

Московский Авиационный Институт  
(Национальный Исследовательский Университет)

Факультет: «Информационных технологий и прикладной математики»

Кафедра: 806 «Вычислительная математика и программирование»

Дисциплина: «Базы данных»

**Курсовой проект**  
**Тема: «Успеваемость студентов»**

Студенты:	Ивенкова Л.В. Королев И.М.
Группа:	М8О-308Б-19
Преподаватель:	Кузнецова Е.В.
Дата:	
Оценка:	

## Оглавление

<b>Техническое задание .....</b>	<b>3</b>
Постановка задачи .....	3
Требования по функциональности .....	3
Требования по интерфейсу .....	3
Инструменты .....	4
Календарный план.....	4
<b>База данных в FoxPro .....</b>	<b>5</b>
<b>Модель базы данных в ErWin .....</b>	<b>6</b>
Логическая модель .....	6
Физическая модель .....	6
<b>Главное меню .....</b>	<b>7</b>
<b>Просмотр студентов и информации о них .....</b>	<b>8</b>
<b>Ведомость по группе .....</b>	<b>9</b>
Печать отчёта ведомости группы .....	11
<b>Поиск задолжников.....</b>	<b>12</b>
Печать отчёта по задолжникам.....	13
<b>Промежуточное меню с выбором изменений в БД.....</b>	<b>14</b>
<b>Добавление и просмотр студентов.....</b>	<b>14</b>
<b>Добавление новой группы вместе со студентами.....</b>	<b>18</b>

## **Техническое задание**

### **Постановка задачи**

Разработать прикладное программное обеспечение для отслеживания успеваемости студентов. Каждый студент имеет имя, фамилию, отчество, дату рождения, зачётную книжку и группу. Каждая группа относится к какой-либо специальности, кафедре и факультету. При проведении какого-либо экзамена каждому студенту группы будет ставиться оценка по заданному предмету. С помощью данного программного обеспечения можно получать ведомости и список должников по заданному предмету, вносить изменения в базу данных. Данное ПО может использоваться, например, деканатом ВУЗа.

### **Требования по функциональности**

Система должна предоставлять следующую функциональность:

1. Обновление базы данных (изменения информации о студентах и группах);
2. Просмотр информации о студентах;
3. Просмотр ведомости для группы по заданному предмету;
4. Поиск задолжников по заданному предмету.

### **Требования по интерфейсу**

Программа должна предоставлять для пользователя возможность лёгкого доступа к данным (на интуитивном уровне). Для этого у программы должен быть развитый графический интерфейс с командами:

1. Добавление новых студентов и новых групп со студентами, просмотр информации о студентах.
2. Печать ведомости по заданной группе с заданным предметом.
3. Печать списка должников по заданному предмету.

У программы должно быть главное меню, из которого пользователь может перейти в нужную ему форму для выполнения определённых действий. Меню будет содержать кнопки:

1. Информация о студентах;
2. Ведомость по группе;
3. Поиск задолжников;
4. Внесение изменений в БД;
5. Выход.

При просмотре информации о студентах должны выводиться все студенты из БД с их персональными данными. Можно переходить к первому и последнему студентам. Также необходимо, чтобы программа могла выводить таблицу со всеми студентами и отчёт по ним.

При поиске ведомости по группе будут задаваться факультет, номер группы, предмет и форма сдачи. В итоге необходимо, чтобы выводилась таблица с данными, а также можно было напечатать отчёт с ней.

При поиске задолжников выбирается предмет. Необходимо, чтобы можно было вывести таблицу с задолжниками, а также преобразовать это в отчёт.

При изменении информации о студентах также можно просматривать информацию о них – это реализовано с помощью списка, в котором удобно одновременно просматривать и быстро выбирать информацию, которую надо изменить. Информацию о студенте можно редактировать, удалить или добавить совершенно нового студента.

При изменении информации о группах надо указать новые данные для группы и ввести данные каждого студента в ней.

