены на рисунках 3.5-3.14.

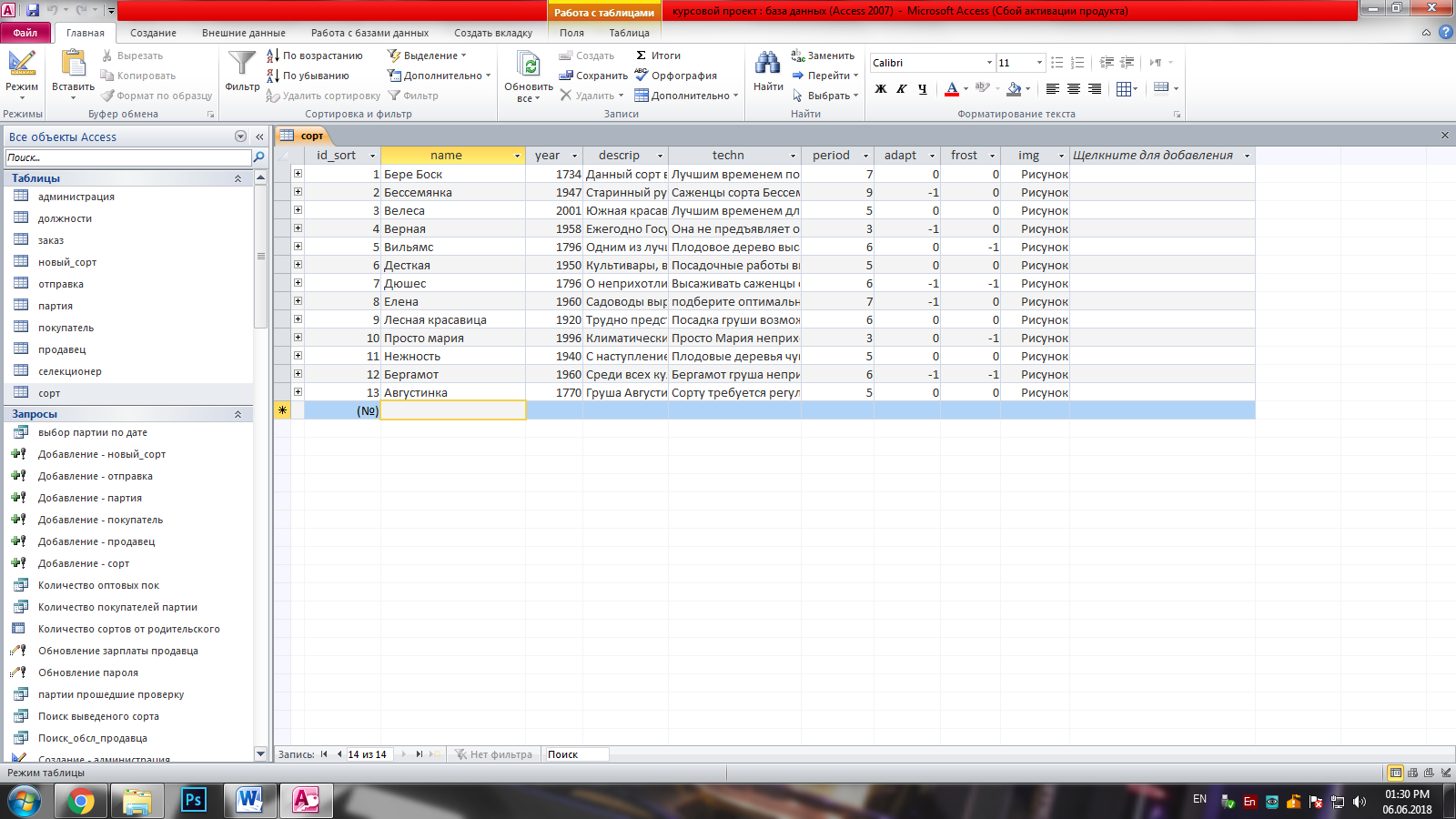


Рисунок 3.5 – Созданная таблица «Сорт»

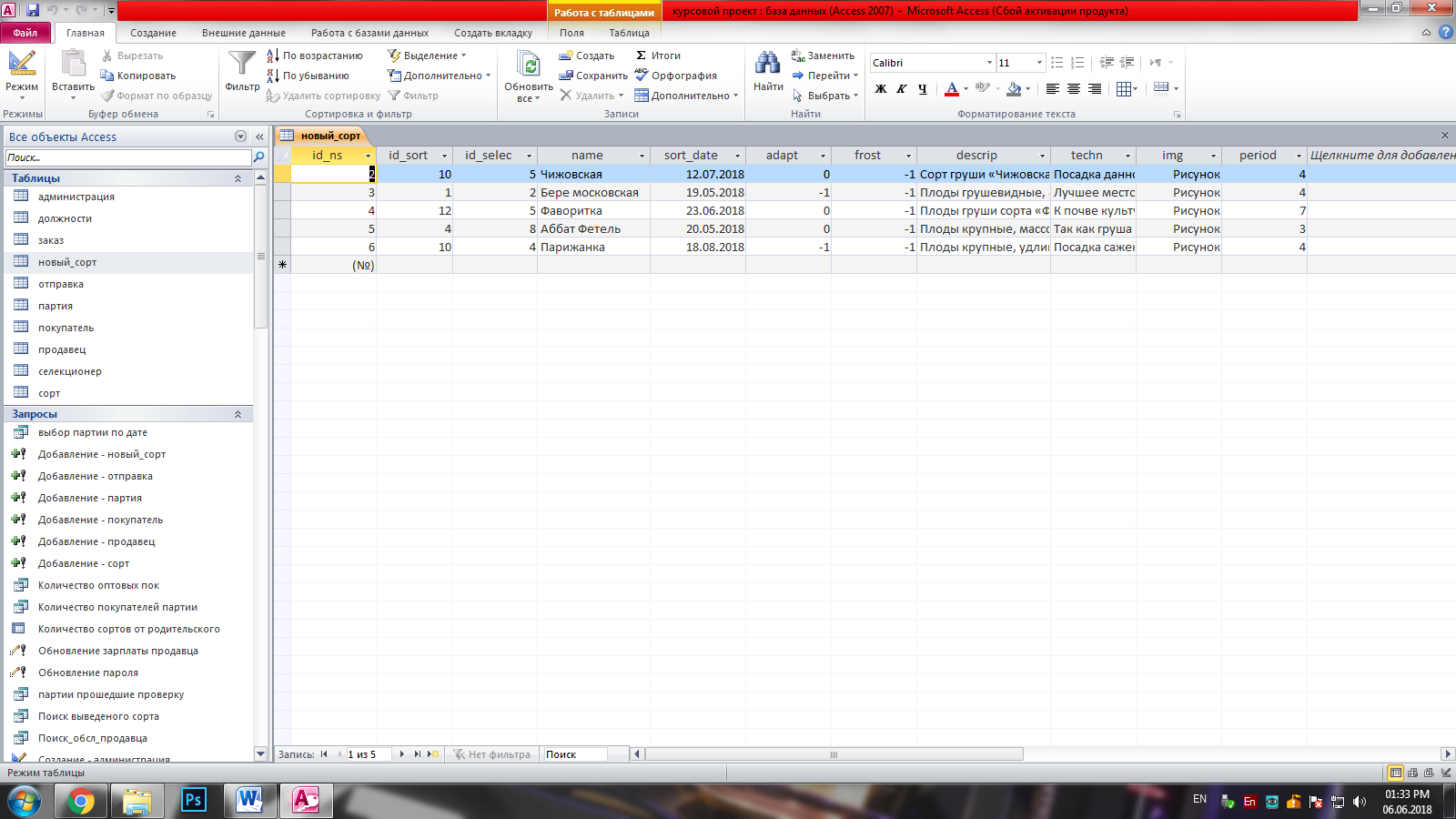


Рисунок 3.6 – Созданная таблица «Новый сорт»

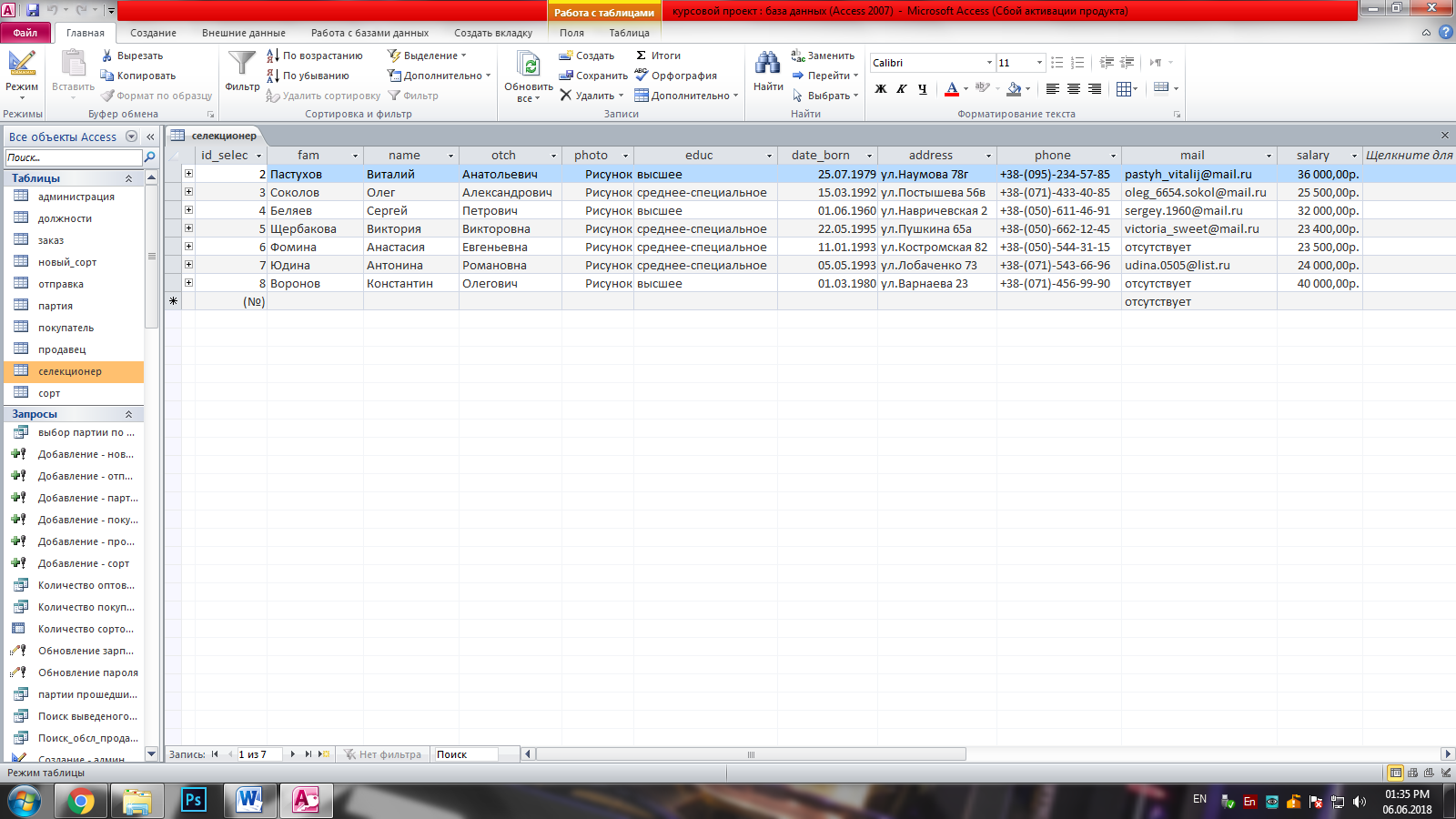


Рисунок 3.7 – Созданная таблица «Селекционер»

