Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №9**

**«**РАБОТА С ТИПИЗИРОВАННЫМИ ФАЙЛАМИ»

**ПО ДИСЦИПЛИНЕ**

**«**РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ**»**

Выполнил: студент учебной группы

ИСПк-101-51-00

Лысков Лев Алексеевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

**В отчете должны отображаться:**

1. **Цель работы**

Целью работы является приобретение и закрепление знаний и умений в организации хранения и использования информации с помощью типизированных файлов, закрепить навыки разработки пользовательского интерфейса

1. **Формулировка задания**

1. Сформировать структуру записи, с которой будут осуществляться работа, и согласовать ее с преподавателем

2. Разработать приложение, позволяющее извлекать набор записей из типизированного файла, визуализировать данный набор, а также позволяющее изменять данные и выполнять сохранение проделанных изменений





1. **Код программы**
2. **unit Main;**
3. **{$mode objfpc}{$H+}**
4. **interface**
5. **uses**
6. **Classes, SysUtils, SQLDB, DB, mysql55conn, mysql80conn, MSSQLConn, Forms,**
7. **Controls, Graphics, Dialogs, ExtCtrls, Buttons, Grids, edit;**
8. **type**
9. **{ TfMain }**
10. **TfMain = class(TForm)**
11. **ButtonPanel: TPanel;**
12. **bAdd: TSpeedButton;**
13. **bEdit: TSpeedButton;**
14. **bDel: TSpeedButton;**
15. **bSort: TSpeedButton;**
16. **SG: TStringGrid;**
17. **procedure bAddClick(Sender: TObject);**
18. **procedure bDelClick(Sender: TObject);**
19. **procedure bEditClick(Sender: TObject);**
20. **procedure bSortClick(Sender: TObject);**
21. **procedure FormClose(Sender: TObject; var CloseAction: TCloseAction);**
22. **procedure FormCreate(Sender: TObject);**
23. **private**
24. **public**
25. **end;**
26. **type**
27. **Contacts = record**
28. **Name: string[100];**
29. **Price: real;**
30. **Factory: string[100];**
31. **Calory: integer;**
32. **HowMany: string[10];**
33. **end;**
34. **var**
35. **fMain: TfMain;**
36. **adres: string; //адрес, откуда запущена программа**
37. **implementation**
38. **{$R \*.lfm}**
39. **{ TfMain }**
40. **procedure TfMain.bAddClick(Sender: TObject);**
41. **begin**
42. **//очищаем поля, если там что-то есть:**
43. **fEdit.fName.Text:= '';**
44. **fEdit.fPrice.Text:= '';**
45. **fEdit.factoryName.Text:= '';**
46. **fEdit.Calory.Text:= '';**
47. **fEdit.HowMany.Text:= '';**
48. **//устанавливаем ModalResult редактора в mrNone:**
49. **fEdit.ModalResult:= mrNone;**
50. **//теперь выводим форму:**
51. **fEdit.ShowModal;**
52. **//если пользователь ничего не ввел - выходим:**
53. **if (fEdit.fName.Text= '') or (fEdit.fPrice.Text= '') then exit;**
54. **//если пользователь не нажал "Сохранить" - выходим:**
55. **if fEdit.ModalResult <> mrOk then exit;**
56. **//иначе добавляем в сетку строку, и заполняем её:**
57. **SG.RowCount:= SG.RowCount + 1;**
58. **SG.Cells[0, SG.RowCount-1]:= fEdit.fName.Text;**
59. **SG.Cells[1, SG.RowCount-1]:= fEdit.fPrice.Text;**
60. **SG.Cells[2, SG.RowCount-1]:= fEdit.factoryName.Text;**
61. **SG.Cells[3, SG.RowCount-1]:= fEdit.Calory.Text;**
62. **SG.Cells[4, SG.RowCount-1]:= fEdit.HowMany.Text;**
63. **end;**
64. **procedure TfMain.bDelClick(Sender: TObject);**
65. **begin**
66. **//если данных нет - выходим:**
67. **if SG.RowCount = 1 then exit;**
68. **//иначе выводим запрос на подтверждение:**
69. **if MessageDlg('Требуется подтверждение',**
70. **'Вы действительно хотите удалить продукт "' +**
71. **SG.Cells[0, SG.Row] + '"?',**
72. **mtConfirmation, [mbYes, mbNo, mbIgnore], 0) = mrYes then**
73. **SG.DeleteRow(SG.Row);**
74. **end;**
75. **procedure TfMain.bEditClick(Sender: TObject);**
76. **begin**
77. **//если данных в сетке нет - просто выходим:**
78. **if SG.RowCount = 1 then exit;**
79. **//иначе записываем данные в форму редактора:**
80. **fEdit.fName.Text:= SG.Cells[0, SG.Row];**
81. **fEdit.fPrice.Text:= SG.Cells[1, SG.Row];**
82. **fEdit.factoryName.Text:= SG.Cells[2, SG.Row];**
83. **fEdit.Calory.Text:= SG.Cells[3, SG.Row];**
84. **fEdit.HowMany.Text:= SG.Cells[4, SG.Row];**
85. **//устанавливаем ModalResult редактора в mrNone:**
86. **fEdit.ModalResult:= mrNone;**
87. **//теперь выводим форму:**
