

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ «ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНО-
ЭКОНОМИЧЕСКИЙ КОЛЛЕДЖ»

Специальность 2-40 01 01 «Программное обеспечение
информационных технологий»

КУРСОВОЙ ПРОЕКТ

по дисциплине: «Основы алгоритмизации и программирования»

**на тему: «Разработка информационно-поисковой системы
«Тарифы мобильных операторов»**

Пояснительная записка

Разработал	<hr/>	<hr/>	<hr/> Величко М.Н.
	(дата)	(подпись)	(ФИО)
Руководитель	<hr/>	<hr/>	<hr/> Глушенок А. В.
	(дата)	(подпись)	(ФИО)

Гомель, 2021

СОДЕРЖАНИЕ

Введение.....3

1 Условные операторы. Логические операции.....4

 1.1 Условный оператор..... 4

 1.2 Оператор «Case»..... 5

 1.3 Логические операции..... 6

2 Описание алгоритмов технологий решения..... 7

 2.1 Постановка задачи..... 7

 2.2 Описание разработанных алгоритмов.....8

 2.3 Организация диалога с пользователем.....9

3 Практическая часть выполнения задачи..... 18

 3.1 Интерфейс приложения..... 18

 3.2 Программная реализация.....27

 3.3 Проверка корректности ввода данных.....37

Заключение..... 73

Приложение Б Блок-схема программы..... 125

					01.2021.2-400101.1.ПО-21.Г-В-16-ПЗ			
Изм	Лист	№ докум.	Подп.	Дата				
Разраб.		Величко			Разработка информационно- поисковой системы «Тарифы мобильных операторов»	Лит.	Лист	Листов
Пров.		Глушенок				У		
						Учреждение образования «Гомельский государственный аграрно- экономический колледж»		
Н. Контр.								
Утв.								

ВВЕДЕНИЕ

Цель данной курсовой работы состоит в разработке информационно-поисковой системы “Тарифы мобильных операторов”.

Информационно-поисковая система – это прикладная компьютерная среда для обработки, хранения, сортировки, фильтрации и поиска больших массивов структурированной информации. Каждая ИПС предназначена для решения определенного класса задач, для которых характерен свой набор объектов и их признаков.

Тариф - ставка или система ставок оплаты за различные производственные и непроизводственные услуги, предоставляемые компаниями, организациями, фирмами и учреждениями.

Задачи курсовой работы:

- Определить ключевые инструменты и этапы в разработке информационно-поисковой системы и рассмотреть особенности их применения в рассматриваемой теме;
- Изучить теоретические сведения о поставленной теме курсового проекта, создание собственных типизированных файлов, создания базы данных с содержащимися в ней данными из типизированного файла, алгоритмов обработки данных;
- Провести анализ поставленной темы и алгоритмов её решения;
- Составить алгоритмы работы с типизированными файлами, входными данными, пользовательским интерфейсом ;
- Разработать готовую программу в которой будут решены поставленные задачи.

При разработке данного курсового проекта будут решены задачи связанные с поставленной темой курсового проекта, такие как: поиск в типизированном файле, добавление новой записи в типизированный файл, изменение записи в типизированном файле, сортировка данных в типизированном файле, фильтрация данных в типизированном файле, а так же сохранение и открытие типизированного файла для последующей работы в данной программе. При разработке информационно-поисковой системы будет создан удобный интерфейс программы, для того, чтобы обычному пользователю было удобно пользоваться данным программный обеспечением в не зависимости от возраста и уровня компьютерной грамотности.

Главной задачей при разработке информационно-поисковой системы будет уделяться работе с данными предоставляемыми пользователем разрабатываемой программы, а именно: корректно обрабатывать входные данные предоставленные пользователем, проверять на наличие ошибок в получаемых данных, корректно создавать и изменять данные в типизированном файле , и как следует из названия курсового проекта, программа должна правильно искать, сортировать, и фильтровать исходные данные для последующего использования изменённых данных пользователем.

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	3
01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ					

В результате выполнения курсового проекта получится готовый программный продукт со своим кругом пользователей.

1 УСЛОВНЫЕ ОПЕРАТОРЫ. ЛОГИЧЕСКИЕ ОПЕРАЦИИ

1.1 Условный оператор

Условные оператор - оператор, конструкция языка программирования, обеспечивающая выполнение определённой команды (набора команд) только при условии истинности некоторого логического выражения, либо выполнение одной из нескольких команд (наборов команд) в зависимости от значения некоторого выражения [3, с. 55].

Условие — это выражение, которое может быть либо истинным, либо ложным. Условие обязательно содержит логические операторы.

Когда выполнение основной ветки программы доходит до условного оператора «if-else», то в зависимости от результата логического выражения в его заголовке выполняются разные блоки кода. Если логическое выражение вернуло true, то выполняется один блок (в Паскале начинается со слова «then»), если «false» – то другой (начинается со слова «else»). После выполнения одного из вложенных блоков кода, ход программы возвращается в основную ветку. Другой вложенный блок не выполняется. Полная форма условного оператора показана на рисунке 1.1.

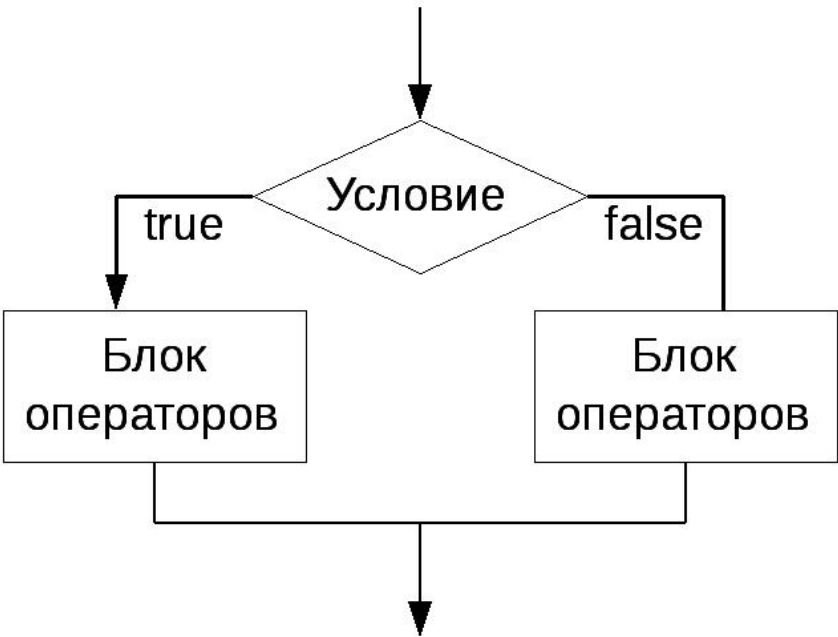


Рисунок 1.1 - Полная форма условного оператора

Есть неполные формы условных операторов. В таком случае вложенный в «if» блок кода выполняется только в случая «true» логическом выражении заголовка. В случае «false» выполнение программы сразу передается в

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		4

основной блок. Понятно, что ветка «else» в таком случае отсутствует. Пример неполного оператора показан на рисунке 1.2.



Рисунок 1.2 - Неполная форма условного оператора

Непосредственно после «then» может стоять только один оператор. При необходимости выполнения нескольких операторов они должны быть заключены в операторные скобки «begin-end».

1.2 Оператор «Case»

Оператор выбора (варианта) используется в тех случаях, когда в зависимости от значения какого-либо выражения необходимо выполнить один из нескольких операторов. Оператор «case» состоит из выражения (селектора) и списка операторов, каждому из которых предшествует одна или более констант (они называются константами выбора) или ключевое слово «else».

Селектор должен иметь порядковый тип размера байт. Строковый тип и тип «LongInt» являются недопустимыми типами селектора. Все константы выбора должны быть уникальными и иметь порядковый тип, совместимый с типом селектора. Оператор «case» приводит к выполнению оператора, которому предшествует константа выбора, равная значению селектора или диапазону выбора, в котором находится значение селектора. Если такой константы выбора или такого диапазона выбора не существует, то выполняется оператор, следующий за ключевым словом «else». Если ветвь «else» отсутствует, то не выполняется никакой оператор [4, с. 103]. Структура оператора «case» показана на рисунке 1.3.

В операторе выбора в качестве константы допускается использование списка констант. В качестве констант также могут использоваться

перечислимые и ограниченные типы данных, но число имен не должно превышать 255.

```
CASE <выражение> OF
константа 1: оператор 1;
константа 2: оператор 2;
.....
константа n: оператор n;
ELSE оператор;
END;
```

Рисунок 1.3 - Оператор «Case»

1.3 Логические операции

Логическая операция (логический оператор, логическая связка, пропозициональная связка) — операция над высказываниями, позволяющая составлять новые высказывания путём соединения более простых. В качестве основных обычно называют конъюнкцию(&), дизъюнкцию, импликацию, отрицание. В смысле классической логики логические связки могут быть определены через алгебру логики. В асинхронной секвинциальной логики определена логико-динамическая связка в виде операции венъюнкции [5, с. 77].

Логическая операция — в программировании операция над выражениями логического (булевского) типа, соответствующая некоторой операции над высказываниями в алгебре логики. Как и высказывания, логические выражения могут принимать одно из двух истинностных значений — «истинно» или «ложно». К логическим операциям относятся бинарные операции «and», «or» и «xor», а также унарная операция «not», имеющие операнды типа «boolean» и возвращающие значение типа «boolean». Эти операции подчиняются стандартным правилам логики: а «and» b истинно только тогда, когда истинны а и b, а «or» b истинно только тогда, когда истинно либо а, либо b, а «xor» b истинно только тогда, когда только одно из а и b истинно, «not» а истинно только тогда, когда а ложно [6, с. 34]. Все эти правила показаны в таблице истинности на рисунке 1.4.

					Лист	
					6	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

a	b	a and b	a or b	a xor b	not a
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0

Рисунок 1.4 – Таблица Истинности

2 ОПИСАНИЕ АЛГОРИТМОВ ТЕХНОЛОГИЙ РЕШЕНИЯ

2.1 Постановка задачи

Основная задача при разработке информационно-поисковой системы «Тарифы мобильных операторов» заключалась в упрощении работы мобильным операторам при работе с большими массивами данных, такими как: стоимость абонентской платы, количество любимых номеров и тд. Программа должна создавать, читать данные из файла и корректно их записывать в файл. Программа может выполнять такие функции, как:

- Создавать файл;
- Сохранять файл;
- Добавлять данные в файл;
- Удалять данный в файле;
- Изменять данные;
- Сортировать данные;
- Фильтровать данные;
- Искать в данных нужную информацию.

Исходные данные – это те данные, который пользователь вводит в программу и те в свою очередь используется в программе. Исходные данные в данной курсовой работе представлены в виде типизированной структуры данных, которая представлена на рисунке 2.1.

```
type
  tele = record
    name: string[10];
    tarif: string[15];
    cost: integer;
    trafic: string[3];
    free_min : integer;
    lovenumber : integer;
    date: string[10];
end;
```


Рисунок 2.1 – Исходные данные в типизированной структуре

Поле «name» имеет строковый тип и хранит в себе название мобильного оператора. Поле «tarif» имеет строковый тип и хранит в себе тариф мобильного оператора. Поле «cost» имеет целочисленный тип и хранит в себе стоимость абонентской платы за тариф предоставляемым мобильным оператором. Поле «trafic» имеет строковый тип и хранит в себе значение интернет трафика. Поле «free_min» имеет целочисленный тип и хранит в себе количество бесплатных минут предоставляемых клиенту. Поле «lovenumber» имеет целочисленный тип и хранит в себе количество любимых номеров. Поле «date» имеет строковый тип и хранит в себе дату появления тарифа у клиента.

2.2 Описание разработанных алгоритмов

При разработке информационно-поисковой системы были решены поставленные задачи работы с данными и разработкой алгоритмов их обработки.

Разработанные алгоритмы обработки данных:

- Сортировка данных;
- Поиск данных;
- Фильтрация данных;
- Добавление данных;
- Удаления данных;
- Изменения данных;
- Сохранение данных в файл;
- Открытие файла с данными.

При вызове функции «Добавление данных» происходит проверка корректности ввода данных, если данные введены верно, то происходит добавление данных в таблицу, а если данные введены не корректно, то об этом сообщается пользователю.

При вызове функции «Изменение данных» происходит изменение данных из выбранной строки таблицы, и проверка на правильность введённых данных, и если введённые данные были введены корректно, то данная строка в таблице изменится на изменённую.

При вызове функции «Удаление данных» в таблице с данными удаляется выбранная запись.

При вызове функции «Сортировка данных» пользователь выбирает конкретную колонки для сортировки и способ сортировки: «по возрастанию», «по убыванию», после чего таблица с данными будет отсортированная по выбранным параметрам.

При вызове функции «Фильтрация данных» пользователь вводит данные для фильтрации, а затем он нажимает на кнопку. После чего, таблица с данными будет содержать только те данные, которые содержат искомое значение.

При вызове функции «Поиск данных» пользователь вводит искомую информацию в поле для ввода, после чего он нажимает на кнопку для начала поиска. После нажатия на кнопку в таблице с данными будет подсвечены ячейки, которые содержат искомое значение.

При вызове функции «Сохранение данных» пользователь выбирает файл, в который он будет сохранять данные из таблицы. После чего, в выбранный файл будут записаны данные из таблицы.

При вызове функции «Открытие файла с данными» пользователь выбирает файл, из которого будут считаны данные в таблицу. После выбора файла, в таблицу с данными будет занесена информация из выбранного файла.

При работе с файлами необходимо использовать такие функции, как: «Assign», «Reset», «Rewrite», «Close».

Функция «Assign» связывает переменную F с файлом какого-либо типа. После этого в программе можно использовать переменную F для работы с указанным файлом.

Функция «Reset» открывает существующий внешний файл с именем, назначенным в переменной F. Происходит ошибка, если внешний файл с данным именем не существует. Если файл F уже открыт, то он сначала закрывается, а затем вновь открывается.

Функция «Rewrite» создает новый внешний файл с именем, назначенным в переменной F. Если внешний файл с тем же самым именем уже существует, то он удаляется, а на его месте создается новый пустой файл [7, с. 47].

Функция «Close» закрывает ранее открытый файл функцией «Assign».

2.3 Организация диалога с пользователем

Для оформления графического интерфейса были использованы следующие компоненты «Delphi»: «Button», «Stringgrid», «Edit», «DateTimePicker», «Combobox», «Label», «ListBox», «Memo».

Для ввода данных использовались компоненты: «Edit», «Combobox», «DateTimePicker».

Для вывода данных использовались компоненты: «Label», «ListBox», «Memo».

Макет формы «Приветствия» в данной курсовой работе представлена на рисунке 2.3.

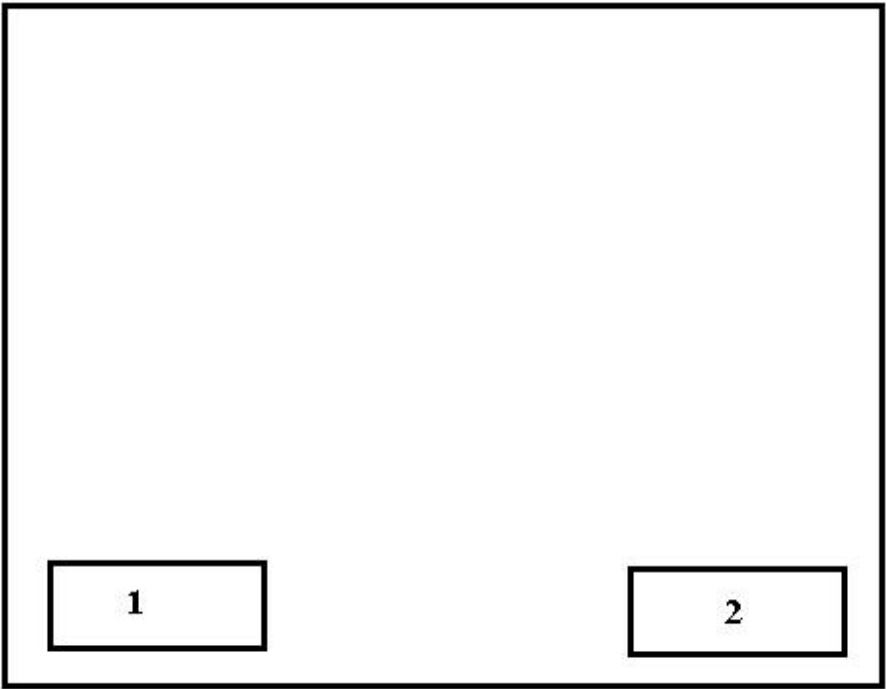


Рисунок 2.3 — макет формы «Приветствия»:
1 — блок выхода; 2 — блок перехода

Макет главной формы курсовой работы представлен на рисунке 2.4.

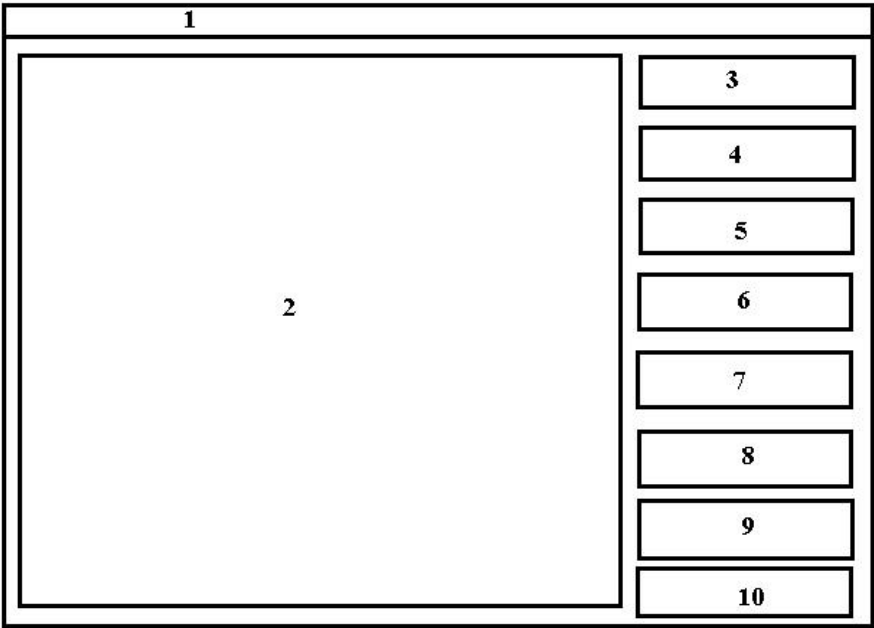


Рисунок 2.4 — Макет главной формы:
1 — блок главного меню; 2 — блок вывода информации; 3 — блок для перехода к блокировке; 4 — блок для перехода к добавлению; 5 — блок для перехода к изменению; 6 — блок для перехода к поиску; 7 — блок для

перехода к сортировке; 8 — блок для перехода к фильтрации; 9 — блок для перехода к статистике; 10 — блок для перехода к удалению

Макет формы добавления и изменения показан на рисунке 2.5.

1

2

Рисунок 2.5 – Макет формы добавления и изменения:
1 – блок ввода информации; 2 – блок сохранения

Макет формы поиска показан на рисунке 2.6.

1

2

3

Рисунок 2.6 – Макет формы «Поиска»:
1 — блок ввода данных; 2 — блок начала поиска; 3 — блок вывода промежуточного результата

Макет формы сортировки показан на рисунке 2.7.

1

2

3

4

5

Рисунок 2.7 – Макет формы «Сортировка»:
1 — блок вывода информации; 2 — блок для перехода к сортировке
выбором; 3 — блок сортировки; 4 — блок сохранения результата; 5 — блок
выхода

Макет формы «Фильтрации данных» показан на рисунке 2.8.

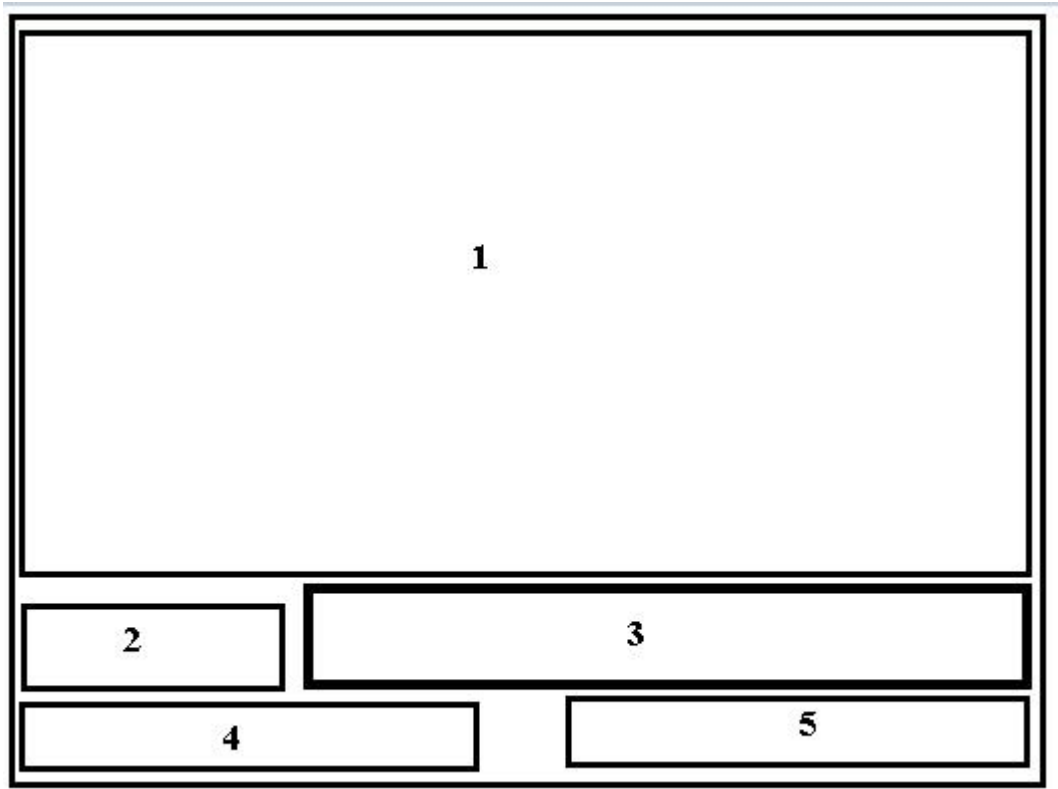


Рисунок 2.8 – Макет формы «Фильтрации данных»:
1 — блок вывода информации; 2 — блок для фильтрации «По любимым номерам»; 3 — блок фильтрации; 4 — блок сохранения результата; 5 — блок выхода

Макет формы «Удаления оператора» показан на рисунке 2.9.

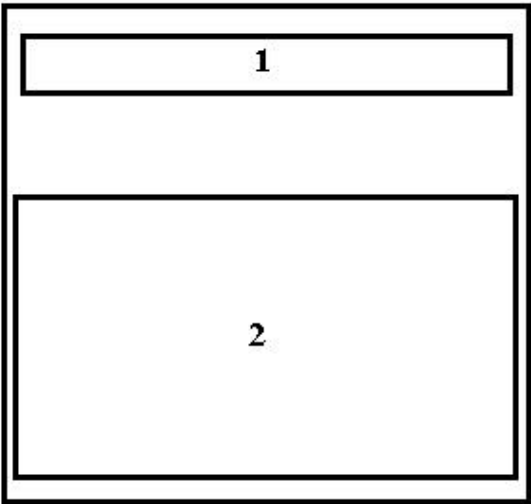


Рисунок 2.9 – Макет формы «Удаления оператора»:
1 — блок выбора информации; 2 — блок удаления

Макет формы «Статистика» показан на рисунке 2.10.

					<div>01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ</div>	Лист
						13
Изм.	Лист	№ докум.	Подп.	Дата		

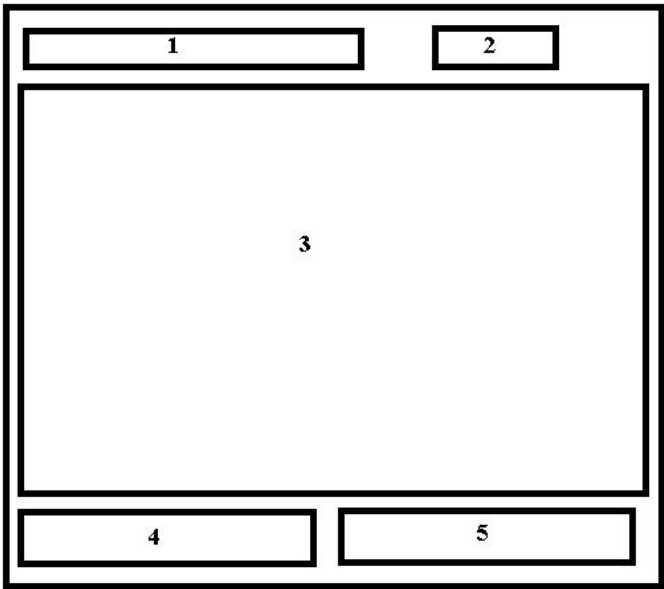


Рисунок 2.10 – Макет формы «Статистика»:
1 — блок выбора информации; 2 — блок очистки поля вывода; 3 — блок вывода информации; 4 — блок просмотра данных; 5 — блок выхода

Макет формы «Открытие файла» показан на рисунке 2.11.

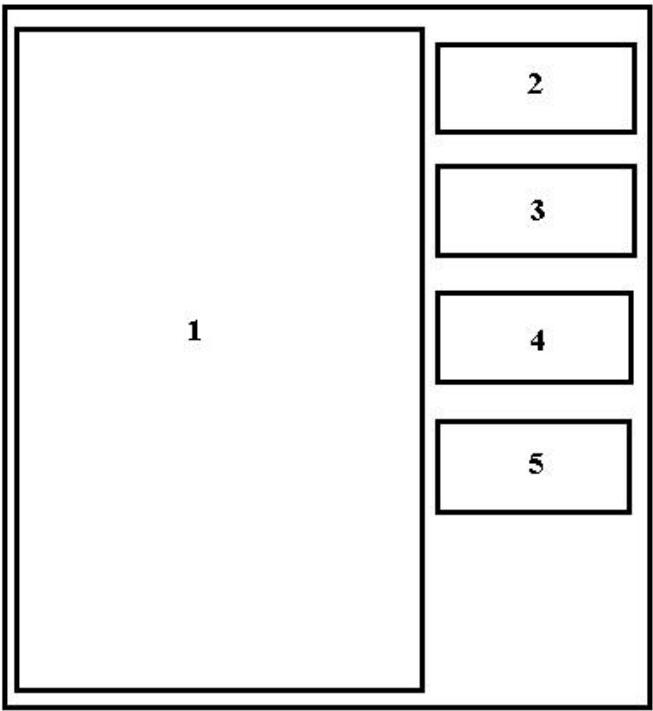


Рисунок 2.11 – Макет формы «Открытие файла»:
1 — блок вывода информации; 2 — блок создания файла; 3 — блок удаления файла; 3 — блок выбора файла; 4 — блок открытия файл;

Макет формы «Об авторе» показан на рисунке 2.12.

1

23

Рисунок 2.12 – Макет формы «Об авторе»:
1 — блок изображения; 2 — блок ссылки на «Вконтакте»; 3 — блок
ссылки на «Instagram»

Макет формы «Заблокировать» показан на рисунке 2.13.

1

2

Рисунок 2.13 – Макет формы «Заблокировать»:
1 — блок ввода информации; 2 — блок проверки информации

Макет формы «Доавбления тарифа» показан на рисунке 2.14.

1

2

Рисунок 2.14 — Макет формы «Добавления тарифа»:
1— блок ввода информации; 2 — блок сохранения данных

Макет формы «Функции» показан на рисунке 2.15.

1

2

3

4

Рисунок 2.15 – Макет формы «Функции»:
1 — блок выбора информации; 2 — блок вывода первого значения; 3 — блок вывода второго значения; 4 — блок вывода результата

Макет формы «Потверждения выхода» показан на рисунке 2.16.

1

23

Рисунок 2.16 — Макет формы «Потверждения выхода»:
1 — блок вывода информации; 2 — блок сохранения; 3 — блок выхода

Макет сайта «Помощь» показан на рисунке 2.17.

1	2
3	4

Рисунок 2.17 — Макет сайта «Помощь»:

1 — блок вывода компонента; 2 — блок вывода информации; 3 — блок вывода информации о компоненте; 4 — блок вывода изображения

3 ПРАКТИЧЕСКАЯ ЧАСТЬ ВЫПОЛНЕНИЯ ЗАДАЧИ

3.1 Интерфейс приложения

Графический интерфейс — система средств для взаимодействия пользователя с компьютером, основанная на представлении всех доступных пользователю системных объектов и функций в виде графических компонентов экрана [8, с. 87].

Форма «Приветствия» встречает пользователя при открытии программы. На данной форме располагаются компоненты: кнопка «Далее» и кнопка «Выйти». Кнопка «Выйти» отвечает за выход из программы до открытия главной формы. Кнопка «Далее» отвечает за переход к главной форме разрабатываемой программы. Форма «Приветствия» показана на рисунке 3.1.

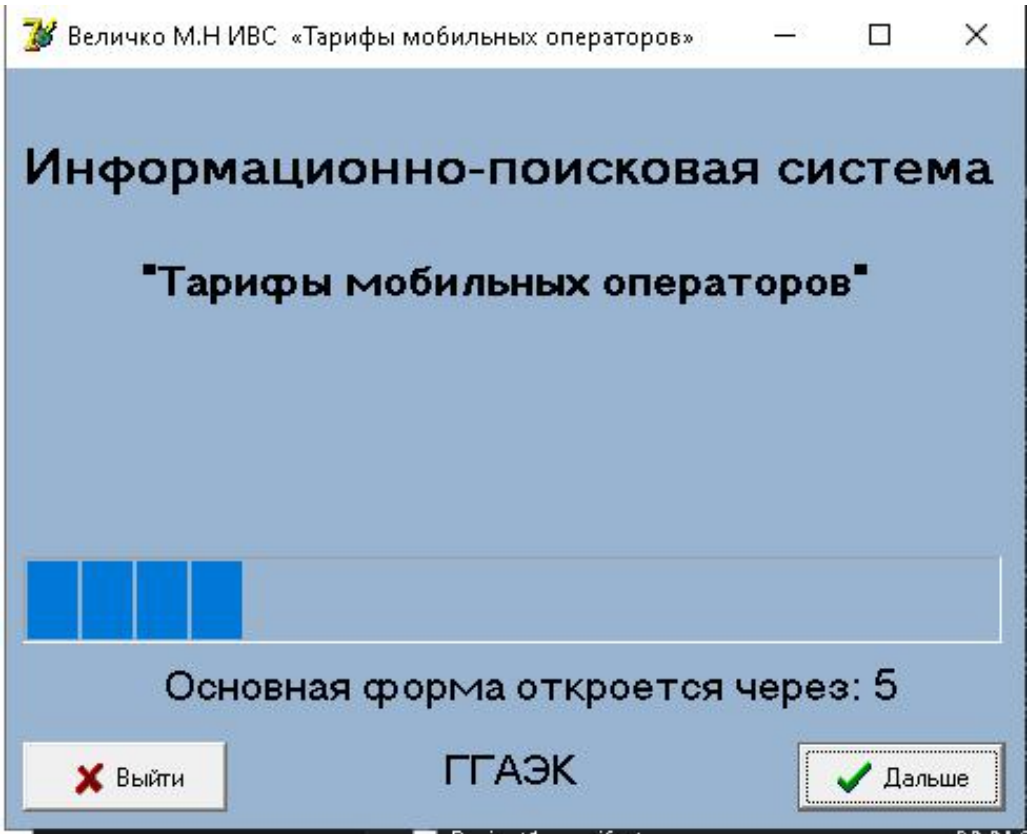


Рисунок 3.1 – Форма «Приветствия»

Главная форма курсовой работы, на которой располагаются основные компоненты для работы с входными данными предоставляемыми пользователем программы и выходными данными, которое создаёт разрабатываемая программа представлена на рисунке 3.2.