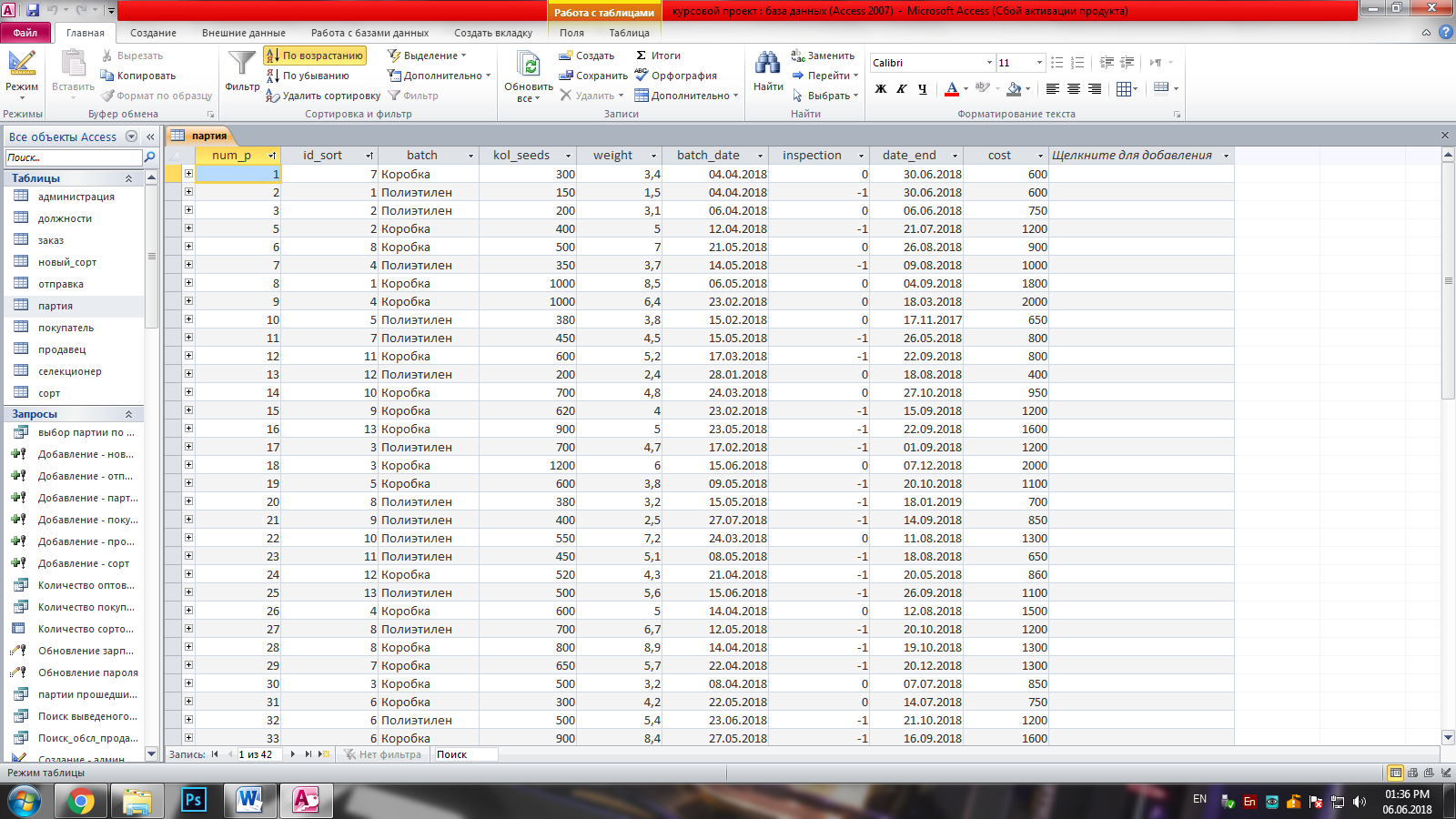


Рисунок 3.8 – Созданная таблица «Партия»



Рисунок 3.9 – Созданная таблица «Отправка»

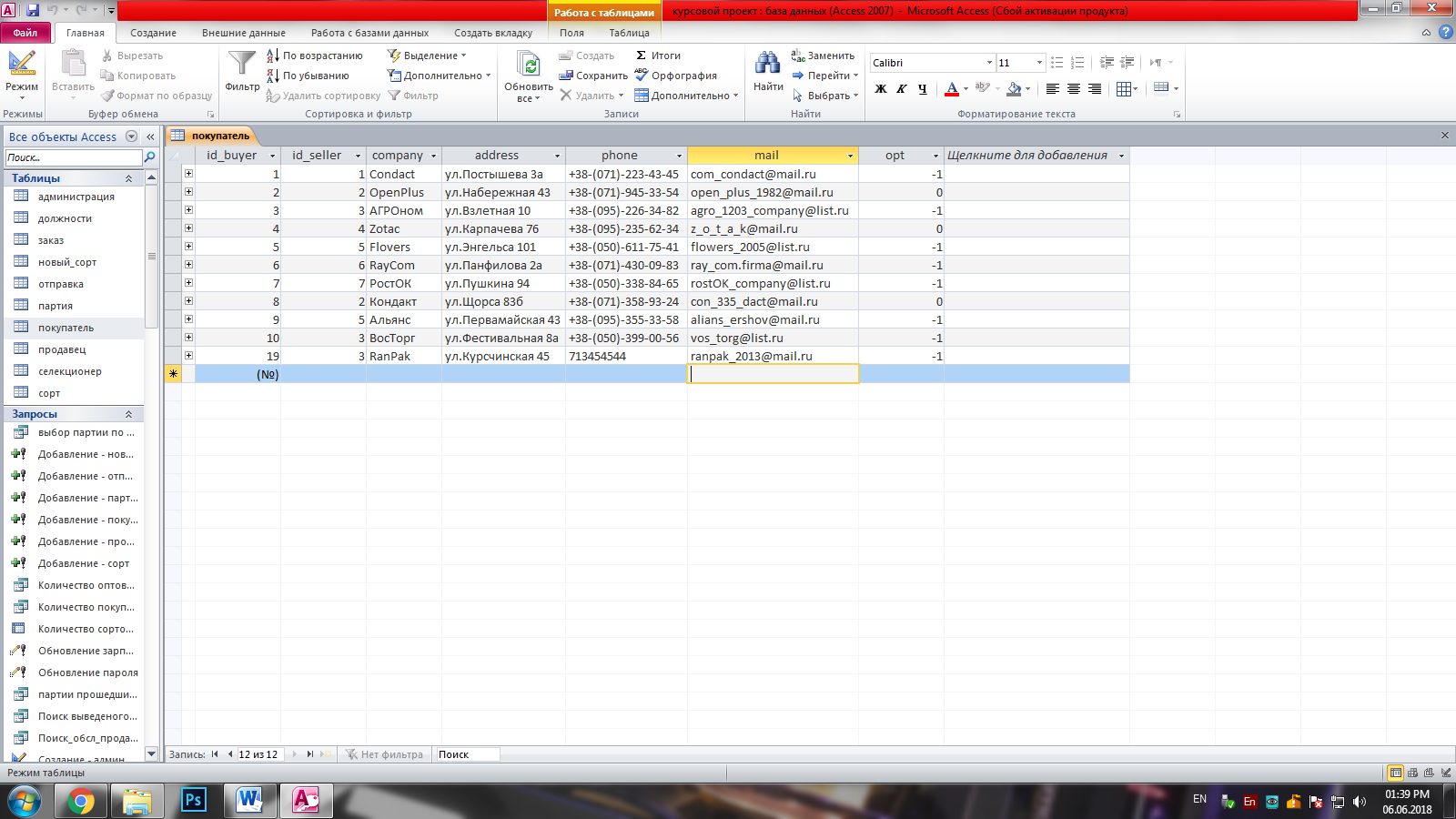


Рисунок 3.10 – Созданная таблица «Покупатель»

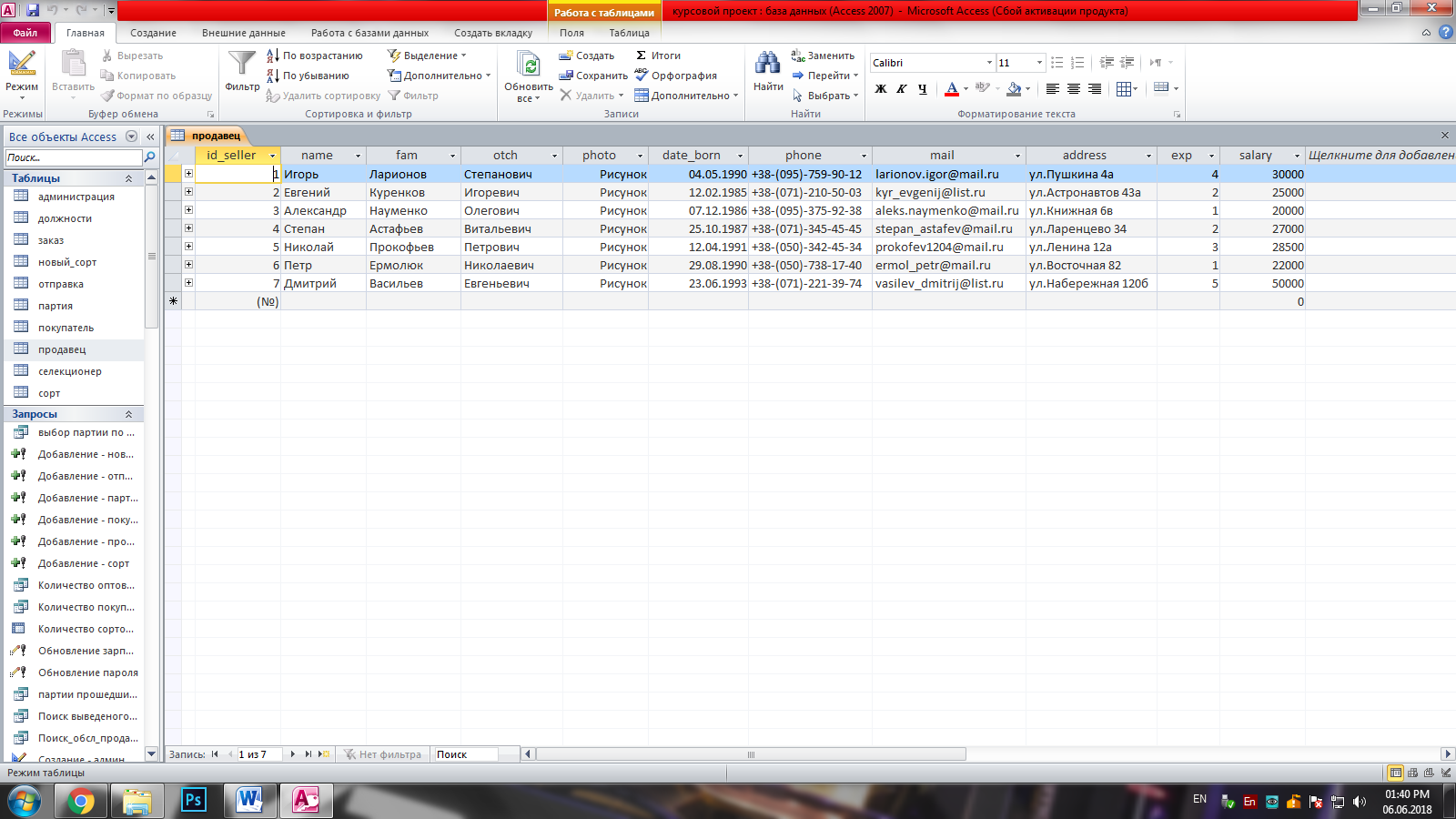


Рисунок 3.11 – Созданная таблица «Продавец»

