# ПРИМЕР ВЫПОЛНЕНИЯ РАБОТЫ

Рассмотрим пример проектирования базы данных в среде СУБД MS Access.

1 Описание предметной области.

В данном проекте выполняется разработка реляционной базы данных для информационной системы «Учет успеваемости школьников».

1.1 Постановка задачи.

Главная задача системы – сохранение в базе данных всех необходимых сведений о школьниках и сведений их успеваемости по итогам учебного года, формирование необходимых печатных форм для отображения и ввода данных, генерация сводных итогов по успеваемости за учебный год для учителей. Также предусмотрена возможность ознакомления школьников со своими оценками, без права редактирования записей. В системе должен использоваться единый справочник предметов для классов всех параллелей. Информационная система предназначена для двух категорий пользователей: учителя и ученики. Учителя могут просматривать информацию об учениках, добавлять новые записи, а также удалять старые записи. Ученики могут получить сведения о своей успеваемости.

2 Концептуальная модель базы данных

При разработке ER-моделей мы должны получить следующую информацию о предметной области:

˗ список сущностей предметной области;

˗ список атрибутов сущностей;

˗ описание взаимосвязей между сущностями.

После анализа предметной области мы выделили три сущности: «Ученик», «Оценка», «Предмет». Следующим шагом определим атрибуты всех сущностей и выделим ключевые атрибуты (рис. 1).

|  |  |  |
| --- | --- | --- |
| Ученик | Оценка | Предмет |
| Номер | Номер | Номер |
| Фамилия  Имя  Отчество  Класс  Возраст  Увлечения | Оценка | Предмет |
|  |  |

Рис. 1. Список абитуриентов

Домены из которых атрибуты берут свои значения, приведены в таблице. Здесь же приведены ограничения для атрибутов на уровне кортежей: повторяемость, обязательность и значения по умолчанию.

Таблица 1. Список атрибутов с ограничениями

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Ограничение | Повторя  е  мость | Обязател  ь  ность | Значение по  умолч  а  нию |
| НомерУченика |  | - | + | нет |
| Фамилия |  | + | + | нет |
| Имя |  | + | + | нет |
| Класс | 1. | + | + | нет |
| Возраст | 2. | + | + | нет |
| Увлечения | 4. | + | - | нет |
| НомерПредмета |  | - | + | нет |
| Оценка | 3. | + | + | нет |
| Предмет |  | + | + | нет |

Примечания:

1. Класс записывается в таком формате: номер класса, буква (без пробелов между ними)
2. 12<=Возраст<=16
3. Целое число от 2 до 5
4. Записываются через запятую в произвольном порядке

Определим типы связей и построим начальную ER-модель данных

(рис. 2).

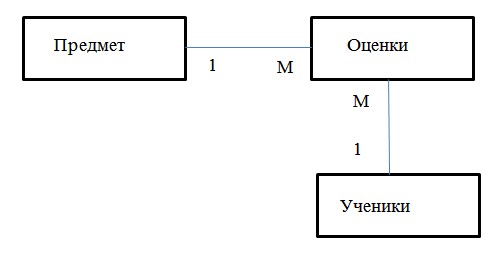


Рис. 2. Начальная ER - модель данных

Преобразование концептуальной модели в концептуальную схему выбранной реляционной СУБД осуществляется в следующей последовательности.

* 1. Для каждой сильной сущности ER-модели создается отдельная таблица, а для каждого атрибута сущности создается столбец таблицы. Ключевой атрибут становится первичным ключом, а дополнительные ключевые атрибуты - потенциальными ключами.
  2. Для каждой слабой сущности также создается отдельная таблица, в которой должны присутствовать ключевые столбцы доминирующих таблиц. В зависимости от вида связи устанавливаются ключевые атрибуты таблицы.
  3. Далее необходимо создать внешние ключи, обеспечивающие ссылочную целостность, по указанному типу связи в ER-модели.

Вполне возможно, что в ER-схеме будет присутствовать избыточность данных, поэтому необходимо нормализировать базу данных, как минимум, до нормальной формы Бойса-Кодда (рис. 3).

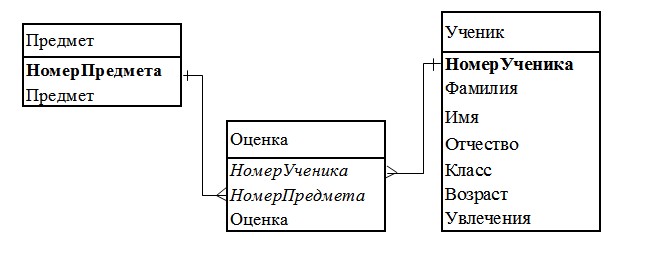


Рис. 3. Концептуальная ER-диаграмма

1. В физической модели каждой сущности будет соответствовать таблица базы данных, а каждому атрибуту – поле таблицы.