					Лист	
					18	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

ИПС "Тарифы мобильных операторов"

ФайлОчистить поискФункцииПомощьСвязьОб авторе

Файл: inp.dat

Статус: открыт

Row: 10

Col: 6

заблокировать

Мобильный оператор	Тариф	Стоимость обоненс	Интернет трафик (М	Количество беспла	Количество любим	Дата появления тар
Life :)	Комфорт L	65	34	23	65	03.05.2021
A1	Без Лимита	12	65	76	23	03.05.2021
МТС	Бизнес	87	2	23	65	03.05.2021
БелСел	Эконом	87	123	23	65	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021
МТС	Без Лимита	76	123	7	0	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021
БелСел	Звонки	87	12	34	87	03.05.2021
Life :)	Комфорт L	87	234	34	6	03.05.2021
МТС	Комфорт L	76	23	456	87	03.05.2021

добавление

изменить

Поиск

Сортировка

Фильтрация

Статистика

Удаление оператора

Рисунок 3.2 – Главная форма программы

При нажатии на кнопку «Добавление» на главной форме разрабатываемой программы открывается форма «Добавление» на которой есть компоненты для ввода данных для добавления и кнопка «Сохранить» при нажатии на которую введенные данные будут сохранены в соответствующие колонки таблицы. Форма «Добавления» разрабатываемой программы показана на рисунке 3.3.

7

Добавление

—□×

Мобильный оператор

Тариф

Стоимость абоненской платы

Интернет трафик

Количество любимых номеров

Количество бесплатных минут

Дата появления тарифа

03.05.2021

✓

OK

Рисунок 3.3 – Форма «Добавление»

При нажатии на кнопку «Изменить» открывается форма «Изменить» на которой располагаются компоненты для ввода данных и кнопка «Применить» при нажатии на которую введенные данные будут изменены в таблице. Форма «Изменить» показана на рисунке 3.4.

					<div>01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ</div>	Лист
						20
Изм.	Лист	№ докум.	Подп.	Дата		

Изменить

Мобильный оператор

Тариф

Стоимость абонентской платы

Интернет трафик

Количество любимых номеров

Количество бесплатных минут

Дата появления тарифа

03.05.2021

?

Проверить

✂

Скопировать

↺

Исправить ошибки

✓

применить

Рисунок 3.4 – Форма «Изменить»

При нажатии на кнопку «Поиск» открывается форма «Поиск» на которой располагаются компоненты для ввода и вывода искомых, которые программа и будет сравнивать с данными в таблице. При нажатии на кнопку «Поиск» происходит поиск по таблице на главной форме разрабатываемой программы с данными для поиска. Форма «Поиск» показана на рисунке 3.5.

Поиск...

✓

Поиск

Рисунок 3.5 – Форма «Поиск»

					Лист	
					21	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

При нажатии на кнопку «Фильтрация» на главной форме разрабатываемой программы открывается форма «Фильтрация», где располагаются компоненты для ввода искомых данных для фильтрации в таблице. При нажатии на кнопку «Сохранить» отфильтрованные данные будут сохранены в таблицу на главной форме разрабатываемой программы. Форма «Фильтрация» показана на рисунке 3.6.

Фильтрация

Очистить Вернуться к изначальной таблице

Мобильный оператор	Тариф	Стоимость обоненс	Интернет трафик	Количество беспла	Количество любим	Дата появления тар
МТС	Комфорт L	65	45	78	765	03.05.2021

Записи с любимыми номерами

Фильтровать по:

Начать фильтрацию

Сохранить

Выйти

Рисунок 3.6 – Форма «Фильтрация»

При нажатии на кнопку «Статистика» открывается форма «Статистика» на которой располагается поле для вывода диаграмм. При нажатии на кнопку «Посмотреть статистику» в поле для вывода диаграмм появится диаграммы в зависимости от значения выбранного для статистики. Форма «Статистика» разрабатываемой программы показана на рисунке 3.7.



Рисунок 3.7 – Форма «Статистика»

При нажатии на кнопку «Удаление оператора» откроется форма «Удаление оператора». При нажатии на кнопку «Удалить» в таблице на главной форме разрабатываемой программы удалятся записи с выбранным оператором. Форма «Удаление оператора» показана на рисунке 3.8.

Рисунок 3.8 – Форма «Удаление оператора»

Форма «Заблокировать» появляется при нажатии на кнопку «Заблокировать» на главной форме курсовой работы. Форма «Заблокировать» служит для блокировки основной формы. Для ввода пароля нужно вводить данные в компонент «Edit» на форме «Заблокировать» после чего нажать на кнопку «Войти». Форма «Заблокировать» показана на рисунке 3.9.

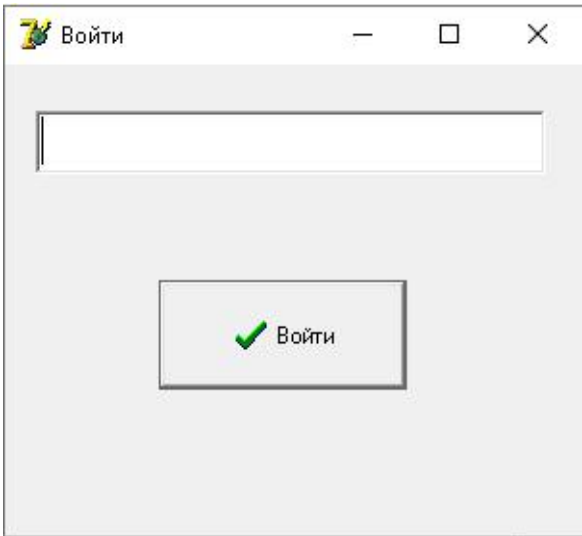
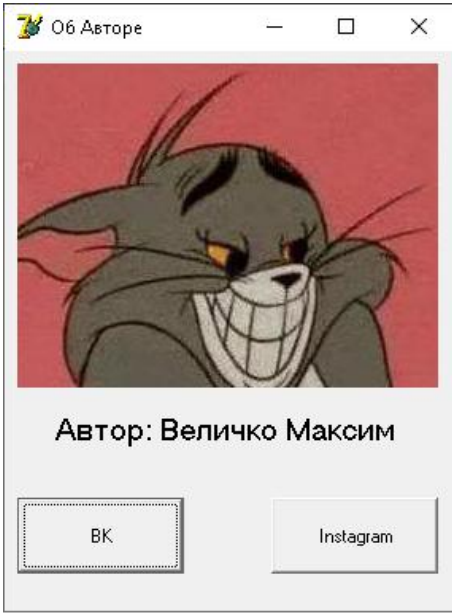


Рисунок 3.9 – Форма «Заблокировать»

При нажатии на кнопку «Об авторе» в компоненте «MeinMenu» откроется форма «Об авторе» на которой располагается изображение и две кнопки: кнопка «ВК» и кнопка «Instagram». При нажатии на кнопку «ВК» откроется браузер со ссылкой на страницу автора в социальной сети «Вконтакте», а при нажатии на кнопку «Instagram» откроется страница автора в социальной сети «Instagram». Форма «Об авторе» показана на рисунке 3.10.



					Лист	
					24	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

Рисунок 3.10. – Форма «Об авторе»

При нажатии на кнопку «Открыть» в компоненте «MeinMenu» в пункте «Файл» откроется форма «Открытие файла» в которой располагает компонент для вывода файлов с расширением «.dat» и кнопки: «Создать новый файл», «Удалить файл», «Выбор файла», «Открыть выбранный файл». При нажатии на кнопку «Создать новый файл» откроется форма «Создать новый файл» в которой можно создать новый файл в расширении «.dat». При нажатии на кнопку «Удалить файл» удаляется выбранный файл из компонента «ListBox». При нажатии на кнопку «Выбор файла» откроется проводник в котором можно выбрать файл для открытия. При нажатии на кнопку «Открыть выбранный файл» в программе откроется файл выбранный в компоненте «ListBox». Форма «Открытие файла» показана на рисунке 3.11.

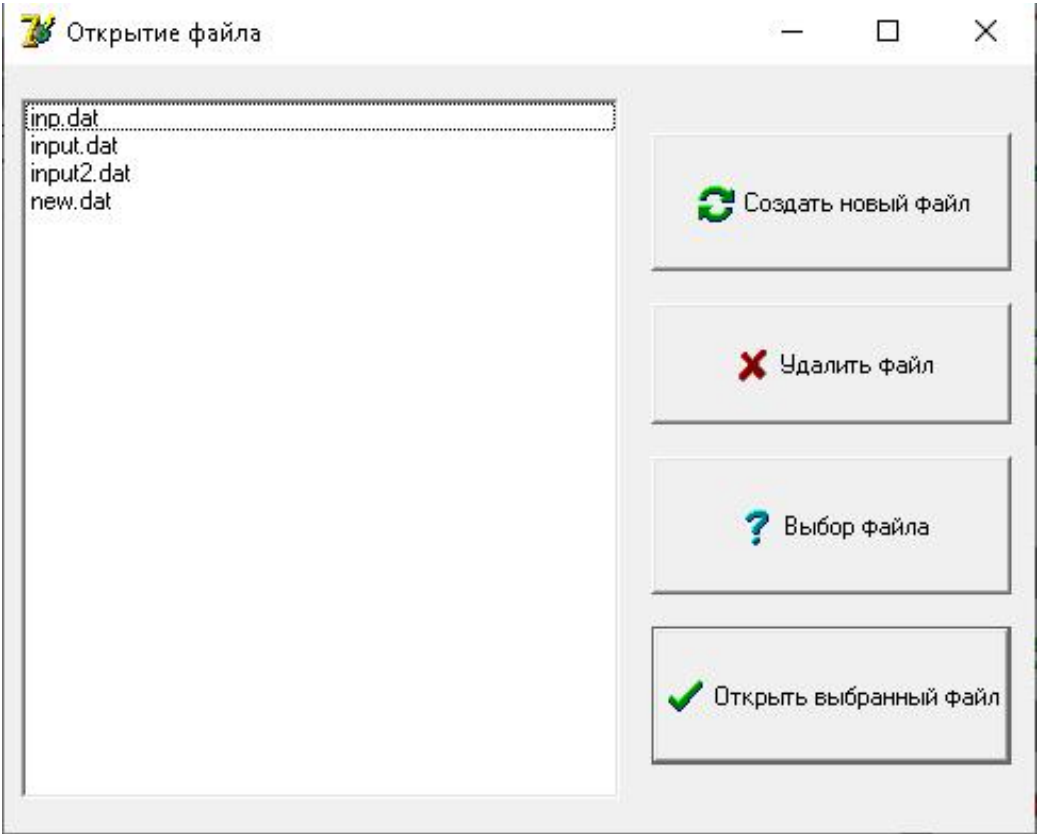


Рисунок 3.11 – Форма «Открытие файла»

При нажатии на кнопку «Создать файл» на форме «Открытие файла» откроется форма «Создание файла» на котором располагается компонент «Edit» для ввода информации и кнопка «Создать файл» при нажатии на которую создаётся файл. Форма «Создание файл» показана на рисунке 3.12.

Создание файла

Введите название файла

Создать файл

Рисунок 3.12 – Форма «Создание файла»

При нажатии на кнопку «Сортировка» на главной форме разрабатываемой программы откроется форма «Сортировка» на которой располагается таблица для вывода информации и компоненты для сортировки данных в таблице, такие как: кнопка «Сортировка по убыванию», кнопка «Сортировка по возрастанию», кнопка «Сортировка выбором». Форма «Сортировка» показана на рисунке 3.13.

Сортировка

Стартовая страница

Мобильный оператор	Тариф	Стоимость обзвон	Интернет трафик	Количество бесплатных	Количество любимых	Дата появления
МТС	Комфорт М	123	321	6	543	03.05.2021

Сортировать выбором

Сортировать по убыванию

Сортировать по возрастанию

Сохранить и выйти

Выйти

Рисунок 3.13 – Форма «Сортировка»

3.2 Программная реализация

Исходный код — текст компьютерной программы на каком-либо языке программирования или языке разметки, который может быть прочтён человеком. В обобщённом смысле — любые входные данные для транслятора.

Исходный код функции добавления, в котором происходит сохранения введённых данных и их проверка на правильность для последующего сохранения в таблицу «Stringgrid» показан на рисунке 3.14.

```
if(combobox1.ItemIndex = 0) then begin
    form1.stringgrid1.cells[0, form1.add_rowindex] := combobox1.items[0];
end;
if(combobox1.ItemIndex = 1) then begin
    form1.stringgrid1.cells[0, form1.add_rowindex] := combobox1.items[1];
end;
if(combobox1.ItemIndex = 2) then begin
    form1.stringgrid1.cells[0, form1.add_rowindex] := combobox1.items[2];
end;
if(combobox1.ItemIndex = 3) then begin
    form1.stringgrid1.cells[0, form1.add_rowindex] := combobox1.items[3];
end;
//таблица
if(combobox2.ItemIndex = 0) then begin
    form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[0];
end;
if(combobox2.ItemIndex = 1) then begin
    form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[1];
end;
if(combobox2.ItemIndex = 2) then begin
    form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[2];
end;
if(combobox2.ItemIndex = 3) then begin
    form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[3];
end;
if(combobox2.ItemIndex = 4) then begin
    form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[4];
end;
if(combobox2.ItemIndex = 5) then begin
    form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[5];
end;
```

Рисунок 3.14 — Исходный код функции добавления

Исходный код функции изменения в котором происходит сохранения введённых данных и их проверки на правильность для последующего

изменения уже существующих данных в таблицы «Stringgrid» показан на рисунке 3.15.

```
edit1.Color := clwhite;
form1.StringGrid1.Cells[0, form1.row_index] := combobox1.Text;
form1.StringGrid1.Cells[1, form1.row_index] := combobox2.Text;
form1.StringGrid1.Cells[2, form1.row_index] := edit3.Text;
form1.StringGrid1.Cells[3, form1.row_index] := edit1.Text;
form1.StringGrid1.Cells[4, form1.row_index] := edit5.Text;
form1.StringGrid1.Cells[5, form1.row_index] := edit6.Text;
form1.StringGrid1.Cells[6, form1.row_index] := datetostr(datetimepicker1.Date);

form5.Hide;
form1.show;
```

Рисунок 3.15 — Исходный код функции изменения

Исходный код функции поиска в котором происходит сохранения введённых данных для функции и последующего поиска по сохранённым данным в таблице с перекрашиванием ячеек таблицы показан на рисунке 3.16.

```
draw_choice := 0;
m := 0;
str := Edit1.text;
for i:=0 to form1.StringGrid1.ColCount - 1 do begin

    for j:=1 to form1.StringGrid1.RowCount - 1 do begin

        str2 := form1.StringGrid1.Cells[i, j];

        n := pos(str, str2);

        if(n = 0) then begin
            continue;
        end
        else if(n <> 0) then begin

            form1.col_index := i;
            form1.row_index := j;
            draw_choice :=1;

            form1.str_grid := str2;
            str3 := form1.stringgrid1.Cells[i, j];
            form1.stringgrid1.Cells[i, j] := str3;
        end;
    end;
end;
form17.Close;
listbox1.Visible := false;
end;
```

Рисунок 3.16 — Исходный код функции поиска

Исходный код функции сортировки в котором происходит выполнения сортировки данных в таблице по заданным параметрам, таким как: сортировка по убыванию, сортировка по возрастанию показана на рисунке 3.17.

```
procedure TForm1.BitBtn2Click(Sender: TObject);
var a, g, h : integer;
var r, q1, w1 : string;
begin
  for a := stringgrid1.RowCount - 1 downto 0 do begin

    for g := 1 to a - 1 do begin

      if strtoint(stringgrid1.Cells[2, g]) < strtoint(stringgrid1.Cells[2, g + 1]) then 1

        r := stringgrid1.Cells[h, g];
        stringgrid1.Cells[h, g] := stringgrid1.Cells[h, g + 1];
        stringgrid1.Cells[h, g + 1] := r;

      end;
    end;
  end;

  for a := stringgrid1.RowCount - 1 downto 0 do begin

    for g := 1 to a - 1 do begin

      if strtoint(stringgrid1.Cells[2, g]) > strtoint(stringgrid1.Cells[2, g + 1]) then 1

        r := stringgrid1.Cells[h, g];
        stringgrid1.Cells[h, g] := stringgrid1.Cells[h, g + 1];
        stringgrid1.Cells[h, g + 1] := r;

      end;
    end;
  end;
end;
```

Рисунок 3.17 — Исходный код функции сортировки

Исходный код функции фильтрации в котором происходит сохранения исходных данных для функции и последующей фильтрации данных в таблице в зависимости от сохранённых данных показана на рисунке 3.18.


```
procedure dell_filtr(col: integer; ch: string);
var i, j, k: integer;
begin

    k := 0;
    for i:=0 to form15.StringGrid1.ColCount - 1 do begin

        for j:=1 to form15.StringGrid1.RowCount - 1 do begin

            if(form15.StringGrid1.Cells[col, j] <> ch) then begin

                k := j;
                DeleteRow_ss(k, form15.StringGrid1);
                break;
            end
            else begin
                continue;
            end;
        end;
    end;
end;
end;
```

Рисунок 3.18 — Исходный код функции фильтрации

Исходный код функции удаления оператора в котором сохраняются введённые данные и удаления записи в таблице в зависимости от сохранённых данных при нажатии на кнопку «Удалить» представлена на рисунке 3.19.

```
str:=combobox1.text;
n := 0;

if(str <> '') then begin

    for i:=0 to form1.stringgrid1.colcount - 1 do begin

        for j:=1 to form1.stringgrid1.rowcount - 1 do begin

            if(form1.stringgrid1.cells[0,j] = str) then begin

                n := j;|
                DeleteRow(n);
            end;
        end;
    end;

    form1.add_rowindex := form1.StringGrid1.RowCount;

    form14.Hide;
```

Рисунок 3.19 — Исходный код функции удаления оператора

Исходный код функции статистики в котором в зависимости от выбранных данных программа строит диаграммы в поле для вывода информации представлена на рисунке 3.20.

```
al := 0;
life := 0;
mts := 0;
belsel := 0;

with Series1 do begin

  if (combobox1.ItemIndex = 0) then begin

    for i:=0 to form1.StringGrid1.ColCount - 1 do begin

      for j:=0 to form1.StringGrid1.RowCount - 1 do begin

        if (form1.StringGrid1.Cells[i, j] = 'A1') then begin
          al := al + 1;
        end;

        if (form1.StringGrid1.Cells[i, j] = 'Life ;)') then begin
          life := life + 1;
        end;

        if (form1.StringGrid1.Cells[i, j] = 'MTC') then begin
          mts := mts + 1;
        end;

        if (form1.StringGrid1.Cells[i, j] = 'БелСел') then begin
          belsel := belsel + 1;
        end;
      end;
    end;
  end;
  clear;

  Add(al, 'A1', clRed);
  Add(life, 'Life ;)', clGreen);
  Add(mts, 'MTC', clYellow);
  Add(belsel, 'БелСел', clBlue);

end;
```

Рисунок 3.20 — Исходный код функции статистики

Исходный код функции открытия файла в котором открытие файла с последующим сохранением данных из файла в таблицу на главной форме разрабатываемой программы показан на рисунке 3.21.

					Лист	
					31	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	


```
for h := 1 to form1.StringGrid1.RowCount - 1 do begin
    DeleteRow(h);
end;

assignfile(input, listbox1.Items[listbox1.Itemindex]);
reset(input);
seek(input, 0);
form1.Label7.Caption := 'Файл: ' + listbox1.Items[listbox1.Itemindex];
form1.Label9.Caption := 'Статус: открыт';
p := 0;
m :=0;
k := 1;
h := 1;

for i:=0 to form1.StringGrid1.ColCount - 1 do begin

    for j:=1 to form1.StringGrid1.RowCount - 1 do begin

        end;
    end;
while not (eof(input)) do begin

    p := p + 1;
    //k := k + 1;
    read(input, arr_tele[p]);

    form1.stringgrid1.Cells[m, k] := arr_tele[p].name;
    m := m + 1;
    form1.stringgrid1.Cells[m, k] := arr_tele[p].tarif;
    m := m + 1;
    form1.stringgrid1.Cells[m, k] := inttostr(arr_tele[p].cost);
    m := m + 1;
    form1.stringgrid1.Cells[m, k] := arr_tele[p].trafic;
    m := m + 1;
    form1.stringgrid1.Cells[m, k] := inttostr(arr_tele[p].free_min);
    m := m + 1;
    form1.stringgrid1.Cells[m, k] := inttostr(arr_tele[p].lovenumber);
    m := m + 1;
    form1.stringgrid1.Cells[m, k] := arr_tele[p].date;
    m := 0;

    if(k > form1.stringgrid1.RowCount -1) then begin

        form1.stringgrid1.RowCount:= form1.stringgrid1.RowCount + 1;
        end;
        k := k + 1;
    end;
closefile(input);
```

Рисунок 3.21 — Исходный код функции открытия файла

Исходный код функции сохранения файла в котором данные из таблицы разрабатываемой программы будут сохранены в типизированный файл ранее открытый в программе представлен на рисунке 3.22.

					Лист	
					32	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

```
if(form1.open_file <> 1) then begin

    form1.N4Click(sender);
end
else begin
n := 0;
assignfile(input, file_name);
rewrite(input);
seek(input, 0);
i := 0;

for i:=0 to stringgrid1.ColCount - 1 do begin

    for j := 1 to stringgrid1.RowCount - 1 do begin

        arr_tele[j].name := stringgrid1.Cells[0, j];
        arr_tele[j].tarif := stringgrid1.Cells[1, j];
        arr_tele[j].cost := strtoint(stringgrid1.Cells[2, j]);
        arr_tele[j].trafic := stringgrid1.Cells[3, j];
        arr_tele[j].free_min := strtoint(stringgrid1.Cells[4, j]);
        arr_tele[j].lovenumber := strtoint(stringgrid1.Cells[5, j]);
        arr_tele[j].date := stringgrid1.Cells[6, j];

    end;

end;
n := n + 1;
b := stringgrid1.RowCount - 1;

for k:=1 to b do begin
    write(input, arr_tele[k]);
end;
closefile(input);
form1.Label9.Caption := 'Статус: сохранён';
timer2.Enabled := true;
form1.open_file := 2;
```

Рисунок 3.21 — Исходный код функции сохранения файла

Исходный код функции сохранения в «Excel» в котором данные из таблицы сохраняются в файл «.xls» для работы с данными в программе «Excel» показан на рисунке 3.22.

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	33
01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ					

```
begin
  ExcelApp := CreateOleObject('Excel.Application');
  try
    ExcelApp.Visible := false;

    ExcelApp.Workbooks.Add;
    Sheet := ExcelApp.ActiveWorkbook.Worksheets[1];

    for Col := 0 to stringgrid1.ColCount - 1 do begin

      for Row := 0 to stringgrid1.RowCount - 1 do begin

        Sheet.Cells[Row + 1, Col + 1] := Stringgrid1.Cells[Col, Row];
      end;
    end;

    ExcelApp.ActiveWorkbook.SaveAs(GetExcelFileName());
    showmessage('данные из stringgrid сохранены в таблицу excel!');

  finally
    ExcelApp.Application.Quit;
    ExcelApp := unassigned;
  end;
end;
```

Рисунок 3.22 — Исходный код функции сохранения в excel

Исходный код функции «Выбора значения» показан на рисунке 3.23.

```
if(form1.fun_code = 1) then begin

  stringgrid1.MouseToCell(X, Y, ACol, ARow);
  str := stringgrid1.Cells[ACol, ARow];
  val(str, ii, jj);
  if(jj <> 0) then begin
    end
  else begin
    i_fun := strtoint(stringgrid1.Cells[ACol, ARow]);
    form23.Label13.Caption := 'Значение: ' + stringgrid1.Cells[ACol, ARow];
    form1.fun_code := 2;
  end;
end;
```

Рисунок 3.23 — Код функции «Выбора значения»

Исходный код функции очистить поиск в котором происходит очистка таблицы от закрашенных ячеек, которое были перекрашены функцией поиска представлен на рисунке 3.24.

```

form17.draw_choice := 2;

for i:=0 to stringgrid1.ColCount - 1 do begin

for j:=1 to stringgrid1.RowCount - 1 do begin

if(stringgrid1.Cells[i, j] = str_grid) then begin
myRect.Left := i;
myRect.Top := j;
myRect.Right := i;
myRect.Bottom := j;

stringgrid1.Selection := myRect;
end;
end;
end;

for i:=0 to stringgrid1.ColCount - 1 do begin

for j:=1 to stringgrid1.RowCount - 1 do begin

myRect.Left := i+1;
myRect.Top := j + 1;
myRect.Right := i + 1;
myRect.Bottom := j + 1;

stringgrid1.Selection := myRect;
end;
end;
bitbtn17.Visible := false;

```

Рисунок 3.24 — Исходный код функции очистки поиска

Исходный код функции создания файла в котором происходит сохранение введённого значения и создания файла с названием, которое соответствует сохранённому значению представлен на рисунке 3.25.

					<div>01.2021.2-</div> <div>400101.1.ПО-21.Г-В-</div> <div>16-ПЗ</div>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		35

```
procedure TForm22.BitBtn1Click(Sender: TObject);
var SR: TSearchRec; // поисковая переменная
var FindRes: Integer;
var str, str2 : string;
var i, j, k, len: integer;
begin
|
  len := length(edit1.text);
  str := edit1.Text;

  if((str[len] = 't') and (str[len - 1] = 'a') and (str[len-2] = 'd') and (str[len-3] = '.')) then begin

    fs := TFileStream.Create(edit1.Text, fmCreate or fmOpenWrite);
    showmessage('Файл ' + ' (' + edit1.Text + ' )' + ' создан в корневой папке');

    form16.listbox1.Clear;
    FindRes := FindFirst('*.dat', faAnyFile, SR);
    while FindRes = 0 do // пока мы находим файлы (каталоги), то выполнять цикл
    begin
      form16.ListBox1.Items.Add(SR.Name); // добавление в список название
      // найденного элемента
      FindRes := FindNext(SR); // продолжение поиска по заданным условиям
    end;
    FindClose(SR); // закрываем поиск
    form22.Hide;

    end
  else begin
    showmessage('Ошибка! тип файла должен быть [.dat]');
  end;
end;
```

Рисунок 3.25 — Исходный код функции создания файла

Исходный код функции заблокировать в котором происходит проверка правильности введенных данных с последующий проверкой на соответствие паролю представлен на рисунке 3.26.

```
procedure TForm18.BitBtn1Click(Sender: TObject);
var prof: string;
begin
  //pass := '';
  prof := 'admin';

  if(pass = prof) then begin

    pass := '';
    edit1.Clear;
    form1.show;
    form18.Close;
  end
  else begin
    pass := '';
    edit1.Clear;
    showmessage('Ошибка! Пароль не верный. Повторите попытку снова.');
```

Рисунок 3.26 — Исходный код функции заблокировать

3.3 Контроль корректности ввода данных

Проверка корректности ввода данных в функции добавлении и изменении заключается в том, что если пользователь введёт некорректные данные в поля для ввода, то появится сообщение об ошибке. Программой предусмотрено чтобы в поля: стоимости абонентской платы, интернет трафик, количество бесплатных минут, количество любимых номеров не возможно было ввести символы или отрицательные значения. Код проверки корректности ввода в функции добавления представлен на рисунке 3.27.

```
str := edit3.Text;
val(str, i, n);
if((n <> 0) or (i < 0)) then begin

    edit3.Color := clred;
    ShowMessage('Ошибка! Нельзя вводить символы или отрицательные значения в [стоимость абоненской платы].');
end
else begin

    edit3.Color := clwhite;
    str2 := edit2.Text;
    val(str2, i, b);
    if((b <> 0) or (i < 0)) then begin
        edit2.Color := clred;
        ShowMessage('Ошибка! Нельзя вводить символы или отрицательные значения в [количество бесплатных минут].');
    end
    else begin

        edit2.Color := clwhite;
        str3 := edit1.Text;
        val(str3, i, h);
        if((h <> 0) or (i < 0)) then begin
            edit1.Color := clred;
            ShowMessage('Ошибка! Нельзя вводить символы или отрицательные значения в [количество любимых номеров].');
        end
        else begin
            edit1.Color := clwhite;
```

Рисунок 3.27 — Проверка корректности ввода данных при добавлении и изменении

Проверка корректности ввода данных в функции создания файла заключается в том, чтобы было не возможно ввести не корректное расширение файла в поле для ввода названия файла. Код проверки корректности ввода данных в функции создания файла представлен на рисунке 3.28.

```
len := length(edit1.text);
str := edit1.Text;

if((str[len] = 't') and (str[len - 1] = 'a') and (str[len-2] = 'd') and (str[len-3] = '.')) then begin
```

Рисунок 3.28 — Проверка корректности ввода данных в функции создания файла

В разрабатываемой программе присутствует проверка на наличие пустого поля для ввода. Если программа обнаружит, что в поле для ввода нет хотя бы одного символа или текста, то она выдаст ошибку. Код проверки на наличие пустого поля для ввода представлен на рисунке 3.29.


```
str:=combobox1.text;
n := 0;

if (str <> '') then begin
```

Рисунок 3.29 — Код проверки пустого поля для ввода

В разрабатываемой программе предусмотрена проверка на правильность введенных данных в поля для ввода пароля на форме «Заблокировать», то есть, если были введены не правильные данные, которое не соответствуют заданному паролю, то программа выдаст ошибку. Код проверки на правильность ввода пароля показан на рисунке 3.30.

```
prof := 'admin';

if (pass = prof) then begin

    pass := '';
    edit1.Clear;
    form1.show;
    form18.Close;
end
else begin
    pass := '';
    edit1.Clear;
    showmessage('Ошибка! Пароль не верный. Повторите попытку снова.');
```

Рисунок 3.30 — Код проверки на правильность ввода пароля на форме «Заблокировать»

В программе предусмотрена проверка на правильность выбранного файла, то есть, если был выбран файл в расширении не поддерживаемой программой, то она выдаст ошибку. Код проверки выбранного файла показан на рисунке 3.31.

```
if (opendialog1.Execute) then begin
    len := length(opendialog1.FileName);
    str := opendialog1.FileName;

    if ((str[len] = 't') and (str[len - 1] = 'a') and (str[len-2] = 'd') and (str[len-3] = '.')) then begin
        assignfile(input, opendialog1.FileName);
```

Рисунок 3.31 — Код проверки выбранного файла

В программе предусмотрены проверка на выход из разрабатываемой программы с последующим появлением окна с просьбой выполнить одно из двух действий: сохранить файл или не сохранять файл. И в зависимости от выбранного значения данные будут сохранены в файл или нет. Код при закрытии программы показан на рисунке 3.32.

```
-
WND := Form1.Handle;
lpCaption := 'Сохранить таблицу ?';
lpText := 'Хотите ли вы сохранить файл ?';
Tip := MB_YESNO + MB_ICONINFORMATION + MB_DEFBUTTON1;
case MessageBox(WND, lpText, lpCaption, Tip) of
  IDYES: begin
    form1.save1Click(Sender);
    cls_qeru := True;
    Application.Terminate;
    end;
  IDNO : Application.Terminate;
end;
end;
```

Рисунок 3.32 — Код вызываемый при закрытии программы

При поиске данных в таблице программа сначала проверяет данные которое находятся в ячейке таблицы с искомыми данными, которые ввёл пользователь для поиска, и при наличии искомого значение в ячейке, он закрашивает её, а если нет, то поиск проходит данную ячейку и не закрашивает её. Код проверки данных в таблицы с искомыми данными для поиска показан на рисунке 3.33.

```
str2 := form1.StringGrid1.Cells[i, j];

n := pos(str, str2);

if (n = 0) then begin
  continue;
end
else if (n <> 0) then begin
```

Рисунок 3.33 — Код проверки данных в таблице с искомыми данными

4 РЕЗУЛЬТАТ РАБОТЫ

Форма «Приветствия» курсовой работы показана на рисунке 4.1.