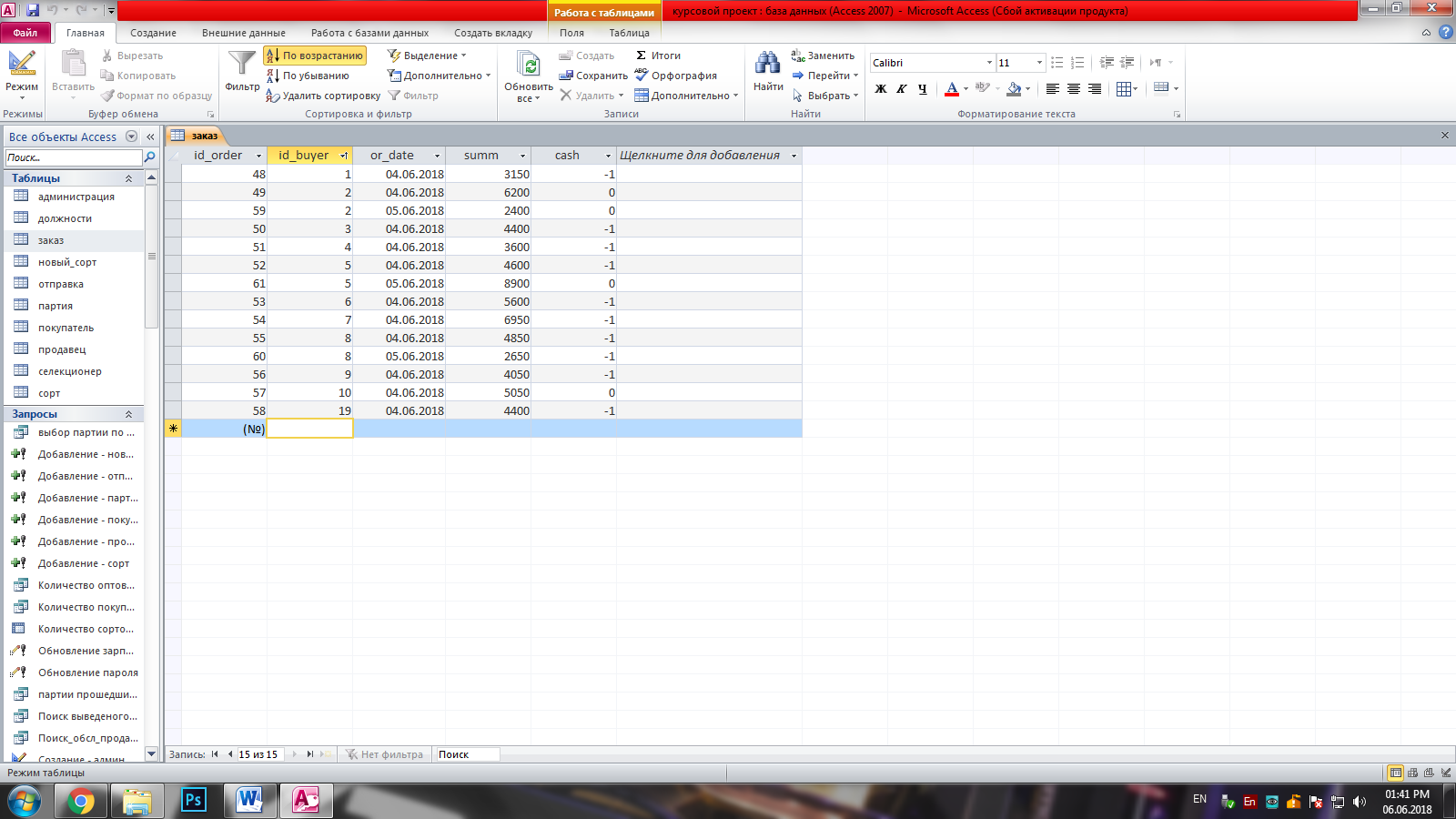


Рисунок 3.12 – Созданная таблица «Заказ»

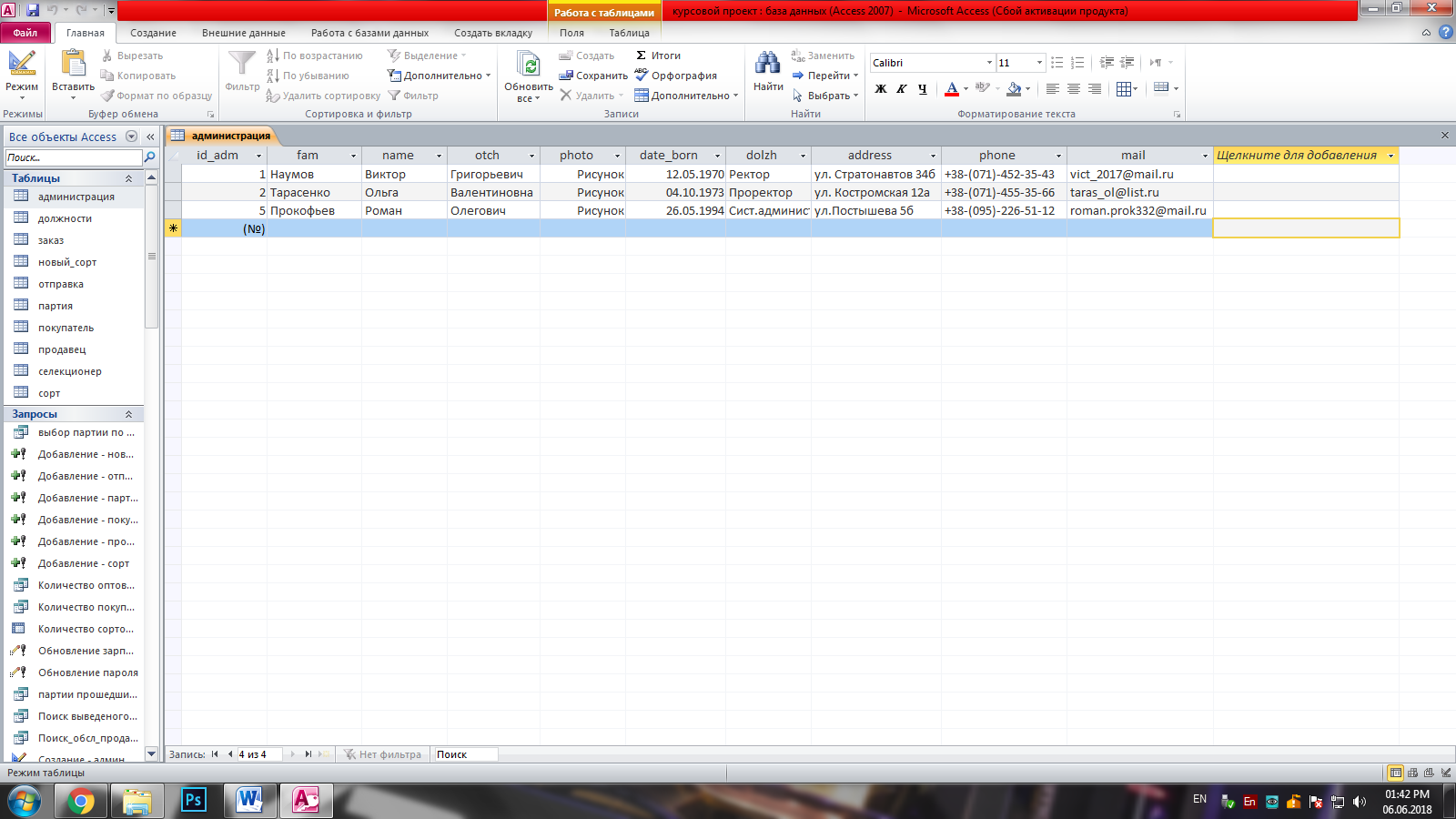


Рисунок 3.13 – Созданная таблица «Администрация»

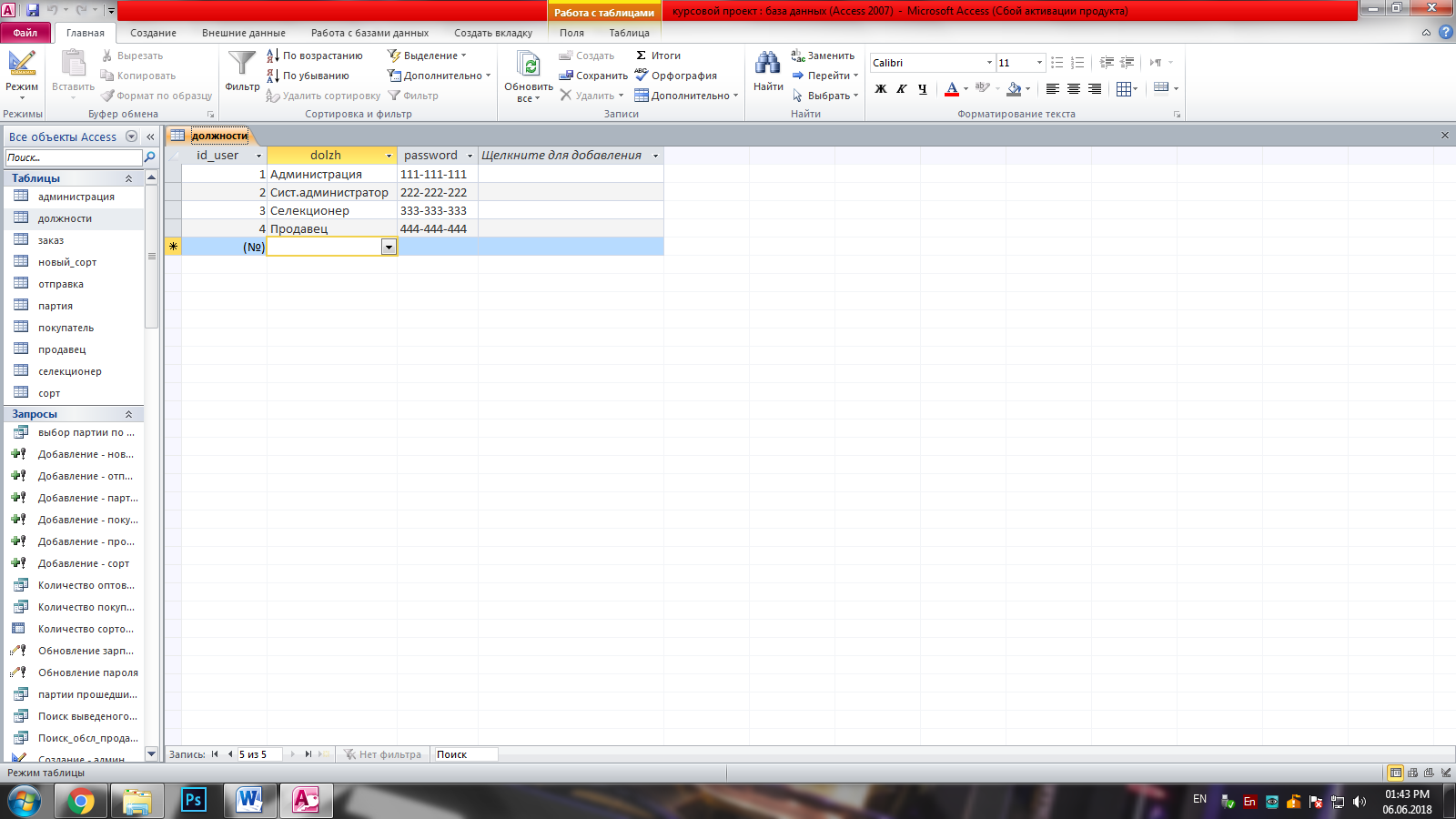


Рисунок 3.14 – Созданная таблица «Должности»

### Разработка запросов

Был создан запрос по трем таблицам, который выводит информацию о проданной партии на текущую дату.