Таблица 2. Ученик

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| **НомерУченика** | Счетчик |
| Фамилия | Текстовый |
| Имя | Текстовый |
| Класс | Текстовый |
| Возраст | Числовой |
| Увлечения | Memo |

Таблица 3. Оценка

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| НомерУченика | Числовой |
| НомерПредмета | Числовой |
| Оценка | Числовой |

Таблица 4. Предмет

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| **НомерПредмета** | Счетчик |
| Предмет | Текстовый |

Физическая ER-диаграмма представлена в приложении 1.

1. Разработка базы данных

*4.1. Создание таблиц*

На основании реляционной модели данных необходимо создать 3 таблицы «Оценка», «Ученик», «Предмет». Создадим таблицы в режиме конструктора, для этого на вкладке ленты «Создание» выберем пункт «Конструктор таблиц». Каждой таблице назначим ключевое поле (рис. 4).

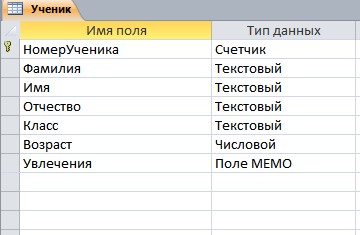


Рис. 4. Таблица «Ученик» в режиме конструктора

В поле «НомерУченика» таблицы «Оценка» выберем тип данных

«Мастер постановок». В появившемся окне выберем команду «Объект «поле подстановки» получит значение из другой таблицы или запроса». В появившемся окне выберем таблицу «Ученик». В списке «Доступные поля» выберем поле «НомерУченика», «Фамилия», «Имя». Порядок сортировки выберем сначала по полю «Фамилия», затем по полю «Имя» по возрастанию, в следующем диалоговом окне установим флажок «Скрывать ключевое поле». Аналогичным образом создадим раскрывающийся список для поля «КодПредмета» в таблице «Оценка», источником данных будет таблица «Предмет».

После создания ключевых полей можно приступить к созданию связей. Создадим связи между таблицами, выполнив команду: вкладка ленты Работа с базами данных – Схема данных. В диалоговом окне схемы связи включим флажов *Обеспечение условия целостности,* выбрав пункты *каскадное обновление связанных полей* и *каскадное удаление связанных полей* (рис. 5)*.*

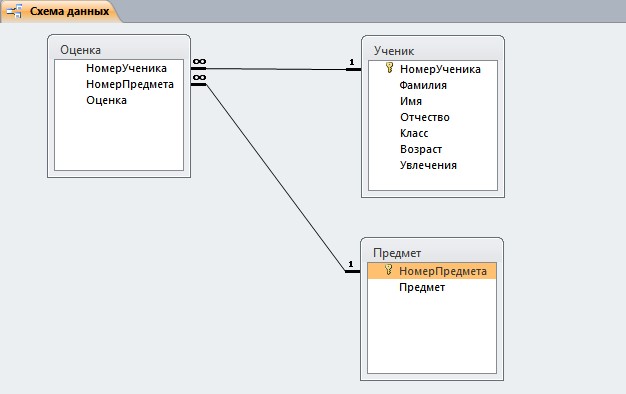


Рис. 5. Схема данных

Заполним таблицы данными. Снимки экранов таблиц представлены в приложении 2.

*4.2. Создание запросов*

Создадим запрос на выборку, используя мастер запросов.

Вкладка ленты Создание – Мастер запросов – Простой запрос

Из списка выберем с помощью SELECT выберем таблицу «Должностной оклад». Далее с помощью FROM выбираем Поля штатные единицы, Должностной оклад. Результаты запросов представлены в приложении 2.

Следующий запрос создадим с помощью конструктора запросов, для этого выполним команду: вкладка ленты Создание – Конструктор запросов. Создадим запрос, показывающий список учеников 8а класса (рис. 6). Результат запроса представлен в приложении 2.

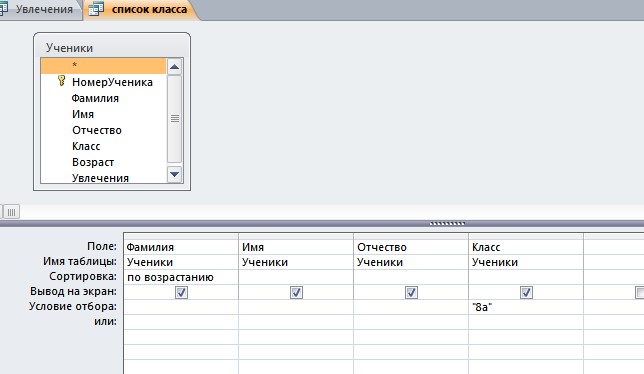


Рис. 6. Запрос на выборку по 8а классу

Следующий запрос создадим с параметром. При запуске такого запроса на экран выводится диалоговое окно для ввода значения в качестве условия отбора. Чтобы создать запрос с параметром необходимо ввести следующий текст сообщения в строке Условие отбора: Like[Текст сообщения].