### **Инструменты**

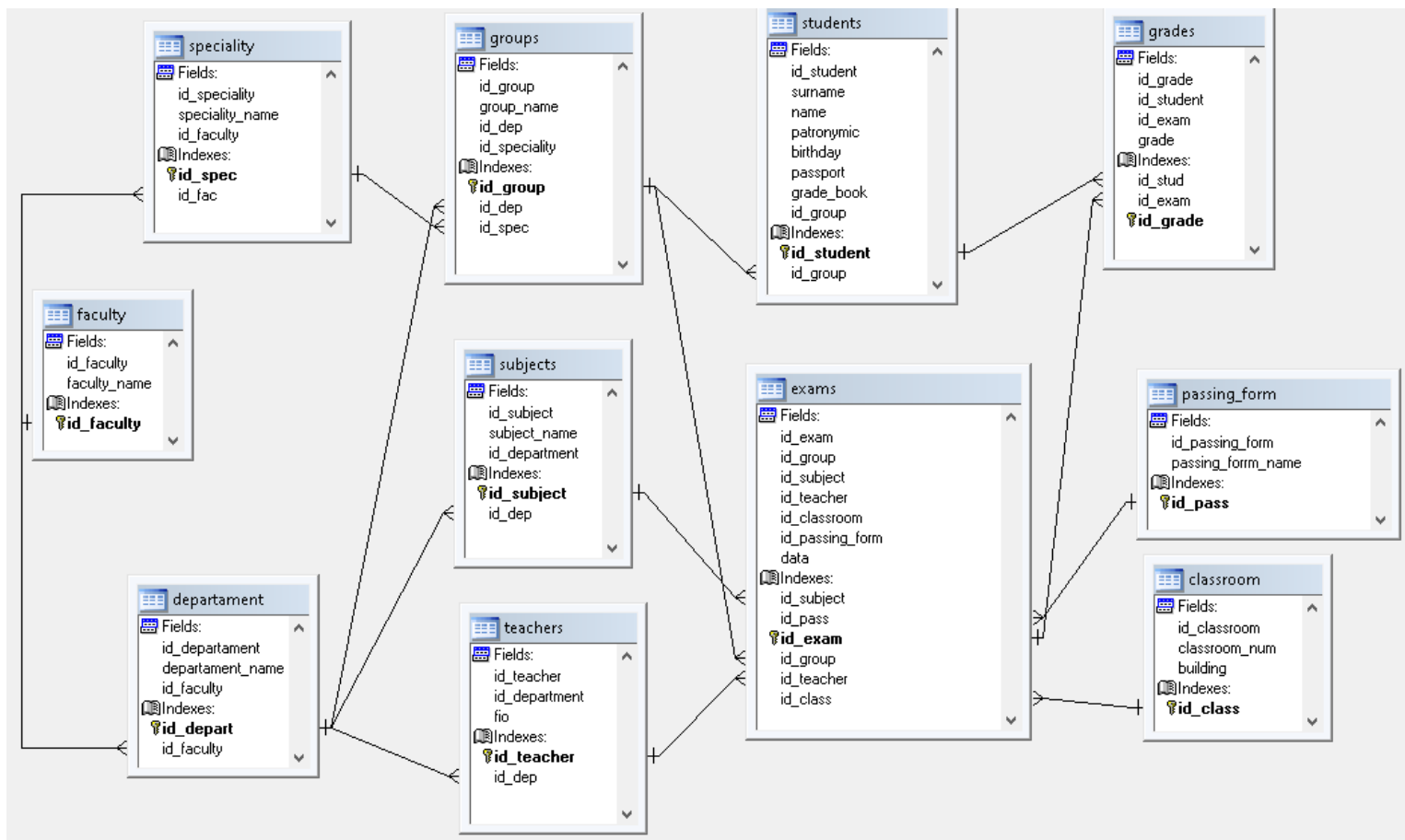
CASE – средство: Erwin Data Modeler

СУБД: Microsoft Visual FoxPro 9.0

### **Календарный план**

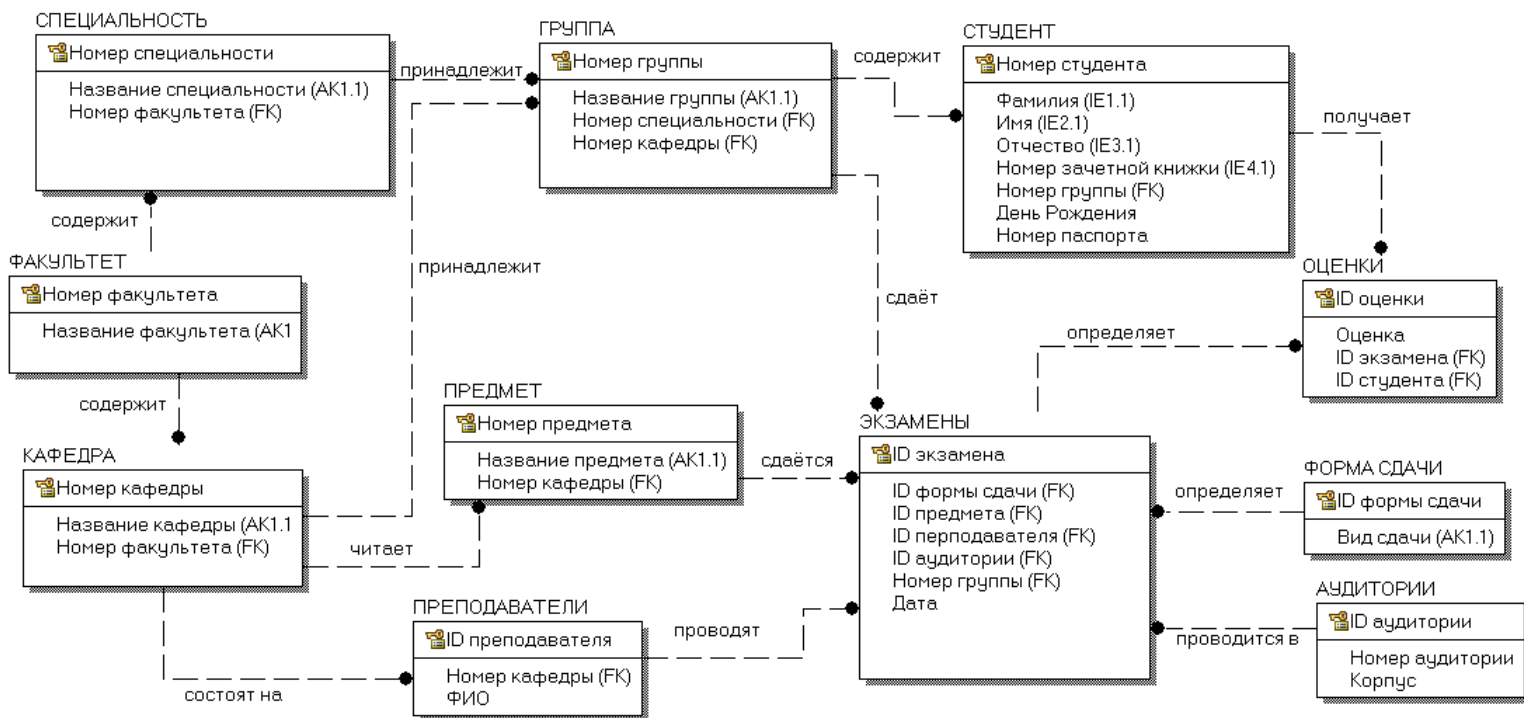
1. Заполнение базы данных - Королев И.М., Ивенкова Л.В.: 23.11.2021
2. Разработка интерфейса - Королев И.М., Ивенкова Л.В.: 24.11.2021
3. Разработка меню поиска ведомости по заданной группе и предмету и её печать; поиска задолжников и печати отчёта по ним - Королев И.М.: 01.12.2021
4. Разработка меню редактирования, обновления и удаления информации о студентах; создание формы для добавления новых групп – Ивенкова Л.В.: 01.12.2021
5. Завершение реализации проекта и создание отчёта - Королев И.М., Ивенкова Л.В.: 03.12.2021