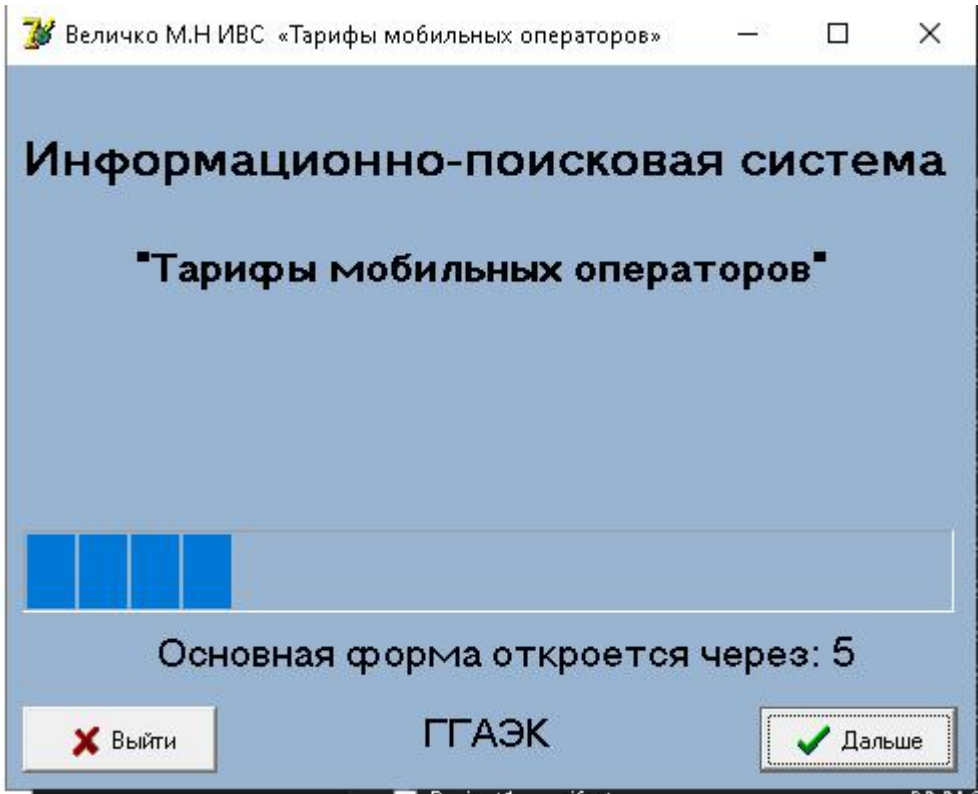


Рисунок 4.1 — Форма «Приветствия»

Форма «Приветствия» встречает пользователя при открытии программы. На данной форме есть такие компоненты, как: Кнопка «Далее», кнопка «Выйти». Кнопка «Выйти» отвечает за выход из программы. Кнопка «Далее» отвечает за то, когда пользователь нажмёт на данную кнопку, откроется главная форма программы без автоматического открытия главной формы таймером, который находится на форме «Приветствия» и отсчёт которого визуализируется в компоненте «ProgressBar».

Главная форма курсовой работы, на которой располагаются основные компоненты для работы с входными данными предоставляемыми пользователем и выходными данными, которое создаёт программа представлена на рисунке 4.2.

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						40
Изм.	Лист	№ докум.	Подп.	Дата		

ИПС "Тарифы мобильных операторов"

ФайлОчистить поискФункцииПомощьСвязьОб авторе

Файл: inp.dat

Статус: открыт

Row: 10

Col: 6

заблокировать

Мобильный оператор	Тариф	Стоимость обоненс	Интернет трафик (М	Количество беспла	Количество любим	Дата появления тар
Life :)	Комфорт L	65	34	23	65	03.05.2021
A1	Без Лимита	12	65	76	23	03.05.2021
МТС	Бизнес	87	2	23	65	03.05.2021
БелСел	Эконом	87	123	23	65	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021
МТС	Без Лимита	76	123	7	0	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021
БелСел	Звонки	87	12	34	87	03.05.2021
Life :)	Комфорт L	87	234	34	6	03.05.2021
МТС	Комфорт L	76	23	456	87	03.05.2021

добавление

изменить

Поиск

Сортировка

Фильтрация

Статистика

Удаление оператора

Рисунок 4.2 — Главная форма курсовой работы

На главной форме курсовой работы располагаются основные компоненты, которые оперируют входными данными предоставляемые пользователем разрабатываемой программы. На основной форме располагаются, такие компоненты для работы со входными данными, как: «Добавление», «Изменение», «Поиск», «Сортировка», «Фильтрация», «Статистика», «Удаление оператора», «Добавления нового тарифа».

При нажатии на кнопку «Добавление» открывается форма «Добавление» в которой происходит добавление новых данных в таблицу «Stringgrid», которая располагается на основной форме данной курсовой работы.

При нажатии на кнопку «Изменение» открывается форма, в которой пользователь изменяет ранее введенные данные из таблицы, или данные, которой содержались в файле, и который пользователь открыл в данной курсовой работе.

После нажатии на кнопку «Поиск» открывается форма «Поиск» в которой пользователь может задать значения для поиска, и найденные данные будут подсвечены в таблице «Stringgrid» на главной форме данной курсовой работы.

При нажатии на кнопку «Сортировка» открывается форма «Сортировка» в которой пользователь может отсортировать данные и сохранить или же не сохранять отсортированные данные в таблицу «Stringgrid» на основной форме программы.

После нажатии на кнопку «Фильтрация» открывается форма «Фильтрация» в которой пользователь может отфильтровать данные и сохранить или же не сохранять отфильтрованные данные в таблицу «Stringgrid» на основной форме программы.

При нажатии на кнопку «Статистика» открывается форма «Статистика» в которой пользователь может посмотреть статистику по данным из основной таблицы с помощью графиков.

После нажатии на кнопку «Удаление оператора» открывается форма «Удаление оператора» в которой пользователь может выбрать мобильного оператора из списка существующих и удалить выбранного оператора из таблицы «Stringgrid», которая располагается на основной форме программы.

После нажатии на кнопку «Заблокировать» открывается форма «Заблокировать» в которой нужно ввести пароль для входа на основную форму для продолжения работы в программы. В компоненте «Файл», который располагается в главном меню данной курсовой работы, располагаются компоненты: «Открыть», «Сохранить», «Сохранить как», «Сохранить в «Excel», «Настройки», «Выйти».

При нажатии на компонент «Очистить поиск» в главном меню программы, очищается, вернее сказать, изменяется цвет выделенных ячеек, которые были использованы функцией «Поиск».

При нажатии на компонент «Помощь» в главном меню программы, открывается сайт, где подробно расписано про каждый компонент данной курсовой работы с описанием данных, которой нужно вводить в данные компоненты.

После нажатии на кнопку «Связь» в главном меню программы, открывается форма «Связь», где пользователь может написать сообщение на почту разработчиков данной программы.

При нажатии на компонент «Об авторе» в главном меню программы, открывается форма «Об авторе», где находятся ссылки на соцсети разработчика данной курсовой работы.

Форма «Добавления» курсовой работы на которой располагаются компоненты для ввода данных и их сохранения показана на рисунке 4.3.

Добавление

Мобильный оператор

Тариф

Стоимость абонентской платы

Интернет трафик

Количество любимых номеров

Количество бесплатных минут

Дата появления тарифа

03.05.2021

OK

Рисунок 4.3 — Форма «Добавление»

На форме «Добавление» разрабатываемой программы представлены компоненты для ввода данных от пользователя. На данной форме есть такие компоненты, как: «Мобильный оператор», «Тариф», «Стоимость абонентской платы», «Интернет трафик», «Количество любимых номеров», «Количество бесплатных минут», «Дата появления тарифа», кнопка «Сохранить». В поле выбора «Тариф» хранятся заранее сохранённые мобильные операторы страны и пользователь выбирает нужного ему оператора из списка предложенных и выбранное значение после сохранения добавляется в колонку «Тариф» в главной таблице данной курсовой работы, которой находится на главной форме программы. В поле ввода «Стоимость абонентской платы» пользователь вводит стоимость платы за конкретный тариф мобильного оператора предложенных и выбранное значение после сохранения добавляется в колонку «Стоимость абонентской платы» в главной таблице разрабатываемой программы, которой находится на главной форме программы. В поле ввода «Интернет трафик» пользователь данной программы вводит количество интернет трафика для конкретного клиента. В поле ввода «Количество любимых номеров» пользователь вводит количество любимых номеров для конкретного клиента мобильного оператора предложенных и выбранное значение после сохранения добавляется в колонку «Количество

любимых номеров» в главной таблице данной курсовой работы, которой находится на главной форме программы. В поле ввода «Количество бесплатных минут» пользователь данной программы вводит количество бесплатных минут, то есть, то количество минут внутри сети мобильного оператора, которой клиент выбранного мобильного оператора может использовать без уменьшения баланса на счёте и выбранное значение после сохранения добавляется в колонку «Количество бесплатных минут» в главной таблице разрабатываемой программы. В поле выбора «Дата появления тарифа» пользователь программы выбирает в календаре конкретный день и год, когда у клиента выбранного мобильного оператора появился выбранный тариф и выбранное значение после сохранения добавляется в колонку «Дата появления тарифа» в главной таблице разрабатываемой программы. Пример выбора даты показан на рисунке 4.6.

Пример ввода данных на форме добавления представлен на рисунке 4.4. При нажатии на кнопку «Сохранить» программа сначала проверяет правильность введённых данных в полях для ввода данных, и если программа не обнаружит ошибок, то она сохранит введённые данные в таблицу «Stringgrid» на главной форме разрабатываемой программы. Пример сохранённых данных в таблицу показан на рисунке 4.5.

Добавление

×

Мобильный оператор

A1

Тариф

Комфорт L

Стоимость абоненской платы

76

Интернет трафик

23

Количество любимых номеров

34

Количество бесплатных минут

76

Дата появления тарифа

03.05.2021

OK

Рисунок 4.4 — Пример введенных данных на форме добавления

Если же программа обнаружит ошибки в введенных данных, она сначала оповестит пользователя, что в конкретном компоненте для ввода присутствует ошибка и подсветит данное поле в красный цвет. Пример вывода ошибки показан на рисунке 4.4.

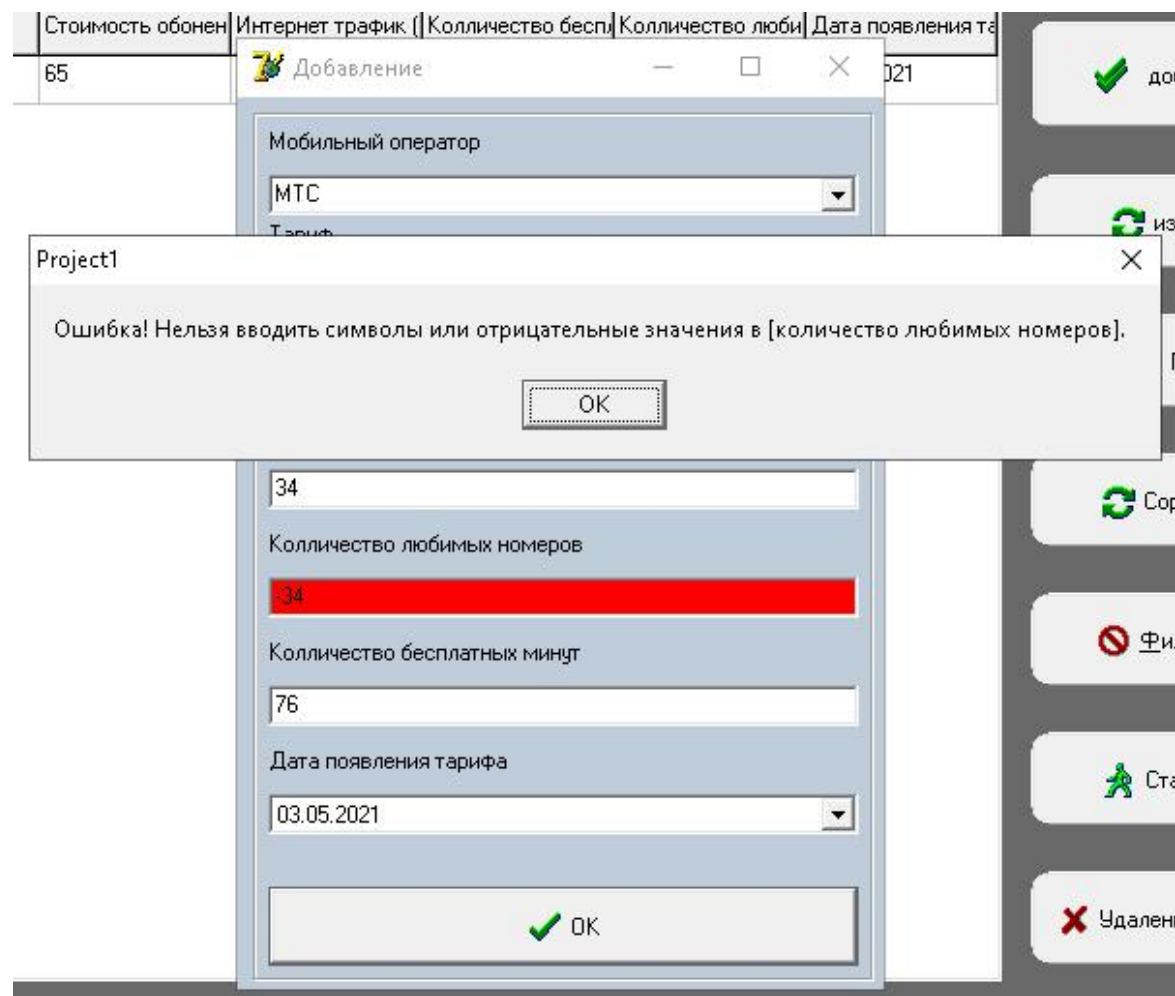


Рисунок 4.5 — Пример вывода сообщения об ошибке

При избавлении от ошибок программа так же сохранит введенные данные в таблицу «Stringgrid» на главной форме курсовой работы. Результат ввода данных на форме добавления показан на рисунке 4.5.

ИПС "Тарифы мобильных операторов"

ФайлОчистить поискФункцииПомощьСвязьОб авторе

Файл: inp.dat

Статус: открыт

Row: 11

Col: 6

заблокировать

Мобильный оператор	Тариф	Стоимость обзвона	Интернет трафик (М	Количество беспла	Количество любим	Дата появления тар
Life :)	Комфорт L	65	34	23	65	03.05.2021
A1	Без Лимита	12	65	76	23	03.05.2021
МТС	Бизнес	87	2	23	65	03.05.2021
БелСел	Эконом	87	123	23	65	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021
МТС	Без Лимита	76	123	7	0	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021
БелСел	Звонки	87	12	34	87	03.05.2021
Life :)	Комфорт L	87	234	34	6	03.05.2021
МТС	Комфорт L	76	23	456	87	03.05.2021
A1	Комфорт L	76	23	76	34	03.05.2021

добавление

изменить

Поиск

Сортировка

Фильтрация

Статистика

Удаление оператора

Рисунок 4.5 — Пример сохранённых данных после добавления

Дата появления тарифа

03.05.2021

май 2021 г.

пн

вт

ср

чт

пт

сб

вс

26

27

28

29

30

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

1

2

3

4

5

6

Сегодня: 20.06.2021

Рисунок 4.6 — Выбора даты появления тарифа

Форма «Изменить» курсовой работы на которой располагаются компоненты для изменения выбранной строки показана на рисунке 4.7.

Изменить

Мобильный оператор

Тариф

Стоимость абонентской платы

Интернет трафик

Количество любимых номеров

Количество бесплатных минут

Дата появления тарифа

03.05.2021

?

Проверить

А

Скопировать

И

Исправить ошибки

✓

применить

Рисунок 4.7 — Форма «Изменить»

На форме «Изменить» данной курсовой работы представлены компоненты для ввода данных от пользователя. На данной форме есть такие компоненты, как: «Мобильный оператор», «Тариф», «Стоимость абонентской платы», «Интернет трафик», «Количество любимых номеров», «Количество бесплатных минут», «Дата появления тарифа», кнопка «Сохранить». В поле выбора «Мобильный оператор» хранится ранее добавленный мобильный оператор, который был введён пользователем на форме «Добавление», и при надобности пользователь данной программы может заменить уже сохранённого оператора на нужного ему из списка предложенных. В поле выбора «Тариф» хранится ранее добавленный тариф мобильного оператора, который был введён пользователем на форме «Добавление», и при надобности пользователь данной программы может заменить уже сохранённый тариф на нужный ему из списка предложенных. В поле ввода «Стоимость абонентской платы» хранится ранее добавленное значение «Стоимости абонентской платы», которое было введено пользователем на форме «Добавление», и при надобности пользователь данной программы может заменить уже сохранённое

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	47
01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ					

значение стоимости на нужное ему. В поле ввода «Интернет трафик» хранится ранее добавленное значение «Интернет трафика», которое было введено пользователем на форме «Добавление», и при надобности пользователь данной программы может заменить уже сохранённое значение на нужное ему. В поле ввода «Количество любимых номеров» хранится ранее добавленное значение, которое было введено пользователем на форме «Добавление», и при надобности пользователь данной программы может заменить уже сохранённое значение на нужное ему. В поле ввода «Количество бесплатных минут» хранится ранее добавленное значение, которое было введено пользователем на форме «Добавление», и при надобности пользователь данной программы может заменить уже сохранённое значение на нужное ему. В поле выбора «Дата появления тарифа» хранится ранее добавленное значение даты появления тарифа у конкретного пользователя данного мобильного оператора, которое было введено пользователем на форме «Добавление», и при надобности пользователь данной программы может заменить уже сохранённое значение на нужное ему выбрав из календаря нужный ему день и год. При нажатии на кнопку «Применить» программа сначала проверяет правильность введённых данных, и если все поля не имеют явных ошибок, связанных, например с вводом отрицательных значений в поля для ввода «Стоимость абонентской платы», «Интернет трафик», «Количество любимых номеров», «Количество бесплатных минут» или же ввода символом в данные поля, то программа корректно сохраняет изменённые данные в те поля в таблице на главной форме курсовой работы, где они ранее содержались. Пример изменения данных на форме изменения показан на рисунке 4.8. А если в полях для ввода данных содержатся ошибки, то программа оповестит пользователя данной программы, где именно содержится ошибка, а точнее сказать, в какой компоненте и ещё программа подсветит компонент где содержится ошибка в красный цвет, что и будет означать, что в данном компоненте есть ошибка.

Изменить

✕

Мобильный оператор

БелСел

Тариф

Звонки

Стоимость абонентской платы

2

Интернет трафик

12

Количество любимых номеров

76

Количество бесплатных минут

34

Дата появления тарифа

03.05.2021

?

Проверить

👤

Скопировать

🔄

Исправить ошибки

🔄

Изм. стоимость

✅

применить

Рисунок 4.8 — Пример изменения данных на форме изменения

При нажатии на кнопку «Проверить» на форме «Изменить» открывается форма «Проверка», где есть поле для вывода информации, где выводится информация о тех полях, где есть ошибки. На данной форме в поля для ввода указывается конкретный компонент в котором есть ошибка, строку с ошибкой и конкретный символ в строке, которой содержит ошибку. Форма «Проверить» показана на рисунке 4.9. При нажатии на кнопку «Исправить ошибки» программа исправит ошибки в тех полях для ввода данных, где содержатся ошибки. При нажатии на кнопку «Применить» программа сохранит изменённые данные в таблицу. Пример изменённых данных показан на рисунке 4.10. При нажатии на кнопку «Изм.стоимость» появится форма «Изменить стоимость», где можно изменить стоимость абонентской платы для выбранного тарифа.

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	49
01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ					

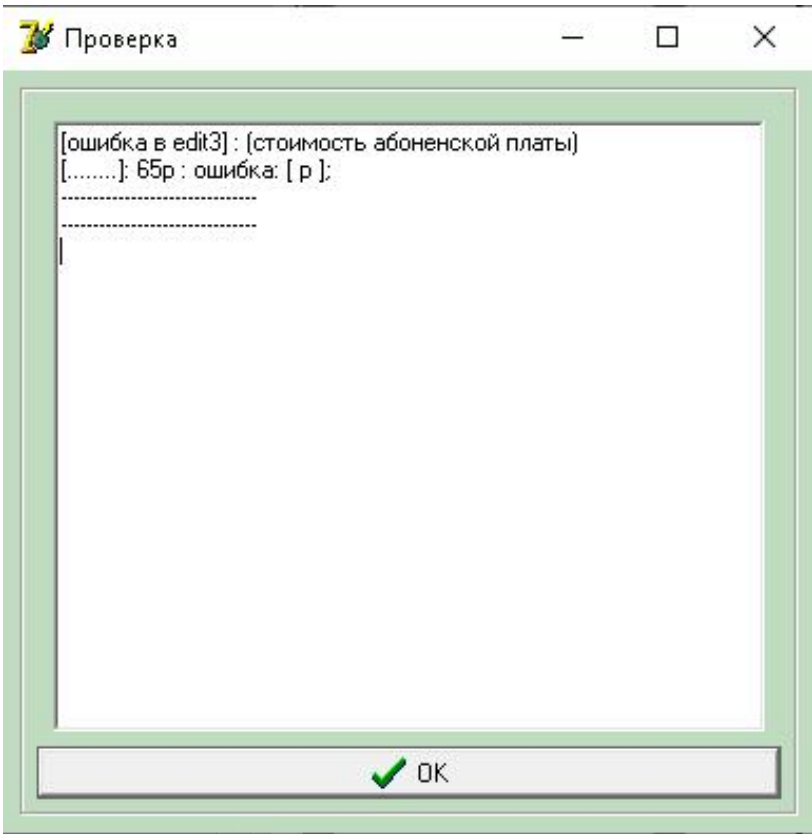


Рисунок 4.9 — Форма «Проверить»

Мобильный оператор	Тариф	Стоимость абоненс	Интернет трафик (М	Количество беспла	Количество любим	Дата появления тар
Life :)	Комфорт L	65	34	23	65	03.05.2021
A1	Без Лимита	12	65	76	23	03.05.2021
МТС	Бизнес	87	2	23	65	03.05.2021
БелСел	Эконом	87	123	23	65	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021
МТС	Без Лимита	76	123	7	0	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021
БелСел	Звонки	2	12	76	34	03.05.2021
Life :)	Комфорт L	87	234	34	6	03.05.2021
МТС	Комфорт L	76	23	456	87	03.05.2021
A1	Комфорт L	76	23	76	34	03.05.2021

Рисунок 4.10 — Пример изменнёх данных в таблице

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	50

Форма «Поиск» данной курсовой работы, где располагаются компоненты для поиска данных в таблице показана на рисунке 4.11.

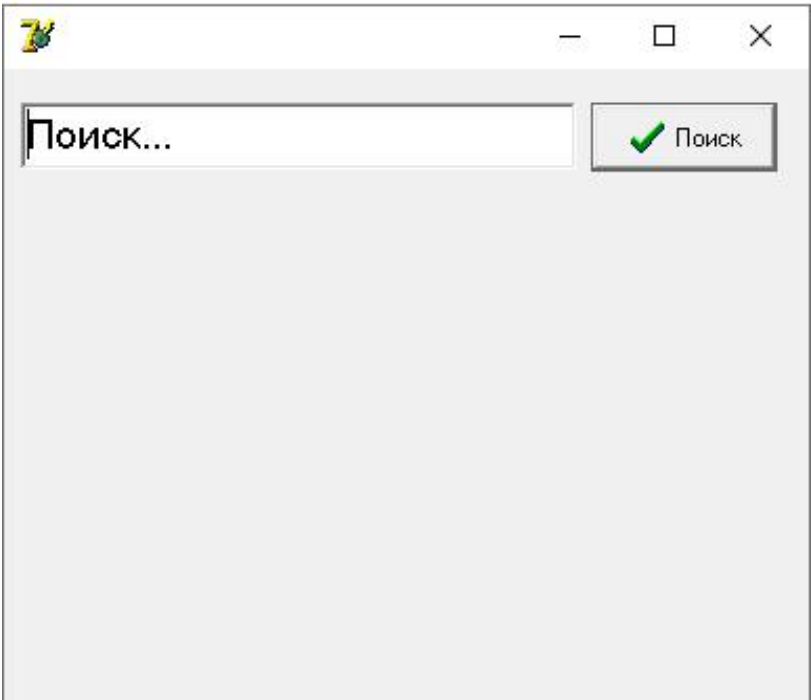


Рисунок 4.11 — Форма «Поиск»

На форме «Поиск» разрабатываемой программы располагаются компоненты для поиска информации. Для поиска по таблице сначала нужно ввести искомое значение в поле «Поиск...» на форме «Поиск». Пример ввода значения показан на рисунке 4.12.




Рисунок 4.12 — Пример ввода данных на форме поиска

					<i>01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ</i>	Лист
						51
Изм.	Лист	№ докум.	Подп.	Дата		

При наборе искомого значение в появившемся компоненте «ListBox» будут появляться только те значение из таблицы «Stringgrid» в которых есть искомое значение, которой вводит пользователь данной программы. После успешного набора искомого значения или выбора значения из компонента «ListBox», пользователь нажимает на кнопку «Поиск» на форме «Поиск» после чего форма «Поиск» закрывается, а в таблице «Stringgrid» на главной форме, ракрашиваются ячейки, который содержали искомое значение. Пример поиска в таблице показана на рисунке 4.13.

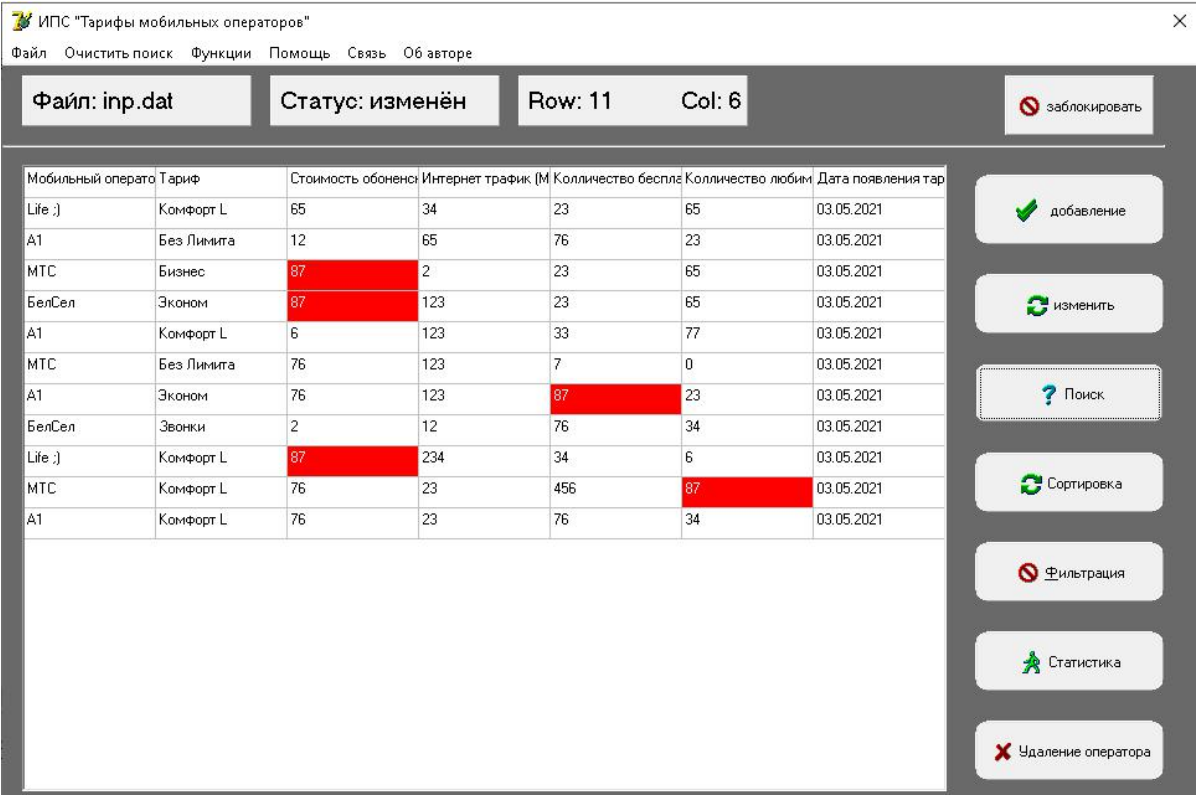


Рисунок 4.13 — Пример поиска в таблице

При нажатии на кнопку «Очистить поиск» в компонента «MeinMenu» на главной форме курсовой работы, таблица, в которой были закрашены ячейки, будет перекрашена в изначальный цвет, то есть, искомые значения, который были выделены при поиске будут очищены.

Форма «Фильтрация» данной курсовой работы, где располагаются компоненты для фильтрации данных в таблице показана на рисунке 4.14.

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						52
Изм.	Лист	№ докум.	Подп.	Дата		

Фильтрация

Очистить Вернуться к изначальной таблице

Мобильный оператор	Тариф	Стоимость обоненс	Интернет трафик	Количество беспла	Количество любим	Дата появления
Life :)	Комфорт L	65	34	23	65	03.05.2021
A1	Без Лимита	12	65	76	23	03.05.2021
МТС	Бизнес	87	2	23	65	03.05.2021
БелСел	Эконом	87	123	23	65	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021
МТС	Без Лимита	76	123	7	0	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021
БелСел	Звонки	2	12	76	34	03.05.2021
Life :)	Комфорт L	87	234	34	6	03.05.2021

Записи с любимыми номерами

Фильтровать по:

Начать фильтрацию

Сохранить

Выйти

Рисунок 4.14 — Форма «Фильтрация»

На форме «Фильтрация» данной курсовой работы представлены компоненты для фильтрации данных в таблице. После нажатия на кнопку «Фильтрация» на главной форме курсовой работы откроется форма «Фильтрация» и данные из основной таблицы будут перенесены в таблицу на форме «Фильтрация». Для фильтрации «По любимым номерам» нужно нажать на кнопку «Записи с любимыми номерами» на форме «Фильтрация» после в таблице «Stringgrid» будут сохранены только те строки, который содержали более нуля любимых номеров из колонки «Количество любимых номеров». Пример фильтарции по любимым номерам покзаан на рисунок 4.15. Для фильтарции по конкретной колонке нужно выбрать в компоненте «Combobox» на форме «Фильтрация» нужную нам колонку, а после ввести в рядом стоящий компонент «Combobox» значение для фильтарции и после чего нажать на кнопку «Начать фильтарция». Пример ввода значений для фильтарции показан на рисунок 4.16.

Фильтрация

Очистить Вернуться к изначальной таблице

Мобильный оператор	Тариф	Стоимость обоненс	Интернет трафик	Количество беспла	Количество любим	Дата появления тар
Life :)	Комфорт L	65	34	23	65	03.05.2021
A1	Без Лимита	12	65	76	23	03.05.2021
МТС	Бизнес	87	2	23	65	03.05.2021
БелСел	Эконом	87	123	23	65	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021
БелСел	Звонки	2	12	76	34	03.05.2021
Life :)	Комфорт L	87	234	34	6	03.05.2021
МТС	Комфорт L	76	23	456	87	03.05.2021

Записи с любимыми номерами

Фильтровать по:

Начать фильтрацию

Сохранить

Выйти

Рисунок 4.15 — Пример фильтрации по любимым номерам

Фильтрация

Очистить Вернуться к изначальной таблице

Мобильный оператор	Тариф	Стоимость обоненс	Интернет трафик	Количество беспла	Количество любим	Дата появления тар
Life :)	Комфорт L	65	34	23	65	03.05.2021
A1	Без Лимита	12	65	76	23	03.05.2021
МТС	Бизнес	87	2	23	65	03.05.2021
БелСел	Эконом	87	123	23	65	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021
БелСел	Звонки	2	12	76	34	03.05.2021
Life :)	Комфорт L	87	234	34	6	03.05.2021
МТС	Комфорт L	76	23	456	87	03.05.2021

Записи с любимыми номерами

Фильтровать по:

Оператору

A1

Начать фильтрацию

Сохранить

Выйти

Рисунок 4.16 — Пример ввода данных для фильтарции

После этого в таблице на форме «Фильтрация» будут сохранены только те строки, которое содержали искомое значение для фильтрации введенное пользователем. Пример фильтрации по значению показан на рисунке 4.17.