SELECT партия.num\_p, отправка.buy\_date, сорт.name, партия.weight, партия.cost

FROM сорт INNER JOIN (партия INNER JOIN отправка ON партия.num\_p = отправка.num\_p) ON сорт.id\_sort = партия.id\_sort

WHERE (((отправка.buy\_date)=Date());

Был создан запрос, который выводит количество оптовых покупателей.

SELECT Count(\*)

FROM покупатель

WHERE opt=-1;

Был создан запрос по трем таблицам, который выводит количество оптовых покупателей определенной партии.

SELECT Count(покупатель.id\_buyer) AS количество\_покупателей

FROM покупатель INNER JOIN (партия INNER JOIN отправка ON партия.num\_p = отправка.num\_p) ON покупатель.id\_buyer = отправка.id\_buyer

WHERE (((партия.num\_p)=[Введеите номер партии]));

Был создан запрос на обновление пароля системного администратора.

UPDATE должности SET должности.[password] = "123456789"

WHERE (((должности.id\_user)=2));

Был создан запрос на обновление значения заработной платы продавца.

UPDATE продавец SET salary = 50000

WHERE id\_seller=7;

Был создан запрос на удаление ненужного сорта.

DELETE сорт.[name]

FROM сорт

WHERE (((сорт.[name])=[Введите название сорта]));

### Разработка форм

При разработке программного обеспечения была разработана 51 форма, главная форма приложения представлена на рисунке 3.15.

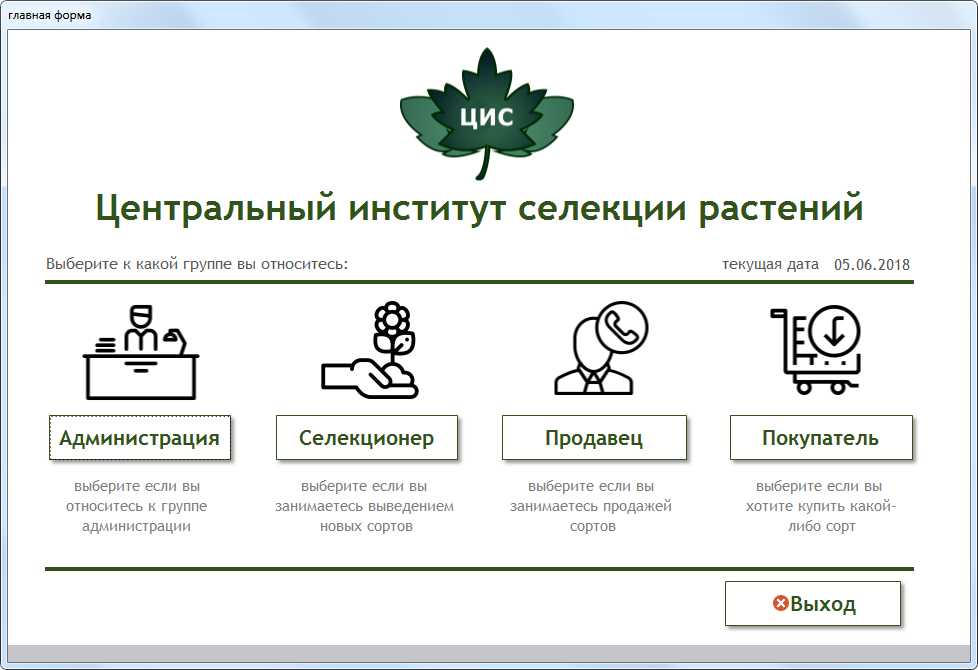


Рисунок 3.15 – Главная форма приложения

При разработке программного обеспечения была разработана форма для оформления заказа покупателя (рис.3.16).

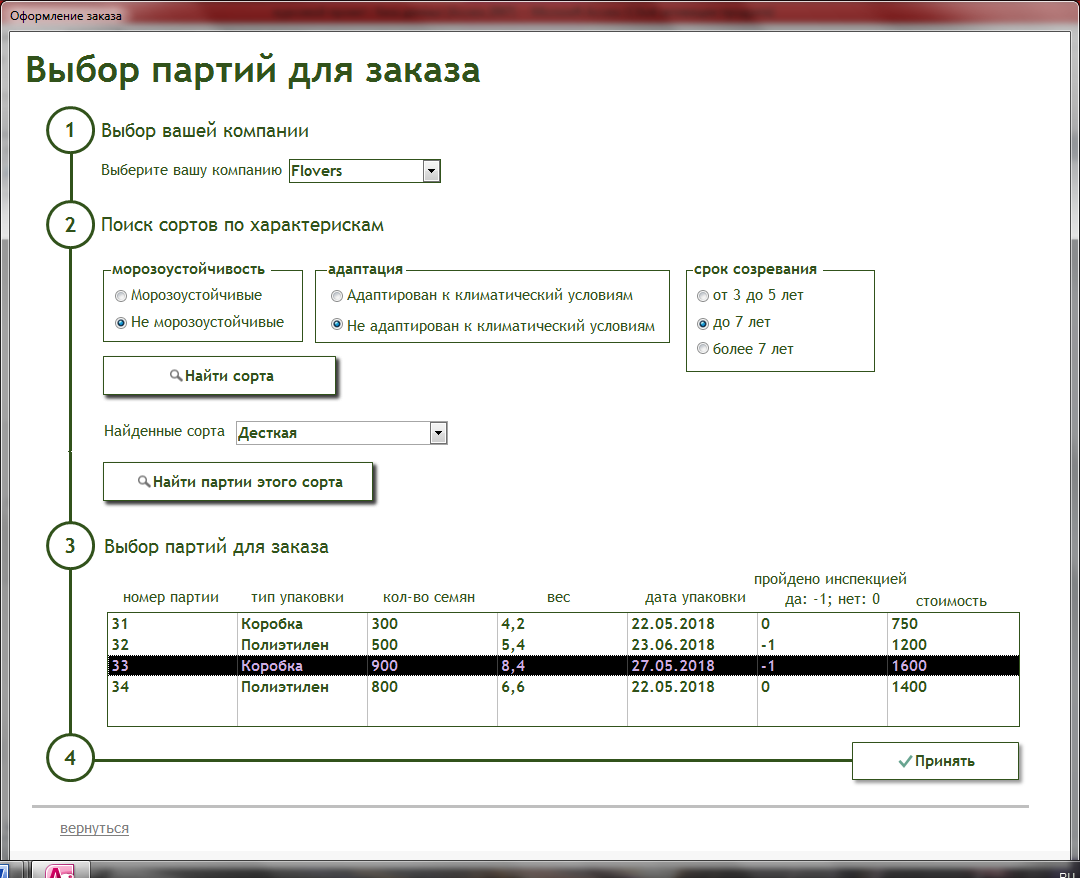


Рисунок 3.16 – Форма для выбора партий для покупки

Также после выбора покупателем партии открывается форма «Корзина» в которой выводится информация о выбранных партиях и выводится сумма заказа покупателя и количество выбранных партий (рис.3.17).

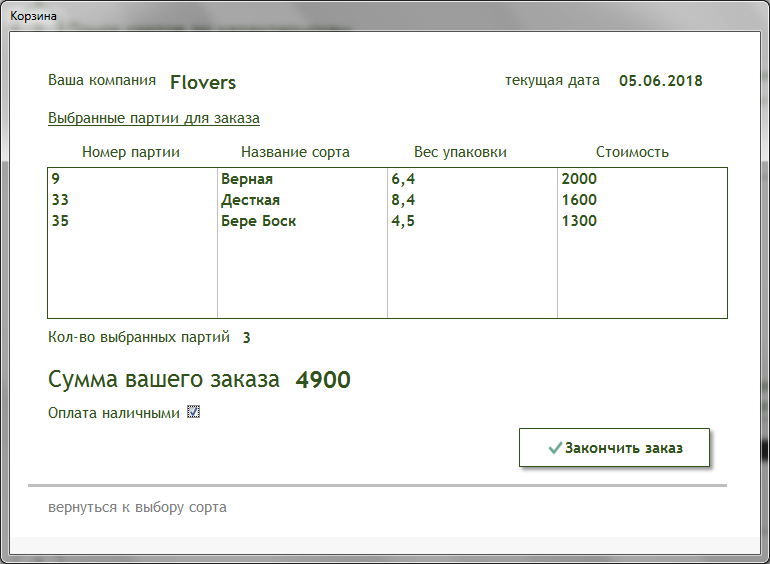


Рисунок 3.17 – Форма для оформления заказа

В ходе разработки программного обеспечения была разработана форма для сотрудников администрации, которая выводит основные статистические данные по всей информации в базе данных (рис.3.18).

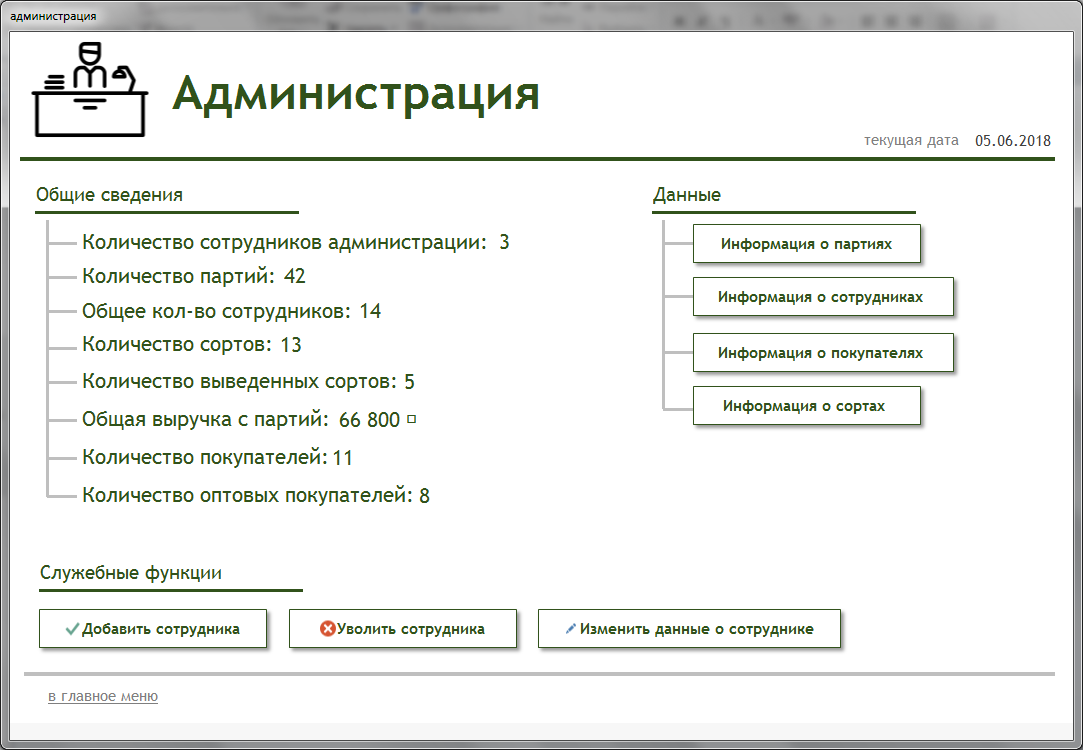


Рисунок 3.18 – Форма администрации

Также была создана форма для селекционеров (рис.3.19).