## База данных в FoxPro

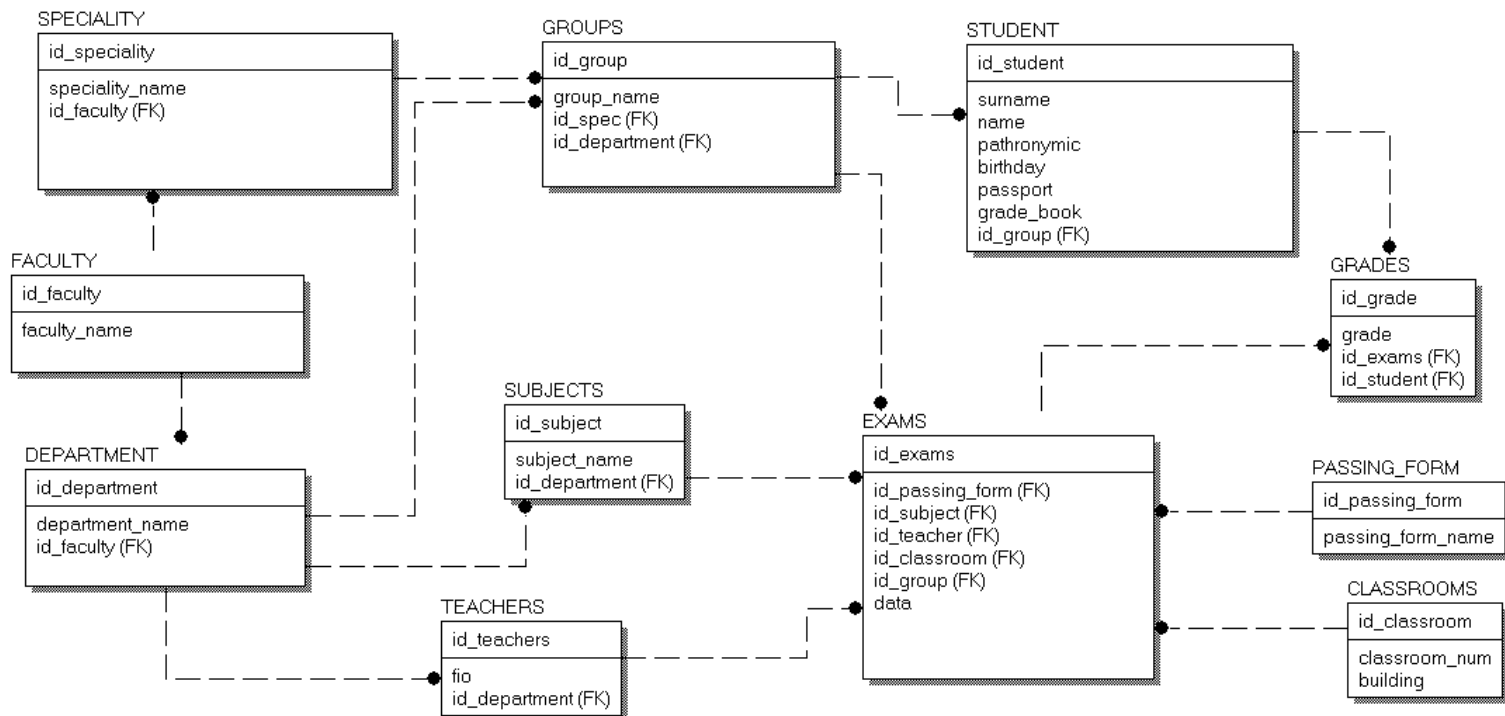


# Модель базы данных в ErWin

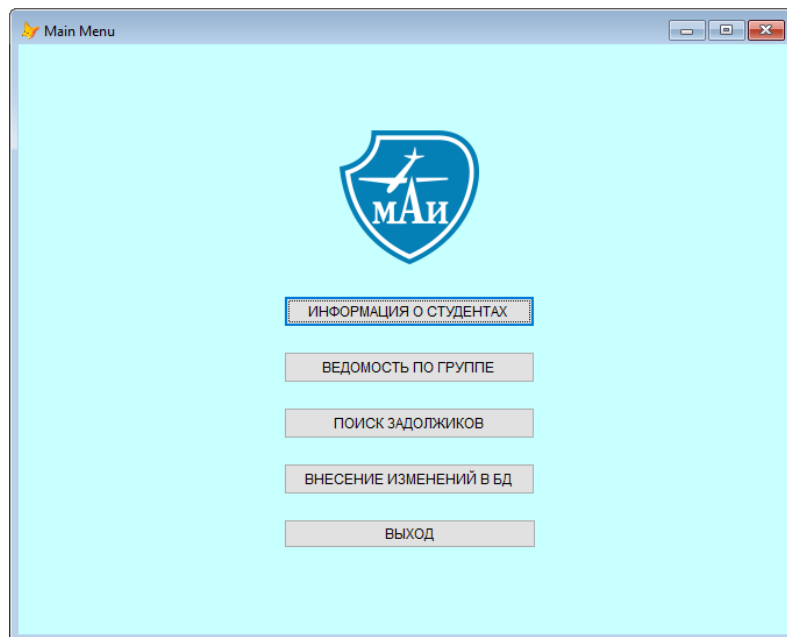
## Логическая модель



## Физическая модель



## Главное меню



Кнопка «Информация о студентах»:

\* Переходим в форму информации о студентах

```
DO FORM "C:\DataBases\StudentsPerformance\forms\studentinfo_form"  
RELEASE THISFORM
```

Кнопка «Ведомость по группе»:

\* Переходим в форму вывода ведомости по группе

```
DO FORM "C:\DataBases\StudentsPerformance\forms\group_list"  
RELEASE THISFORM
```

Кнопка «Поиск задолжников»:

\* Переходим в форму информации о задолжниках среди студентов

```
DO FORM "C:\DataBases\StudentsPerformance\forms\debtors_search"  
RELEASE THISFORM
```

Кнопка «Внесение изменений в БД»:

\* Переходим в форму внесения изменений в БД

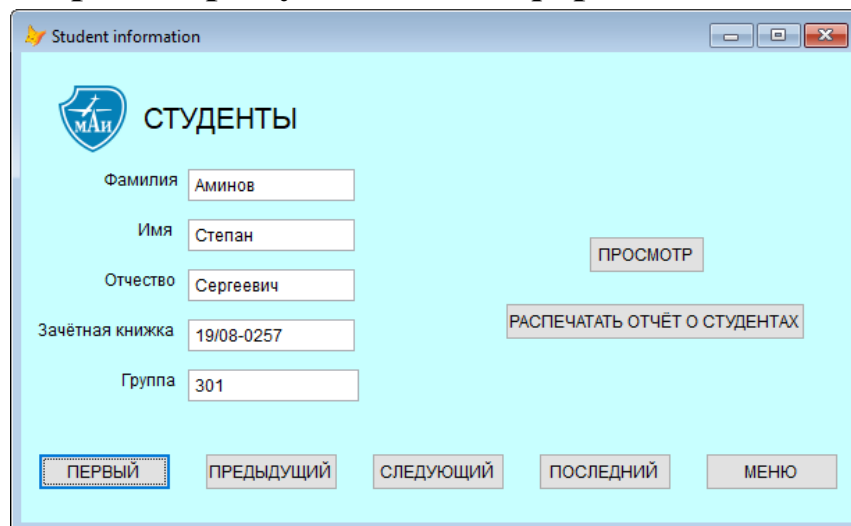
```
DO FORM "C:\DataBases\StudentsPerformance\forms\db_changes"  
RELEASE THISFORM
```

Кнопка «Выход»:

\* Запрашиваем и выходим, если Да

```
IF MESSAGEBOX("Выйти из формы?", 4+32+256, "Выход")=6  
ThisForm.Release()  
ELSE _  
ThisForm.Refresh()  
ENDIF
```

## Просмотр студентов и информации о них



Кнопка «Первый»:

\* Переходим на первую запись и обновляем информацию в форме

```
IF !BOF()  
GO TOP  
ENDIF _  
ThisForm.Refresh()
```

Кнопка «Предыдущий»:

\* Переходим на предыдущую запись и обновляем информацию в форме

```
IF !BOF()  
SKIP - 1  
ENDIF _  
ThisForm.Refresh()
```

Кнопка «Следующий»:

\* Переходим на следующую запись и обновляем информацию в форме

```
IF !EOF()  
SKIP  
ENDIF _  
ThisForm.Refresh()
```

Кнопка «Последний»:

\* Переходим на последнюю запись и обновляем информацию в форме

```
IF !EOF()  
GO BOTTOM  
ENDIF _  
ThisForm.Refresh()
```

Кнопка «Просмотр»:

Browse

Кнопка «Распечатать отчёт о студентах»:

REPORT FORM "C:\DataBases\StudentsPerformance\reports\studentinfo\_report.frx" PREVIEW

Кнопка «Меню»:

```
DO FORM C:\DataBases\StudentsPerformance\forms\main_menu  
RELEASE THISFORM
```



## Ведомость по группе

The screenshot shows a web application window titled "Group list". Inside the window, there is a header with a logo and the text "ВЕДОМОСТЬ ПО ГРУППЕ". Below the header, there are four dropdown menus for selection: "ФАКУЛЬТЕТ" (Faculty) with the value "Информационные технологии и прикладная математика", "НОМЕР ГРУППЫ" (Group Number) with the value "308", "ПРЕДМЕТ" (Subject) with the value "Комплексный анализ", and "ФОРМА СДАЧИ" (Exam Form) with the value "Экзамен". Below these menus are three buttons: "ПОКАЗАТЬ ВЕДОМОСТЬ" (Show Report), "РАСПЕЧАТАТЬ ВЕДОМОСТЬ" (Print Report), and "МЕНЮ" (Menu).

Кнопка «Показать ведомость»:

```
IF faculty_selected_flag = 0
    MESSAGEBOX("Необходимо выбрать факультет!", 0, "Предупреждение")
    RETURN
ENDIF
```

```
IF group_selected_flag = 0
    MESSAGEBOX("Необходимо выбрать группу!", 0, "Предупреждение")
    RETURN
ENDIF
```

```
IF subject_selected_flag = 0
    MESSAGEBOX("Необходимо выбрать предмет!", 0, "Предупреждение")
    RETURN
ENDIF
```

```
IF pass_flag = 0
    MESSAGEBOX("Необходимо выбрать форму сдачи!", 0, "Предупреждение")
    RETURN
ENDIF
```

```
SELECT passing_form.id_passing_form;
FROM passing_form;
WHERE passing_form.passing_form_name = pass_value;
INTO CURSOR _pass
```

```
SELECT subjects.id_subject, subjects.subject_name;
FROM subjects;
WHERE subjects.subject_name = subject_value;
INTO CURSOR _subject
```

```
SELECT exams.id_exam, _subject.subject_name;
FROM exams, _subject;
WHERE exams.id_subject = _subject.id_subject AND exams.id_passing_form = _pass.id_passing_form;
INTO CURSOR _exam
```

```

SELECT grades.id_student, grades.grade, _exam.subject_name;
FROM grades, _exam;
WHERE grades.id_exam = _exam.id_exam;
INTO CURSOR _grades