88. **fEdit.ShowModal;**
89. **//сохраняем в сетку возможные изменения,**
90. **//если пользователь нажал "Сохранить":**
91. **if fEdit.ModalResult = mrOk then begin**
92. **SG.Cells[0, SG.Row]:= fEdit.fName.Text;**
93. **SG.Cells[1, SG.Row]:= fEdit.fPrice.Text;**
94. **SG.Cells[2, SG.Row]:= fEdit.factoryName.text;**
95. **SG.Cells[3, SG.Row]:= fEdit.Calory.Text;**
96. **SG.Cells[4, SG.Row]:= fEdit.HowMany.text;**
97. **end;**
98. **end;**
99. **procedure TfMain.bSortClick(Sender: TObject);**
100. **begin**
101. **//если данных в сетке нет - просто выходим:**
102. **if SG.RowCount = 1 then exit;**
103. **//иначе сортируем список:**
104. **SG.SortColRow(true, 0);**
105. **end;**
106. **procedure TfMain.FormClose(Sender: TObject; var CloseAction: TCloseAction);**
107. **var**
108. **MyCont: Contacts; //для очередной записи**
109. **f: file of Contacts; //файл данных**
110. **i: integer; //счетчик цикла**
111. **begin**
112. **//если строки данных пусты, просто выходим:**
113. **if SG.RowCount = 1 then exit;**
114. **//иначе открываем файл для записи:**
115. **try**
116. **AssignFile(f, adres + 'food.dat');**
117. **Rewrite(f);**
118. **//теперь цикл - от первой до последней записи сетки:**
119. **for i:= 1 to SG.RowCount-1 do begin**
120. **//получаем данные текущей записи:**
121. **MyCont.Name:= SG.Cells[0, i];**
122. **MyCont.Price:= StrToFloat(SG.Cells[1,i]);**
123. **MyCont.Factory:= SG.Cells[2, i];**
124. **MyCont.Calory:= StrToInt(SG.Cells[3,i]);**
125. **MyCont.HowMany:= SG.Cells[4, i];**
126. **//записываем их:**
127. **Write(f, MyCont);**
128. **end;**
129. **finally**
130. **CloseFile(f);**
131. **end;**
132. **end;**
133. **procedure TfMain.FormCreate(Sender: TObject);**
134. **var**
135. **MyCont: Contacts; //для очередной записи**
136. **f: file of Contacts; //файл данных**
137. **i: integer; //счетчик цикла**
138. **begin**
139. **//сначала получим адрес программы:**
140. **adres:= ExtractFilePath(ParamStr(0));**
141. **//настроим сетку:**
142. **SG.Cells[0, 0]:= 'МОДЕЛЬ';**
143. **SG.Cells[1, 0]:= 'ДАТА ВЫПУСКА';**
144. **SG.Cells[2, 0]:= 'ЦВЕТ';**
145. **SG.Cells[3, 0]:= 'ОБЪЕМ ДВИГАТЕЛЯ';**
146. **SG.Cells[4, 0]:= 'МАКСИМАЛЬНАЯ СКОРОСТЬ';**
147. **SG.ColWidths[0]:= 360;**
148. **SG.ColWidths[1]:= 100;**
149. **SG.ColWidths[2]:= 150;**
150. **SG.ColWidths[3]:= 110;**
151. **SG.ColWidths[4]:= 100;**
152. **//если файла данных нет, просто выходим:**
153. **if not FileExists(adres + 'food.dat') then exit;**
154. **//иначе файл есть, открываем его для чтения и**
155. **//считываем данные в сетку:**
156. **try**
157. **AssignFile(f, adres + 'food.dat');**
158. **Reset(f);**
159. **//теперь цикл - от первой до последней записи сетки:**
160. **while not Eof(f) do begin**
161. **//считываем новую запись:**
162. **Read(f, MyCont);**
163. **//добавляем в сетку новую строку, и заполняем её:**
164. **SG.RowCount:= SG.RowCount + 1;**
165. **SG.Cells[0, SG.RowCount-1]:= MyCont.Name;**
166. **SG.Cells[1, SG.RowCount-1]:= FloatToStr(MyCont.Price);**
167. **SG.Cells[2, SG.RowCount-1]:= MyCont.Factory;**
168. **SG.Cells[3, SG.RowCount-1]:= IntToStr(MyCont.Calory);**
169. **SG.Cells[4, SG.RowCount-1]:= MyCont.HowMany;**
170. **end;**
171. **finally**
172. **CloseFile(f);**
173. **end;**
174. **end;**
175. **end.**

**Результат выполнения программа.**

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рисунок 1 результат выполнения программы

Изображение выглядит как текст, снимок экрана, дисплей, число

Автоматически созданное описание

Рисунок 2 добавление новой записи

**Вывод**

В данном отчете предоставлены данные и решения задач, которые были даны в домашней контрольной работе для написания и решения на языке программирования Pascal и Lazarus. Для выполнения были применены знания, которые давались на лекциях и практиках по МДК 05.02 «Разработка кода информационных систем». Так же, помогли базовые и более углубленные знания в математике, в частности, в алгебре.

Во время выполнения данной работы пришлось столкнуться с многими трудностями. Для решения данных проблем пришлось изучить заново несколько презентаций и лекций, изучить информацию в интернете.