Создадим запрос, показывающий информацию по ученику при вводе его фамилии (рис. 7).

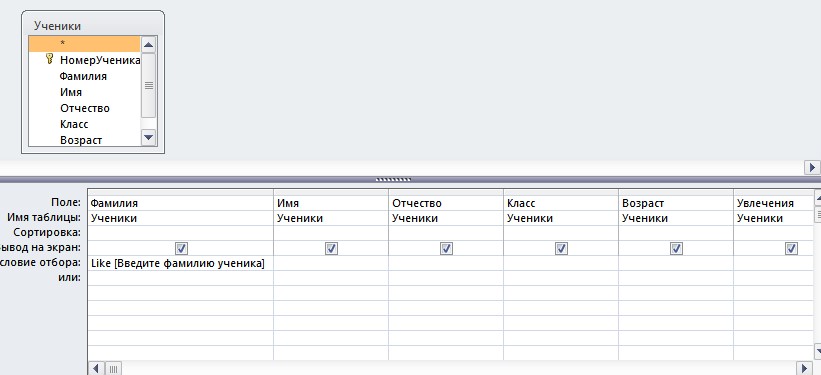


Рис. 7 Запрос с параметром

При запуске запроса появится диалоговое окно, в которое следует ввести интересующего ученика (рис. 8).

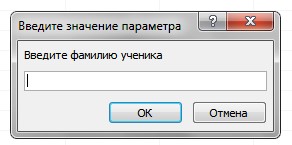


Рис. 8. Окно ввода параметра

Создадим многотабличный запрос, который показывает список учеников хорошистов с оценками по предметам 4 или 5 с сортировкой по полю «Фамилия» (рис. 9).

Результат выполнения запроса представлен в приложении 2.

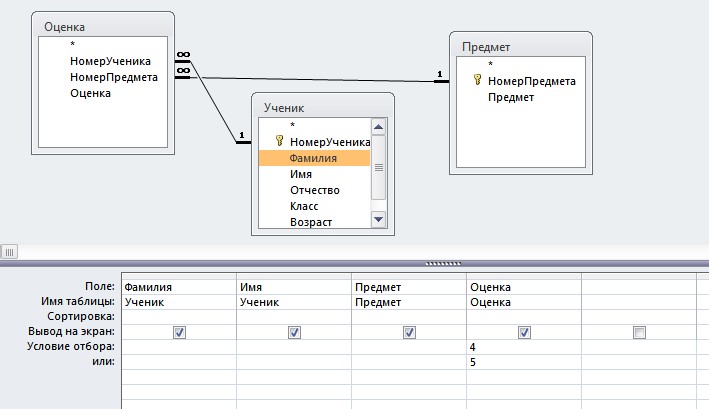


Рис. 9. Запрос в режиме конструктора

Иногда в запросах требуется произвести некоторые вычисления, например, посчитать средний бал по каждому ученику (рис. 10). Для этого создадим запрос в режиме конструктора, добавим две таблицы «Предмет» и «Оценка», правой кнопкой мыши щелкнем по полю «Оценка» и выберем из контекстного меню пункт «Итоги», после чего в строке «Групповая операция» в поле «Оценка» выберем функцию Avg.