```

```

SELECT students.surname as Фамилия, students.name as Имя, students.patronymic as Отчество, faculty_value as
Факультет, students.id_group as Группа, _grades.subject_name as Предмет, _grades.grade as Оценка, pass_value as
Форма сдачи;
FROM students, _grades;
WHERE _grades.id_student = students.id_student AND STR(students.id_group) = group_value

```

**Кнопка «Распечатать ведомость»:**

```

IF faculty_selected_flag = 0
    MESSAGEBOX("Необходимо выбрать факультет!", 0, "Предупреждение")
    RETURN
ENDIF

```

```

IF group_selected_flag = 0
    MESSAGEBOX("Необходимо выбрать группу!", 0, "Предупреждение")
    RETURN
ENDIF

```

```

IF subject_selected_flag = 0
    MESSAGEBOX("Необходимо выбрать предмет!", 0, "Предупреждение")
    RETURN
ENDIF

```

```

IF pass_flag = 0
    MESSAGEBOX("Необходимо выбрать форму сдачи!", 0, "Предупреждение")
    RETURN
ENDIF

```

```

SELECT passing_form.id_passing_form;
FROM passing_form;
WHERE passing_form.passing_form_name = pass_value;
INTO CURSOR _pass

```

```

SELECT subjects.id_subject, subjects.subject_name;
FROM subjects;
WHERE subjects.subject_name = subject_value;
INTO CURSOR _subject

```

```

SELECT exams.id_exam, _subject.subject_name;
FROM exams, _subject;
WHERE exams.id_subject = _subject.id_subject AND exams.id_passing_form = _pass.id_passing_form;
INTO CURSOR _exam

```

```

SELECT grades.id_student, grades.grade, _exam.subject_name;
FROM grades, _exam;
WHERE grades.id_exam = _exam.id_exam;
INTO CURSOR _grades

```

```

SELECT students.surname as Фамилия, students.name as Имя, students.patronymic as Отчество, faculty_value as
Факультет, students.id_group as Группа, _grades.subject_name as Предмет, _grades.grade as Оценка, pass_value as
Форма сдачи;
FROM students, _grades;
WHERE _grades.id_student = students.id_student AND STR(students.id_group) = group_value;
INTO TABLE group_list_report

```

```

SELECT Оценка, COUNT(Оценка) as Количество;

```

```

FROM group_list_report;
GROUP BY Оценка;
INTO CURSOR _group_list_report_statistics

SELECT _group_list_report_statistics.Количество FROM _group_list_report_statistics;
WHERE (Оценка = (2) OR Оценка = (0));
INTO CURSOR _grade_false

SELECT _group_list_report_statistics.Количество FROM _group_list_report_statistics;
WHERE Оценка = (3);
INTO CURSOR _grade_three


SELECT _group_list_report_statistics.Количество FROM _group_list_report_statistics;
WHERE Оценка = (4);
INTO CURSOR _grade_four

SELECT _group_list_report_statistics.Количество FROM _group_list_report_statistics;
WHERE (Оценка = (5) OR Оценка = (1));
INTO CURSOR _grade_five

REPORT FORM "C:\DataBases\StudentsPerformance\reports\group_list_report.frx" PREVIEW

```

## Печать отчёта ведомости группы



ВЕДОМОСТЬ ГРУППЫ

12/03/21

Факультет

Информационные технологии и прикладная математика

Предмет

Комплексный анализ

Форма сдачи

Экзамен

Группа

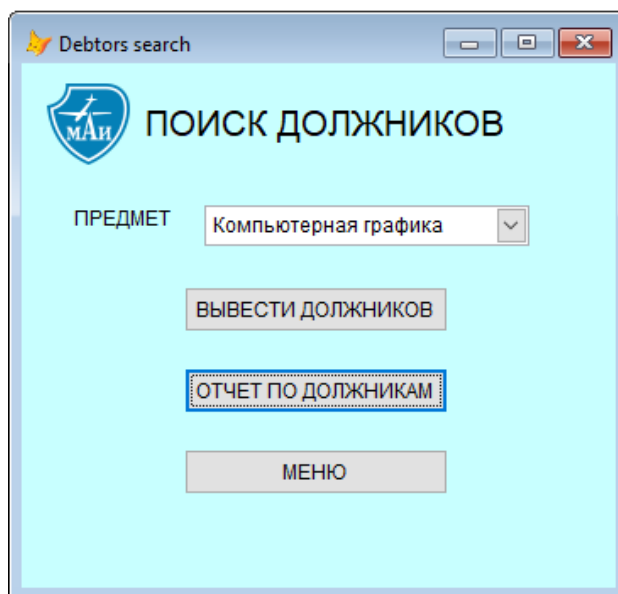
308

Фамилия	Имя	Отчество	Оценка
Ивенкова	Любовь	Васильевна	5
Королев	Игорь	Михайлович	4