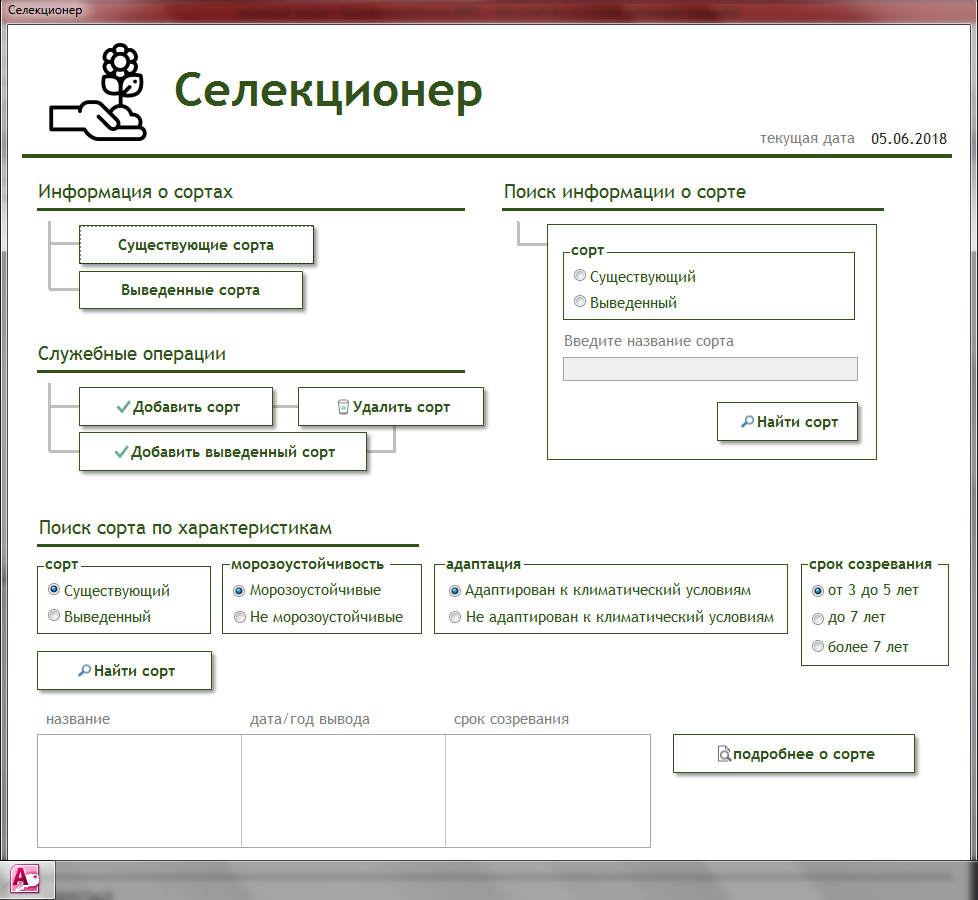


Рисунок 3.19 – Форма селекционера

Также была сформирована форма для продавцов, которая выводит основную информацию по продажам партий (рис.3.20).

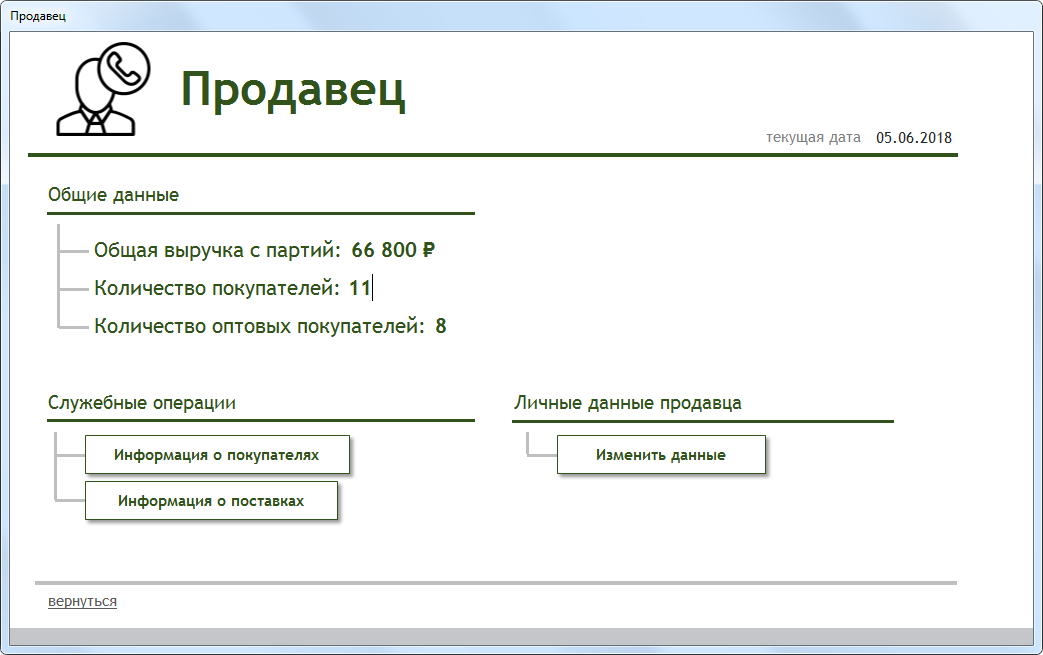


Рисунок 3.20 – Форма продавца

Форма для добавления покупателя представлена на рисунке 3.21.

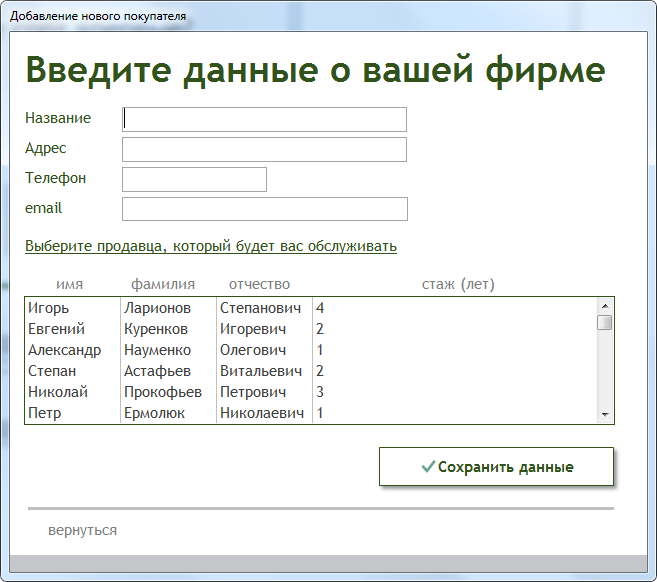


Рисунок 3.21 – Форма добавления покупателя

### Разработка отчетов

При разработки был разработан отчет «Чек», в котором выводится основная информация заказа покупателя (рис.3.22).

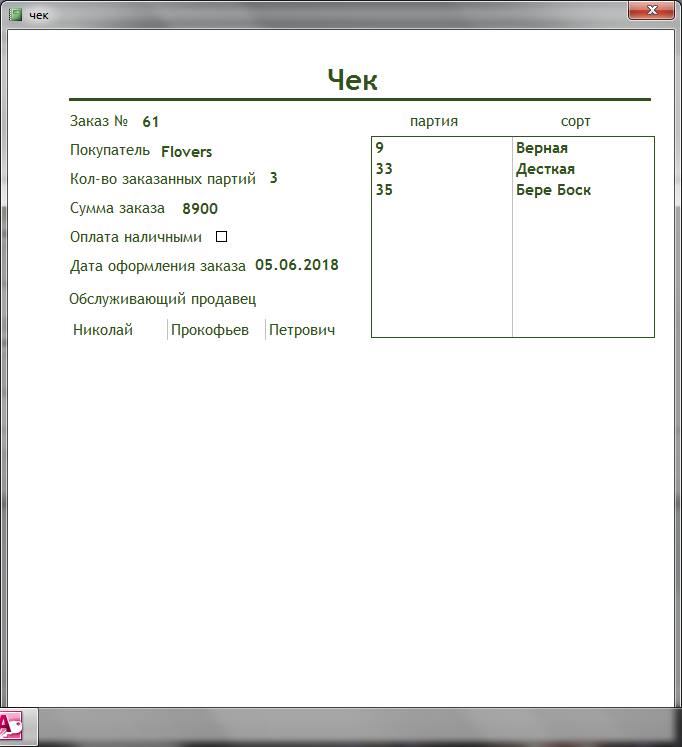


Рисунок 3.22 – Отчет «Чек»

Также был создан отчет о партиях, в котором выводится информация о партиях и покупателях, купившие данную партию (рис.3.23).

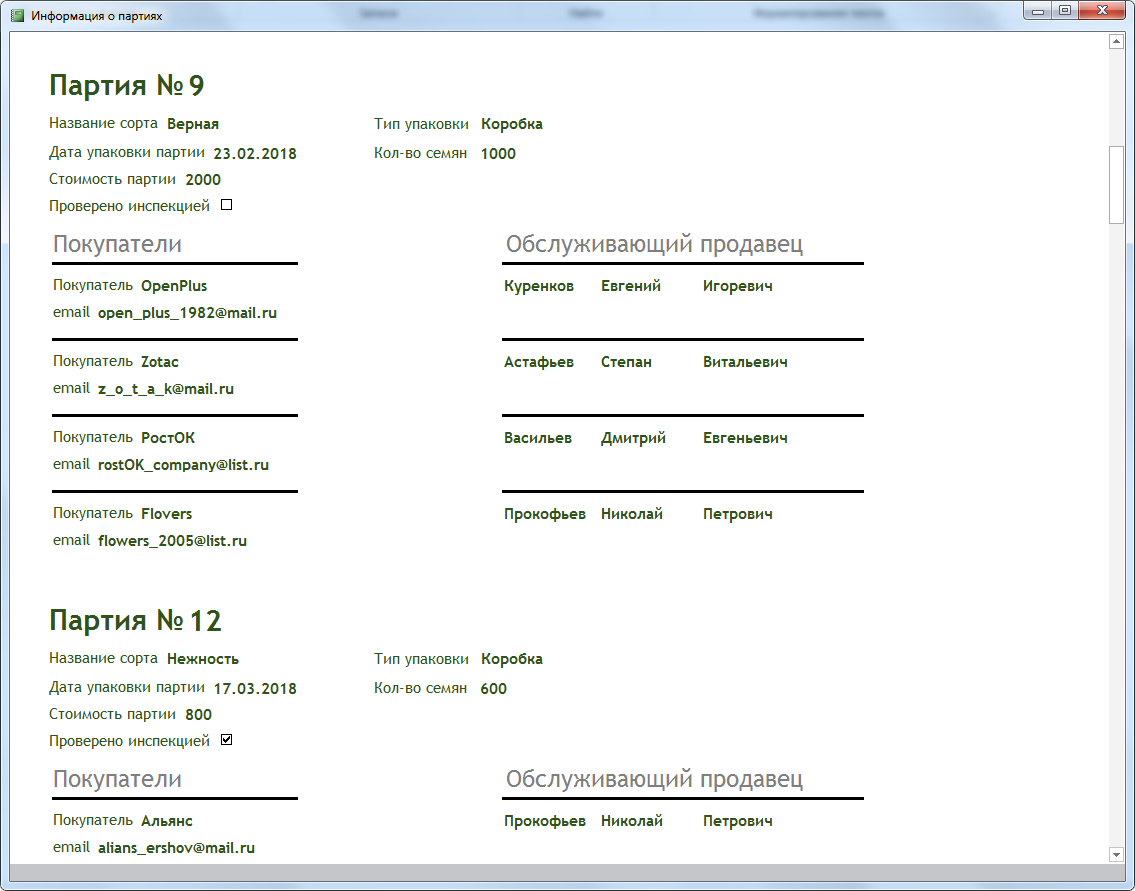


Рисунок 3.23 – Отчет «информация о партия»

Сформировал отчет для вывода данных о покупателях (рис.3.24).