Фильтрация

Очистить Вернуться к изначальной таблице

Мобильный оператор	Тариф	Стоимость обонес	Интернет трафик	Количество беспла	Количество любим	Дата появления тар
A1	Без Лимита	12	65	76	23	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021

Записи с любимыми номерами

Фильтровать по:

Оператору

A1

Начать фильтрацию

Сохранить

Выйти

Рисунок 4.17 — Фильтрация по значению

При нажатии на кнопку «Вернуться к изначальной таблице» таблица на форме «Фильтрация» будет содержать все строки из таблицы «Stringgrid», которая располагается на главной форме программы, то есть, в таблице на форме «Фильтрация» будут содержаться строки из изначальной таблицы до начала фильтрации. При нажатии на кнопку «Выйти» форма «Фильтрация» закроется без сохранения таблицы с отфильтрованными данными. При нажатии на кнопку «Сохранить» таблица содержащая отфильтрованные данные будет сохранена в таблицу «Stringgrid» на главной форме программы. Пример сохранения данных в таблицу показан на рисунке 4.18.

ИПС "Тарифы мобильных операторов"

ФайлОчистить поискФункцииПомощьСвязьОб авторе

Файл: inp.dat

Статус: изменён

Row: 9

Col: 6

заблокировать

Мобильный оператор	Тариф	Стоимость обонеси	Интернет трафик (М	Количество беспла	Количество любим	Дата появления тар
Life :)	Комфорт L	65	34	23	65	03.05.2021
A1	Без Лимита	12	65	76	23	03.05.2021
МТС	Бизнес	87	2	23	65	03.05.2021
БелСел	Эконом	87	123	23	65	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021
БелСел	Звонки	2	12	76	34	03.05.2021
Life :)	Комфорт L	87	234	34	6	03.05.2021
МТС	Комфорт L	76	23	456	87	03.05.2021

добавление

изменить

Поиск

Сортировка

Фильтрация

Статистика

Удаление оператора

Рисунок 4.18 — Сохранённые данные после фильтрации

Форма «Статистика» данной курсовой работы, где располагаются компоненты для вывода статистической информации показана на рисунке 4.19.

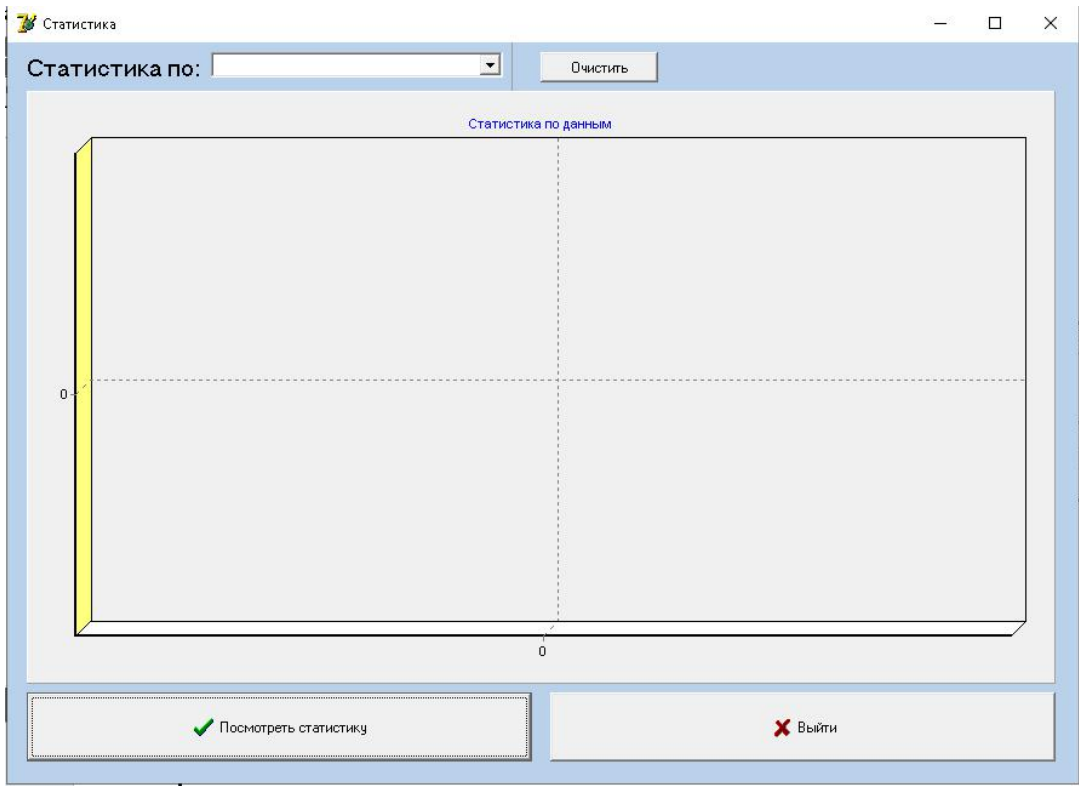


Рисунок 4.19 — Форма статистики

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	56
01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ					

На форме статистики разрабатываемой программы представлены компоненты для вывода графиков и диаграмм. Для вывода статистической информации в виде графиков необходимо выбрать по какому значению будет проводиться статистика. Для этого необходимо выбрать данное значение из списка уже сохранённых их компонента «Combobox», после чего нажать на кнопку «Посмотреть статистику». Пример выбора данных представлен на рисунке 4.20. После нажатия на кнопку «Посмотреть статистику» в поле для вывода диаграмм появится диаграмма, которая будет содержать статистическую информацию по выбранному значению для статистики. Пример вывода диаграммы представлен на рисунке 4.21. При нажатии на кнопку «Выйти» форма «Статистика» будет закрыта, а диаграммы будут очищены.

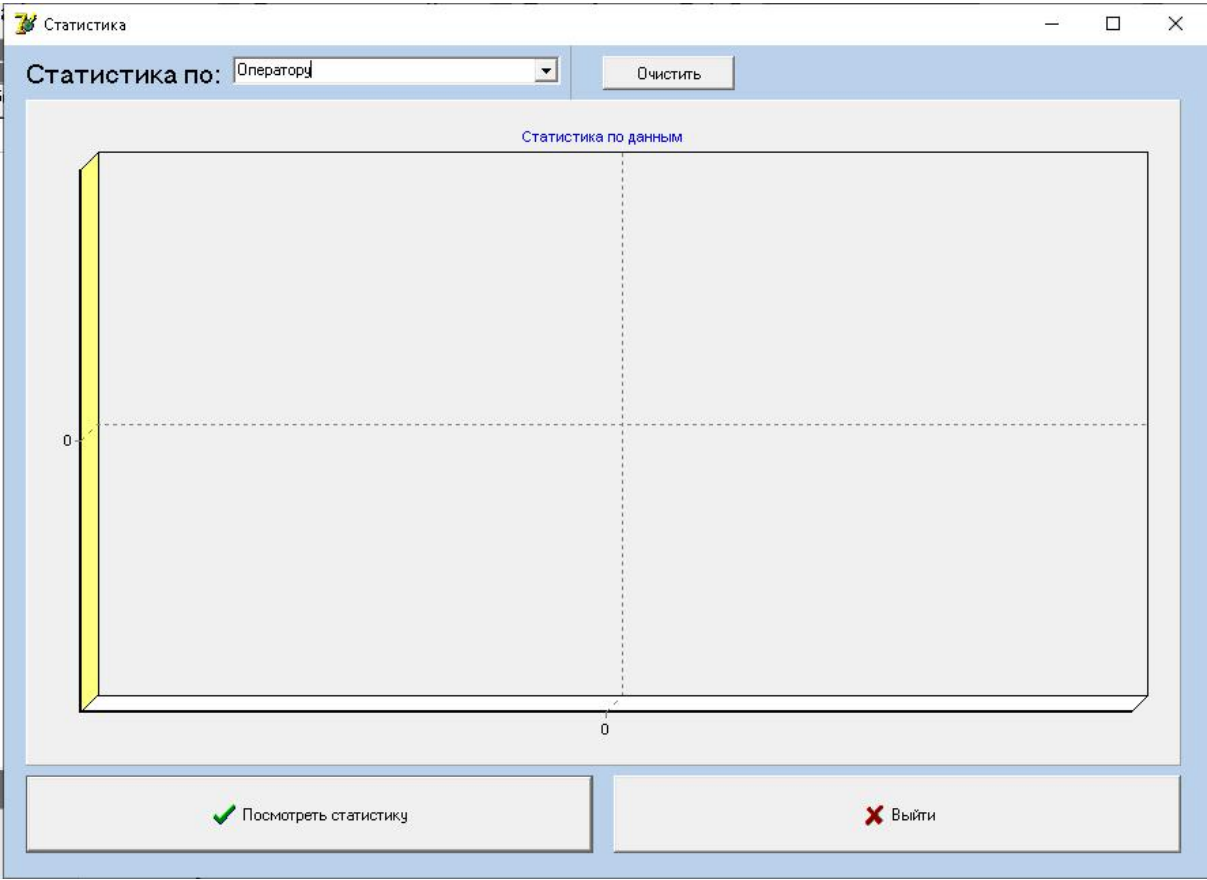


Рисунок 4.20 — Пример выбора значения для статистики



Рисунок 4.21 — Вывод диаграммы

Форма «Удаление оператора» показана на рисунке 4.22.

Рисунок 4.22 — Форма «Удаление оператора»

На форме «Удаление оператора» находятся компоненты для удаление оператора из таблицы. Для удаления конкретного оператора из таблицы необходимо выбрать его из выпадающего списка компонента «Combobox», который содержит всех операторов данной программы. Пример выбора мобильного оператора из списка показан на рисунке 4.23.

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	58
01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ					

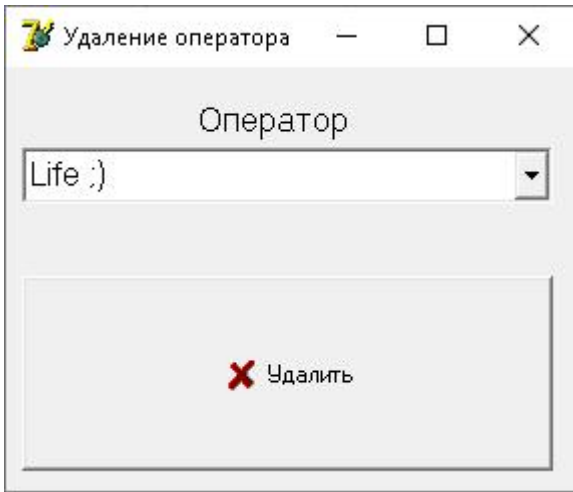
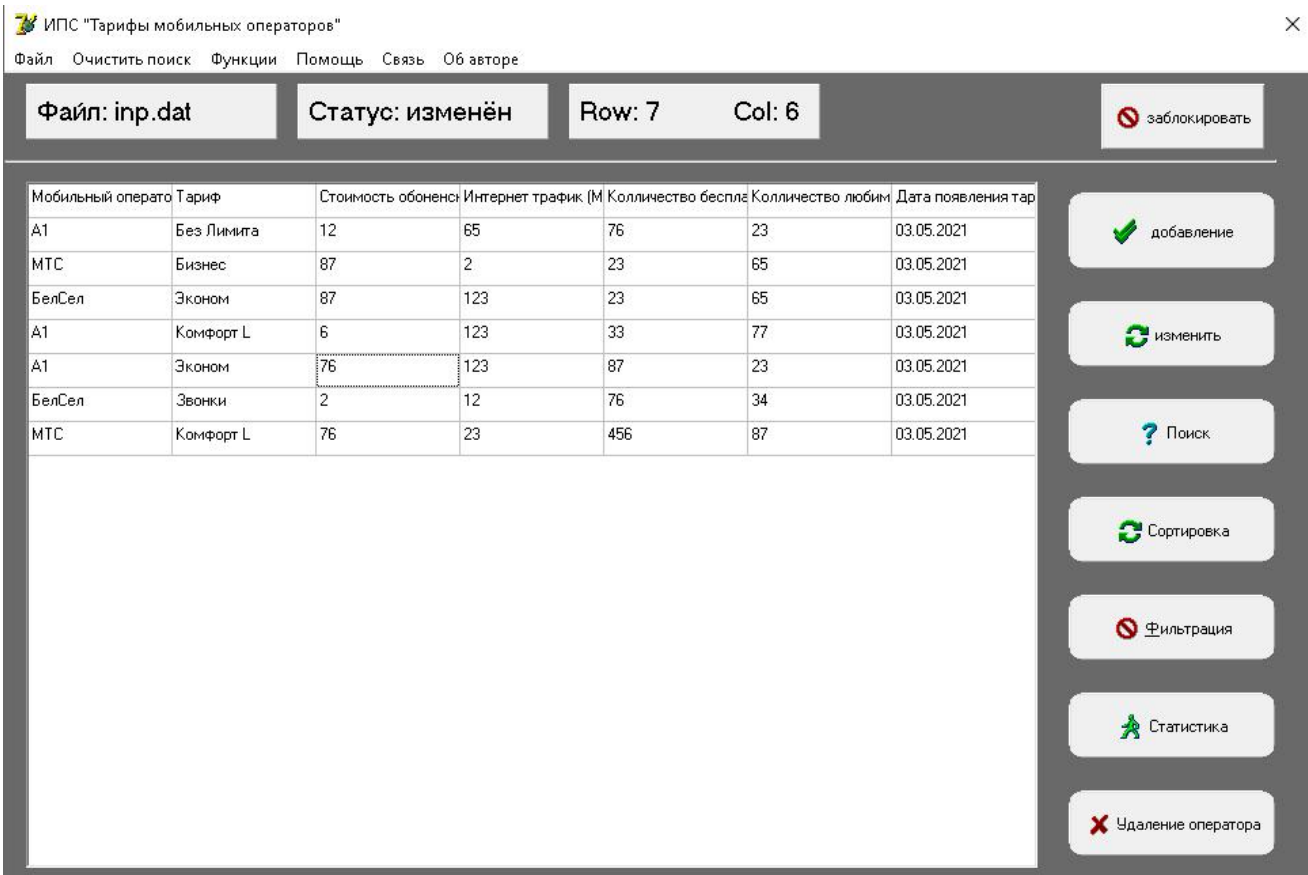


Рисунок 4.23 — Выбор оператора из списка

После чего нажать на кнопку «Удалить». После нажатия на кнопку «Удалить» форма «Удаление оператора» автоматически закроется, а в таблице «Stringgrid» на главной форме будут сохранены только те строки, которые не содержат оператора выбранного для удаления. Пример удаления оператора из таблицы показан на рисунке 4.24.



					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		59

Рисунок 4.24 — Удаления оператора из таблицы

Форма «Заблокировать» показана на рисунке 4.25.

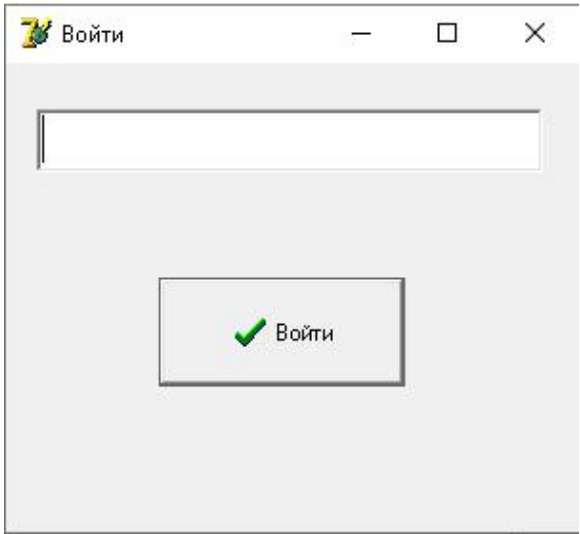


Рисунок 4.25 — Форма «Заблокировать»

Форма «Заблокировать» появляется при нажатии на кнопку «Заблокировать» на главной форме курсовой работы. Для ввода пароля нужно вводить данные в компонент «Edit» на форме «Заблокировать» после чего нужно нажать на кнопку «Войти». Пример ввода значения на форме заблокировать показн на рисунке 4.26.

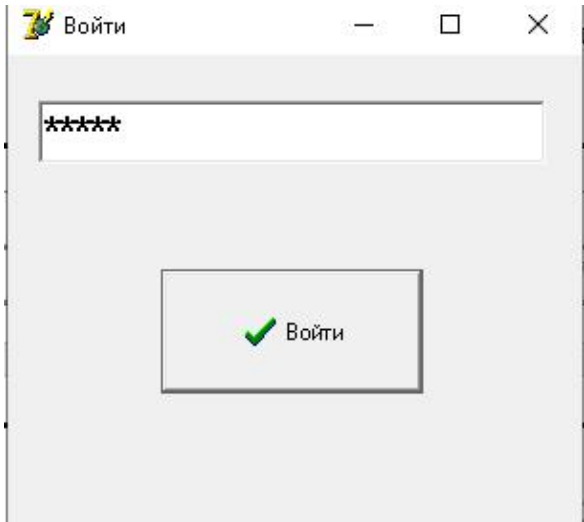


Рисунок 4.26 — Ввода данных на форме заблокировать

После ввода данных и нажатии на кнопку программа проверит правильность введённого пароля, и если пароль правильный, то откроет основную форму, а если нет, то на форме «Заблокировать» появится

					<i>01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ</i>	Лист
						60
Изм.	Лист	№ докум.	Подп.	Дата		

сообщение, что пароль введен неверно, и данные, которые были введены в компонент «Edit» будут очищены. Пример сообщения о ошибке при вводе пароля показан на рисунке 4.27.

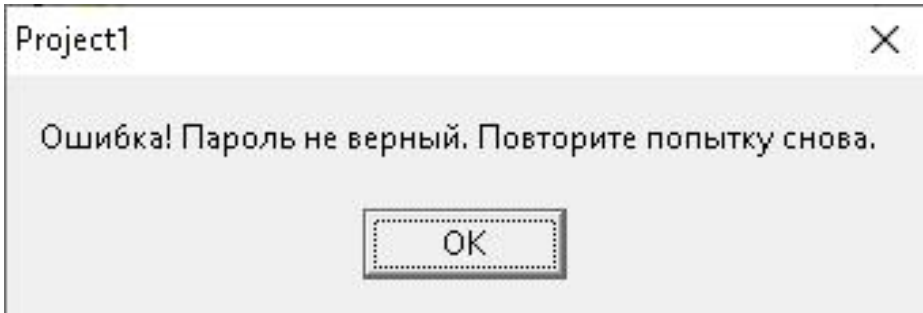


Рисунок 4.27 — Пример сообщения об ошибке ввода пароля

Форма «Об авторе» показана на рисунке 4.28.

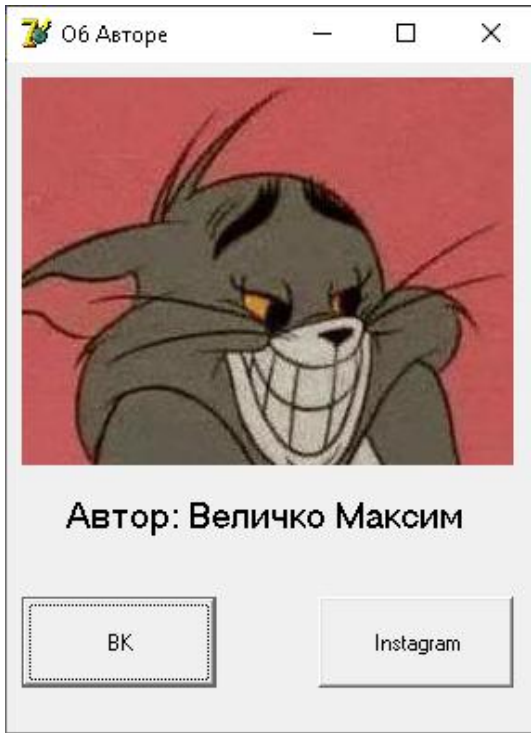


Рисунок 4.28 — Форма «Об авторе»

На форме «Об авторе» содержится информация об создателе данной программы и ссылок на его социальные сети. При нажатии на кнопку «ВК» у пользователя данной программы откроется браузер со ссылкой на страницу автора в социальной сети «Вконтакте». Пример нажатия на кнопку «ВК» показан на рисунке 4.29. При нажатии на кнопку «Instagram» у пользователя данной программы откроется браузер со ссылкой на страницу автора в социальной сети «Instagram».

					Лист	
					61	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

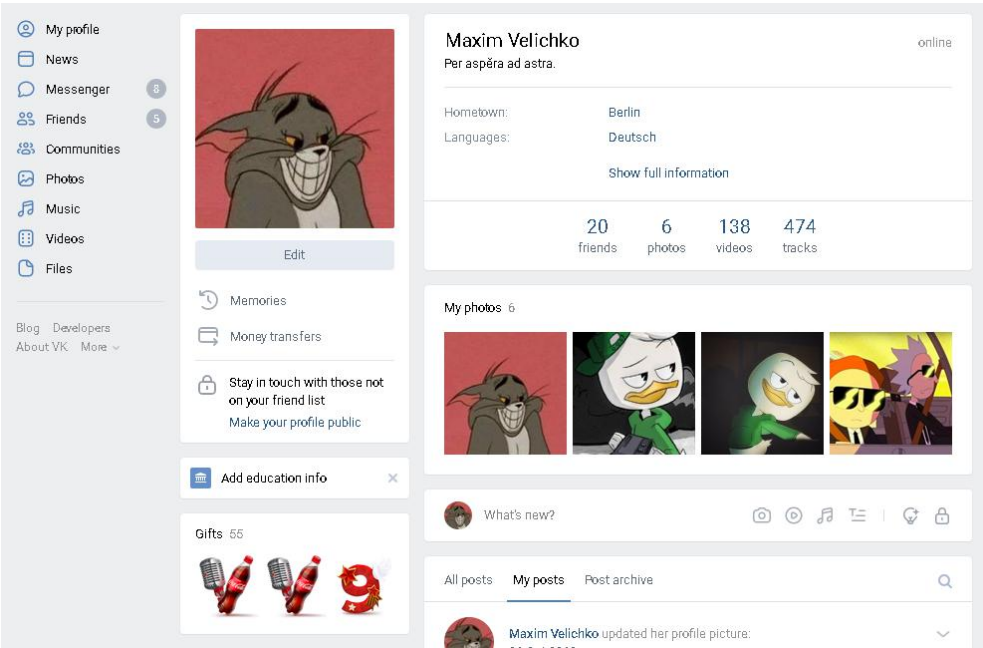


Рисунок 4.29 — Переход к социальной сети «Вконтакте»

Форма «Открытие файла» показана на рисунке 4.30.

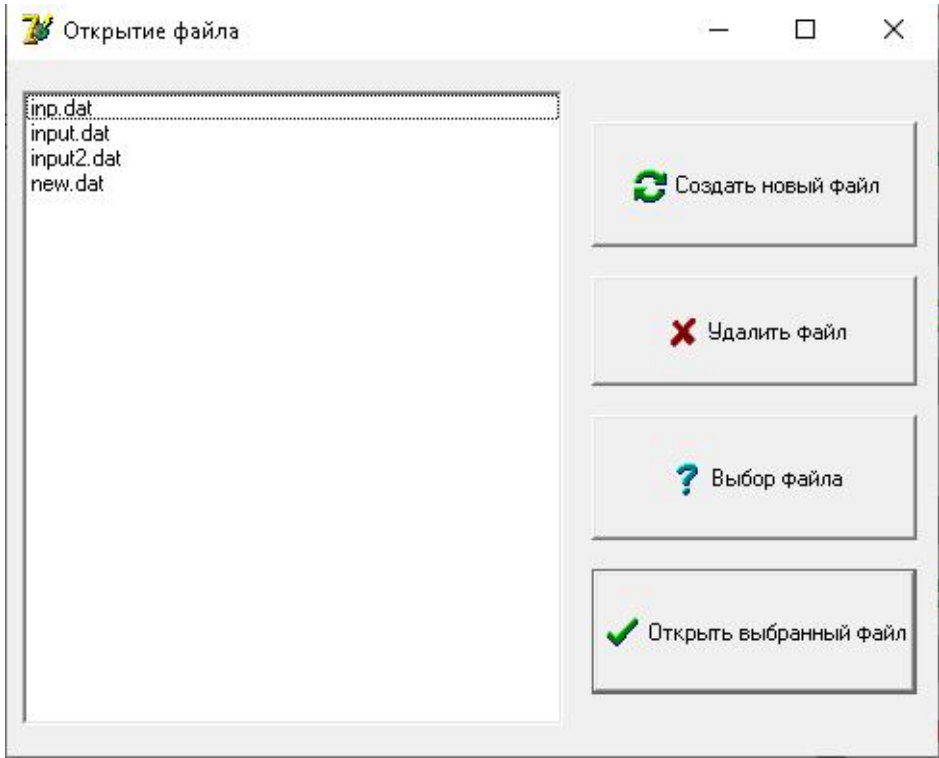


Рисунок 4.30 — Форма «Открытие файла»

На форме «Открытие файла» располагаются компоненты содержащие в себе функции для работы с файлами, а именно: «Создать новый файл»,

					Лист	
					62	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

«Удалить файл», «Выбор файла», «Открыть выбранный файл». Для открытия файл необходимо в компоненте «ListBox» нажать на тот файл, который необходимо открыть. Пример выбора файла из списка показан на рисунке 4.31.

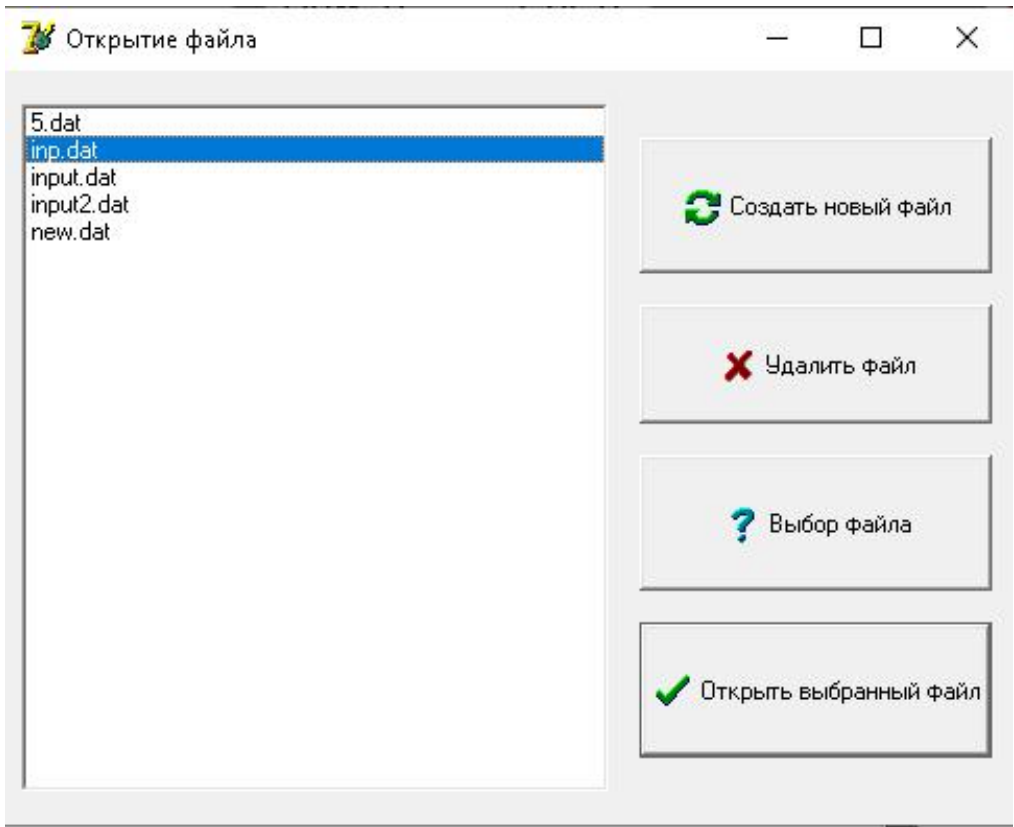


Рисунок 4.31 — Выбора файла из списка вывозимых

Файлы, которые содержатся в компоненте подгружаются в программу автоматически, то есть, если создать файл с расширением «.dat» в папке с программой, программа автоматически найдёт его и добавит название файла в компонент «ListBox». После выбора файла в компоненте «ListBox» необходимо нажать на кнопку «Открыть выбранный файл» после чего форма «Открытие файла» автоматически закроется и откроется главная форма программы, где в таблице уже будут содержаться данные из выбранного ранее файла. Для удаления файла необходимо выбрать файл в компоненте «ListBox», который нужно удалить после чего нажать на кнопку «Удалить файл». Для выбора файла из проводника операционной системы, необходимо нажать на кнопку «Выбор файла» после чего откроется проводник, где можно выбрать необходимый файл. Пример выбора файла из проводника показан на рисунок 4.32.

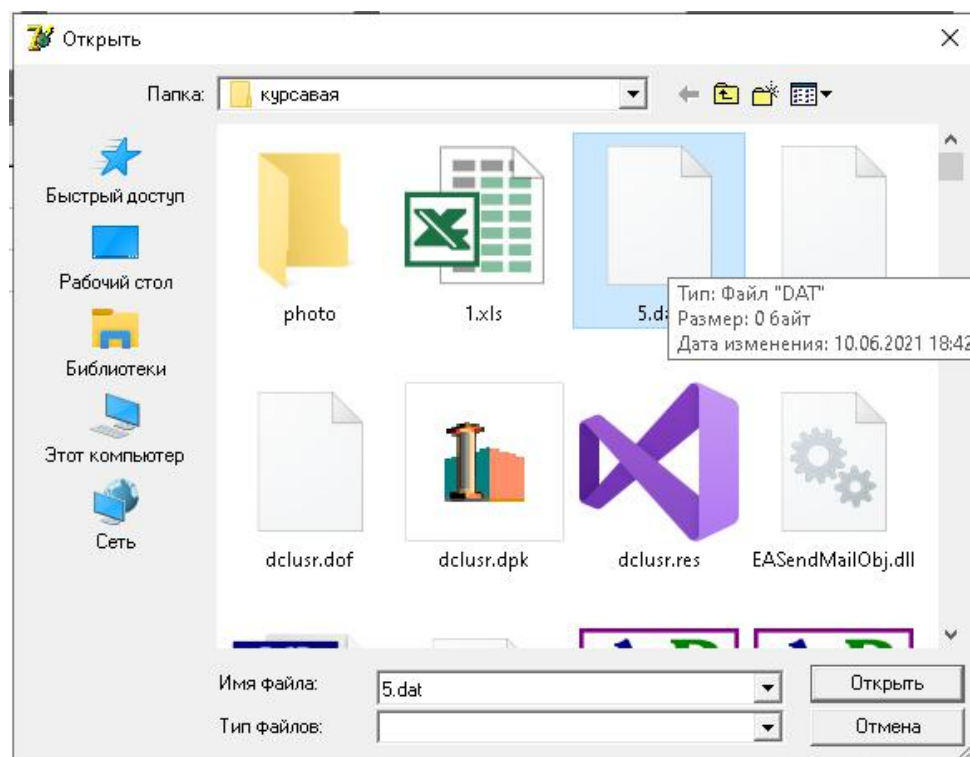


Рисунок 4.32 — Выбор файла из проводника операционной системы

Для создания нового файла необходимо нажать на кнопку «Создать новый файл» и в появившемся окне с формой «Создания файла» ввести название файла, который необходимо создать, при выполнении этих действий файл с введенным названием будет создан в папке с программой, и автоматически добавлен в компонент «ListBox» на форме «Открытие файла». Пример создания нового файла показан на рисунке 4.33.

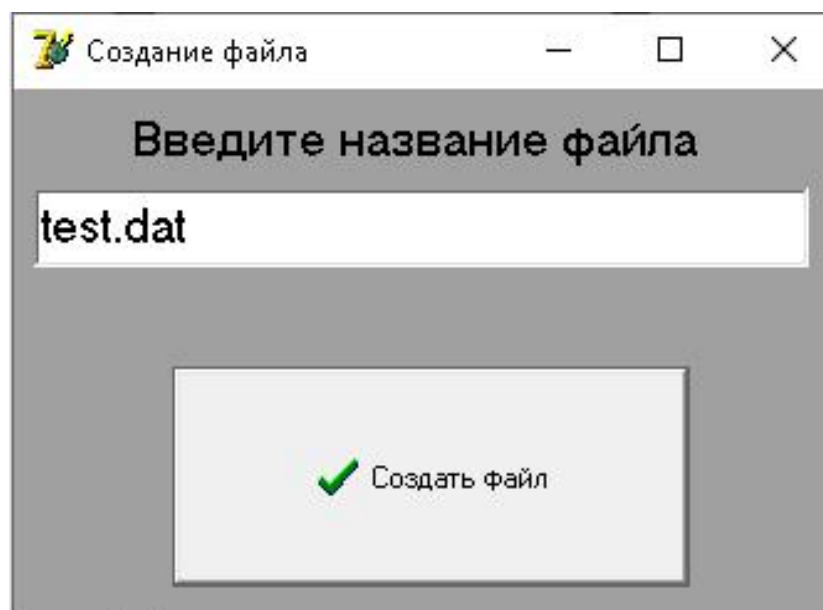


Рисунок 4.33 — Ввод названия файла на форме «Создания файла»

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	64
01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ					

Если на форме «Создания файла» ввести расширения для файла не поддерживаемой программой, то появится окно с сообщением, что введенное расширение не поддерживается программой. Сообщение с ошибкой показано на рисунке 4.34.