Статистика оценок группы

Оценка	2	3	4	5
Количество	0	0	1	1

## Поиск задолжников



Кнопка «Вывести должников»:

```
IF subject_flag = 0
    MESSAGEBOX("Необходимо выбрать предмет!", 0, "Предупреждение")
    RETURN
ENDIF
```

```
SELECT id_subject, subject_name;
FROM subjects;
WHERE subject_name = subject_value;
INTO CURSOR _subject
```

```
SELECT exams.id_exam, _subject.subject_name, passing_form.passing_form_name;
FROM exams, passing_form, _subject;
WHERE exams.id_subject = _subject.id_subject AND exams.id_passing_form = passing_form.id_passing_form;
INTO CURSOR _exam
```

```
SELECT grades.id_student, _exam.passing_form_name, _exam.subject_name;
FROM grades, _exam;
WHERE grades.id_exam = _exam.id_exam AND (grades.grade = (0) OR grades.grade = (2));
INTO CURSOR _grades
```

```
SELECT students.id_group as Группа, students.surname as Фамилия, students.name as Имя, students.patronymic as
Отчество, _grades.subject_name as Предмет, _grades.passing_form_name as Форма_сдачи;
FROM students, _grades;
WHERE _grades.id_student = students.id_student
```

Кнопка «Отчёт по должникам»:

```
IF subject_flag = 0
    MESSAGEBOX("Необходимо выбрать предмет!", 0, "Предупреждение")
    RETURN
ENDIF
```

```
SELECT id_subject, subject_name;
FROM subjects;
WHERE subject_name = subject_value;
INTO CURSOR _subject
```

```
SELECT exams.id_exam, _subject.subject_name, passing_form.passing_form_name;
```


```
FROM exams, passing_form, _subject;
WHERE exams.id_subject = _subject.id_subject AND exams.id_passing_form = passing_form.id_passing_form;
INTO CURSOR _exam
```

```
SELECT grades.id_student, _exam.passing_form_name, _exam.subject_name;
FROM grades, _exam;
WHERE grades.id_exam = _exam.id_exam AND (grades.grade = (0) OR grades.grade = (2));
INTO CURSOR _grades
```

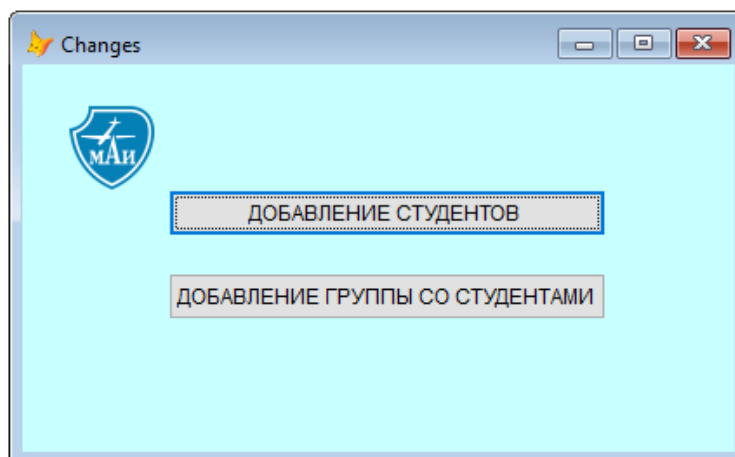
```
SELECT students.id_group as Группа, students.surname as Фамилия, students.name as Имя, students.patronymic as
Отчество, _grades.subject_name as Предмет, _grades.passing_form_name as Форма_сдачи;
FROM students, _grades;
WHERE _grades.id_student = students.id_student;
INTO TABLE debtors
```

REPORT FORM "C:\DataBases\StudentsPerformance\reports\debtors\_report.frx" [PREVIEW](#)

## Печать отчёта по задолжникам

<div>  <div> <b>ЗАДОЛЖНИКИ</b> Предмет Компьютерная графика </div> <div>12/03/21</div> </div>				
Группа	Фамилия	Имя	Отчество	Форма Сдачи
<b>301</b>				
	Аминов	Степан	Сергеевич	Рейтинг
<b>308</b>				
	Шубин	Григорий	Сергеевич	Курсовая работ а

## Промежуточное меню с выбором изменений в БД



Кнопка «Добавление студентов»:

DO FORM "C:\DataBases\StudentsPerformance\forms\students\_editing"

RELEASE ThisForm

Кнопка «Добавление группы со студентами»:

DO FORM "C:\DataBases\StudentsPerformance\forms\new\_group"

RELEASE ThisForm

## Добавление и просмотр студентов

A screenshot of a Windows application window titled 'Form1'. It contains a table with student data. The table has columns for name, date, and ID. The data is as follows:

Имя	Фамилия	Патроним	Дата	Идентификатор	Номер	
Каплина	Анастасия	Валерьевна	04/05/02	7895 454585	20/08-0211	302
Арапов	Степан	Александров	11/12/03	7862 302355	21/08-0250	109
Артамонов	Олег	Алексеевич	10/11/01	4589 454545	19/08-0256	306
Горохов	Михаил	Антонович	09/10/01	4589 652031	19/08-0255	4
Гапулхаков	Руслан	Рахматович	08/09/01	1234 123456	19/08-0244	7
Двоеглазова	Анна	Владимировна	07/08/03	5689 552200	21/08-0232	304
Иванов	Федор	Алексеевич	06/07/01	1256 896631	19/08-0233	5
Иванова	Любовь	Васильевна	04/06/01	7845 120032	19/08-0289	308
Каплина	Анастасия	Валерьевна	04/05/02	7895 454585	20/08-0211	3
Короткевич	Леонид	Витальевич	03/04/02	4595 123567	20/08-0157	211
Королев	Игорь	Михайлович	02/03/01	4578 125845	19/08-0157	308
Шубин	Григорий	Сергеевич	01/02/01	4523 789545	19/08-0057	308
М	М	М	??			308
М	М	М	??			308
К	М	М	??			308
К	М	М	??			308
К	М	М	??			308
Г	Хех	Мем	??	6732	25412	0
К	И	М	??	121	312	301
К	И	М	??	121	312	308
Ж	И	Э	??	321	423432	301
Ж	И	Э	??	321	423432	303

Кнопка «Отмена»:

DO FORM "C:\DataBases\StudentsPerformance\forms\db\_changes"