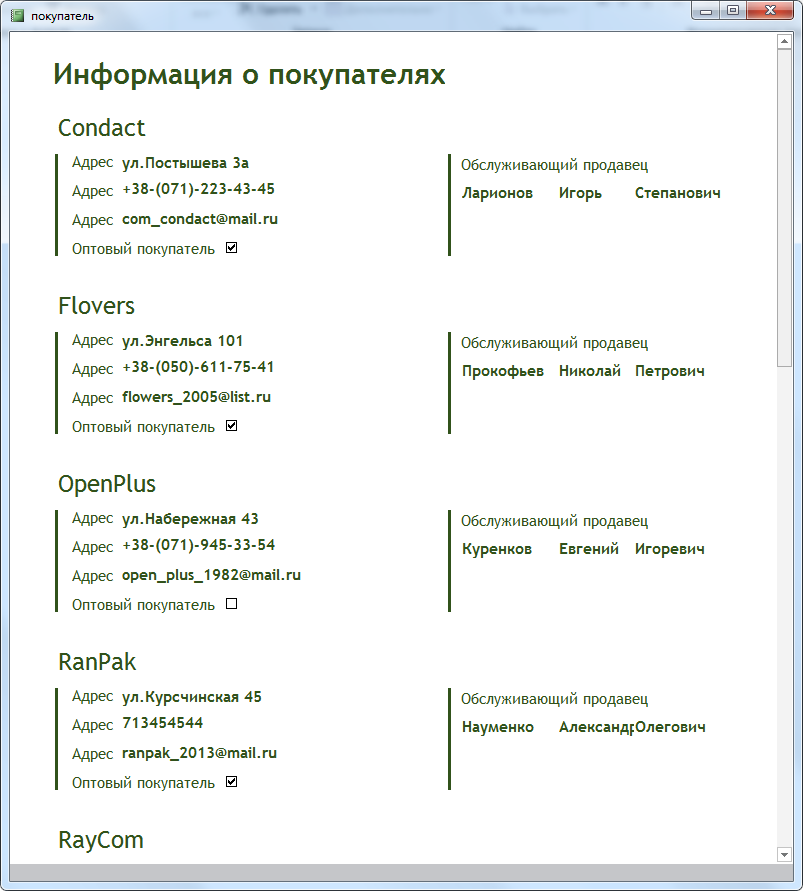


Рисунок 3.24 – Отчет «Покупатель»

Разработал отчет который выводит информацию о покупателях и выводит партии, которые были приобретены покупателем (рис.3.25).

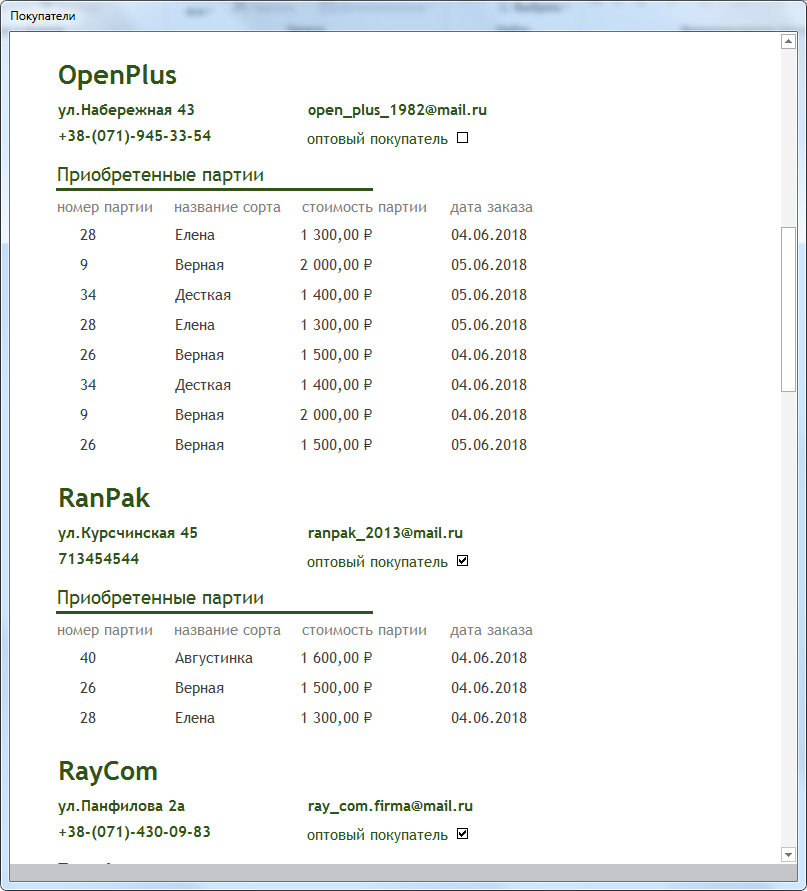


Рисунок 3.25 – Отчет «Покупатели»

# РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

При открытии базы данных откроется главная форма на которой можете выбрать к какой группе пользователя относитесь. Есть возможность зайти в базу данных как сотрудник администрации, системный администратор, селекционер, продавец или покупатель.

Выбрав группу «Администрация» необходимо будет ввести пароль для авторизации. Пароль по умолчанию для группы администрации – 111111111, пароль может быть изменен системным администратором.

Успешно авторизовавшись на форме предоставляются статистические данные по деятельности института и кроме этого выполнение таких функций как:

* добавление сотрудника;
* увольнение сотрудника;
* изменение данных о сотруднике;
* вывод данных о партиях;
* вывод информации о сотрудниках;
* вывод информации о покупателях;
* вывод информации о сортах.

При добавлении сотрудника откроется диалоговое окно в котором необходимо будет выбрать должность добавляемого сотрудника.

Выбрав определенную должность откроется окно для добавление сотрудника.

При увольнении сотрудника откроется диалоговое окно в котором необходимо будет выбрать должность увольняемого сотрудника. После выбора должности станет доступна секция для увольнения сотрудника.

При изменении данных сотрудника откроется диалоговое окно в котором необходимо будет выбрать должность изменяемого сотрудника.

Выбрав определенную должность откроется окно для выбора сотрудников выбранной должности. Затем после выбора сотрудника откроется окно для изменения данных о нем.

При выборе вывода данных о партиях откроется отчет с информациях о партиях и их покупателях.

При выборе данных о сотрудниках откроется окно выбора должности сотрудника, а затем вывод данных о сотрудниках выбранной должности.

При выборе вывода данных о покупателях откроется отчет с информацией о покупателях.

При выборе вывода данных о сортах откроется диалоговое окно для выбора необходимого сорта, существующего или выведенного. Затем после выбора группы сорта откроется форма для выбора сорта по определенным характеристикам.

Выбрав группу «Администрация» необходимо будет ввести пароль для авторизации. Если вы заходите как системный администратор необходимо будет ввести пароль 222222222, он же является паролем по умолчанию, пароль может быть изменен.

Успешно авторизовавшись на форме предоставляются функции для изменения системных, таких функции как:

* изменение данных сотрудника администрации;
* изменение данных селекционера;
* изменение данных продавца;
* изменение пароля;

После выбора изменения данных сотрудников администрации откроется список сотрудников администрации, выбрав необходимого сотрудника откроется окно для изменения данных о нем.

После выбора изменения данных селекционера откроется список селекционеров, выбрав необходимого сотрудниках откроется окно для изменения данных о нем.

После выбора изменения данных продавца откроется список продавцов, выбрав необходимого сотрудника откроется окно для изменения данных о нем.

После выбора изменения пароля откроется окно для выбора группы сотрудников для кого меняется пароль.

Выбрав группу «Селекционер» вам необходимо будет ввести пароль для авторизации. Пароль по умолчанию для группы селекционер – 333333333, пароль может быть изменен системным администратором.

Успешно авторизовавшись вам предоставляется поиск сорта по определенным характеристикам и кроме этого выполнение таких функций как:

* добавление сорта;
* добавление выведенного сорта;
* удаление сорта;
* вывод информации о существующий сортах;
* вывод информации о выведенных сортах.

При выборе добавления сорта откроется окно для ввода информации о новом сорте.

При выборе добавления выведенного сорта откроется окно для ввода информации о новом сорте.

При выборе удаления сорта откроется окно для удаления сорта, выведенного или существующего

При выборе вывода информации о существующих сортах откроется отчет с информацией о каждом существующем сорте.

При выборе вывода информации о выведенных сортах откроется отчет с информацией о каждом выведенном сорте.

Выбрав группу «Продавец» вам необходимо будет ввести пароль для авторизации. Пароль по умолчанию для группы продавец – 444444444, пароль может быть изменен системным администратором.

Успешно авторизовавшись вам предоставляются статистические данные по деятельности института и кроме этого выполнение таких функций как:

* вывод данных о партиях;
* вывод информации о сотрудниках;
* вывод информации о покупателях;
* изменение данных продавца.

При выборе вывода данных о партиях откроется отчет с информациях о партиях и их покупателях.

При выборе данных о сотрудниках откроется окно выбора должности сотрудника, а затем вывод данных о сотрудниках выбранной должности.

При выборе вывода данных о покупателях откроется отчет с информацией о покупателях.

После выбора изменения данных продавца откроется список продавцов, выбрав необходимого сотрудника откроется окно для изменения данных о нем.

Выбрав группу «Покупатель» вам откроется окно выполнения таких функций как:

* поиск информации о сорте;
* заказать сорт.

При выборе вывода данных о сортах откроется диалоговое окно для выбора необходимого сорта, существующего или выведенного. Затем после выбора группы сорта откроется форма для выбора сорта по определенным характеристикам.

При выборе заказа сорта вам откроется окно выбора необходимой партии, партии ищутся по характеристикам сорта который содержится в партии. После выбора необходимой партии откроется окно в котором отображаются выбранные партии для заказа, их количество и общая сумма заказа. После окончания заказа откроется отчет в котором указываются выбранные партии, сумма заказа, дата и остальные данные.

# ЗАКЛЮЧЕНИЕ

При разработке программного обеспечения для института селекции растений был проведен анализ предметной области в которой было произведено обозначение границ моделируемой системы, построение контекстной диаграммы и её декомпозиция.

Также было сформировано техническое задание и проведен дальнейший его анализ, а затем проектирование базы данных, которое включает в себя концептуальное проектирование, логическое проектирование, физическое проектирование и сама разработка приложения, создания таблиц, запросов, форм и отчетов.

После окончания разработки программного обеспечения было установлено руководство пользователя в котором описана работа приложения и все выполняемые функции.

# ПЕРЕЧЕНЬ ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Интернет журнал. Декоративный сад [электронный ресурс] / Режим доступа: www/URL: [http://www.udec.ru/derevo/grusha-sorta.php / 10.05.2018](http://www.udec.ru/derevo/grusha-sorta.php%20/%2010.05.2018) г.
2. Про ферму [электронный ресурс] / Режим доступа: www/URL:  [http://profermu.com/sad/derevia/grushi/sorta-g/sorta-g.html / 12.05.2018](%20http://profermu.com/sad/derevia/grushi/sorta-g/sorta-g.html%20/%2012.05.2018) г.
3. Про ферму [электронный ресурс] / Режим доступа: www/URL:  [http://profermu.com/sad/derevia/grushi/sorta-g/sorta-g.html / 12.05.2018](%20http://profermu.com/sad/derevia/grushi/sorta-g/sorta-g.html%20/%2012.05.2018) г.
4. AGRONOMU[, журнал агронома №1](%20http://profermu.com/sad/derevia/grushi/sorta-g/sorta-g.html%20/%2012.05.2018) [электронный ресурс] / Режим доступа: www/URL: [https://agronomu.com/bok/335-opisaniya-populyarnyh-sortov-grush-foto.html / 12.05.2018](%20https://agronomu.com/bok/335-opisaniya-populyarnyh-sortov-grush-foto.html%20/%2012.05.2018) г.
5. Село.guru, для тех, кого тянет к земле [электронный ресурс] / Режим доступа: www/URL: https://selo.guru/sadovodstvo/grushi/sorta-g/osennie-g/bere-bosk.html [/ 12.05.2018](%20https://agronomu.com/bok/335-opisaniya-populyarnyh-sortov-grush-foto.html%20/%2012.05.2018) г.
6. Бурмистров Л.А. Грушевый сад [Текст]: [Леонид Бурмистров](https://www.ozon.ru/person/249566/) «Лениздат» 1991 г. – 128 с.