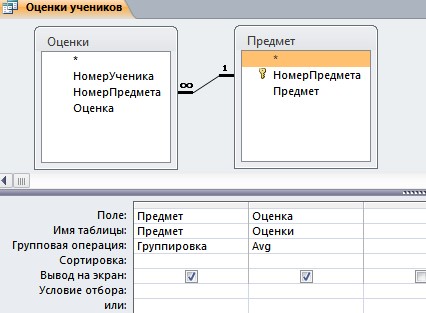


Рис. 10. Запрос по средним оценкам в режиме конструктора

*4.3. Создание форм*

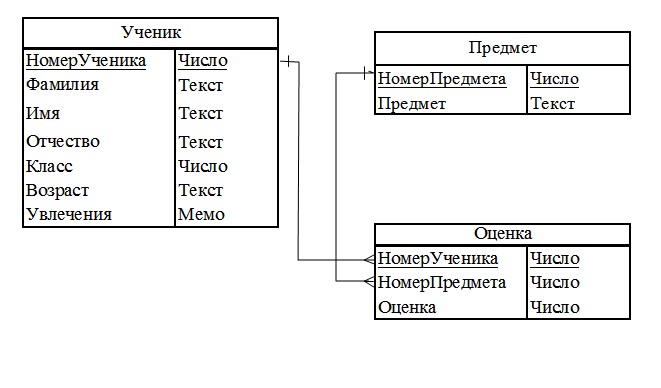
Создадим с помощью конструктора форм нужные для проекта формы, для этого выполним команду на вкладке ленты Создание – Конструктор форм. Создадим главную кнопочную форму, на которой будет предложено выбрать пользователя базы данных. Создадим на этой форме три кнопки: Учитель – ведет на форму «Учитель», Ученик – ведет на форму «Ученик» и кнопку Выход – закрытие главной формы. Далее с помощью мастера форм создадим форму «Ученик», на которой будут располагаться данные из таблицы «Ученик», откроем форму в режиме конструктора и дополнительно разместим кнопку закрытия формы и кнопку вывода ведомости оценок по указанной фамилии. В свойствах кнопки «поиск по фамилии» на вкладке «События» в поле «Нажатие кнопки» укажем имя макроса «Макрос\_поиск». В свойстве «Блокировка» всех элементов управления формы, которые берут данные из таблицы «Ученик» укажем «Да». Это означает, что данные будут доступны только в режиме чтения, без права исправления. В режиме конструктора создадим форму «Учитель», на которой будут располагаться три кнопки: показывающая список учеников по классам, показывающая ведомость оценок, кнопка закрытия формы. При нажатии кнопки «Показать список класса» будет открыта дополнительная форма «Список всех учеников», при нажатии кнопки «Показать ведомость оценок» будет открыт отчет «Итоговая ведомость». На форме «Список всех учеников» разместим кнопки удаления и добавления новой записи в таблицу, а также кнопку закрытия формы. Снимки экранов форм представлены в приложении 2.

*4.4. Создание отчетов*

Для создания отчетов можно воспользоваться мастером отчетов, для этого выполним команду: вкладка ленты Создание – Мастер отчетов. С помощью мастера создадим отчет об успеваемости школьников на основе запроса «Ведомость всех оценок», создадим отчет на основе перекрестного запроса. Затем создадим итоговый отчет, где с помощью функции Avg подсчитаем среднюю оценку за предмет.

Отчеты представлены в приложении 2.

# ПРИЛОЖЕНИЕ 1 СХЕМА ДАННЫХ



**ПРИЛОЖЕНИЕ 2**

**ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ**

Таблица «Предмет»

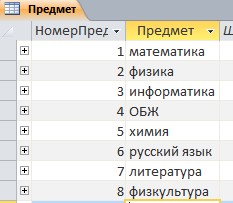
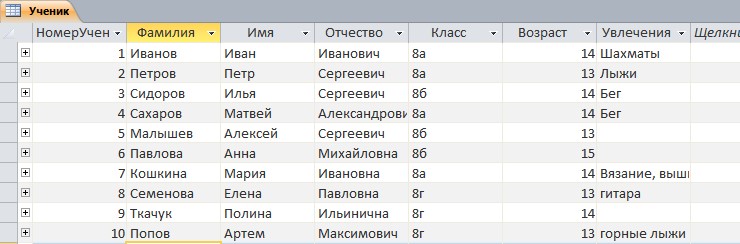


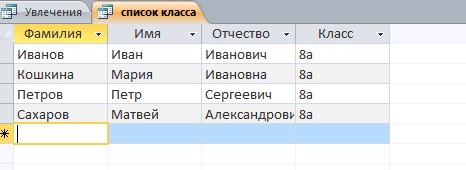
Таблица «Оценки»



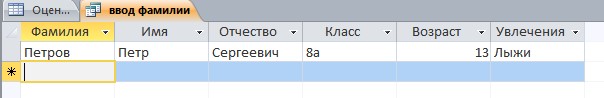
Таблица «Ученик»



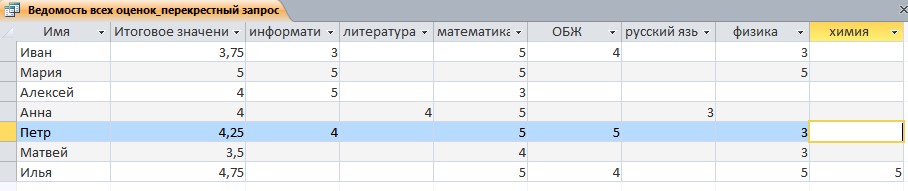
Результат выполнения запроса на выбору по классу «8а»



Результат выполнения запроса с параметром «фамилия Петров»



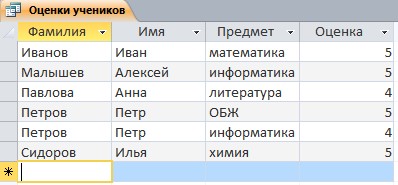
Результат выполнения перекрестного запроса с подсчетом средней оценки



Результат выполнения запроса по всем ученикам всех классов



Результат выполнения запроса на выборку по оценкам «4» и «5»



Результат выполнения запроса на выборку по увлечениям