ThisForm.Release

Кнопка «Редактировать»:

```
ThisForm.Pageframe1.Page2.Command3.Visible = .T.
ThisForm.Pageframe1.Page2.Command3.Enabled = .T.
ThisForm.Pageframe1.Page2.Command2.Enabled = .F.
ThisForm.Pageframe1.Page2.Command1.Enabled = .F.
```

```
ThisForm.Pageframe1.Page2.txtsurname.Enabled = .T.
ThisForm.Pageframe1.Page2.txtname.Enabled = .T.
ThisForm.Pageframe1.Page2.txtpatronymic.Enabled = .T.
ThisForm.Pageframe1.Page2.txtBirthday.Enabled = .T.
ThisForm.Pageframe1.Page2.txtpassport.Enabled = .T.
ThisForm.Pageframe1.Page2.txtgrade_book.Enabled = .T.
ThisForm.Pageframe1.Page2.txtid_group.Enabled = .F.
ThisForm.Pageframe1.Page2.Combo1.Enabled = .T.
```

Кнопка «Сохранить» (становится видимой и доступной при нажатии кнопки «Редактировать»):

```
IF EMPTY(ThisForm.Pageframe1.Page2.txtsurname.Value) = .T.
  =MESSAGEBOX('Вы забыли ввести фамилию студента!',48,'Предупреждение!')
  ThisForm.Pageframe1.Page2.txtsurname.SetFocus
  RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.Page2.txtname.Value) = .T.
  =MESSAGEBOX('Вы забыли ввести имя студента!',48,'Предупреждение!')
  ThisForm.Pageframe1.Page2.txtname.SetFocus
  RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.Page2.txtBirthday.Value) = .T.
  =MESSAGEBOX('Вы забыли ввести дату рождения студента!',48,'Предупреждение!')
  ThisForm.Pageframe1.Page2.txtBirthday.SetFocus
  RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.Page2.txtpassport.Value) = .T.
```

```

=MESSAGEBOX('Вы забыли ввести номер паспорта студента!',48,'Предупреждение!')
ThisForm.Pageframe1.Page2.txtpassport.SetFocus
RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.Page2.txtgrade_book.Value) = .T.
=MESSAGEBOX('Вы забыли ввести номер зачётной книжки!',48,'Предупреждение!')
ThisForm.Pageframe1.Page2.txtgrade_book.SetFocus
RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.Page2.Combo1.Value) = .T.
=MESSAGEBOX('Вы забыли ввести номер группы!',48,'Предупреждение!')
ThisForm.Pageframe1.Page2.Combo1.SetFocus
RETURN
ENDIF

surn = ThisForm.Pageframe1.Page2.txtsurname.Value
n = ThisForm.Pageframe1.Page2.txtname.Value
patr = ThisForm.Pageframe1.Page2.txtpatronymic.Value
birthd = CAST (ThisForm.Pageframe1.Page2.txtBirthday.Value AS Date)
pas = ThisForm.Pageframe1.Page2.txtpassport.Value
grb = ThisForm.Pageframe1.Page2.txtgrade_book.Value
gr_name = ThisForm.Pageframe1.Page2.Combo1.Value

UPDATE students SET surname=surn, name=n, patronymic=patr,;
birthday=birthd, passport=pas, grade_book=grb,;
id_group=(SELECT id_group FROM groups WHERE group_name=gr_name);
WHERE id_student = ids

=MESSAGEBOX('Информация изменена!',48,'SUCCESS!')

ThisForm.Pageframe1.page2.Command1.Enabled = .T.
ThisForm.Pageframe1.page2.Command2.Enabled = .T.
GOTO TOP
ThisForm.pageframe1.ActivePage = 1
ThisForm.PageFrame1.Page1.Refresh
ThisForm.Pageframe1.Page1.List1.SetFocus
ThisForm.Pageframe1.Page1.List1.Refresh

```

**Кнопка «Удалить»:**

```

SELECT students
DELETE
GOTO TOP

=MESSAGEBOX('Студент удалён!',48,'SUCCESS!')

ThisForm.pageframe1.ActivePage = 1
SET DELETED ON
ThisForm.PageFrame1.Page1.Refresh
ThisForm.Pageframe1.Page1.List1.SetFocus
ThisForm.Pageframe1.Page1.List1.Refresh

```



Кнопка «Вставить»:

```

IF EMPTY(ThisForm.Pageframe1.Page3.Text1.Value) = .T.
=MESSAGEBOX('Вы забыли ввести фамилию студента!',48,'Предупреждение!')
ThisForm.Pageframe1.Page3.Text1.SetFocus
RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.Page3.Text2.Value) = .T.
=MESSAGEBOX('Вы забыли ввести имя студента!',48,'Предупреждение!')
ThisForm.Pageframe1.Page3.Text2.SetFocus
RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.Page3.Text4.Value) = .T.
=MESSAGEBOX('Вы забыли ввести дату рождения студента!',48,'Предупреждение!')
ThisForm.Pageframe1.Page3.Text4.SetFocus
RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.Page3.Text5.Value) = .T.
=MESSAGEBOX('Вы забыли ввести номер паспорта студента!',48,'Предупреждение!')
ThisForm.Pageframe1.Page3.Text5.SetFocus
RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.Page3.Text6.Value) = .T.
=MESSAGEBOX('Вы забыли ввести номер зачётной книжки!',48,'Предупреждение!')
ThisForm.Pageframe1.Page3.Text6.SetFocus
RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.Page3.Combo2.Value) = .T.
=MESSAGEBOX('Вы забыли ввести номер группы!',48,'Предупреждение!')
ThisForm.Pageframe1.Page3.Combo2.SetFocus
RETURN
ENDIF

```

```

sum = ThisForm.Pageframe1.Page3.Text1.Value
n = ThisForm.Pageframe1.Page3.Text2.Value
patr = ThisForm.Pageframe1.Page3.Text3.Value

```

```

birthd = CAST (ThisForm.Pageframe1.Page3.Text4.Value AS Date)
pas = ThisForm.Pageframe1.Page3.Text5.Value
grb = ThisForm.Pageframe1.Page3.Text6.Value
gr_name = ThisForm.Pageframe1.Page3.Combo2.Value

```

```

INSERT INTO students (surname, name, patronymic, birthday, passport, grade_book, id_group);
SELECT surn, n, patr, birthd, pas, grb, id_group;
FROM groups WHERE groups.group_name = gr_name

```

```

=MESSAGEBOX('Студент добавлен!',48,'SUCCESS!')