Приложение А

Техническое задание

Введение

Настоящее техническое задание распространяется на разработку программы для института селекции растений, занимающийся выведением и продажей сортов.

## Наименование и область применения

## Наименование

Прикладная программа «Селекция растений»

## Область применения

Данная разработка предназначена для применения в институте селекции растений.

## Основание для разработки

## Основание

Программа разрабатывается на основе необходимости ведения учета выведения сортов растений и контроля продаж сортов семян.

## Тема разработки

Разработка прикладной программы «Селекция растений».

## Исполнитель:

Ермоленко Евгений

## Соисполнители

Нет.

## Назначение разработки

Программа предназначена для использования администрации института и для продажи сортов потенциальным покупателям.

## Технические требования к программе или программному изделию

## Требования к функциональным характеристикам

## Функциональные требования

Программа должна обеспечивать возможность выполнения следующих функций:

* Возможность добавления сотрудника в базу данных;
* Возможность удаления сотрудника из базы данных;
* Возможность добавления покупателя в базу данных;
* Добавление сорта в базу данных;
* Добавление выведенного сорта в базу данных;
* Удаление сорта и базы данных;
* Поиск сорта по заданным характеристикам;
* Возможность выбора покупателю определенного сорта по заданным характеристикам для покупки;
* Возможность выбора определенной партии определенного сорта для покупки;
* Реализовать разгруппировку данных по группе пользователя;
* Реализовать ввод пароля для каждой группы пользователя;
* Реализовать изменения пароля для входа в базу данных каждой группы пользователя;
* Реализовать изменения данных сотрудника каждой группы;
* Вывод статистических данных для пользователя каждой группы;

## Исходные данные

* Данные о сорте;

## Требования к надежности

В разрабатываемой системе необходимо предусмотреть следующие меры защиты:

* контроль вводимой информации;
* разграничение прав доступа;
* защиту от несанкционированного доступа посредствам паролей.

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем операционной системы, не должно превышать времени, необходимого на перезагрузку операционной системы и запуск программы.

Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем (крахом) операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

## Условия эксплуатации

Минимальное количество персонала, требуемого для работы программы, должно составлять не менее 3 штатных единиц – сотрудник администрации института, системный администратор и потенциальный покупатель.

Системный программист должен иметь минимум среднее техническое образование.

В перечень задач, выполняемых системным программистом, должны входить:

* задача поддержания работоспособности технических средств;
* задачи установки (инсталляции) и поддержания работоспособности системных программных средств - операционной системы;
* задача установки (инсталляции) программы.

## Требования к составу и параметрам технических средств

В состав технических средств должен входить IBM-совместимый персональный компьютер (ПЭВМ), включающий в себя:

* процессор Intel Core 2 Dou и выше с тактовой частотой, 2,7 МГц , не менее;
* оперативную память объемом, 2 Гб, не менее;
* жесткий диск объемом 40 Гб, и выше;
* манипулятор типа «мышь».

## Требования к информационной и программной совместимости

Системные программные средства, используемые программой, должны быть представлены локализованной версией операционной системы Windows 7-10.

## Требования к маркировке и упаковке

Не предъявляются.

## Требования к транспортированию и хранению

Не предъявляются.

## Специальные требования

Программа должна быть снабжена убодным пользовательским интерфейсом.

## Технико-экономические показатели

Ориентировочная экономическая эффективность не рассчитывается.

Предполагаемое число использования программы в год – ежедневное использование программы, за исключением выходных дней, в течение рабочего дня.

## Стадии и этапы разработки

## Стадии разработки

Разработка должна быть проведена в три стадии:

* разработка технического задания;
* рабочее проектирование;
* внедрение.

## Этапы разработки

На стадии разработки технического задания должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

* изучение предметной области
* проектирование системы;
* разработка программы;
* разработка программной документации;
* тестирование и отладка программы;
* внедрение программы.

## Порядок контроля и приемки

После проведения испытаний в полном объеме, на основании «Протокола испытаний» утверждают «Свидетельство о приемке», после чего программный продукт считается принятым.

Приложение Б

VBA код для формы выбора партии для заказа

Option Compare Database  
Private Sub adapt\_Click()  
name\_sort.Enabled = False  
l3.BorderColor = RGB(49, 82, 27)  
butt1.Enabled = True  
End Sub  
  
Private Sub apply\_Click()  
Module1.nump = vibor\_p.Value  
Module1.idb = num\_b.Value  
  
Application.SetOption "Confirm Action Queries", False  
DoCmd.RunSQL ("INSERT INTO отправка (num\_p,id\_buyer,buy\_date) VALUES (" & nump & "," & idb & ",Date())")  
MsgBox "Партия добавлена в корзину"  
DoCmd.OpenForm "Корзина", acNormal  
  
el3.Enabled = False  
el4.Enabled = False  
el3.BorderColor = RGB(191, 191, 191)  
el4.BorderColor = RGB(191, 191, 191)  
l2.BorderColor = RGB(191, 191, 191)  
l3.BorderColor = RGB(191, 191, 191)  
l4.BorderColor = RGB(191, 191, 191)  
l5.BorderColor = RGB(191, 191, 191)  
l6.BorderColor = RGB(191, 191, 191)  
l7.BorderColor = RGB(191, 191, 191)  
butt1.Enabled = False  
name\_sort.Enabled = False  
butt2.Enabled = False  
vibor\_p.Enabled = False  
  
End Sub  
  
Private Sub butt1\_Click()  
name\_sort.Enabled = True  
  
If moroz.Value = 1 And adapt.Value = 1 And srok.Value = 1 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=-1 and [сорт].[adapt]=-1 and [сорт].[period]<=5;"  
  
ElseIf moroz.Value = 1 And adapt.Value = 1 And srok.Value = 2 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=-1 and [сорт].[adapt]=-1 and [сорт].[period]<=7;"  
  
ElseIf moroz.Value = 1 And adapt.Value = 1 And srok.Value = 3 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=-1 and [сорт].[adapt]=-1 and [сорт].[period]>7;"

ElseIf moroz.Value = 2 And adapt.Value = 1 And srok.Value = 1 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=0 and [сорт].[adapt]=-1 and [сорт].[period]<=5;"  
  
ElseIf moroz.Value = 2 And adapt.Value = 1 And srok.Value = 2 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=0 and [сорт].[adapt]=-1 and [сорт].[period]<=7;"  
  
ElseIf moroz.Value = 2 And adapt.Value = 1 And srok.Value = 3 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=0 and [сорт].[adapt]=-1 and [сорт].[period]>7;"  
  
ElseIf moroz.Value = 1 And adapt.Value = 2 And srok.Value = 1 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=-1 and [сорт].[adapt]=0 and [сорт].[period]<=5;"  
  
ElseIf moroz.Value = 1 And adapt.Value = 2 And srok.Value = 2 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=-1 and [сорт].[adapt]=0 and [сорт].[period]<=7;"  
  
ElseIf moroz.Value = 1 And adapt.Value = 2 And srok.Value = 3 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=-1 and [сорт].[adapt]=0 and [сорт].[period]>7;"  
  
ElseIf moroz.Value = 2 And adapt.Value = 2 And srok.Value = 1 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=0 and [сорт].[adapt]=0 and [сорт].[period]<=5;"  
  
ElseIf moroz.Value = 2 And adapt.Value = 2 And srok.Value = 2 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=0 and [сорт].[adapt]=0 and [сорт].[period]<=7;"  
  
ElseIf moroz.Value = 2 And adapt.Value = 2 And srok.Value = 3 Then  
name\_sort.RowSource = "SELECT [сорт].[id\_sort], [сорт].[name] FROM [сорт] WHERE [сорт].[frost]=0 and [сорт].[adapt]=0 and [сорт].[period]>7;"  
  
Else  
MsgBox "Сорта с такими характеристиками нет..."  
  
End If  
  
name\_sort.Enabled = True  
  
End Sub  
  
Private Sub butt2\_Click()  
  
el3.BorderColor = RGB(49, 82, 27)  
el3.ForeColor = RGB(49, 82, 27)  
n3.ForeColor = RGB(49, 82, 27)  
  
Dim a As Recordset  
Dim name As String  
Dim id As Integer  
  
Dim sql As String  
  
Set a = CurrentDb.OpenRecordset("партия", dbOpenDynaset)  
  
id = name\_sort.Value  
'MsgBox "sort" + Str(id)  
  
a.MoveFirst  
  
Do Until a.EOF  
  
If a![id\_sort] = id Then  
nump = a![num\_p]  
'MsgBox "partia" + Str(nump)  
End If  
  
sql = "SELECT [партия].[num\_p], [партия].[batch], [партия].[kol\_seeds], [партия].[weight], [партия].[batch\_date], [партия].[inspection], [партия].[cost] " & "FROM партия " & "WHERE (((партия.id\_sort)=" & id & "));"  
vibor\_p.RowSource = sql  
  
a.MoveNext  
  
Loop  
  
vibor\_p.Enabled = True  
  
  
End Sub  
  
Private Sub moroz\_Click()  
name\_sort.Enabled = False  
l2.BorderColor = RGB(49, 82, 27)  
butt1.Enabled = True  
  
End Sub  
  
Private Sub Кнопка0\_Click()  
  
Module1.nump = vibor\_p.Value  
Module1.idb = num\_b.Value  
  
Application.SetOption "Confirm Action Queries", False  
DoCmd.RunSQL ("INSERT INTO отправка (num\_p,id\_buyer,buy\_date) VALUES (" & nump & "," & idb & ",Date())")  
MsgBox "Партия добавлена в корзину"  
DoCmd.OpenForm "Корзина", acNormal  
  
End Sub  
  
Private Sub name\_sort\_AfterUpdate()  
l2.BorderColor = RGB(49, 82, 27)  
l3.BorderColor = RGB(49, 82, 27)  
l4.BorderColor = RGB(49, 82, 27)  
l5.BorderColor = RGB(49, 82, 27)  
butt2.Enabled = True  
  
End Sub  
  
Private Sub num\_b\_AfterUpdate()  
  
l1.BorderColor = RGB(49, 82, 27)  
el2.BorderColor = RGB(49, 82, 27)  
el2.ForeColor = RGB(49, 82, 27)  
n2.ForeColor = RGB(49, 82, 27)  
moroz.Enabled = True  
adapt.Enabled = True  
srok.Enabled = True  
  
End Sub  
  
Private Sub srok\_Click()  
name\_sort.Enabled = False  
l4.BorderColor = RGB(49, 82, 27)  
butt1.Enabled = True  
End Sub  
  
Private Sub vibor\_p\_AfterUpdate()  
  
l6.BorderColor = RGB(49, 82, 27)  
l7.BorderColor = RGB(49, 82, 27)  
el4.BorderColor = RGB(49, 82, 27)  
el4.ForeColor = RGB(49, 82, 27)  
apply.Enabled = True  
  
End Sub

Код для формы «Корзина»

Option Compare Database  
  
Private Sub Form\_Load()  
  
Dim a As Recordset  
Set a = CurrentDb.OpenRecordset("отправка", dbOpenDynaset)  
Dim l As Recordset  
Set l = CurrentDb.OpenRecordset("покупатель", dbOpenDynaset)  
  