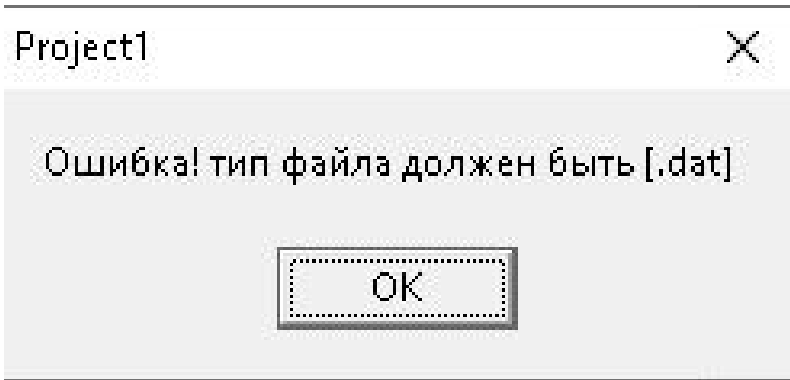


Рисунок 4.34 — Сообщения об ошибке

Форма «Сортировка» показана на рисунке 4.35.

The screenshot displays a web application window titled 'Сортировка'. It features a table with the following data:

Мобильный оператор	Тариф	Стоимость обонен	Интернет трафик	Количество беспл	Количество любии	Дата появления та
Life ;)	Комфорт L	65	34	23	65	03.05.2021
A1	Без Лимита	12	65	76	23	03.05.2021
МТС	Бизнес	87	2	23	65	03.05.2021
БелСел	Эконом	87	123	23	65	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021
МТС	Без Лимита	76	123	7	0	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021
БелСел	Звонки	87	12	34	87	03.05.2021
Life ;)	Комфорт L	87	234	34	6	03.05.2021
МТС	Комфорт L	65	23	76	23	03.05.2021

Below the table, there are three buttons: 'Сортировать выбором', 'Сортировать по убыванию', and 'Сортировать по возрастанию'. At the bottom of the form, there are two buttons: 'Сохранить и выйти' (with a green checkmark icon) and 'Выйти' (with a red X icon).

Рисунок 4.35 — Форма сортировки данных

На форме «Сортировка» располагаются компоненты для сортировки данных в таблице. Для выполнения сортировки по убыванию необходимо нажать на кнопку «Сортировка по убыванию» после чего данные в таблице будут отсортированы по убыванию в зависимости от тех данных, которой находятся в колонке «Стоимость абонентской платы». Пример сортировки по убыванию показан на риснке 4.36.

Сортировка

×

Стартовая страница

Мобильный оператор	Тариф	Стоимость абонентской платы	Интернет трафик	Количество бесплатных минут	Количество бесплатных SMS	Дата появления тарифа
МТС	Бизнес	87	2	23	65	03.05.2021
БелСел	Эконом	87	123	23	65	03.05.2021
БелСел	Звонки	87	12	34	87	03.05.2021
Life :)	Комфорт L	87	234	34	6	03.05.2021
МТС	Без Лимита	76	123	7	0	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021
Life :)	Комфорт L	65	34	23	65	03.05.2021
МТС	Комфорт L	65	23	76	23	03.05.2021
A1	Без Лимита	12	65	76	23	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021

Сортировать выбором

Сортировать по убыванию

Сортировать по возрастанию

Сохранить и выйти

Выйти

Рисунок 4.36 — Сортировка по убыванию

Для выполнения сортировки по возрастанию необходимо нажать на кнопку «Сортировка по возрастанию» после чего данные в таблице будут отсортированы по возрастанию в зависимости от тех данных, которой находятся в колонке «Стоимость абонентской платы». Для выполнения сортировки по выбору необходимо нажать на кнопку «Сортировать выбором» на форме «Сортировка». Открытия формы «Сортировки выбором» показан на рисунке 4.37.

Сортировка выбором

Стартовая страница страница сортировки

Мобильный опер	Тариф	Стоимость обоне	Интернет трафик	Количество бес	Количество люд	Дата появления
МТС	Бизнес	87	2	23	65	03.05.2021
БелСел	Эконом	87	123	23	65	03.05.2021
БелСел	Звонки	87	12	34	87	03.05.2021
Life :)	Комфорт L	87	234	34	6	03.05.2021
МТС	Без Лимита	76	123	7	0	03.05.2021
A1	Эконом	76	123	87	23	03.05.2021
Life :)	Комфорт L	65	34	23	65	03.05.2021
МТС	Комфорт L	65	23	76	23	03.05.2021
A1	Без Лимита	12	65	76	23	03.05.2021
A1	Комфорт L	6	123	33	77	03.05.2021

Колонка

Способ фильтрации

Сортировать

Сохранить и выйти

Выйти

Рисунок 4.37 — Форма сортировки выбором

На форме сортировки выбором в компоненте «MeinMenu» есть два пункта: «Стартовая страница» и «Страница сортировки». При нажатии на пункт «Стартовая страница» данные из таблица на главной форме разрабатываемой программы будут перенесены на форму сортировки выбором. При нажатии на пункт «Страница сортировки» данные из таблица на форме «Сортировка» будут перенесены на форме сортировки выбором. Для выбора значений для сортировки выбором необходимо выбрать колонку для сортировки и метод сортировки: по убыванию, по возрастанию. Пример выбора значений показан на рисунке 4.38.

Сортировка выбором

Стартовая страница страница сортировки

Мобильный опер	Тариф	Стоимость обоне	Интернет трафик	Количество бес	Количество люд	Дата появления
Life :)	Комфорт L	123	654	87	34	03.05.2021
БелСел	Бизнес	98	88	67	65	03.05.2021
МТС	Без Лимита	87	777	45	4587	03.05.2021
Life :)	Без Лимита	76	568	34	98	03.05.2021
БелСел	Комфорт M	34	87	76	123	03.05.2021

Колонка

Интернет трафик

Способ фильтрации

По убыванию

Сортировать

Сохранить и выйти

Выйти

Рисунок 4.38 — Выбора значений для сортировки выбором

При нажатии на кнопку «Сортировать» на форме сортировки выбором данные из таблица будут отсортированы в зависимости от выбранных значений в компонентах «Combobox». Пример сортировки выбором показан на рисунке 4.39.

Сортировка выбором

Стартовая страница страница сортировки

Мобильный опер.	Тариф	Стоимость обоне	Интернет трафик	Количество бес	Количество люб	Дата появления
МТС	Без Лимита	87	777	45	4587	03.05.2021
Life.)	Комфорт L	123	654	87	34	03.05.2021
Life.)	Без Лимита	76	568	34	98	03.05.2021
БелСел	Бизнес	98	88	67	65	03.05.2021
БелСел	Комфорт М	34	87	76	123	03.05.2021

Колонка

Интернет трафик

Способ фильтрации

По убыванию

✔ Сортировать

✔ Сохранить и выйти

✖ Выйти

Рисунок 4.39 — Сортировка выбором

При нажатии на кнопку «Сохранить и выйти» в таблицу на главной форме программы будут занесены отсортированные данные в отсортированном порядке. При нажатии на кнопку «Выйти» форма «Сортировка» будет закрыта и отсортированные данные будут очищены. Пример сохранения отсортированных данных в таблицу показан на рисунке 4.40.

Мобильный оператор	Тариф	Стоимость обзона	Интернет трафик (МБ)	Количество бесплатных минут	Количество бесплатных SMS	Дата появления тарифа
МТС	Без Лимита	87	777	45	4587	03.05.2021
Life .)	Комфорт L	123	654	87	34	03.05.2021
Life .)	Без Лимита	76	568	34	98	03.05.2021
БелСел	Бизнес	98	88	67	65	03.05.2021
БелСел	Комфорт M	34	87	76	123	03.05.2021

Рисунок 4.40 — Сохранения отсортированных данных в таблицу

При нажатии на пункт «Файл» в компоненте «MeinMenu» появится всплывающая форма, где располагаются кнопки отвечающие за функции работы с файлом. Всплывающая формы с компонентами для работы с файлом показана на рисунке 4.41.

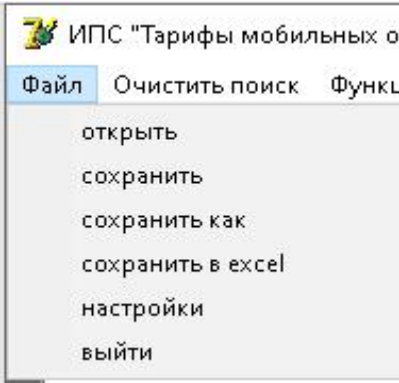


Рисунок 4.41 — Функции для работы с файлом

При нажатии на кнопку «Сохранить» ранее выбранный файл в программе будет сохранён. Если нажать на данную кнопку без открытого файла в программе, то пользователя перебросит на «сохранить как», где в проводнике необходимо выбрать файл для сохранения.

При нажатии на кнопку «сохранить в excel» данные из таблицы на главной форме будут сохранены в файл «.xls», который можно открыть в программе «Excel» и там с ним работать. Пример сохранения данных из таблицы в файл «.xls» показан на рисунке 4.42.

					<div>01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ</div>	Лист
						69
Изм.	Лист	№ докум.	Подп.	Дата		

1.xls - Excel

Файл Главная Вставка Разметка страницы Формулы Данные Рецензирование Вид Команда Что вы хотите сделать?

A1

Мобильный оператор

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Мобильн	Тариф	Стоимост	Интернет	Колличес	Колличес	Дата появления тарифа									
2	МТС	Комфорт	76	34	23	76	03.05.2021									
3	БелСел	Без Лими	76	34	7677	23	03.05.2021									
4	A1	Бизнес	888	12	34	67	03.05.2021									
5	МТС	Звонки	77	256	3487	456	03.05.2021									
6	Life;)	Эконом	98	78	87	3	03.05.2021									
7																
8																
9																
10																
11																
12																
13																
14																
15																
16																
17																
18																
19																
20																
21																
22																
23																
24																
25																
26																
27																
28																
29																

Рисунок 4.42 – Сохранение данных из таблицы в файл «.xls»

После нажатия появится сообщения об успешном сохранении таблицы в файл «.xls». Сообщение об успешном сохранении показан на рисунке 4.43.

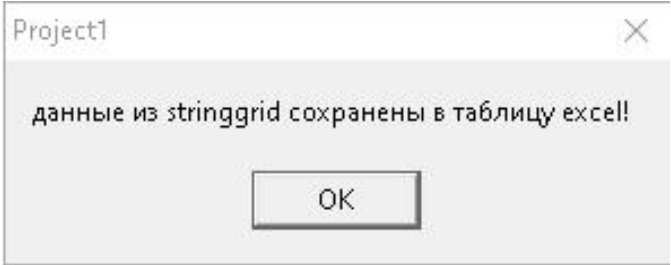


Рисунок 4.43 — Сообщения об успешном сохранении в файл «.xls»

При нажатии на кнопку «Настройки» появится форма с настройками прогаммы, где можно настроить цветовую гамму программы, размеры окна программы, устаность пароль для формы «Заблокировать». Форма настройки показана на рисунке 4.44.

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ		Лист
Изм.	Лист	№ докум.	Подп.	Дата			70

Рисунок 4.44 — Форма настройки программы

При нажатии на кнопку «Очистить поиск» в компоненте «MeinMenu» таблице будут очищены ячейки, которые были раскрашены функцией поиска. Пример до и после нажатия на кнопку «Очистить поиск» показан на рисунках 4.45, 4.46.

ИПС "Тарифы мобильных операторов"

ФайлОчистить поискФункцииПомощьСвязьОб авторе

Файл:

Статус:

Row: 6

Col: 6

заблокировать

Мобильный оператор	Тариф	Стоимость обоненс	Интернет трафик (М	Количество беспла	Количество любим	Дата появления тар
Life .)	Без Лимита	65	45	78	76	03.05.2021
МТС	Бизнес	76	45	76	34	03.05.2021
БелСел	Эконом	23	56	23	65	03.05.2021
Life .)	Комфорт L	65	123	87	65	03.05.2021
БелСел	Бизнес	76	123	76	23	03.05.2021
A1	Бизнес	76	321	76	32	03.05.2021

добавление

изменить

Поиск

Сортировка

Фильтрация

Статистика

Удаление оператора

Рисунок 4.45 — Таблица после поиска данных

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						71
Изм.	Лист	№ докум.	Подп.	Дата		

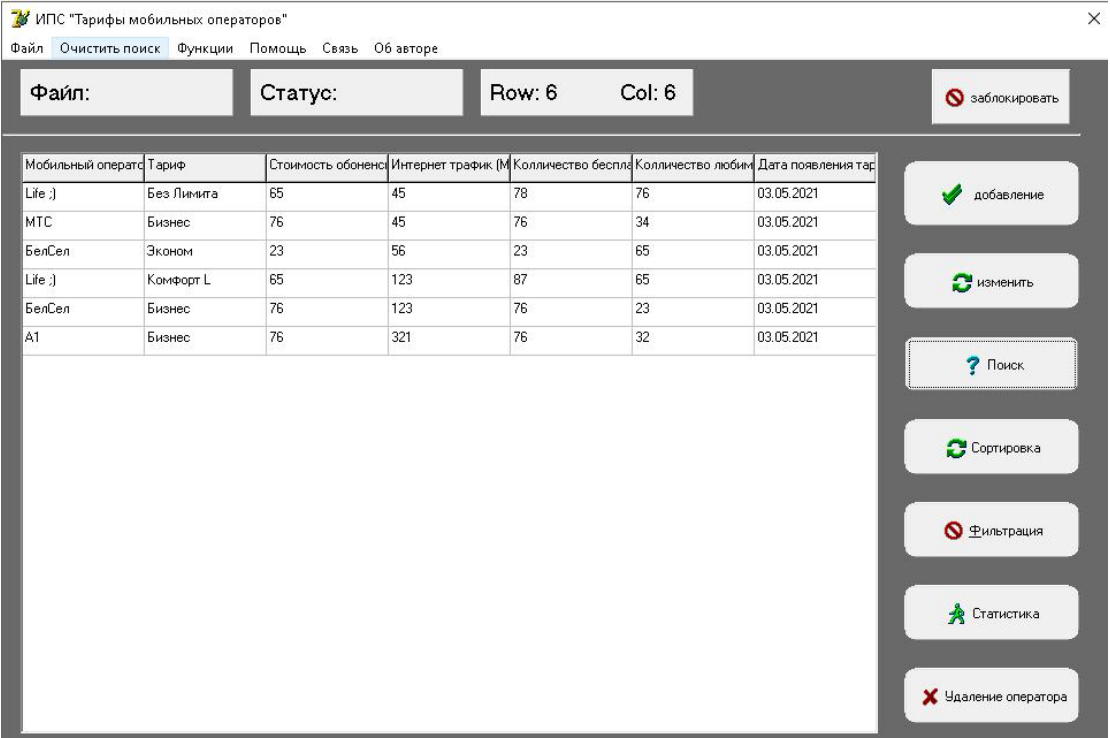
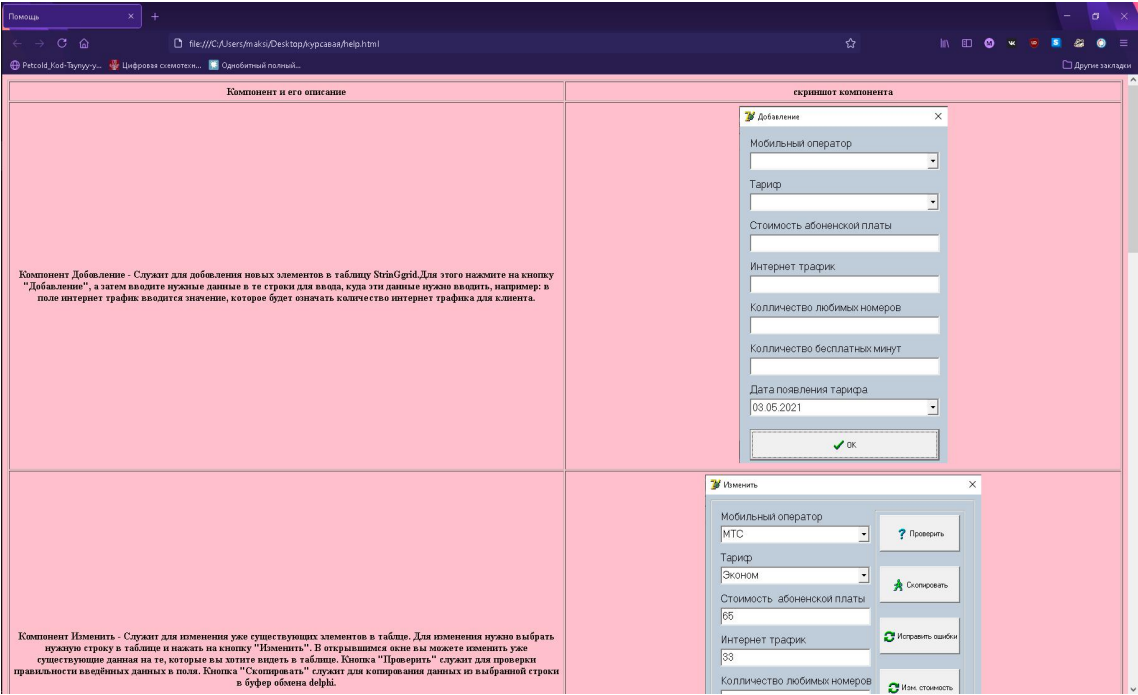


Рисунок 4.46 — Таблица после нажатия на кнопку «Очистить поиск

При нажатии на кнопку «Помощь» в компоненте «MeinMenu» откроется стандартный браузер операционной системы где автоматически откроется сайт «Помощь», где описанны функции и как с ними работать пользователь, а также приведены скриншоты форм, которые и выполняют данные функции. Сайт «Помощь» показан на рисунке 4.46.



ЗАКЛЮЧЕНИЕ

Исходя из цели курсового проекта, изучен теоретический материал на тему «Условные и логические операции» и разработана информационно-поисковая система «Тарифы мобильных операторов» на языке программирования «Delphi».

Для достижения цели были поставлены и решены следующие основные задачи, которые необходимо было решить в ходе работы: раскрыты понятия условных и логических операторов и созданы процедуры и функции поисковой системы.

Для реализации всей разработки был выбран язык программирования «Delphi», с помощью которого можно быстро и удобно разрабатывать приложения. Визуальное построение приложений способствует быстрому и качественному созданию интерфейса программы.

Актуальность курсового проекта заключается в том, что при помощи программы работники мобильных операторов могут быстро добавлять сведения о новых тарифах.

Система «Delphi» позволяет решать множество задач таких как: быстро создавать оконный интерфейс для любых приложений; создавать системы работы базами данных;

В процессе выполнения курсового проекта было усвоено некоторые возможности «Delphi».

При выполнении курсового проекта: была изучена теория и технология работы с средой программирования «Delphi»; была изучена и проанализирована соответствующая литература; был проведён сравнительный анализ языков этого класса и показаны преимущества и недостатки языка «Delphi».

В ходе разработки курсовой работы были решены поставленные задачи:

- создания файла;
- открытия файла;
- добавления данных в типизированный файл;
- изменение данных в типизированном файле;
- поиск данных;
- сортировка данных;
- фильтарция данных.

С помощью данного курсового проекта были закреплены полученные знания за курс обучения, улучшены навыки решения поставленных задач. Данный курсовой проект помогает закрепить полученные знания на протяжении всего курса обучения.

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		73

Список использованных источников

1. ГОСТ 2.105-95 Общие требования общие требования к текстовым документам [Электронный ресурс]. – Электронные данные. – Режим доступа: – <http://gostexpert.ru/gost/gost-2.105-95>. - Дата доступа 21.04.2021

2. Тариф [Электронный ресурс] / Википедия. – Режим доступа: <https://ru.wikipedia.org/wiki/%D0%A2%D0%B0%D1%80%D0%B8%D1%84>. – Дата доступа: 21.04.2021.

3. Федотова, С.В. Курс Delphi для начинающих. Полигон нестандартных задач / С.В.Федотова. – М.: Наука и техника, 2010. – 187 с.

4. Чеснокова, О.В. Delphi 7. Алгоритмы и программы / О.В.Чеснокова. – М.: НТ Пресс, 2008. – 237 с.

5. Бакнелл, Д.А. Фундаментальные алгоритмы и структуры данных в Delphi / Д.А.Бакнелл. – М.: Питер, 2006. – 203 с.

6. Бобровский, С.А. Технологии Delphi 2006. Новые возможности / С.А.Бобровский. – М.: Питер, 2006. – 157 с.

7. Павловская, Т. А. Паскаль. Программирование на языке Турбо Паскаль / Т. А. Павловская. – СПб: Петербург, 2007. – 317 с

8. Алексеев, Е. Р. Турбо Паскаль 7.0 / Е. Р. Алексеев. – М.: НТ Пресс, 2007. – 83 с.

					<i>01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ</i>	<i>Лист</i>
						74
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		

ПРИЛОЖЕНИЕ А
(обязательное)
Листинг программ

Исходный код формы приветствия находящийся в модуле Unit2.pas

```
unit Unit2;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ExtCtrls, StdCtrls, Buttons, ComCtrls, MPlayer;
type
  TForm2 = class(TForm)
    Timer1: TTimer;
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    ProgressBar1: TProgressBar;
    procedure Timer1Timer(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure BitBtn2Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
    sec, sec2: integer;
  end;
var
  Form2: TForm2;
implementation
uses Unit1;
{$R *.dfm}
procedure TForm2.Timer1Timer(Sender: TObject);
var i,j, h: integer;
begin
  label4.Caption := ('îñîîâîàÿ ôîðîà îèèðîâðñÿ ÷âðâç: ' + inttostr(sec));
  sec := sec - 1;
  sec2 := sec2 + 1;
  progressbar1.Position := sec2;
  if(sec = 1) then begin
    sec := 30;
    form1.Show;
    form2.hide;
  end;
end;
procedure TForm2.BitBtn1Click(Sender: TObject);
begin
```

					<div>01.2021.2-</div> <div>400101.1.ПО-21.Г-В-</div> <div>16-ПЗ</div>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		75

```

    timer1.Enabled := false;
    form2.Hide;
    form1.Show;
end;
procedure TForm2.BitBtn2Click(Sender: TObject);
begin
    form2.Close;
end;
procedure TForm2.FormCreate(Sender: TObject);
begin
    form2.BorderIcons := [biSystemMenu];
    sec := 30;
    sec2 := 1;
    timer1.Enabled := true;
end;
end.
```

Исходный код главной формы программы находящийся в модуле Unit1.pas

```

unit Unit1;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, StdCtrls, ExtCtrls, Buttons, Menus, Grids, ShellAPI, OleCtnrs,
    ComObj, DateUtils,
    Outline, ImgList;
type
    TForm1 = class(TForm)
        Edit1: TEdit;
        Edit2: TEdit;
        Label1: TLabel;
        Label2: TLabel;
        Bevel1: TBevel;
        ComboBox1: TComboBox;
        MainMenu1: TMainMenu;
        open1: TMenuItem;
        open2: TMenuItem;
        save1: TMenuItem;
        exit1: TMenuItem;
        StringGrid1: TStringGrid;
        RadioGroup1: TRadioGroup;
        ComboBox2: TComboBox;
        Label3: TLabel;
        Label4: TLabel;
        Label5: TLabel;
        Bevel2: TBevel;
        OpenDialog1: TOpenDialog;
        SaveDialog1: TSaveDialog;
        Bevel4: TBevel;
        BitBtn1: TBitBtn;
        BitBtn2: TBitBtn;
        BitBtn3: TBitBtn;
        BitBtn4: TBitBtn;
        Edit3: TEdit;
        BitBtn5: TBitBtn;
        BitBtn6: TBitBtn;
        BitBtn7: TBitBtn;
        Edit4: TEdit;
        BitBtn8: TBitBtn;
        Memo1: TMemo;
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						76
Изм.	Лист	№ докум.	Подп.	Дата		

```
BitBtn9: TBitBtn;
help1: TMenuItem;
N1: TMenuItem;
BitBtn10: TBitBtn;
BitBtn11: TBitBtn;
BitBtn12: TBitBtn;
BitBtn13: TBitBtn;
settings1: TMenuItem;
savetoexcel1: TMenuItem;
OleContainer1: TOleContainer;
BitBtn15: TBitBtn;
N2: TMenuItem;
BitBtn16: TBitBtn;
BitBtn14: TBitBtn;
Bevel3: TBevel;
N3: TMenuItem;
BitBtn17: TBitBtn;
ComboBox3: TComboBox;
Label6: TLabel;
Timer1: TTimer;
Timer2: TTimer;
Timer3: TTimer;
N4: TMenuItem;
PopupMenu1: TPopupMenu;
N5: TMenuItem;
hhh1: TMenuItem;
Panel1: TPanel;
Panel2: TPanel;
Label7: TLabel;
Label8: TLabel;
N6: TMenuItem;
Panel3: TPanel;
Label9: TLabel;
N7: TMenuItem;
procedure exit1Click(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure open2Click(Sender: TObject);
procedure save1Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
procedure BitBtn5Click(Sender: TObject);
procedure BitBtn6Click(Sender: TObject);
procedure BitBtn7Click(Sender: TObject);
procedure BitBtn8Click(Sender: TObject);
procedure BitBtn9Click(Sender: TObject);
procedure N1Click(Sender: TObject);
procedure BitBtn10Click(Sender: TObject);
procedure StringGrid1SelectCell(Sender: TObject; ACol, ARow: Integer; Rect:
TRect;
    var CanSelect: Boolean);
procedure help1Click(Sender: TObject);
procedure BitBtn13Click(Sender: TObject);
procedure settings1Click(Sender: TObject);
procedure BitBtn11Click(Sender: TObject);
procedure BitBtn12Click(Sender: TObject);
procedure savetoexcel1Click(Sender: TObject);
procedure BitBtn15Click(Sender: TObject);
procedure BitBtn14Click(Sender: TObject);
procedure StringGrid1DrawCell(Sender: TObject; ACol, ARow: Integer;
    Rect: TRect; State: TGridDrawState);
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		77

```
procedure N2Click(Sender: TObject);
procedure BitBtn16Click(Sender: TObject);
procedure BitBtn17Click(Sender: TObject);
procedure Edit5Change(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
procedure Timer2Timer(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure Timer3Timer(Sender: TObject);
procedure FormCloseQuery(Sender: TObject; var CanClose: Boolean);
procedure N4Click(Sender: TObject);
//procedure N5Click(Sender: TObject);
procedure hhh1Click(Sender: TObject);
procedure N3Click(Sender: TObject);
procedure N6Click(Sender: TObject);
procedure Panell1MouseMove(Sender: TObject; Shift: TShiftState; X,
    Y: Integer);
procedure N7Click(Sender: TObject);
procedure StringGrid1MouseDown(Sender: TObject; Button: TMouseButton;
    Shift: TShiftState; X, Y: Integer);
procedure ComboBox3Change(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }

    col_index, row_index: integer;
    col_col, row_col : integer;
    start_time, stop_time : TDateTime;
    elapsed: integer;
    add_rowindex: integer;
    str_grid, file_name: string;
    find_row, fun_code, i_fun, j_fun: integer;
    str_draw: string;
    cls_qeru: boolean;
    open_file : integer;
    dell_j: array[1..100] of integer;

end;

    TMyGrid = class(TCustomGrid);
type
film = record
    name : string[20];
    genre : string[20];
    cost : integer;
    country : string[20];
    rating: string[3];
end;
type
tele = record
    name: string[10];
    tarif: string[15];
    cost: integer;
    trafic: string[3];
    free_min : integer;
    lovenumber : integer;
    date: string[10];
end;
const
    EXCEL_FILE_EXT = '.xls';
var
    Form1: TForm1;
    var movie : film;
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		78

```
var k, m, n, j, i, h, p, g, f, d, s, l, tmp, z: integer;
kk, jj, mm, hh, ll, gg, vv, bb, nn: string;
kkk, mmm, q2, q3, q4: integer;
arr: array[1..10] of film;
mas: array[1..10] of integer;
arr_tele: array[1..100] of tele;
input: file of tele;
find_start_tele: array[1..100] of tele;
find_tele: array[1..100] of tele;
col: integer;
implementation
uses Unit3, Unit2, Unit4, Unit5, Unit8, Unit9, Unit10, Unit11, Unit13,
    Unit14, Unit15, Unit16, Unit17, Unit18, Unit19, Unit20, Unit21, Unit23;
{$R *.dfm}
procedure dell(grid: TStringGrid; num: integer);
begin
    TMyGrid(grid).DeleteRow(num);
end;
procedure del(num : integer; Grid: TstringGrid);
var i : integer;
begin
    //i := 0;
    for i:=num to num do begin

        grid.Rows[i] := grid.Rows[i + 1];
        grid.RowCount := grid.RowCount - 1;
    end;
    q2 := q2 + 1;
end;
procedure TForm1.exit1Click(Sender: TObject);
begin
    form1.Close;
    //k := 1;
    //m := 0;
end;
procedure TForm1.BitBtn1Click(Sender: TObject);
begin
    for i:=1 to stringgrid1.RowCount - 1 do begin
        if(stringgrid1.Cells[4, i] = 'fantasy') then begin
            stringgrid1.Cells[4, i] := 'fantazy';
        end;
    end;
end;
procedure TForm1.BitBtn2Click(Sender: TObject);
begin
    arr[h].name := edit1.Text;
    arr[h].country := combobox1.text;
    arr[h].cost := strtoint(edit2.text);
    arr[h].genre := combobox2.Text;
    if(radiogroup1.ItemIndex = 0) then arr[h].rating := '0';
    if(radiogroup1.ItemIndex = 1) then arr[h].rating := '3';
    if(radiogroup1.ItemIndex = 2) then arr[h].rating := '6';
    if(radiogroup1.ItemIndex = 3) then arr[h].rating := '9';
    edit1.Clear;
    edit2.Clear;
    //stringgrid1.Cells[0, g] := inttostr(col);;
    stringgrid1.Cells[m, k] := arr[h].name;
    m := m + 1;
    stringgrid1.Cells[m, k] := inttostr(arr[h].cost);
    m := m + 1;
    stringgrid1.Cells[m, k] := arr[h].country;
    m := m + 1;
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		79


```
if(strtoint(arr[h].rating) = 0) then
  stringgrid1.Cells[m, k] := '0+';
if(strtoint(arr[h].rating) = 3) then
  stringgrid1.Cells[m, k] := '3+';
if(strtoint(arr[h].rating) = 6) then
  stringgrid1.Cells[m, k] := '6+';
if(strtoint(arr[h].rating) = 9) then
  stringgrid1.Cells[m, k] := '9+';
if(strtoint(arr[h].rating) = 12) then
  stringgrid1.Cells[m, k] := '12+';
//mas[g] := strtoint(stringgrid1.Cells[m, k]);
m := m + 1;
stringgrid1.Cells[m, k] := arr[h].genre;
m := 0;
k := k + 1;
h := h + 1;
g := g + 1;
col := col + 1;
if(k > stringgrid1.RowCount) then begin
  stringgrid1.RowCount:= stringgrid1.RowCount + 1;
end;
movie.name := '';
movie.cost := 0;
movie.rating := '0';
movie.genre := '';
movie.country := '';
end;
procedure DrawRounded(Control: TWinControl) ;
var
  R: TRect;
  Rgn: HRGN;
begin
  with Control do
    begin
      R := ClientRect;
      rgn := CreateRoundRectRgn(R.Left, R.Top, R.Right, R.Bottom, 20, 20) ;
      Perform(EM_GETRECT, 0, lParam(@r)) ;
      InflateRect(r, - 4, - 4) ;
      Perform(EM_SETRECTNP, 0, lParam(@r)) ;
      SetWindowRgn(Handle, rgn, True) ;
      Invalidate;
    end;
end;
procedure TForm1.FormCreate(Sender: TObject);
begin
  cls_geru := False;
  form2.Hide;
  start_time := Now;
  add_rowindex := 1;
  timer2.Enabled := true;
  form1.BorderIcons := [biSystemMenu];
  form1.fun_code := 0;
  col := 0;
  k := 1;
  m:= 0;
  h:=1;
  g := 1;
  col := 0;
  q2 := 1;
  if(form1.open_file <> 1) then begin
    bitbtn11.Enabled := false;
    bitbtn10.Enabled := false;
```