```

```

ThisForm.Pageframe1.Page3.Text1.Value= ""
ThisForm.Pageframe1.Page3.Text2.Value= ""
ThisForm.Pageframe1.Page3.Text3.Value= ""
ThisForm.Pageframe1.Page3.Text4.Value= ""
ThisForm.Pageframe1.Page3.Text5.Value= ""
ThisForm.Pageframe1.Page3.Text6.Value= ""
ThisForm.Pageframe1.Page3.Combo2.Value= ""

```

GOTO TOP

```

ThisForm.pageframe1.ActivePage = 1
ThisForm.PageFrame1.Page1.Refresh
ThisForm.Pageframe1.Page1.List1.SetFocus
ThisForm.Pageframe1.Page1.List1.Refresh

```

## Добавление новой группы вместе со студентами

Кнопка «Вставить студентов» (вставка группы):

```

gr_name = ThisForm.Pageframe1.Page1.Text1.Value
gr_dep = CAST (ThisForm.Pageframe1.Page1.Combo2.Value AS Integer)
gr_sp = ThisForm.Pageframe1.Page1.Combo1.Value
PUBLIC k

```

```

SELECT COUNT(id_group) AS c;
FROM groups INTO CURSOR tmp;
WHERE groups.group_name = gr_name

IF tmp.c <> (0)
=MESSAGEBOX("Такая группа уже существует!", 48, "Предупреждение!")
ThisForm.Pageframe1.Page1.Text1.SetFocus
RETURN
ENDIF

INSERT INTO groups (group_name, id_dep, id_speciality);
SELECT gr_name, gr_dep, id_speciality;
FROM speciality;
WHERE speciality_name = gr_sp

=MESSAGEBOX("Группа добавлена!", 48, "SUCCESS!")

ThisForm.Pageframe1.Page1.Text1.Value = ""
ThisForm.Pageframe1.Page1.Combo2.Value = ""
ThisForm.Pageframe1.Page1.Combo1.Value = ""

ThisForm.Pageframe1.Page1.Combo1.Refresh
ThisForm.PageFrame1.page2.Enabled=.T.
ThisForm.pageframe1.ActivePage = 2

```

Кнопка «Сохранить»:

```

IF EMPTY(ThisForm.Pageframe1.page2.txtSurname.Value) = .T.
=MESSAGEBOX("Вы забыли ввести фамилию студента!", 48, "Предупреждение!")
ThisForm.PageFrame1.Page2.TxtSurname.SetFocus
RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.page2.txtName.Value) = .T.

```

```

=MESSAGEBOX('Вы забыли ввести имя студента!',48,'Предупреждение!')
ThisForm.PageFrame1.Page2.TxtName.SetFocus
RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.page2.txtbirthday.Value) = .T.
=MESSAGEBOX('Вы забыли ввести дату рождения студента!',48,'Предупреждение!')
ThisForm.PageFrame1.Page2.txtbirthday.SetFocus
RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.page2.txtpassport.Value) = .T.
=MESSAGEBOX('Вы забыли ввести номер паспорта студента!',48,'Предупреждение!')
ThisForm.PageFrame1.Page2.txtpassport.SetFocus
RETURN
ENDIF
IF EMPTY(ThisForm.Pageframe1.page2.txtgrade_book.Value) = .T.
=MESSAGEBOX('Вы забыли ввести номер зачётной книжки!',48,'Предупреждение!')
ThisForm.PageFrame1.Page2.txtgrade_book.SetFocus
RETURN
ENDIF

sum = ThisForm.Pageframe1.Page2.txtsurname.Value
n = ThisForm.Pageframe1.Page2.txtname.Value
patr = ThisForm.Pageframe1.Page2.txtpatronymic.Value
birthd = CAST (ThisForm.Pageframe1.Page2.txtbirthday.Value AS Date)
pas = ThisForm.Pageframe1.Page2.txtpassport.Value
grb = ThisForm.Pageframe1.Page2.txtgrade_book.Value

INSERT INTO students (surname, name, patronymic, birthday, passport, grade_book, id_group);
SELECT sum, n, patr, birthd, pas, grb, id_group;
FROM groups;
WHERE groups.group_name = gr_name

=MESSAGEBOX('Студент добавлен!',48,'SUCCESS!')

ThisForm.Pageframe1.Page2.txtsurname.Value= ""
ThisForm.Pageframe1.Page2.txtname.Value= ""
ThisForm.Pageframe1.Page2.txtpatronymic.Value= ""
ThisForm.Pageframe1.Page2.txtbirthday.Value= ""
ThisForm.Pageframe1.Page2.txtpassport.Value= ""
ThisForm.Pageframe1.Page2.txtgrade_book.Value= ""

```

Кнопка «Добавить ещё одну группу»:

```

ThisForm.PageFrame1.page1.Enabled=.T.
ThisForm.PageFrame1.Page1.Refresh
ThisForm.pageframe1.ActivePage = 1

```

Кнопка «Завершить»:

```

DO FORM " C:\DataBases\StudentsPerformance\forms\main_menu"
RELEASE THISFORM

```