Dim cost, k As Integer  
Dim par As Recordset  
Dim ot As Recordset  
Dim c As String  
  
a.MoveFirst  
Do Until a.EOF  
  
If a![id\_buyer] = Int(idb) Then  
  
spisok.RowSource = "SELECT [партия].[num\_p], [сорт].[name], [партия].[weight], [партия].[cost] FROM (сорт INNER JOIN партия ON сорт.id\_sort = партия.id\_sort) INNER JOIN отправка ON партия.num\_p = отправка.num\_p WHERE отправка.id\_buyer=" & idb & " and отправка.buy\_date=Date()"  
  
End If  
  
a.MoveNext  
  
Loop  
  
l.MoveFirst  
Do Until l.EOF  
  
If l![id\_buyer] = Int(idb) Then  
  
c = l![company]  
  
End If  
  
l.MoveNext  
  
Loop  
  
pok.Value = c  
  
Set par = CurrentDb.OpenRecordset("партия", dbOpenDynaset)  
Set ot = CurrentDb.OpenRecordset("отправка", dbOpenDynaset)  
  
'поиск стоимости выбранной партии  
par.MoveFirst  
Do Until par.EOF  
  
If par![num\_p] = Int(nump) Then  
  
cost = par![cost]  
  
End If  
  
par.MoveNext  
  
Loop  
  
Module1.a = cost + Module1.summ  
summ.Value = Module1.a  
Module1.summ = Module1.a  
  
Dim b As Recordset  
Dim vs\_k As Integer  
  
Set b = CurrentDb.OpenRecordset("отправка", dbOpenDynaset)  
  
vs\_k = 0  
  
b.MoveFirst  
Do Until b.EOF  
  
If b![buy\_date] = date And b![id\_buyer] = Int(idb) Then  
  
vs\_k = vs\_k + 1  
  
End If  
  
b.MoveNext  
  
Loop  
  
vs\_kol.Value = vs\_k  
Module1.kol = vs\_k  
  
If vs\_k >= 3 Then  
Module1.opt = -1  
ElseIf vs\_k < 3 Then  
Module1.opt = 0  
End If  
  
  
End Sub  
  
Private Sub Кнопка25\_Click()  
  
If opt.Value = 0 Then  
Module1.cash = 0  
ElseIf opt.Value = -1 Then  
Module1.cash = -1  
End If  
Application.SetOption "Confirm Action Queries", False  
DoCmd.RunSQL ("INSERT INTO заказ (id\_buyer,or\_date,summ,cash) VALUES (" & Module1.idb & ", Date()," & Module1.summ & "," & Module1.cash & ")")  
Application.SetOption "Confirm Action Queries", False  
DoCmd.RunSQL ("UPDATE покупатель SET opt=" & Module1.opt & " WHERE id\_buyer=" & Module1.idb & " ")  
  
DoCmd.OpenReport "Чек", acViewReport  
DoCmd.OpenForm "Уведомление о заказе", acNormal  
  
DoCmd.Close acForm, "Корзина", acSaveYes  
DoCmd.Close acForm, "Оформление заказа", acSaveYes  
  
End Sub

VBA код для отчета «Чек»

Option Compare Database  
  
Private Sub Report\_Load()  
Dim q, s As Integer  
Dim c, sql, d As String  
  
Dim a As Recordset  
Dim b As Recordset  
Set a = CurrentDb.OpenRecordset("заказ", dbOpenDynaset)  
Set b = CurrentDb.OpenRecordset("покупатель", dbOpenDynaset)  
  
'поиск номера заказа  
a.MoveFirst  
Do Until a.EOF  
  
If a![id\_buyer] = Int(idb) And a![summ] = Int(summ) Then  
  
q = a![id\_order]  
  
End If  
  
a.MoveNext  
  
Loop  
  
'поиск суммы заказа  
a.MoveFirst  
Do Until a.EOF  
  
If a![id\_order] = q Then  
  
s = a![summ]  
  
End If  
  
a.MoveNext  
  
Loop  
  
b.MoveFirst  
Do Until b.EOF  
  
If b![id\_buyer] = Int(idb) Then  
  
MsgBox "1"  
c = b![company]  
  
End If  
  
b.MoveNext  
  
Loop  
  
sql = "SELECT продавец.name, продавец.fam, продавец.otch FROM продавец INNER JOIN покупатель ON продавец.id\_seller = покупатель.id\_seller WHERE покупатель.id\_buyer=" & idb & ""  
  
'поиск даты заказа  
a.MoveFirst  
Do Until a.EOF  
  
If a![id\_buyer] = Int(idb) And a![summ] = Int(summ) Then  
  
d = a![or\_date]  
  
End If  
  
a.MoveNext  
  
Loop  
  
Dim k As Recordset  
Set k = CurrentDb.OpenRecordset("отправка", dbOpenDynaset)  
  
k.MoveFirst  
Do Until k.EOF  
  
If k![id\_buyer] = Int(idb) Then  
  
sp.RowSource = "SELECT партия.num\_p,сорт.name FROM (сорт INNER JOIN партия ON сорт.id\_sort = партия.id\_sort) INNER JOIN отправка ON партия.num\_p = отправка.num\_p WHERE отправка.id\_buyer=" & idb & " and отправка.buy\_date=Date()"  
  
End If  
  
k.MoveNext  
  
Loop  
  
z.Value = Str(q)  
pok.Value = c  
p.Value = Str(s)  
kol.Value = Str(Module1.kol)  
cash.Value = Module1.cash  
pr.RowSource = sql  
  
End Sub

Код VBA для формы удаления сорта

Option Compare Database  
  
Private Sub Form\_Load()  
sp.RowSource = "SELECT [сорт].[name] FROM [сорт];"  
End Sub  
  
Private Sub sorts\_AfterUpdate()  
  
If sorts.Value = 1 Then  
sp.RowSource = "SELECT [сорт].[name] FROM [сорт];"  
ElseIf sorts.Value = 2 Then  
sp.RowSource = "SELECT [новый\_сорт].[name] FROM [новый\_сорт];"  
End If  
  
End Sub  
  
Private Sub sorts\_Click()  
  
sp.Enabled = True  
sp.Value = ""  
  
End Sub  
  
Private Sub Кнопка13\_Click()  
Dim a As String  
  
If sorts.Value = 1 Then  
a = sp.Value  
  
DoCmd.RunSQL ("DELETE FROM сорт WHERE name='" + a + "' ")  
  
ElseIf sorts.Value = 2 Then  
  
DoCmd.RunSQL ("DELETE FROM новый\_сорт WHERE name='" + a + "' ")  
  
End If  
  
End Sub

Код VBA для формы изменения пароля

Option Compare Database  
  
Private Sub Form\_Load()  
sp.Value = "Администрация"  
pass\_adm.Enabled = True  
pass\_sist.Enabled = False  
pass\_sel.Enabled = False  
pass\_prod.Enabled = False  
End Sub  
  
Private Sub sp\_Change()  
  
If sp.Value = "Администрация" Then  
pass\_adm.Enabled = True  
pass\_sist.Enabled = False  
pass\_sel.Enabled = False  
pass\_prod.Enabled = False  
  
ElseIf sp.Value = "Сист.администратор" Then  
pass\_adm.Enabled = False  
pass\_sist.Enabled = True  
pass\_sel.Enabled = False  
pass\_prod.Enabled = False  
  
ElseIf sp.Value = "Селекционер" Then  
pass\_adm.Enabled = False  
pass\_sist.Enabled = False  
pass\_sel.Enabled = True  
pass\_prod.Enabled = False  
  
ElseIf sp.Value = "Продавец" Then  
pass\_adm.Enabled = False  
pass\_sist.Enabled = False  
pass\_sel.Enabled = False  
pass\_prod.Enabled = True  
  
End If  
  
End Sub  
  
Private Sub Кнопка11\_Click()  
Dim a As String  
  
If sp.Value = "Администрация" Then  
a = pass\_adm.Value  
DoCmd.RunSQL ("UPDATE должности SET password=" + a + " WHERE id\_user=1 ")  
  
ElseIf sp.Value = "Сист.администратор" Then  
a = pass\_sist.Value  
DoCmd.RunSQL ("UPDATE должности SET password=" + a + " WHERE id\_user=2 ")  
  
ElseIf sp.Value = "Селекционер" Then  
a = pass\_sel.Value  
DoCmd.RunSQL ("UPDATE должности SET password=" + a + " WHERE id\_user=3 ")  
  
ElseIf sp.Value = "Продавец" Then  
a = pass\_prod.Value  
DoCmd.RunSQL ("UPDATE должности SET password=" + a + " WHERE id\_user=4 ")  
  
End If  
  
End Sub

Код VBA для формы увольнения сотрудника

Option Compare Database  
  
Private Sub Form\_Load()  
sp.Value = " "  
pass\_adm.Enabled = True  
pass\_sist.Enabled = False  
pass\_sel.Enabled = False  
pass\_prod.Enabled = False  
End Sub  
  
Private Sub sp\_Change()  
  
If sp.Value = " " Then  
pass\_adm.Enabled = True  
pass\_sist.Enabled = False  
pass\_sel.Enabled = False  
pass\_prod.Enabled = False  
  
ElseIf sp.Value = " . " Then  
pass\_adm.Enabled = False  
pass\_sist.Enabled = True  
pass\_sel.Enabled = False  
pass\_prod.Enabled = False  
  
ElseIf sp.Value = " " Then  
pass\_adm.Enabled = False  
pass\_sist.Enabled = False  
pass\_sel.Enabled = True  
pass\_prod.Enabled = False  
  
ElseIf sp.Value = " " Then  
pass\_adm.Enabled = False  
pass\_sist.Enabled = False  
pass\_sel.Enabled = False  
pass\_prod.Enabled = True  
  
End If  
  
End Sub  
  
Private Sub 11\_Click()  
Dim a As String  
  
If sp.Value = " " Then  
a = pass\_adm.Value  
DoCmd.RunSQL ("UPDATE SET password=" + a + " WHERE id\_user=1 ")  
  
ElseIf sp.Value = " . " Then  
a = pass\_sist.Value  
DoCmd.RunSQL ("UPDATE SET password=" + a + " WHERE id\_user=2 ")  
  
ElseIf sp.Value = " " Then  
a = pass\_sel.Value  
DoCmd.RunSQL ("UPDATE SET password=" + a + " WHERE id\_user=3 ")  
  
ElseIf sp.Value = " " Then  
a = pass\_prod.Value  
DoCmd.RunSQL ("UPDATE SET password=" + a + " WHERE id\_user=4 ")  
  
End If  
  
End Sub

Код VBA для авторизации сотрудника авторизации

Option Compare Database  
  
Private Sub enter\_Click()  
  
Dim a As Recordset  
Dim b, c As Boolean  
  
b = False  
  
Set a = CurrentDb.OpenRecordset("должности", dbOpenDynaset)  
  
  
a.MoveFirst  
  
Do Until a.EOF  
  
If a![Password] = pass.Value And a![id\_user] = 1 Then  
  
b = True  
  
ElseIf a![Password] = pass.Value And a![id\_user] = 2 Then  
  
c = True  
  
End If  
  
a.MoveNext  
  
Loop  
  
If b = True Then  
  
MsgBox "Вход выполнен для сотрудника администрации!"  
DoCmd.OpenForm "Администрация", acNormal  
DoCmd.Close acForm, "Авторизация - администрация"  
  
ElseIf c = True Then  
  
MsgBox "Вход выполнен для системного администратора!"  
DoCmd.OpenForm "Системный администратор", acNormal  
DoCmd.Close acForm, "Авторизация - администрация"  
  
ElseIf b = flase And c = False Then  
  
MsgBox "Пароль введен неверно!"  
  
End If  
  
  
End Sub