					Лист	
					80	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

```

        bitbtn12.Enabled := false;
        bitbtn14.Enabled := false;
        bitbtn13.Enabled := false;
        bitbtn15.Enabled := false;
    end;
    //-----закругляем углы кнопок-----
    DrawRounded(bitbtn1);
    DrawRounded(bitbtn2);
    DrawRounded(bitbtn3);
    DrawRounded(bitbtn4);
    DrawRounded(bitbtn5);
    DrawRounded(bitbtn6);
    DrawRounded(bitbtn7);
    DrawRounded(bitbtn8);
    DrawRounded(bitbtn9);
    DrawRounded(bitbtn10);
    DrawRounded(bitbtn11);
    DrawRounded(bitbtn12);
    DrawRounded(bitbtn13);
    DrawRounded(bitbtn14);
    DrawRounded(bitbtn15);
    //-----закругляем углы кнопок-----
    stringgrid1.RowCount := 2;
    stringgrid1.Cells[0, 0] := 'Мобильный оператор';
    stringgrid1.Cells[1, 0] := 'Тариф';
    stringgrid1.Cells[2, 0] := 'Стоимость обоненской платы';
    stringgrid1.Cells[3, 0] := 'Интернет трафик (МБ)';
    stringgrid1.Cells[4, 0] := 'Количество бесплатных минут';
    stringgrid1.Cells[5, 0] := 'Количество любимых номеров';
    stringgrid1.Cells[6, 0] := 'Дата появления тарифа';

end;
procedure TForm1.open2Click(Sender: TObject);
var WND: HWND;
var lpText, lpCaption: PChar;
var Tip, k: integer;
var Sendera: TObject;
begin
    //for i:=0 to stringgrid1.ColCount - 1 do begin

        //for j:=0 to stringgrid1.RowCount - 1 do begin

            //stringgrid1.Cells[i, j] := '';
        //end;
    //end;
if(form1.open_file = 1) then begin
    WND := Form1.Handle;
    lpCaption := 'Сохранить таблицу ?';
    lpText := 'Хотите ли вы сохранить файл ?';
    Tip := MB_YESNO + MB_ICONINFORMATION + MB_DEFBUTTON1;
    case MessageBox(WND, lpText, lpCaption, Tip) of
        IDYES: begin
            form1.sav1Click(Sendera);
            cls_geru := True;
            //form1.Close;
            //Application.Terminate;
            form16.Show;
            form16.Position := poScreenCenter;
            end;
        IDNO : form16.show;
    end;
end;
end;

```

					<div>01.2021.2-</div> <div>400101.1.ПО-21.Г-В-</div> <div>16-ПЗ</div>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		81

```
if(form1.open_file = 2) then begin

    form16.Show;
    form16.Position := poScreenCenter;
end;
if(form1.open_file = 0) then begin
    form16.Show;
end;
end;
procedure TForm1.savelClick(Sender: TObject);
var i, j, n, k, b: integer;
begin
    if(form1.open_file <> 1) then begin

        form1.N4Click(sender);
    end
    else begin
        n := 0;
        assignfile(input, file_name);
        rewrite(input);
        seek(input, 0);
        i := 0;
        for i:=0 to stringgrid1.ColCount - 1 do begin
            for j := 1 to stringgrid1.RowCount - 1 do begin
                arr_tele[j].name := stringgrid1.Cells[0, j];
                arr_tele[j].tarif := stringgrid1.Cells[1, j];
                arr_tele[j].cost := strtoint(stringgrid1.Cells[2, j]);
                arr_tele[j].trafic := stringgrid1.Cells[3, j];
                arr_tele[j].free_min := strtoint(stringgrid1.Cells[4, j]);
                arr_tele[j].lovenumber := strtoint(stringgrid1.Cells[5, j]);
                arr_tele[j].date := stringgrid1.Cells[6, j];
            end;
        end;
        n := n + 1;
        b := stringgrid1.RowCount - 1;
        for k:=1 to b do begin
            write(input, arr_tele[k]);
        end;
        closefile(input);
        form1.Label9.Caption := 'Статус: сохранён';
        timer2.Enabled := true;
        form1.open_file := 2;
    end;
end;
procedure TForm1.BitBtn3Click(Sender: TObject);
begin
{
    g := 1;
    for i:=0 to stringgrid1.RowCount - 1 do begin

        mas[g] := strtoint(stringgrid1.Cells[3, g]);

        if(stringgrid1.Cells[3, g] = '') then begin
            break;
        end;
        g := g + 1;
    end;
}
{
    for i:=1 to g-1 do begin
        for j:=1 to g - i do begin
            if(mas[j] < mas[j + 1]) then begin
```

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	82
01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ					

```

        tmp := mas[j];
        mas[j] := mas[j + 1];
        mas[j + 1] := tmp;
    end;
end;
end;
for i:=1 to g do begin
    memol.Lines.add(inttostr(mas[i]));
end;
for i:=0 to stringgrid1.ColCount - 1 do begin
    for j:=0 to stringgrid1.RowCount - 1 do begin
        stringgrid1.Cells[i, j] := '';
    end;
end;
//insert sort
for i:=0 to stringgrid1.ColCount - 1 do begin

    for j:=0 to stringgrid1.RowCount - 1 do begin
        end;
    end;
}
s := 0;
z := 0;
for i:=0 to stringgrid1.ColCount - 1 do begin
    for j:=1 to stringgrid1.RowCount - 1 do begin
        vv := stringgrid1.Cells[3, j];
        gg := stringgrid1.Cells[3, j + 1];
        delete(vv, 2, 1);
        delete(gg, 2, 1);
        //memol.Lines.add(vv);
        //memol.Lines.add(gg);
        if (vv < gg)then begin
            kk := stringgrid1.Cells[0, j];
            stringgrid1.Cells[0, j] := stringgrid1.Cells[0, j + 1];
            stringgrid1.Cells[0, j + 1] := kk;
            kkk := strtoint(stringgrid1.Cells[1, j]);
            stringgrid1.Cells[1, j] := stringgrid1.Cells[1, j + 1];
            stringgrid1.Cells[1, j + 1] := inttostr(kkk);
            jj := stringgrid1.cells[2, j];
            stringgrid1.Cells[2, j] := stringgrid1.Cells[2, j + 1];
            stringgrid1.Cells[2, j + 1] := jj;
            mmm := strtoint(vv);
            stringgrid1.Cells[3, j] := stringgrid1.Cells[3, j + 1];
            stringgrid1.Cells[3, j + 1] := inttostr(mmm) + '+';
            mm := stringgrid1.Cells[4, j];
            stringgrid1.Cells[4, j] := stringgrid1.Cells[4, j + 1];
            stringgrid1.Cells[4, j + 1] := mm;
        end;
    end;
    end;
    kk := '';
    jj := '';
    mm := '';
    kkk := 0;
    mmm := 0;
    vv:='';
    gg := '';
end;
end;
procedure TForm1.BitBtn4Click(Sender: TObject);
begin
    edit3.Visible := true;
    bitbtn5.Visible := true;
end;
```

```
end;
procedure TForm1.BitBtn5Click(Sender: TObject);
begin
  l := strtoint(edit3.Text);
  stringgrid1.Cells[0, 0] := 'name';
  stringgrid1.Cells[1, 0] := 'cost';
  stringgrid1.Cells[2, 0] := 'country';
  stringgrid1.Cells[3, 0] := 'rating';
  stringgrid1.Cells[4, 0] := 'genre';
  if(l = 0) then showmessage('error: = 0');
  stringgrid1.cells[0, 1] := '';
  stringgrid1.Cells[1, 1] := '';
  stringgrid1.Cells[2, 1] := '';
  stringgrid1.Cells[3, 1] := '';
  stringgrid1.Cells[4, 1] := '';
  k:= k - 1;
  //k := 1;
  f := l + 1;
  if(l = stringgrid1.RowCount - 1) then begin
    stringgrid1.cells[0, 1] := '';
    stringgrid1.Cells[1, 1] := '';
    stringgrid1.Cells[2, 1] := '';
    stringgrid1.Cells[3, 1] := '';
    stringgrid1.Cells[4, 1] := '';
  end
  else begin
    for i:=0 to stringgrid1.ColCount - 1 do begin
      for j:=f to stringgrid1.RowCount - 1 do begin
        stringgrid1.Cells[i, j - 1] := stringgrid1.Cells[i, j];
      end;
    end;
    stringgrid1.Cells[0, stringgrid1.RowCount - 1] := '';
    stringgrid1.Cells[1, stringgrid1.RowCount - 1] := '';
    stringgrid1.Cells[2, stringgrid1.RowCount - 1] := '';
    stringgrid1.Cells[3, stringgrid1.RowCount - 1] := '';
    stringgrid1.Cells[4, stringgrid1.RowCount - 1] := '';
    stringgrid1.RowCount := stringgrid1.RowCount - 1;
  end;
  col := col - 1;
end;
procedure TForm1.BitBtn6Click(Sender: TObject);
var a, g, h : integer;
var r, q1, w1 : string;
begin
  for a := stringgrid1.RowCount - 1 downto 0 do begin
    for g := 1 to a - 1 do begin
      if strtoint(stringgrid1.Cells[1, g]) < strtoint(stringgrid1.Cells[1, g
+ 1]) then for h:=stringgrid1.ColCount -1 downto 0 do begin
        r := stringgrid1.Cells[h, g];
        stringgrid1.Cells[h, g] := stringgrid1.Cells[h, g + 1];
        stringgrid1.Cells[h, g + 1] := r;
      end;
    end;
  end;
end;
procedure TForm1.BitBtn7Click(Sender: TObject);
var a, g, h : integer;
var r, q1, w1 : string;
begin
  for a := stringgrid1.RowCount - 1 downto 0 do begin
    for g := 1 to a - 1 do begin
```

```

        if strtoint(stringgrid1.Cells[1, g]) > strtoint(stringgrid1.Cells[1, g
+ 1]) then for h:=stringgrid1.ColCount -1 downto 0 do begin
            r := stringgrid1.Cells[h, g];
            stringgrid1.Cells[h, g] := stringgrid1.Cells[h, g + 1];
            stringgrid1.Cells[h, g + 1] := r;
        end;
    end;
end;
end;
procedure TForm1.BitBtn8Click(Sender: TObject);
var a, g, h, n: integer;
var r, q1, w1 : string;
begin
    r := edit4.Text;
    for i:=stringgrid1.RowCount - 1 downto stringgrid1.FixedRows do begin
        q1 := stringgrid1.Cells[3, i];
        delete(q1, 2, 1);
        memol.Lines.add(q1);
        if(q1 <> r) then dell(stringgrid1, i);
    end;
end;
procedure TForm1.BitBtn9Click(Sender: TObject);
begin
    form3.Show;
    form3.Position := poScreenCenter;
end;
procedure TForm1.N1Click(Sender: TObject);
begin
    form4.show();
end;
procedure TForm1.BitBtn10Click(Sender: TObject);
begin
    row_index := stringgrid1.Row;
    if(row_index = 0) then begin
        ShowMessage('Ошибка! Нельзя изменять фиксированные колонки в таблице.');
```

					Лист	
					85	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

```

        stringgrid1.Canvas.Font.Color := clblack;
        stringgrid1.Canvas.TextOut(rect.Left+2,          rect.Top          +          2,
stringgrid1.Cells[ACol, ARow]);
    end;
    if(ACol = col_index) and (ARow = row_index) then begin

        stringgrid1.Canvas.Brush.Color := clblue;
        stringgrid1.Canvas.FillRect(stringgrid1.CellRect(col_index, row_index));
        stringgrid1.Canvas.Font.Color := clblack;
        stringgrid1.Canvas.TextOut(rect.Left+2,          rect.Top          +          2,
stringgrid1.Cells[ACol, ARow]);
    end;
    }
    row_index := ARow;
end;
procedure TForm1.help1Click(Sender: TObject);
begin
    ShellExecute(0, 'open', PChar('help.html'), nil, nil, SW_SHOWNORMAL);
end;
procedure TForm1.BitBtn13Click(Sender: TObject);
begin
    form8.show;
    form8.Position := poScreenCenter;
end;
procedure TForm1.settings1Click(Sender: TObject);
begin
    form9.show;
end;
procedure TForm1.BitBtn11Click(Sender: TObject);
begin
    col_col := stringgrid1.ColCount - 1;
    row_col := stringgrid1.RowCount - 1;
    //form10.show;
    form20.show;
    //form17.Position := poScreenCenter;
end;
procedure TForm1.BitBtn12Click(Sender: TObject);
begin
    form11.show;
    form11.Timer1.Enabled := true;
    form11.Position := poScreenCenter;
end;
function GetExcelFileName: String;
begin
    Result := ExtractFilePath(Application.ExeName) + '1.xls';
    if LowerCase(ExtractFileExt(Result)) <> EXCEL_FILE_EXT then
        Result := Result + EXCEL_FILE_EXT;
end;
procedure TForm1.savetoexcel1Click(Sender: TObject);
var
    ExcelApp, Sheet: variant;
    Col, Row: Word;
begin
    ExcelApp := CreateOleObject('Excel.Application');
    try
        ExcelApp.visible := false;
        ExcelApp.Workbooks.Add;
        Sheet := ExcelApp.ActiveWorkbook.Worksheets[1];
        for Col := 0 to stringgrid1.ColCount - 1 do begin
            for Row := 0 to stringgrid1.RowCount - 1 do begin
                Sheet.Cells[Row + 1, Col + 1] := Stringgrid1.Cells[Col, Row];
            end;
        end;
    end;
end;
```

					Лист	
					86	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

```

end;
ExcelApp.ActiveWorkbook.SaveAs (GetExcelFileName());
showmessage('данные из stringgrid сохранены в таблицу excel!');
finally
  ExcelApp.Application.Quit;
  ExcelApp := unassigned;
end;
end;
procedure TForm1.BitBtn15Click(Sender: TObject);
begin
  //DrawRounded(bitbtn15);
  form14.Show;
  form14.Position := poScreenCenter;
end;
procedure TForm1.BitBtn14Click(Sender: TObject);
begin
  //form1.Show;
  //form15.Visible := true;
  for i:=0 to form1.StringGrid1.ColCount - 1 do begin

    for j:=1 to form1.StringGrid1.RowCount - 1 do begin

      form15.combobox2.items.add(form1.StringGrid1.Cells[i, j]);
    end;
  end;
  form15.show;
  form15.Timer1.Enabled := true;
  form15.Position := poScreenCenter;
end;
procedure TForm1.StringGrid1DrawCell(Sender: TObject; ACol, ARow: Integer;
  Rect: TRect; State: TGridDrawState);
var i,j: integer;
  FontColor : TColor;
  FontStyle : TFontStyles;
  BrColor : TColor;
  myRect: Grids.TGridRect;
begin
  //if(form17.draw_choice = 1) then begin

    //stringgrid1.Canvas.Brush.Color := clwhite;
    //stringgrid1.Canvas.FillRect(stringgrid1.CellRect(ACol, ARow));
    //stringgrid1.Canvas.Font.Color := clblack;
    //stringgrid1.Canvas.TextOut(rect.Left+2,          rect.Top          +          2,
stringgrid1.Cells[ACol, ARow]);
  //end;
  //if(ACol = form17.col_index) and (ARow = form17.row_index) then begin

    //stringgrid1.Canvas.Brush.Color := clblue;
    //stringgrid1.Canvas.FillRect(stringgrid1.CellRect(col_index, row_index));
    //stringgrid1.Canvas.Font.Color := clblack;
    //stringgrid1.Canvas.TextOut(rect.Left+2,          rect.Top          +          2,
stringgrid1.Cells[ACol, ARow]);
  //end;
  //FontColor := Font.Color;
  //FontStyle := Font.Style;
  //BrColor := Brush.Color;
  if(form17.draw_choice = 1) then begin
  with stringgrid1 do begin
  //if(form17.draw_choice = 1) then begin
    for i:=0 to ColCount - 1 do begin
      for j:=1 to RowCount - 1 do begin
        if(Cells[i, j] = form17.str2) then begin

```

					<div>01.2021.2-</div> <div>400101.1.ПО-21.Г-В-</div> <div>16-ПЗ</div>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		87


```

        end;
    end;
end;
for i:=0 to stringgrid1.ColCount - 1 do begin
    for j:=1 to stringgrid1.RowCount - 1 do begin
        myRect.Left := i+1;
        myRect.Top := j + 1;
        myRect.Right := i + 1;
        myRect.Bottom := j + 1;
        stringgrid1.Selection := myRect;
    end;
end;
bitbtn17.Visible := false;
end;
procedure TForm1.BitBtn16Click(Sender: TObject);
begin
    form18.show;
    form8.Position := poScreenCenter;
    form1.hide;
end;
procedure DeleteRow(ARow: Integer; grid: TstringGrid);
var i, j: Integer;
begin
    with grid do
        begin
            for i:=ARow+1 to RowCount-1 do
                for j:=0 to ColCount-1 do
                    Cells[j, i-1]:=Cells[j, i];
                for i:=0 to ColCount-1 do
                    Cells[i, RowCount-1]:='';
                RowCount:=RowCount-1;
            end;
        end;
end;
procedure TForm1.BitBtn17Click(Sender: TObject);
var i, j, k, err, l, n: integer;
var str: string;
begin
    //form19.Show;
    k := 1;
    err := 0;
    n := 0;
    str := combobox3.Text;
    for i:=1 to form1.stringgrid1.RowCount - 1 do begin
        for j:=0 to form1.StringGrid1.ColCount - 1 do begin
            if(form1.StringGrid1.Cells[j, i] <> str) then begin
                err := 1;
            end
            else if(form1.StringGrid1.Cells[j, i] = str) then begin
                err := 2;
            end;
        end;
    end;
    if(err = 1) then begin
        {
            dell_j[k] := i;
            err := 0;
            k := k + 1;
        }
        DeleteRow(i, form1.StringGrid1);
        break;
    end;
    if(err = 2) then begin
        err := 0;
    end;
end;

```

					<div>01.2021.2-</div> <div>400101.1.ПО-21.Г-В-</div> <div>16-ПЗ</div>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		89

```

        k := k + 1;
        //continue;
    end;
end;
{
for i:= 1 to k do begin
    DeleteRow(dell_j[i], form1.StringGrid1);
end;
}
end;
procedure TForm1.Edit5Change(Sender: TObject);
var i, j, k, n : integer;
begin
end;
procedure TForm1.Timer1Timer(Sender: TObject);
var i, j : integer;
var str, str1 : string;
begin
    for i:=0 to stringgrid1.ColCount - 1 do begin
        for j:=1 to stringgrid1.RowCount - 1 do begin
            if(stringgrid1.Cells[i, j] = form17.str) then begin
                str:=form17.str;
                str_draw := str;

                str1 := '>> ' + str + ' <<';
                stringgrid1.Cells[i, j] := str1;
            end;
        end;
    end;
    timer2.Enabled := true;
    timer1.Enabled := false;
end;
procedure TForm1.Timer2Timer(Sender: TObject);
var i, j : integer;
var str, str1 : string;
begin
    //timer1.Enabled := false;
    {
    for i:=0 to stringgrid1.ColCount - 1 do begin

        for j:=1 to stringgrid1.RowCount - 1 do begin

            if(stringgrid1.Cells[i, j] = form17.str) then begin

                str :=form17.str;
                label6.Caption := str;
                stringgrid1.Cells[i, j] := 'hhh';
            end;
        end;
    end;
    timer1.Enabled := true;
    timer2.Enabled := false;
    }
    form1.Label8.Caption := 'Row: ' + inttostr(form1.StringGrid1.RowCount - 1) +
    '
        Col: ' + inttostr(form1.stringgrid1.ColCount - 1);
end;
procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
var WND: HWND;
var lpText, lpCaption: PChar;
var Tip, k: integer;
var Sendera: TObject;
begin
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						90
Изм.	Лист	№ докум.	Подп.	Дата		

```

{
  WND := Form1.Handle;
  lpCaption := 'Сохранить таблицу?';
  lpText := 'Хотите ли вы сохранить файл?';
  Tip := MB_YESNO + MB_ICONINFORMATION + MB_DEFBUTTON1;
  case MessageBox(WND, lpText, lpCaption, Tip) of
    IDYES: form1.save1Click(Sendera);
    IDNO : timer3.Enabled := true;
  end;

  //form2.Show;
  //form1.Show;
}
end;
procedure TForm1.Timer3Timer(Sender: TObject);
begin
  timer2.Enabled := false;
  //ShellExecute(0, nil, 'Project1.exe', '/C TASKKILL /IM Project.exe', nil,
SW_HIDE);
  form1.Hide;
  timer3.Enabled := false;
end;
procedure TForm1.FormCloseQuery(Sender: TObject; var CanClose: Boolean);
var WND: HWND;
var lpText, lpCaption: PChar;
var Tip, k: integer;
var Sendera: TObject;
begin
  WND := Form1.Handle;
  lpCaption := 'Сохранить таблицу?';
  lpText := 'Хотите ли вы сохранить файл?';
  Tip := MB_YESNO + MB_ICONINFORMATION + MB_DEFBUTTON1;
  case MessageBox(WND, lpText, lpCaption, Tip) of
    IDYES: begin
      form1.save1Click(Sendera);
      cls_qeru := True;
      Application.Terminate;
      end;
    IDNO : Application.Terminate;
  end;
end;
procedure TForm1.N4Click(Sender: TObject);
var i, j, n, k, b: integer;
begin
  n := 0;
  if(savedialog1.Execute) then begin
    assignfile(input, savedialog1.filename);
    rewrite(input);
    seek(input, 0);
    i := 0;
    for i:=0 to stringgrid1.ColCount - 1 do begin
      for j := 1 to stringgrid1.RowCount - 1 do begin
        arr_tele[j].name := stringgrid1.Cells[0, j];
        arr_tele[j].tarif := stringgrid1.Cells[1, j];
        arr_tele[j].cost := strtoint(stringgrid1.Cells[2, j]);
        arr_tele[j].trafic := stringgrid1.Cells[3, j];
        arr_tele[j].free_min := strtoint(stringgrid1.Cells[4, j]);
        arr_tele[j].lovenumber := strtoint(stringgrid1.Cells[5, j]);
        arr_tele[j].date := stringgrid1.Cells[6, j];
        //n := n + 1;
      end;
      //n := n + 1;
    end;
  end;
end;

```

					<div>01.2021.2-</div> <div>400101.1.ПО-21.Г-В-</div> <div>16-ПЗ</div>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		91

```

end;
n := n + 1;
form1.open_file := 2;
end;
b := stringgrid1.RowCount - 1;

for k:=1 to b do begin

    write(input, arr_tele[k]);
end;
closefile(input);
form1.Label9.Caption := 'Статус: сохранён';
timer2.Enabled := true;
end;
procedure TForm1.hhh1Click(Sender: TObject);
var i, j, k, l : integer;
begin
    label6.Caption := inttostr(form1.StringGrid1.Row);
    j := form1.StringGrid1.Row;

    if(j = 0) then begin
        ShowMessage('Ошибка! Нельзя удалить фиксированную строку');
    end
    else begin
        DeleteRow(j, form1.StringGrid1);
        form1.add_rowindex := form1.StringGrid1.RowCount;
    end;
end;
procedure TForm1.N3Click(Sender: TObject);
begin
    form21.show;
end;
procedure TForm1.N6Click(Sender: TObject);
begin
    row_index := stringgrid1.Row;
    if(row_index = 0) then begin
        ShowMessage('Ошибка! Нельзя изменять фиксированные колонки в таблице.');
```

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	92
01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ					

```

form23.show;
end;
procedure TForm1.StringGrid1MouseDown(Sender: TObject;
  Button: TMouseButton; Shift: TShiftState; X, Y: Integer);
  var ACol, ARow, ARow2, ACol2, i, j: integer;
  var ii, jj, k : integer;
  var str, str2: string;
begin
  if(form1.fun_code = 2) then begin
    stringgrid1.MouseToCell(X, Y, ACol2, ARow2);
    form23.Label5.Caption := 'Значение: ' + stringgrid1.Cells[ACol2, ARow2];
    j_fun := strtoint(stringgrid1.Cells[ACol2, ARow2]);
    if(form23.ComboBox1.ItemIndex = 0) then begin
      form23.label6.Caption := 'Результат функции: ' + inttostr(i_fun +
j_fun);
    end;
    if(form23.ComboBox1.ItemIndex = 1) then begin
      form23.label6.Caption := 'Результат функции: ' + inttostr(i_fun -
j_fun);
    end;
    if(form23.ComboBox1.ItemIndex = 2) then begin
      form23.label6.Caption := 'Результат функции: ' + inttostr(i_fun *
j_fun);
    end;
    if(form23.ComboBox1.ItemIndex = 3) then begin
      form23.label6.Caption := 'Результат функции: ' +
floattostr(Sin(i_fun + j_fun));
    end;
    if(form23.ComboBox1.ItemIndex = 4) then begin
      form23.label6.Caption := 'Результат функции: ' +
floattostr(Cos(i_fun + j_fun));
    end;
  end;
  if(form1.fun_code = 1) then begin

    stringgrid1.MouseToCell(X, Y, ACol, ARow);
    str := stringgrid1.Cells[ACol, ARow];
    val(str, ii, jj);
    if(jj <> 0) then begin

    end
    else begin
      i_fun := strtoint(stringgrid1.Cells[ACol, ARow]);
      form23.Label3.Caption := 'Значение: ' + stringgrid1.Cells[ACol, ARow];

      form1.fun_code := 2;
    end;
  end;

end;

procedure TForm1.ComboBox3Change(Sender: TObject);
var i, j, h, k, size: integer;
var str, str2, str3: string;
begin
  form1.N2Click(sender);
  size := length(combobox3.text);
  str := combobox3.Text;
  for i:= 0 to form1.StringGrid1.ColCount - 1 do begin

    for j:=1 to form1.StringGrid1.RowCount - 1 do begin

      str2 := form1.StringGrid1.Cells[i, j];

```

					<div>01.2021.2-</div> <div>400101.1.ПО-21.Г-В-</div> <div>16-ПЗ</div>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		93

```

        h := pos(str, str2);

        if(h = 0) then begin
            continue;
        end
        else if(h <> 0) then begin

            form17.str2 := str2;

            str3 := form1.StringGrid1.Cells[0, 1];
            form1.StringGrid1.Cells[0, 1] := str3;

        end;
    end;
end;
end.

```

Исходный код формы добавления находится в модуле Unit3.pas

```

unit Unit3;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, StdCtrls, Buttons, ExtCtrls, Mask, ComCtrls;
type
    TForm3 = class(TForm)
        Label1: TLabel;
        BitBtn1: TBitBtn;
        Label2: TLabel;
        Label3: TLabel;
        Edit3: TEdit;
        Label4: TLabel;
        ComboBox1: TComboBox;
        ComboBox2: TComboBox;
        Label5: TLabel;
        Edit1: TEdit;
        Label6: TLabel;
        //MaskEdit1: TMaskEdit;
        Label7: TLabel;
        Edit2: TEdit;
        DateTimePicker1: TDateTimePicker;
        Edit4: TEdit;
        procedure BitBtn1Click(Sender: TObject);
        procedure FormCreate(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;
var
    Form3: TForm3;
    en_erorr_char : set of char;
    ru_error_char : set of char;
    lower_en_error_char : set of char;
    lower_ru_error_char: set of char;
implementation

```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						94
Изм.	Лист	№ докум.	Подп.	Дата		

```
uses Unit1;
{$R *.dfm}
procedure TForm3.BitBtn1Click(Sender: TObject);
var size, size2 , i, j, n, k, b, h, l: integer;
var str, str2, str3, str4: string;
begin
  form1.bitbtn11.Enabled := true;
  form1.bitbtn10.Enabled := true;
  form1.bitbtn12.Enabled := true;
  form1.bitbtn14.Enabled := true;
  form1.bitbtn13.Enabled := true;
  form1.bitbtn15.Enabled := true;
  form1.stringgrid1.Cells[0, 0] := 'İiaëëüiúé iiaðàòîð';
  form1.stringgrid1.Cells[1, 0] := 'Òàðèè';
  form1.stringgrid1.Cells[2, 0] := 'Ñòìèìñòù íáííáíñêîé îèàòù';
  form1.stringgrid1.Cells[3, 0] := 'Ëìòâðíâð òðàðèè (ÌÁ)';
  form1.stringgrid1.Cells[4, 0] := 'Êîëëè-âñòâî áâñîèàòíüò ìèíóò';
  form1.stringgrid1.Cells[5, 0] := 'Êîëëè-âñòâî ëðàèíüò íííâðíâ';
  form1.stringgrid1.Cells[6, 0] := 'Ààòà îîÿâèáîÿ òàðèè';
  en_errr_char := ['A'..'Z'];
  ru_error_char := ['À'..'ß'];
  lower_en_error_char := ['a'..'z'];
  lower_ru_error_char := ['à'..'ÿ'];
  str := edit3.Text;
  val(str, i, n);
  if((n <> 0) or (i < 0)) then begin
    edit3.Color := clred;
    ShowMessage('Îøéâè! Íâëüçü ââîäèò ñèíîëî èèè îððèòàòâèíüâ çíà-âîëü â [Ñòìèìñòù íáííáíñêîé îèàòù].');
  end
  else begin
    edit3.Color := clwhite;
    str2 := edit2.Text;
    val(str2, i, b);
    if((b <> 0) or (i < 0)) then begin
      edit2.Color := clred;
      ShowMessage('Îøéâè! Íâëüçü ââîäèò ñèíîëî èèè îððèòàòâèíüâ çíà-âîëü â [Êîëëè-âñòâî áâñîèàòíüò ìèíóò].');
    end
    else begin
      edit2.Color := clwhite;
      str3 := edit1.Text;
      val(str3, i, h);
      if((h <> 0) or (i < 0)) then begin
        edit1.Color := clred;
        ShowMessage('Îøéâè! Íâëüçü ââîäèò ñèíîëî èèè îððèòàòâèíüâ çíà-âîëü â [Êîëëè-âñòâî ëðàèíüò íííâðíâ].');
      end
      else begin
        edit1.Color := clwhite;
        str4 := edit4.text;
        val(str4, i, l);
        if((l <> 0) or (i < 0)) then begin
          edit4.Color := clred;
          ShowMessage('Îøéâè! Íâëüçü ââîäèò ñèíîëî èèè îððèòàòâèíüâ çíà-âîëü â [Ëìòâðíâð òðàðèè].');
        end
        else begin
          edit4.Color := clwhite;

          //-----
          //-----
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		95


```
if(combobox1.ItemIndex = 0) then begin
  form1.stringgrid1.cells[0, form1.add_rowindex] := combobox1.items[0];
end;
if(combobox1.ItemIndex = 1) then begin
  form1.stringgrid1.cells[0, form1.add_rowindex] := combobox1.items[1];
end;
if(combobox1.ItemIndex = 2) then begin
  form1.stringgrid1.cells[0, form1.add_rowindex] := combobox1.items[2];
end;
if(combobox1.ItemIndex = 3) then begin
  form1.stringgrid1.cells[0, form1.add_rowindex] := combobox1.items[3];
end;
//òàðèè
if(combobox2.ItemIndex = 0) then begin
  form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[0];
end;
if(combobox2.ItemIndex = 1) then begin
  form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[1];
end;
if(combobox2.ItemIndex = 2) then begin
  form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[2];
end;
if(combobox2.ItemIndex = 3) then begin
  form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[3];
end;
if(combobox2.ItemIndex = 4) then begin
  form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[4];
end;
if(combobox2.ItemIndex = 5) then begin
  form1.StringGrid1.Cells[1, form1.add_rowindex] := combobox2.Items[5];
end;
//ñòìèìñòü àáíáíñêîé ìèàòü
form1.StringGrid1.Cells[2, form1.add_rowindex] := edit3.Text;
//èíòáðíâð òðàðèè
{
if(combobox3.ItemIndex = 0) then begin
  form1.StringGrid1.Cells[3, form1.add_rowindex] := combobox3.Items[0];
end;
if(combobox3.ItemIndex = 1) then begin
  form1.StringGrid1.Cells[3, form1.add_rowindex] := combobox3.Items[1];
end;
if(combobox3.ItemIndex = 2) then begin
  form1.StringGrid1.Cells[3, form1.add_rowindex] := combobox3.Items[2];
end;
if(combobox3.ItemIndex = 3) then begin
  form1.StringGrid1.Cells[3, form1.add_rowindex] := combobox3.Items[3];
end;
}
form1.StringGrid1.Cells[3, form1.add_rowindex] := edit4.Text;
//êìèèè÷âñòâì áâñìèàðíüò ìèíóò
form1.StringGrid1.Cells[4, form1.add_rowindex] := edit2.text;
//êìèèè÷âñòâì ëðàèíüò ììâðíâ
form1.StringGrid1.Cells[5, form1.add_rowindex] := edit1.Text;
//âàòà ìÿâèèèè òàðèèà
form1.stringGrid1.Cells[6, form1.add_rowindex] :=
datetostr(datetimepicker1.Date);
form1.add_rowindex := form1.add_rowindex + 1;
if(form1.add_rowindex > form1.StringGrid1.RowCount) then begin
  form1.StringGrid1.RowCount := form1.StringGrid1.RowCount + 1;
end;
edit1.Clear;
//maskedit1.Clear;
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		96

```
form3.close;
    end;
    end;
    end;
end;
end;
procedure DrawRounded(Control: TWinControl) ;
var
    R: TRect;
    Rgn: HRGN;
begin
    with Control do
    begin
        R := ClientRect;
        rgn := CreateRoundRectRgn(R.Left, R.Top, R.Right, R.Bottom, 20, 20) ;
        Perform(EM_GETRECT, 0, lParam(@r)) ;
        InflateRect(r, - 4, - 4) ;
        Perform(EM_SETRECTNP, 0, lParam(@r)) ;
        SetWindowRgn(Handle, rgn, True) ;
        Invalidate;
    end;
end;
procedure TForm3.FormCreate(Sender: TObject);
begin
    DrawRounded(bitbtn1);
    form3.BorderIcons := [biSystemMenu];
end;
end.
```

Исходный код формы изменения находится в модуля Unit5.pas

```
unit Unit5;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, StdCtrls, Buttons, ExtCtrls, Mask, ComCtrls;
type
    TForm5 = class(TForm)
        Bevel1: TBevel;
        BitBtn1: TBitBtn;
        Label1: TLabel;
        Label2: TLabel;
        Label3: TLabel;
        Edit3: TEdit;
        Bevel2: TBevel;
        Label4: TLabel;
        Label5: TLabel;
        Edit5: TEdit;
        BitBtn2: TBitBtn;
        BitBtn3: TBitBtn;
        Label6: TLabel;
        Edit6: TEdit;
        Label7: TLabel;
        ComboBox1: TComboBox;
        ComboBox2: TComboBox;
        DateTimePicker1: TDateTimePicker;
        BitBtn4: TBitBtn;
        Edit1: TEdit;
        BitBtn5: TBitBtn;
        procedure BitBtn1Click(Sender: TObject);
        procedure BitBtn2Click(Sender: TObject);
        procedure BitBtn3Click(Sender: TObject);
    end;
```

					Лист	
					97	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

```

    procedure FormCreate(Sender: TObject);
    procedure BitBtn4Click(Sender: TObject);
    procedure BitBtn5Click(Sender: TObject);
    //procedure BitBtn4Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;
var
    Form5: TForm5;
    en_erorr_char : set of char;
    ru_error_char : set of char;
    lower_en_error_char : set of char;
    lower_ru_error_char: set of char;
implementation
uses Unit1, Unit6, Unit7, Unit3, Unit24;
{$R *.dfm}
procedure TForm5.BitBtn1Click(Sender: TObject);
var size, size2 , i, j, n, k, l, b, v: integer;
var str, str2, str3, str4, str5: string;
begin
    form1.stringgrid1.Cells[0, 0] := 'İîáëëüîüé îîâðàðîð';
    form1.stringgrid1.Cells[1, 0] := 'Òàðèè';
    form1.stringgrid1.Cells[2, 0] := 'Ñòìèìîñòü íáííáíñêîé îèàòü';
    form1.stringgrid1.Cells[3, 0] := 'Êìðâðíâð òðàðèè (ÌÁ)';
    form1.stringgrid1.Cells[4, 0] := 'Êìèèè÷âñòâî áâñîèàðîüò îèíóò';
    form1.stringgrid1.Cells[5, 0] := 'Êìèèè÷âñòâî ëðàèüò íîîâðíâ';
    form1.stringgrid1.Cells[6, 0] := 'Äààà îîÿâèèèÿ òàðèèà';
    en_erorr_char := ['A'..'Z'];
    ru_error_char := ['À'..'ß'];
    lower_en_error_char := ['a'..'z'];
    lower_ru_error_char := ['à'..'ÿ'];
    //-----
    str := edit3.Text;
    val(str, i, n);
    if((n <> 0) or (i < 0)) then begin
        edit3.Color := clred;
        showmessage('İøèáèà! Íâëüçÿ ââîâèèü ñèîîèü èèè îðèèàðâèüîâ çíà÷âèèÿ â [còìèìîñòü àáííáíñêîé îèàòü].');
    end
    else begin
        edit3.Color := clwhite;
        str2 := edit5.text;
        val(str2, i, b);
        if((b <> 0) or (i < 0)) then begin
            edit5.Color := clred;
            showmessage('İøèáèà! Íâëüçÿ ââîâèèü ñèîîèü èèè îðèèàðâèüîâ çíà÷âèèÿ â [êìèèè÷âñòâî ëðàèüò íîîâðíâ].');
        end
        else begin
            edit5.Color := clwhite;
            str3 := edit6.Text;
            val(str3, i, v);
            if((v <> 0) or (i < 0)) then begin
                edit6.Color := clred;
                showmessage('İøèáèà! Íâëüçÿ ââîâèèü ñèîîèü èèè îðèèàðâèüîâ çíà÷âèèÿ â [êìèèè÷âñòâî áâñîèàðîüò îèíóò].');
            end
            else begin
                edit6.Color := clwhite;
                str4 := edit1.Text;
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		98

```

        val(str4, i, v);
        if((v <> 0) or (i < 0)) then begin
            edit1.Color := clred;
            showMessage('Îøéáèà!      Íäëüçÿ      ââîäèòü      ñèìâîëû      èèèè
îòðèèòàòäëüíû çíà÷áìèÿ â [Èìòäðíàò òðàðèè].');
            end
            else begin
                edit1.Color := clwhite;
                str5 := edit1.text;
                val(str5, i, l);
                if((l <> 0) or (i < 0)) then begin
                    edit1.Color := clred;
                    showMessage('Îøéáèà!      Íäëüçÿ      ââîäèòü      ñèìâîëû      èèèè
îòðèèòàòäëüíû çíà÷áìèÿ â [Èìòäðíàò òðàðèè].');
                    end
                    else begin
                        edit1.Color := clwhite;
                        form1.StringGrid1.Cells[0, form1.row_index] := combobox1.Text;
                        form1.StringGrid1.Cells[1, form1.row_index] := combobox2.Text;
                        form1.StringGrid1.Cells[2, form1.row_index] := edit3.Text;
                        form1.StringGrid1.Cells[3, form1.row_index] := edit1.Text;
                        form1.StringGrid1.Cells[4, form1.row_index] := edit5.Text;
                        form1.StringGrid1.Cells[5, form1.row_index] := edit6.Text;
                        form1.StringGrid1.Cells[6, form1.row_index] :=
datetostr(datetimepicker1.Date);
                        form5.Hide;
                        form1.show;
                        end;
                    end;
                end;
            end;
        end;
        //-----
        {
            form1.StringGrid1.Cells[0, form1.row_index] := combobox1.Text;
            form1.StringGrid1.Cells[1, form1.row_index] := combobox2.Text;
            form1.StringGrid1.Cells[2, form1.row_index] := edit3.Text;
            form1.StringGrid1.Cells[3, form1.row_index] := combobox3.Text;
            form1.StringGrid1.Cells[4, form1.row_index] := edit5.Text;
            form1.StringGrid1.Cells[5, form1.row_index] := edit6.Text;
            form1.StringGrid1.Cells[6, form1.row_index] :=
datetostr(datetimepicker1.Date);
            form5.Hide;
            form1.show;
        }
        form1.Label9.Caption := 'Ñòàðòñ: èçìâí,í';
    end;
    procedure TForm5.BitBtn2Click(Sender: TObject);
    var n, k, i, h, f, d, s: integer;
    var b, b1, b2: integer;
    var str, str1, str2, str3: string;
    begin
        form6.Memo1.Clear;
        str := form5.edit3.text;
        val(str, i, n);
        if(n <> 0) then begin
            form5.Edit3.Color := clred;
            form6.memo1.lines.add(['îøéáèà â edit3] : (ñòìèìñòü äáííâíñèé íèàòü)');
            for h:=1 to length(str) do begin
                val(str[h], d, f);
                if(f <> 0) then begin

```

```

        form6.Memo1.Lines.Add('[.....]: ' + str + ' : îøéâêà: [ ' +
str[h] + ' ]');
        end;
    end;
end
else begin
    form5.edit3.Color := clwhite;
end;
form6.Memo1.Lines.Add('-----');
str1 := form5.Edit5.Text;
val(str1, i, n);
if(n <> 0) then begin
    form5.Edit5.Color := clred;
    form6.memo1.lines.add('[îøéâêà â edit5] : (êîëëë÷ãñòâî ëþáëìûõ íîîâðîâ)');
    for h:=1 to length(str1) do begin
        val(str1[h], d, f);
        if(f <> 0) then begin
            form6.Memo1.Lines.Add('[.....]: ' + str1 + ' : îøéâêà: [ ' +
str1[h] + ' ]');
            end;
        end;
    end
else begin
    edit5.Color := clwhite;
end;
form6.Memo1.Lines.Add('-----');
str2 := form5.Edit6.Text;
val(str2, i, n);
if(n <> 0) then begin
    form5.Edit6.Color := clred;
    form6.memo1.lines.add('[îøéâêà â edit6] : (êîëëë÷ãñòâî áãñîëàðîíûõ
îèíîð)');
    for h:=1 to length(str2) do begin
        val(str2[h], d, f);
        if(f <> 0) then begin
            form6.Memo1.Lines.Add('[.....]: ' + str2 + ' : îøéâêà: [ ' +
str2[h] + ' ]');
            end;
        end;
    end
else begin
    edit6.Color := clwhite;
end;
form6.Memo1.Lines.Add('-----');
str3 := form5.Edit1.Text;
val(str3, i, n);
if(n <> 0) then begin
    form5.Edit1.Color := clred;
    form6.memo1.lines.add('[îøéâêà â edit1] : ( èíðâðîíàð òðàðëë )');
    for h:=1 to length(str3) do begin

        val(str3[h], d, f);
        if(f <> 0) then begin
            form6.Memo1.Lines.Add('[.....]: ' + str3 + ' : îøéâêà: [ ' +
str3[h] + ' ]');
            end;
        end;
    end
else begin
    edit1.Color := clwhite;
end;
form6.show;
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						100
Изм.	Лист	№ докум.	Подп.	Дата		

```
end;
procedure TForm5.BitBtn3Click(Sender: TObject);
begin
  //form7.label1.caption := edit1.text;
  //form7.label2.Caption := Edit2.Text;
  //form7.Label3.caption := Edit3.Text;
  //form7.Label4.caption := Edit4.text;
  //form7.label5.Caption := Edit5.Text;
  //form7.show;
end;
procedure DrawRounded(Control: TWinControl) ;
var
  R: TRect;
  Rgn: HRGN;
begin
  with Control do
    begin
      R := ClientRect;
      rgn := CreateRoundRectRgn(R.Left, R.Top, R.Right, R.Bottom, 20, 20) ;
      Perform(EM_GETRECT, 0, lParam(@r)) ;
      InflateRect(r, - 4, - 4) ;
      Perform(EM_SETRECTNP, 0, lParam(@r)) ;
      SetWindowRgn(Handle, rgn, True) ;
      Invalidate;
    end;
  end;
procedure TForm5.FormCreate(Sender: TObject);
begin
  DrawRounded(bitbtn1);
  DrawRounded(bitbtn2);
  DrawRounded(bitbtn3);
  DrawRounded(bitbtn4);
  form5.BorderIcons := [biSystemMenu];
end;
procedure TForm5.BitBtn4Click(Sender: TObject);
var n, k, i, h, f, d, s: integer;
var b, b1, b2: integer;
var str, str1, str2: string;
begin
  str := '';
  str1 := '';
  str2 := '';
  h :=1 ;
  str := form5.edit3.text;
  val(str, i, n);
  if(n <> 0) then begin
    for h:=1 to length(str) do begin

      val(str[h], d, f);
      if(f <> 0) then begin
        delete(str, h, 1);
      end;
    end;
  end
  else begin
    form5.edit3.Color := clwhite;
  end;
  h := 1;
  str1 := form5.Edit5.Text;
  val(str1, i, n);
  if(n <> 0) then begin
    for h:=1 to length(str1) do begin
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						101
Изм.	Лист	№ докум.	Подп.	Дата		

```

        val(str1[h], d, f);
        if(f <> 0) then begin
            delete(str1, h, 1);
        end;
    end;
end
else begin
    edit5.Color := clwhite;
end;
h:=1;
str2 := form5.Edit6.Text;
val(str2, i, s);
if(s <> 0) then begin
    for h:=1 to length(str2) do begin
        val(str2[h], d, f);
        if(f <> 0) then begin
            delete(str2, h, 1);
        end;
    end;
end
else begin
    edit6.Color := clwhite;
end;
edit3.text := str;
edit5.text := str1;
edit6.text := str2;
end;
procedure TForm5.BitBtn5Click(Sender: TObject);
begin
    form24.show;
end;
end.
procedure TForm5.BitBtn4Click(Sender: TObject);
var n, k, i, h, f, d, s: integer;
var b, b1, b2: integer;
var str, str1, str2: string;
begin
end;
```

Исходный код формы проверки данных представлен в модуле Unit7.pas

```

unit Unit6;
interface
uses
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,Dialogs, StdCtrls, Buttons, ExtCtrls;
type
    TForm6 = class(TForm)
        Bevel1: TBevel;
        Memo1: TMemo;
        BitBtn1: TBitBtn;
        procedure FormCreate(Sender: TObject);
        procedure BitBtn1Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;
var
    Form6: TForm6;
implementation
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						102
Изм.	Лист	№ докум.	Подп.	Дата		

```
uses Unit5;
{$R *.dfm}
procedure TForm6.FormCreate(Sender: TObject);
var n, k, i: integer;
begin
end;
procedure TForm6.BitBtn1Click(Sender: TObject);
begin
memo1.clear;
form6.hide;
end;
end.
```

Исходный код формы изменения стоимости представлен в модуле Unit24.pas

```
unit Unit24;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, StdCtrls, Buttons;
type
  TForm24 = class(TForm)
    ComboBox1: TComboBox;
    Label1: TLabel;
    Edit1: TEdit;
    Label2: TLabel;
    BitBtn1: TBitBtn;
    procedure FormCreate(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure Edit1KeyPress(Sender: TObject; var Key: Char);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form24: TForm24;
implementation
uses Unit1;
{$R *.dfm}
procedure TForm24.FormCreate(Sender: TObject);
begin
  form24.BorderIcons := [biSystemMenu];
end;
procedure TForm24.BitBtn1Click(Sender: TObject);
var i, j, k : integer;
var str, str2: string;
begin
  if(combobox1.Text = '') then begin
    ShowMessage('Îèëâè! çìà÷àìèâ [ îîâðàðîðà ] äìëæîî áùòù âûáðàîî');
  end
  else begin
    str := combobox1.Text;
    str2 := edit1.Text;
    for i:=0 to form1.stringgrid1.colcount - 1 do begin
      for j:=1 to form1.stringgrid1.rowcount - 1 do begin
        if(form1.stringgrid1.cells[1, j] = str) then begin
          form1.stringgrid1.cells[2, j] := str2;
        end;
      end;
    end;
  end;
end;
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		103


```
        end;
    end;
end;
procedure TForm24.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
    if not (Key in ['0'..'9', #8]) then Key := #0;
end;
end.
```

Исходный код формы поиска представлен в модуле Unit17.pas

```
unit Unit17;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, StdCtrls, Buttons;
type
    TForm17 = class(TForm)
        Edit1: TEdit;
        BitBtn1: TBitBtn;
        ListBox1: TListBox;
        procedure FormCreate(Sender: TObject);
        procedure Edit1KeyPress(Sender: TObject; var Key: Char);
        procedure Edit1Change(Sender: TObject);
        procedure BitBtn1Click(Sender: TObject);
        procedure ListBox1Click(Sender: TObject);
        procedure Edit1Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
        col_index, row_index : integer;
        draw_choice: integer;
        str, str2, tmp: string;
    end;
var
    Form17: TForm17;
    draw_choice: integer;
implementation
uses Unit1;
{$R *.dfm}
procedure DrawRounded(Control: TWinControl) ;
var
    R: TRect;
    Rgn: HRGN;
begin
    with Control do
    begin
        R := ClientRect;
        rgn := CreateRoundRectRgn(R.Left, R.Top, R.Right, R.Bottom, 20, 20) ;
        Perform(EM_GETRECT, 0, lParam(@r)) ;
        InflateRect(r, - 4, - 4) ;
        Perform(EM_SETRECTNP, 0, lParam(@r)) ;
        SetWindowRgn(Handle, rgn, True) ;
        Invalidate;
    end;
end;
procedure TForm17.FormCreate(Sender: TObject);
begin
    form17.Left := 570;
    form17.Top := 275;
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						104
Изм.	Лист	№ докум.	Подп.	Дата		

```
//Form17.Brush.Style := bsClear;
//Form17.BorderStyle := bsNone;
//Form17.TransparentColorValue := clRed;
//Form17.transparentcolor := true;
//Form17.Color := clRed;
form17.BorderIcons := [biSystemMenu];
end;
procedure TForm17.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
  listbox1.Visible := true;
end;
procedure TForm17.Edit1Change(Sender: TObject);
  var i, j, k, size, n, ii, jj, hh: integer;
  var str3: string;
begin
  for i:=0 to listbox1.Items.Count - 1 do begin
    listbox1.Items.Delete(i);
  end;
  str := AnsiLowerCase(Edit1.Text);
  if(str = ' ') then begin
    for i:=0 to form1.StringGrid1.ColCount - 1 do begin

      for j:=1 to form1.StringGrid1.RowCount - 1 do begin

        form1.stringgrid1.Cells[i, j] := form1.StringGrid1.Cells[i, j];
      end;
    end;
  end;
  for i:=0 to form1.StringGrid1.ColCount - 1 do begin
    for j:=1 to form1.StringGrid1.RowCount - 1 do begin
      str2 := AnsiLowerCase(form1.stringgrid1.Cells[i, j]);
      str3 := form1.StringGrid1.Cells[i, j];
      n := pos(str, str2);
      if(n = 0) then begin
        //ridDeleteRow(j, stringgrid1);
        //edit1.Color := clred;
        continue;
      end
      else if(n <> 0) then begin
        //col_index := i;
        //row_index := j;
        for ii:=0 to listbox1.Items.Count - 1 do begin
          listbox1.Items.Delete(ii);
        end;
        listbox1.Items.Add(str3);
        break;
      end;
    end;
  end;
end;
procedure TForm17.BitBtn1Click(Sender: TObject);
var size, index, i, j, k, s, n, m: integer;
var str, str2, str3: string;
begin
  //listbox1.Clear;
  //listbox1.Visible := false;
  //edit1.Clear;
//stringgrid1.Cells[0, 0] := 'îîáëëüîúé îîäðàðîð';
//stringgrid1.Cells[1, 0] := 'Òàðèèð';
//stringgrid1.Cells[2, 0] := 'Ñòîèèîñòü íáííáíñéíé îèàòû';
//stringgrid1.Cells[3, 0] := 'Èíðáðíáð òðàòèè';
//stringgrid1.Cells[4, 0] := 'Êíèèè÷áñòáí ááñíèàòíúð ìèíóð';
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		105

```
//stringgrid1.Cells[5, 0] := 'Êîëëë÷ãñðâî ëþáëìüõ ïîîåðîâ';
//stringgrid1.Cells[6, 0] := 'Ààðà ïîÿâëåíîÿ òàðëòà';
draw_choice := 0;
m := 0;
str := Edit1.text;
for i:=0 to form1.StringGrid1.ColCount - 1 do begin
  for j:=1 to form1.StringGrid1.RowCount - 1 do begin
    str2 := form1.StringGrid1.Cells[i, j];
    n := pos(str, str2);
    if(n = 0) then begin
      continue;
    end
    else if(n <> 0) then begin
      form1.col_index := i;
      form1.row_index := j;
      draw_choice :=1;
      form1.str_grid := str2;
      str3 := form1.stringgrid1.Cells[i, j];
      form1.stringgrid1.Cells[i, j] := str3;
    end;
  end;
end;
form17.Close;
listbox1.Visible := false;
end;
procedure TForm17.ListBox1Click(Sender: TObject);
begin
  edit1.Text := listbox1.Items[listbox1.Itemindex];
end;
procedure TForm17.Edit1Click(Sender: TObject);
begin
  edit1.Clear;
end;
end.
```

Исходный код формы сортировки представлен в модуле Unit11.pas

```
unit Unit11;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Menus, StdCtrls, Buttons, Grids, ExtCtrls;
type
  TForm11 = class(TForm)
    StringGrid1: TStringGrid;
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    BitBtn3: TBitBtn;
    MainMenu1: TMainMenu;
    N1: TMenuItem;
    BitBtn4: TBitBtn;
    Timer1: TTimer;
    BitBtn5: TBitBtn;
    procedure N1Click(Sender: TObject);
    procedure BitBtn4Click(Sender: TObject);
    procedure BitBtn2Click(Sender: TObject);
    procedure BitBtn3Click(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure BitBtn5Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						106
Изм.	Лист	№ докум.	Подп.	Дата		

```
private
{ Private declarations }
function RunCaptured(const _dirName, _exeName, _cmdLine: string): Boolean;
public
{ Public declarations }
end;
var
Form11: TForm11;
implementation
uses Unit1, Unit12;
{$R *.dfm}
function TForm11.RunCaptured(const _dirName, _exeName, _cmdLine: string):
Boolean;
var
start: TStartupInfo;
procInfo: TProcessInformation;
tmpName: string;
tmp: Windows.THandle;
tmpSec: TSecurityAttributes;
res: TStringList;
return: Cardinal;
begin
Result := False;
try
{ Setze ein Temporares File }
{ Set a temporary file }
tmpName := 'Test.tmp';
FillChar(tmpSec, SizeOf(tmpSec), #0);
tmpSec.nLength := SizeOf(tmpSec);
tmpSec.bInheritHandle := True;
tmp := Windows.CreateFile(PChar(tmpName),
Generic_Write, File_Share_Write,
@tmpSec, Create_Always, File_Attribute_Normal, 0);
try
FillChar(start, SizeOf(start), #0);
start.cb := SizeOf(start);
start.hStdOutput := tmp;
start.dwFlags := StartF_UseStdHandles or StartF_UseShowWindow;
start.wShowWindow := SW_Minimize;
{ Starte das Programm }
{ Start the program }
if CreateProcess(nil, PChar(_exeName + ' ' + _cmdLine), nil, nil, True,
0, nil, PChar(_dirName), start, procInfo) then
begin
SetPriorityClass(procInfo.hProcess, Idle_Priority_Class);
WaitForSingleObject(procInfo.hProcess, Infinite);
GetExitCodeProcess(procInfo.hProcess, return);
Result := (return = 0);
CloseHandle(procInfo.hThread);
CloseHandle(procInfo.hProcess);
Windows.CloseHandle(tmp);
{ Die Ausgaben hinzufügen }
{ Add the output }
res := TStringList.Create;
try
res.LoadFromFile(tmpName);
//Memol.Lines.AddStrings(res);
finally
res.Free;
end;
Windows.DeleteFile(PChar(tmpName));
end
end
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		107

```

else
begin
    Application.MessageBox(PChar(SysErrorMessage(GetLastError())),
        'RunCaptured Error', MB_OK);
end;
except
    Windows.CloseHandle(tmp);
    Windows.DeleteFile(PChar(tmpName));
    raise;
end;
finally
end;
end;
procedure TForm11.N1Click(Sender: TObject);
var i, j : integer;
begin
    stringgrid1.ColCount := 7;
    stringgrid1.RowCount := form1.StringGrid1.RowCount;
    stringgrid1.Cells[0, 0] := 'İîáèëüíúé îîâðàðîð';
    stringgrid1.Cells[1, 0] := 'Òàðèèð';
    stringgrid1.Cells[2, 0] := 'Ñòìèìñòü íáííáíñêîé îëàòü';
    stringgrid1.Cells[3, 0] := 'Ëìòâðíàð òðàðèè';
    stringgrid1.Cells[4, 0] := 'Êîëëè÷ñòîâî äîðîâîüò ìèíóò';
    stringgrid1.Cells[5, 0] := 'Êîëëè÷ñòîâî èðàèüò ííàððíà';
    stringgrid1.Cells[6, 0] := 'Àààà îîÿâèèèè òàðèèà';
    for i:=0 to 6 do begin
        for j:=1 to form1.StringGrid1.RowCount - 1 do begin
            stringgrid1.Cells[i, j] := ' ';
        end;
    end;
    for i:=0 to 6 do begin
        for j:=1 to form1.StringGrid1.RowCount - 1 do begin
            stringgrid1.Cells[i, j] := form1.StringGrid1.Cells[i, j];
        end;
    end;
end;
procedure TForm11.BitBtn4Click(Sender: TObject);
var i, j : integer;
begin
    for i:=0 to form1.StringGrid1.ColCount - 1 do begin
        for j:=1 to form1.StringGrid1.RowCount - 1 do begin
            form1.StringGrid1.Cells[i, j] := form11.StringGrid1.Cells[i, j];
        end;
    end;
    form1.add_rowindex := form1.StringGrid1.RowCount;
    form11.Hide;
end;
procedure TForm11.BitBtn2Click(Sender: TObject);
var a, g, h : integer;
var r, q1, w1 : string;
begin
    for a := stringgrid1.RowCount - 1 downto 0 do begin
        for g := 1 to a - 1 do begin
            if strtoint(stringgrid1.Cells[2, g]) < strtoint(stringgrid1.Cells[2, g
+ 1]) then for h:=stringgrid1.ColCount -1 downto 0 do begin
                r := stringgrid1.Cells[h, g];
                stringgrid1.Cells[h, g] := stringgrid1.Cells[h, g + 1];
                stringgrid1.Cells[h, g + 1] := r;
            end;
        end;
    end;
end;
end;
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		108

```
procedure TForm11.BitBtn3Click(Sender: TObject);
var a, g, h : integer;
var r, ql, wl : string;
begin
  for a := stringgrid1.RowCount - 1 downto 0 do begin
    for g := 1 to a - 1 do begin
      if strtoint(stringgrid1.Cells[2, g]) > strtoint(stringgrid1.Cells[2, g
+ 1]) then for h:=stringgrid1.ColCount -1 downto 0 do begin
        r := stringgrid1.Cells[h, g];
        stringgrid1.Cells[h, g] := stringgrid1.Cells[h, g + 1];
        stringgrid1.Cells[h, g + 1] := r;
      end;
    end;
  end;
end;
procedure TForm11.BitBtn1Click(Sender: TObject);
begin
  form12.show;
end;
procedure TForm11.Timer1Timer(Sender: TObject);
var i, j: integer;
begin
  stringgrid1.ColCount := 7;
  stringgrid1.RowCount := form1.StringGrid1.RowCount;
  stringgrid1.Cells[0, 0] := 'İîáèëüíúé îîâðàðîð';
  stringgrid1.Cells[1, 0] := 'Òàðèèð';
  stringgrid1.Cells[2, 0] := 'Ñòìèìñòü íáííáíñêîé îèàòû';
  stringgrid1.Cells[3, 0] := 'Èìðàðíàð òðàðèè';
  stringgrid1.Cells[4, 0] := 'Êîëëë-âñòâî áâñîèàðîüð îèíóð';
  stringgrid1.Cells[5, 0] := 'Êîëëë-âñòâî ëðàèüð íîîðîîâ';
  stringgrid1.Cells[6, 0] := 'Äààà îîÿâëîëÿ òàðèèà';
  for i:=0 to 6 do begin
    for j:=1 to form1.StringGrid1.RowCount - 1 do begin
      stringgrid1.Cells[i, j] := ' ';
    end;
  end;
  for i:=0 to 6 do begin
    for j:=1 to form1.StringGrid1.RowCount - 1 do begin
      stringgrid1.Cells[i, j] := form1.StringGrid1.Cells[i, j];
    end;
  end;
  timer1.Enabled := false;
end;
procedure TForm11.BitBtn5Click(Sender: TObject);
begin
  form11.Close;
end;
procedure TForm11.FormCreate(Sender: TObject);
begin
  timer1.Enabled := false;
  form11.BorderIcons := [biSystemMenu];
end;
end.
```

Исходный код формы фильтрации представлен в модуле Unit15.pas

```
unit Unit15;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ExtCtrls, StdCtrls, Buttons, Grids, Menus;
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		109

```
type
  TForm15 = class(TForm)
    StringGrid1: TStringGrid;
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    BitBtn3: TBitBtn;
    Bevel1: TBevel;
    Timer1: TTimer;
    Bevel2: TBevel;
    BitBtn4: TBitBtn;
    ComboBox1: TComboBox;
    Label1: TLabel;
    MainMenu1: TMainMenu;
    N1: TMenuItem;
    N2: TMenuItem;
    ComboBox2: TComboBox;
    procedure Timer1Timer(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure BitBtn3Click(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure BitBtn4Click(Sender: TObject);
    procedure BitBtn2Click(Sender: TObject);
    procedure N2Click(Sender: TObject);
    procedure N1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

  TMyGrid = class(TCustomGrid);

var
  Form15: TForm15;
implementation
uses Unit1;
{$R *.dfm}
procedure dell(grid: TStringGrid; num: integer);
begin
  TMyGrid(grid).DeleteRow(num);
end;
procedure TForm15.Timer1Timer(Sender: TObject);
var i, j, k : integer;
begin
  stringgrid1.RowCount := form1.stringgrid1.RowCount;
  for i:=0 to form1.stringgrid1.colcount - 1 do begin
    for j:=1 to form1.stringgrid1.rowcount - 1 do begin
      stringgrid1.Cells[i, j] := form1.stringgrid1.Cells[i, j];
    end;
  end;
  timer1.Enabled := false;
  //timer1.Destroy;
end;
procedure TForm15.FormCreate(Sender: TObject);
begin
  timer1.Enabled := false;
  stringgrid1.Cells[0, 0] := 'Ìíáëëüíúé îîâðàðîð';
  stringgrid1.Cells[1, 0] := 'Òàðèè';
  stringgrid1.Cells[2, 0] := 'Ñòìèìîñòü íáííáíñêíé îèàòû';
  stringgrid1.Cells[3, 0] := 'Èíðàðíàð òðàðèè';
  stringgrid1.Cells[4, 0] := 'Êìèèè÷ãñðâî áãñîèàðíûð îèíóð';
  stringgrid1.Cells[5, 0] := 'Êìèèè÷ãñðâî èðàèíûð íííàðíâ';
  stringgrid1.Cells[6, 0] := 'Äàðà îîÿâèèíèÿ òàðèèà';
  form15.BorderIcons := [biSystemMenu];
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						110
Изм.	Лист	№ докум.	Подп.	Дата		

```
end;
procedure GridDeleteRow(RowNumber: Integer; Grid: TstringGrid);
var
  i: Integer;
begin
  Grid.Row := RowNumber;
  if (Grid.Row = Grid.RowCount - 1) then
    { On the last row}
    Grid.RowCount := Grid.RowCount - 1
  else
    begin
      { Not the last row}
      for i := RowNumber to Grid.RowCount - 1 do
        Grid.Rows[i] := Grid.Rows[i + 1];
      Grid.RowCount := Grid.RowCount - 1;
    end;
  end;
end;
procedure DeleteRow_ss(ARow: Integer; grid: TstringGrid);
var i, j: Integer;
begin
  with grid do
    begin
      for i:=ARow+1 to RowCount-1 do
        for j:=0 to ColCount-1 do
          Cells[j, i-1]:=Cells[j, i];
        for i:=0 to ColCount-1 do
          Cells[i, RowCount-1]:='';
        RowCount:=RowCount-1;
      end;
    end;
  end;
  procedure DeleteRow(ARow: Integer);
  var i, j: Integer;
  begin
    with form15.StringGrid1 do
      begin
        for i:=ARow+1 to RowCount-1 do
          for j:=0 to ColCount-1 do
            Cells[j, i-1]:=Cells[j, i];
          for i:=0 to ColCount-1 do
            Cells[i, RowCount-1]:='';
          RowCount:=RowCount-1;
        end;
      end;
    end;
  procedure TForm15.BitBtn3Click(Sender: TObject);
  var i, j, k: integer;
  var n, str: string;
  begin
    //i := 5;
    //n := inttostr(0);
    n := '0';
    k := 0;
    str := inttostr(0);
    for i:=0 to form15.stringgrid1.ColCount - 1 do begin
      for j:=1 to form15.stringgrid1.RowCount - 1 do begin

        if(stringgrid1.cells[5, j] = str) then begin
          k := j;
          DeleteRow_ss(k, StringGrid1);
          //k := 0;
          break;
          //ShowMessage('g: ' + inttostr(j));
        end
      end
    end
  end
end;
```

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	111
01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ					


```

        else begin
            //ShowMessage('g: ' + inttostr(j));
            //DeleteRow_ss(k, stringgrid1);
            continue;
        end;
    end;
end;
end;
end;
procedure TForm15.BitBtn1Click(Sender: TObject);
var i, j : integer;
begin
    form1.StringGrid1.RowCount := stringgrid1.RowCount;
    for i:=0 to stringgrid1.ColCount - 1 do begin
        for j:=1 to stringgrid1.RowCount - 1 do begin
            form1.StringGrid1.Cells[i, j] := stringgrid1.Cells[i, j];
        end;
    end;
    form15.Close;
    form1.Show;
    form1.add_rowindex := stringgrid1.RowCount;
end;
procedure dell_filtr(col: integer; ch: string);
var i, j, k: integer;
begin
    k := 0;
    for i:=0 to form15.StringGrid1.ColCount - 1 do begin
        for j:=1 to form15.StringGrid1.RowCount - 1 do begin
            if(form15.StringGrid1.Cells[col, j] <> ch) then begin
                k := j;
                DeleteRow_ss(k, form15.StringGrid1);
                break;
            end
            else begin
                continue;
            end;
        end;
    end;
end;
end;
procedure filtr(str: string; ch: string);
var str1, str2, str3, str4, str5: string;
var i, j, k, m, h, l, n : integer;
begin
    if(str = 'a') then begin
        dell_filtr(0, ch);
        //ShowMessage('a');
    end;
    if(str = 'b') then begin
        dell_filtr(1, ch);
    end;
    if(str = 'c') then begin
        dell_filtr(2, ch);
    end;
    if(str = 'd') then begin
        dell_filtr(3, ch);
    end;
    if(str = 'e') then begin
        dell_filtr(4, ch);
    end;
    if(str = 'f') then begin
        dell_filtr(5, ch);
    end;
    if(str = 'g') then begin

```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						112
Изм.	Лист	№ докум.	Подп.	Дата		

```
        dell_filtr(5, ch);
    end;
end;
procedure TForm15.BitBtn4Click(Sender: TObject);
var i, j, k : integer;
var str, s, str2 : string;
begin
    if(combobox1.Text = '') then begin
        ShowMessage('Îøèáêà! Îîëå ñ [òèëüòðàöèéé îî èàëîé êîëîíáêå] äîëæíà áýòü
çàîëèíáíà.');
    end
    else begin
        if(combobox1.ItemIndex = 0) then begin
            filtr('a', combobox2.Text);
            //ShowMessage('1');
        end;
        if(combobox1.ItemIndex = 1) then begin
            filtr('b', combobox2.Text);
            //ShowMessage('2');
        end;
        if(combobox1.ItemIndex = 2) then begin
            filtr('c', combobox2.Text);
            //ShowMessage('3');
        end;
        if(combobox1.ItemIndex = 3) then begin
            filtr('d', combobox2.Text);
            //ShowMessage('4');
        end;
        if(combobox1.ItemIndex = 4) then begin
            filtr('e', combobox2.Text);
            //ShowMessage('5');
        end;
        if(combobox1.ItemIndex = 5) then begin
            filtr('f', combobox2.Text);
            //ShowMessage('6');
        end;
        if(combobox1.ItemIndex = 6) then begin
            filtr('g', combobox2.Text);
            //ShowMessage('7');
        end;
    end;
end;
{
    str := edit1.Text;
    for i:=0 to stringgrid1.ColCount - 1 do begin
        for j:=1 to stringgrid1.RowCount - 1 do begin
            if(stringgrid1.Cells[i, j] <> str) then begin
                k := j;
                DeleteRow(k);
            end
            else begin
                continue;
            end;
        end;
    end;
    edit1.Clear;
}
end;
procedure TForm15.BitBtn2Click(Sender: TObject);
begin
    form15.Close;
end;
procedure TForm15.N2Click(Sender: TObject);
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						113
Изм.	Лист	№ докум.	Подп.	Дата		

```
var i, j, k, n: integer;
begin
  k := 1;
  form15.StringGrid1.RowCount := form1.StringGrid1.RowCount;
  for i:= 0 to form1.StringGrid1.ColCount - 1 do begin
    for j:=1 to form1.StringGrid1.RowCount - 1 do begin
      form15.StringGrid1.Cells[i, j] := form1.StringGrid1.cells[i, j];
    end;
  end;
end;
procedure TForm15.N1Click(Sender: TObject);
var i, j : integer;
begin
  for i:=0 to form15.StringGrid1.ColCount - 1 do begin
    for j:=1 to form15.StringGrid1.RowCount - 1 do begin
      form15.StringGrid1.Cells[i, j] := '';
    end;
  end;
end;
end.
end.
```

Исходный код формы статистики представлен в модуле Unit8.pas

```
unit Unit8;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, Buttons, ExtCtrls, TeeProcs, TeEngine, Chart, Series,
  DateUtils;
type
  TForm8 = class(TForm)
    Chart1: TChart;
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    Series1: TBarSeries;
    Label1: TLabel;
    ComboBox1: TComboBox;
    Bevel1: TBevel;
    BitBtn3: TBitBtn;
    procedure BitBtn2Click(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure BitBtn3Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form8: TForm8;
implementation
uses Unit1;
{$R *.dfm}
procedure TForm8.BitBtn2Click(Sender: TObject);
begin
  form8.Hide;
end;
procedure TForm8.BitBtn1Click(Sender: TObject);
```

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	114

01.2021.2-
400101.1.ПО-21.Г-В-
16-ПЗ

```
var max, min, i, j, n, size: integer;
var str: string;
var a1, life, mts, belsel : integer;
begin
  a1 := 0;
  life := 0;
  mts := 0;
  belsel := 0;
  with Series1 do begin
    if (combobox1.ItemIndex = 0) then begin
      for i:=0 to form1.StringGrid1.ColCount - 1 do begin
        for j:=0 to form1.StringGrid1.RowCount - 1 do begin
          if (form1.StringGrid1.Cells[i, j] = 'A1') then begin
            a1 := a1 + 1;
          end;
          if (form1.StringGrid1.Cells[i, j] = 'Life ;') then begin
            life := life + 1;
          end;
          if (form1.StringGrid1.Cells[i, j] = 'ÌÒÑ') then begin
            mts := mts + 1;
          end;
          if (form1.StringGrid1.Cells[i, j] = 'ÁâëÑä') then begin
            belsel := belsel + 1;
          end;
        end;
      end;
      clear;
      Add(a1, 'A1', clRed);
      Add(life, 'Life ;', clGreen);
      Add(mts, 'ÌÒÑ', clYellow);
      Add(belsel, 'ÁâëÑä', clBlue);
    end;
    if (combobox1.ItemIndex = 1) then begin
      max := -320000;
      min := 320000;
      for i:= 1 to form1.StringGrid1.ColCount - 1 do begin
        for j:=1 to form1.StringGrid1.RowCount - 1 do begin
          if (i = 2) then begin
            if (strtoint(form1.StringGrid1.Cells[2, j]) > max) then begin
              max := strtoint(form1.StringGrid1.Cells[i, j]);
            end;
          end;
        end;
      end;
      for i:= 1 to form1.StringGrid1.RowCount - 1 do begin
        for j:=1 to form1.StringGrid1.RowCount - 1 do begin
          if (i = 2) then begin
            if (strtoint(form1.StringGrid1.Cells[2, j]) < min) then begin
              min := strtoint(form1.StringGrid1.Cells[i, j]);
            end;
          end;
        end;
      end;
      clear;
      Add(max, 'MAX', clRed);
      Add(min, 'MIN', clGreen);
    end;
    if (combobox1.ItemIndex = 2) then begin
      max := -320000;
      min := 320000;
      for i:= 1 to form1.StringGrid1.ColCount - 1 do begin
        for j:=1 to form1.StringGrid1.RowCount - 1 do begin
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		115

```

        if(i = 4) then begin
            if(strtoint(form1.StringGrid1.Cells[2, j]) > max) then begin
                max := strtoint(form1.StringGrid1.Cells[i, j]);
            end;
        end;
    end;
end;
for i:= 1 to form1.StringGrid1.RowCount - 1 do begin

    for j:=1 to form1.StringGrid1.RowCount - 1 do begin
        if(i = 4) then begin
            if(strtoint(form1.StringGrid1.Cells[2, j]) < min) then begin
                min := strtoint(form1.StringGrid1.Cells[i, j]);
            end;
        end;
    end;
end;
clear;
Add(max, 'MAX', clRed);
Add(min, 'MIN', clGreen);
end;
end;
end;
procedure DrawRounded(Control: TWinControl) ;
var
    R: TRect;
    Rgn: HRGN;
begin
    with Control do
    begin
        R := ClientRect;
        rgn := CreateRoundRectRgn(R.Left, R.Top, R.Right, R.Bottom, 20, 20) ;
        Perform(EM_GETRECT, 0, lParam(@r)) ;
        InflateRect(r, - 4, - 4) ;
        Perform(EM_SETRECTNP, 0, lParam(@r)) ;
        SetWindowRgn(Handle, rgn, True) ;
        Invalidate;
    end;
end;
procedure TForm8.FormCreate(Sender: TObject);
begin
    DrawRounded(bitbtn1);
    DrawRounded(bitbtn2);
end;
procedure TForm8.BitBtn3Click(Sender: TObject);
begin
    Series1.Clear;
end;
end.
```

Исходный код формы удаления оператора представлен в модуле Unit14.pas

```

unit Unit14;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, StdCtrls, Buttons, Grids;
type
    TForm14 = class(TForm)
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		116

```
Label1: TLabel;
ComboBox1: TComboBox;
BitBtn1: TBitBtn;
procedure BitBtn1Click(Sender: TObject);
procedure FormCreate(Sender: TObject);
//procedure DeleteRow(ARow: Integer);
private
{ Private declarations }
public
{ Public declarations }
end;
type TMyGrid = class(TCustomGrid);
var
Form14: TForm14;
implementation
uses Unit1;
{$R *.dfm}
procedure dell(grid: TStringGrid; num: integer);
begin
TMyGrid(grid).DeleteRow(num);
end;
procedure GridDeleteRow(RowNumber: Integer; Grid: TStringGrid);
var
i: Integer;
begin
Grid.Row := RowNumber;
if (Grid.Row = Grid.RowCount - 1) then
{ On the last row}
Grid.RowCount := Grid.RowCount - 1
else
begin
{ Not the last row}
for i := RowNumber to Grid.RowCount - 1 do
Grid.Rows[i] := Grid.Rows[i + 1];
Grid.RowCount := Grid.RowCount - 1;
end;
end;
procedure DeleteRow(ARow: Integer);
var i, j: Integer;
begin
with form1.StringGrid1 do
begin
for i:=ARow+1 to RowCount-1 do
for j:=0 to ColCount-1 do
Cells[j, i-1]:=Cells[j, i];
for i:=0 to ColCount-1 do
Cells[i, RowCount-1]:='';
RowCount:=RowCount-1;
end;
end;
procedure TForm14.BitBtn1Click(Sender: TObject);
var str: string;
var i, j, n: integer;
begin
str:=combobox1.text;
n := 0;
if(str <> '') then begin
for i:=0 to form1.stringgrid1.colcount - 1 do begin
for j:=1 to form1.stringgrid1.rowcount - 1 do begin
if(form1.stringgrid1.cells[0,j] = str) then begin
n := j;
DeleteRow(n);
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		117

```
        end;
    end;
end;
form1.add_rowindex := form1.StringGrid1.RowCount;
form14.Hide;
end
else begin
    showmessage('Îøéáêà! Ìëå äëÿ ââîàà íõîîâ');
end;
end;
procedure TForm14.FormCreate(Sender: TObject);
begin
    form14.BorderIcons := [biSystemMenu];
end;
end.
```

Исходный код формы открытия файла представлен в модуле Unit16.pas

```
unit Unit16;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, StdCtrls, Buttons;
type
    TForm16 = class(TForm)
        ListBox1: TListBox;
        BitBtn1: TBitBtn;
        BitBtn2: TBitBtn;
        OpenFileDialog1: TOpenDialog;
        SaveDialog1: TSaveDialog;
        BitBtn3: TBitBtn;
        BitBtn4: TBitBtn;
        procedure BitBtn1Click(Sender: TObject);
        procedure FormCreate(Sender: TObject);
        procedure BitBtn2Click(Sender: TObject);
        procedure BitBtn3Click(Sender: TObject);
        procedure BitBtn4Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;
type
    tele = record
        name: string[10];
        tarif: string[15];
        cost: integer;
        trafic: string[3];
        free_min : integer;
        lovenumber : integer;
        date: string[10];
    end;
var
    Form16: TForm16;
    input: file of tele;
    arr_tele: array[1..100] of tele;
    p, m, n, h, k : integer;
    fs: TFileStream;
implementation
uses Unit1, Unit22;
{$R *.dfm}
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						118
Изм.	Лист	№ докум.	Подп.	Дата		

```
procedure TForm16.BitBtn1Click(Sender: TObject);
var len: integer;
var str: string;
begin
if(opendialog1.Execute) then begin
    len := length(opendialog1.FileName);
    str := opendialog1.FileName;
    if((str[len] = 't') and (str[len - 1] = 'a') and (str[len-2] = 'd') and
(str[len-3] = '.')) then begin
        assignfile(input, opendialog1.Filename);
        reset(input);
        seek(input, 0);
        form1.Label7.Caption := 'Ôàëë: ' + opendialog1.FileName;
        form1.Label9.Caption := 'Ñòàòóñ: îðêðóò';
        p := 0;
        m :=0;
        k := 1;
        h := 1;
        while not(eof(input)) do begin
            p := p + 1;
            //k := k + 1;
            read(input, arr_tele[p]);
            form1.stringgrid1.Cells[m, k] := arr_tele[p].name;
            m := m + 1;
            form1.stringgrid1.Cells[m, k] := arr_tele[p].tarif;
            m := m + 1;
            form1.stringgrid1.Cells[m, k] := inttostr(arr_tele[p].cost);
            m := m + 1;
            form1.stringgrid1.Cells[m, k] := arr_tele[p].trafic;
            m := m + 1;
            form1.stringgrid1.Cells[m, k] := inttostr(arr_tele[p].free_min);
            m := m + 1;
            form1.stringgrid1.Cells[m, k] := inttostr(arr_tele[p].lovenumber);
            m := m + 1;
            form1.stringgrid1.Cells[m, k] := arr_tele[p].date;
            //m := m + 1;
            m := 0;
            //h := h + 1;
            //p := p + 1;
            if(k > form1.stringgrid1.RowCount -1) then begin
                form1.stringgrid1.RowCount:= form1.stringgrid1.RowCount + 1;
            end;
            k := k + 1;
        end;
        form1.add_rowindex := k;
        closefile(input);
        form1.open_file := 1;
        form1.bitbtn11.Enabled := true;
        form1.bitbtn12.Enabled := true;
        form1.bitbtn14.Enabled := true;
        form1.bitbtn13.Enabled := true;
        form1.bitbtn15.Enabled := true;
        form1.BitBtn10.Enabled := true;
        form16.Close;
    end
    //m := 0;
    else begin
        showmessage('Îðåáà! Âóàðáííúê ôàëë ñíàâðæèò ðàññèðåäíëå íà ííàââðæèàââííà
îðíàððàíííé');
    end;
end;
end;
end;
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						119
Изм.	Лист	№ докум.	Подп.	Дата		


```

procedure TForm16.FormCreate(Sender: TObject);
var
  SR: TSearchRec; // iîeñêîâây iãðãlãíiây
  FindRes: Integer;
begin
  form16.BorderIcons := [biSystemMenu];
  listbox1.Clear;
  FindRes := FindFirst('*.dat*', faAnyFile, SR);
  while FindRes = 0 do // iîêà iû iãðlãèi ôàééû (êàòàèiãè), ôi âûiîéiÿòû òèèè
  begin
    ListBox1.Items.Add(SR.Name); // äiáääèäièä â ñièñiê iàçääièä
    // iàèääiîiãl ýèãlãìòà
    FindRes := FindNext(SR); // iðlãlèæãièä iîeñêà iî çäääiîiûi óñèiãèÿi
  end;
  FindClose(SR); // çàèðûäääi iîeñê
end;

procedure DeleteRow(ARow: Integer);
var i, j: Integer;
begin
  with form1.StringGrid1 do
  begin
    for i:=ARow+1 to RowCount-1 do
    for j:=0 to ColCount-1 do
      Cells[j, i-1]:=Cells[j, i];
    for i:=0 to ColCount-1 do
      Cells[i, RowCount-1] := '';
    RowCount:=RowCount-1;
  end;
end;

procedure TForm16.BitBtn2Click(Sender: TObject);
var i, j, h, k, l: integer;
begin
  for h := 1 to form1.StringGrid1.RowCount - 1 do begin
    DeleteRow(h);
  end;
  assignfile(input, listbox1.Items[listbox1.Itemindex]);
  reset(input);
  seek(input, 0);
  form1.Label7.Caption := 'Ôàéé: ' + listbox1.Items[listbox1.Itemindex];
  form1.Label9.Caption := 'Ñòàòóñ: îðèðûò';
  p := 0;
  m := 0;
  k := 1;
  h := 1;
  for i:=0 to form1.StringGrid1.ColCount - 1 do begin

    for j:=1 to form1.StringGrid1.RowCount - 1 do begin
      end;
    end;
  while not(eof(input)) do begin
    p := p + 1;
    //k := k + 1;
    read(input, arr_tele[p]);
    form1.stringgrid1.Cells[m, k] := arr_tele[p].name;
    m := m + 1;
    form1.stringgrid1.Cells[m, k] := arr_tele[p].tarif;
    m := m + 1;
    form1.stringgrid1.Cells[m, k] := inttostr(arr_tele[p].cost);
    m := m + 1;
    form1.stringgrid1.Cells[m, k] := arr_tele[p].trafic;
    m := m + 1;
    form1.stringgrid1.Cells[m, k] := inttostr(arr_tele[p].free min);
  end;
end;

```

```
m := m + 1;
form1.stringgrid1.Cells[m, k] := inttostr(arr_tele[p].lovenumber);
m := m + 1;
form1.stringgrid1.Cells[m, k] := arr_tele[p].date;
m := 0;
if(k > form1.stringgrid1.RowCount -1) then begin
    form1.stringgrid1.RowCount:= form1.stringgrid1.RowCount + 1;
end;
k := k + 1;
end;
closefile(input);
form1.add_rowindex := k;
form16.Close;
form1.file_name := listbox1.Items[listbox1.Itemindex];
form1.open_file := 1;
form1.bitbtn11.Enabled := true;
form1.bitbtn12.Enabled := true;
form1.bitbtn14.Enabled := true;
form1.bitbtn13.Enabled := true;
form1.bitbtn15.Enabled := true;
form1.BitBtn10.Enabled := true;
end;
procedure TForm16.BitBtn3Click(Sender: TObject);
var
    SR: TSearchRec; // ììèñêîâàÿ ìàðàìàíàÿ
    FindRes: Integer;
begin
    form22.show;
    {
    fs := TFileStream.Create('new.dat', fmCreate or fmOpenWrite);
    showmessage('Ôàëë (new.dat) ñîçäàè â êîäîâíêå ìàìå');
    listbox1.Clear;
    FindRes := FindFirst('*.dat*', faAnyFile, SR);
    while FindRes = 0 do // ììåà ìó ìàðìàè òàëëó (åàòàëìå), òì àìììëìÿòó òèëë
    begin
        ListBox1.Items.Add(SR.Name); // àìàààëëàìåà â ñìèñë èàçààìåà
        // ìàëëàìììì ÿëàìàòà
        FindRes := FindNext(SR); // ìðìììåàìåà ììèñëè ìì çàààììì òñììåÿì
    end;
    FindClose(SR); // çàëðóçààì ììèñë
    }
end;
procedure TForm16.BitBtn4Click(Sender: TObject);
var
    SR: TSearchRec; // ììèñëîâàÿ ìàðàìàíàÿ
    FindRes: Integer;
begin
    DeleteFile(listbox1.Items[listbox1.Itemindex]);
    showmessage('Ôàëë ( ' + listbox1.Items[listbox1.Itemindex] + ' )' + ' áóë
    óààë, ì èç êîäîâíêå àëàëììè');
    listbox1.Clear;
    FindRes := FindFirst('*.dat*', faAnyFile, SR);
    while FindRes = 0 do // ììåà ìó ìàðìàè òàëëó (åàòàëìå), òì àìììëìÿòó òèëë
    begin
        ListBox1.Items.Add(SR.Name); // àìàààëëàìåà â ñìèñë èàçààìåà
        // ìàëëàìììì ÿëàìàòà
        FindRes := FindNext(SR); // ìðìììåàìåà ììèñëè ìì çàààììì òñììåÿì
    end;
    FindClose(SR); // çàëðóçààì ììèñë
end;
end.
```

					<div>01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ</div>	Лист
						121
Изм.	Лист	№ докум.	Подп.	Дата		

Исходный код формы создания файла представлен в модуле Unit22.pas

```
unit Unit22;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, Buttons;
type
  TForm22 = class(TForm)
    Edit1: TEdit;
    Label1: TLabel;
    BitBtn1: TBitBtn;
    procedure BitBtn1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form22: TForm22;
  fs: TFileStream;
implementation
uses Unit16;
{$R *.dfm}
procedure TForm22.BitBtn1Click(Sender: TObject);
  var SR: TSearchRec; // ииенѐиâăÿ iâðâiâiâÿ
  var FindRes: Integer;
  var str, str2 : string;
  var i, j, k, len: integer;
begin
  len := length(edit1.text);
  str := edit1.Text;
  if((str[len] = 't') and (str[len - 1] = 'a') and (str[len-2] = 'd') and
(str[len-3] = '.')) then begin
    fs := TFileStream.Create(edit1.Text, fmCreate or fmOpenWrite);
    showmessage('Ôâéè ' + '(' + edit1.Text + ' )' + 'ñîçââi â êiðîââiê iâiêâ');
    form16.listbox1.Clear;
    FindRes := FindFirst('*.dat*', faAnyFile, SR);
    while FindRes = 0 do // ииêâ îû iâðîâèî ôâéèû (êâðâèîâè), òi âûiîiÿòû öèèè
    begin
      form16.ListBox1.Items.Add(SR.Name); // âiâââèâièâ â ñièñiè iâçââièâ
      // iâéââiîiâi ýèâiâiðâ
      FindRes := FindNext(SR); // iðîâièæâièâ iîñèâèâ iî çâââiîiîi óñèiâèÿi
    end;
    FindClose(SR); // çâèðûâââi iîñèâ
    form22.Hide;
  end
  else begin
    showmessage('Îèèâè! òèi ôâéèâ âièæâi âûòû [.dat]');
  end;
end;
procedure TForm22.FormCreate(Sender: TObject);
begin
end;
end.
```

Исходный код формы заблокировать представлен в модуле Unit18.pas

```
unit Unit18;
interface
```

					01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	Лист
						122
Изм.	Лист	№ докум.	Подп.	Дата		

```
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, Buttons;
type
  TForm18 = class(TForm)
    Edit1: TEdit;
    BitBtn1: TBitBtn;
    procedure BitBtn1Click(Sender: TObject);
    procedure Edit1KeyPress(Sender: TObject; var Key: Char);
    procedure Edit1KeyUp(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
    pass: string;
  public
    { Public declarations }
  end;
var
  Form18: TForm18;
implementation
uses Unit1;
{$R *.dfm}
procedure TForm18.BitBtn1Click(Sender: TObject);
var prof: string;
begin
  //pass := '';
  prof := 'admin';
  if(pass = prof) then begin
    pass := '';
    edit1.Clear;
    form1.show;
    form18.Close;
  end
  else begin
    pass := '';
    edit1.Clear;
    showmessage('Îøéâè! Ìàðîëü íà ââðîüé. Îîâîðèòà ïîðîòó ñîîâ.');
```

Исходный код формы формулы предстален в модуле Unit23.pas

					Лист	
					123	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ	

```
unit Unit23;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ExtCtrls, StdCtrls;
type
  TForm23 = class(TForm)
    Label1: TLabel;
    ComboBox1: TComboBox;
    Label2: TLabel;
    Panel1: TPanel;
    Label3: TLabel;
    Label4: TLabel;
    Panel2: TPanel;
    Label5: TLabel;
    Panel3: TPanel;
    Label6: TLabel;
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
    fun_row : integer;
    fun_col : integer;
  end;
var
  Form23: TForm23;
implementation
uses Unit1;
{$R *.dfm}
procedure TForm23.FormCreate(Sender: TObject);
begin
  form1.fun_code := 1;
end;
end.
```

Исходный код функции добавления нового тарифа представлен в модуле Unit25.pas.

```
unit Unit25;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, Buttons;
type
  TForm25 = class(TForm)
    Edit1: TEdit;
    Label1: TLabel;
    BitBtn1: TBitBtn;
    procedure BitBtn1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
```

					<div>Лист</div> <div>400101.1.ПО-21.Г-В-16-ПЗ</div>	<div>124</div>
Изм.	Лист	№ докум.	Подп.	Дата		

```

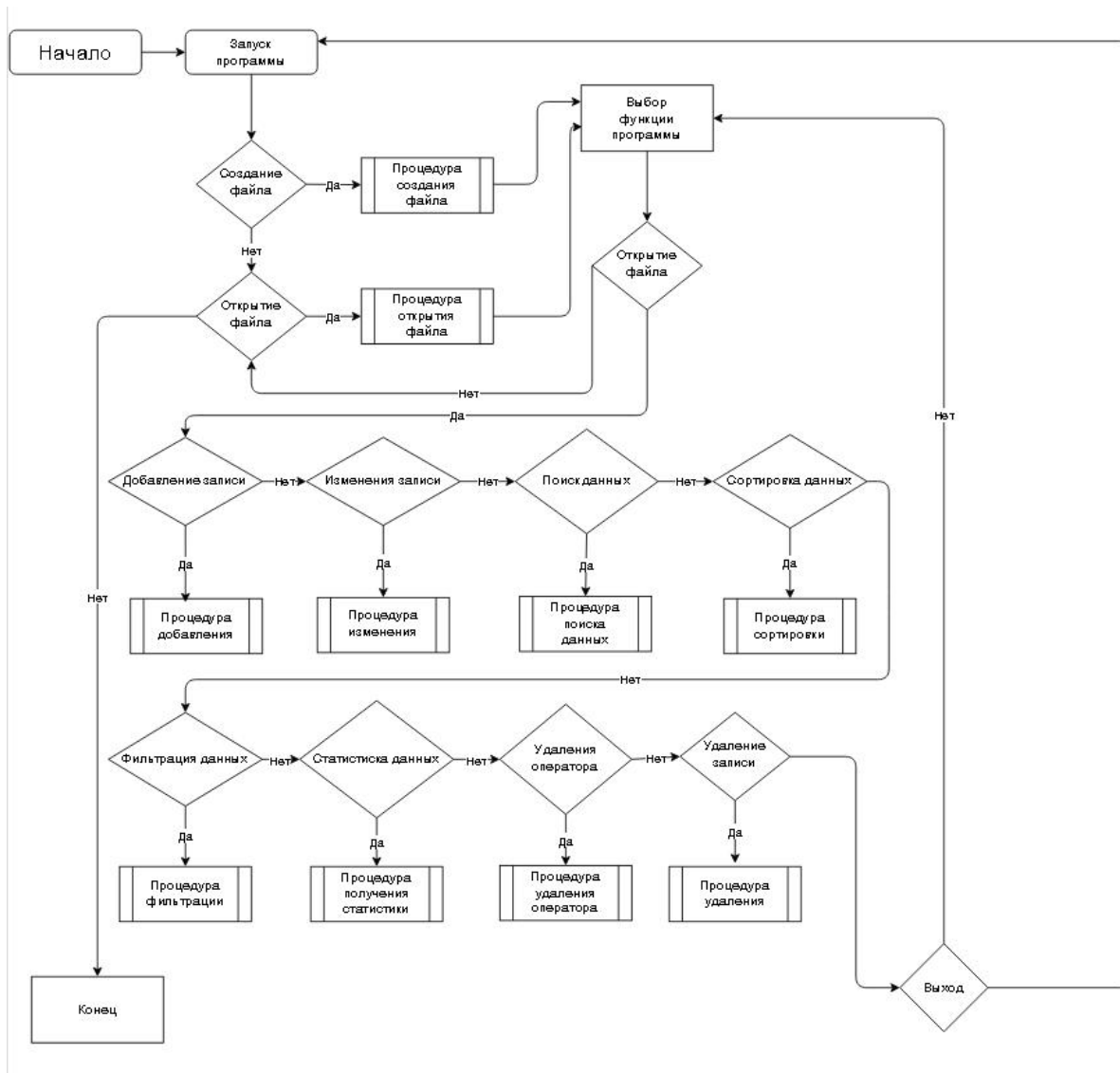
    { Public declarations }
end;
var
    Form25: TForm25;
implementation
uses Unit3, Unit5, Unit15;
{$R *.dfm}
procedure TForm25.BitBtn1Click(Sender: TObject);
var i, j, k : integer;
var str, str2, str3: string;
var err: integer;
begin
    str := edit1.Text;
    err := 1;
    for i:=0 to form3.combobox2.items.Count - 1 do begin
        if(ansilowercase(form3.ComboBox2.Items[i]) = ansilowercase(str)) then
begin
            err := 2;
            break;
        end;
    end;
    if(err = 2) then begin

        showmessage('Ошибка! Данный тариф уже есть в списке');
    end
    else begin
        form3.ComboBox2.Items.Add(str);
        form5.combobox2.items.add(str);
        form15.combobox2.items.add(str);

        showmessage('Тариф [ ' + str + ' ] был добавлен в список');
    end;
end;
procedure TForm25.FormCreate(Sender: TObject);
begin
end;
end.
```

ПРИЛОЖЕНИЕ Б
(обязательное)
Блок-схема программы

					<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>	125
01.2021.2- 400101.1.ПО-21.Г-В- 16-ПЗ					



Форматированный[maksimvendetta658@gmail.com]: Шрифт: 14 пт, Полужирный

Форматированный[maksimvendetta658@gmail.com]: Обычный, Слева

					Лист	
					126	
Изм.	Лист	№ докум.	Подп.	Дата	01.2021.2-400101.1.ПО-21.Г-В-16-ПЗ	