**Задание**

Разработать информационную подсистему АБИТУРИЕНТЫ для автоматизации работы приемной комиссии вуза. Требуется обработка анкетных данных абитуриентов. Анкета включает следующие данные об абитуриенте: регистрационный номер; фамилию, имя и отчество; дату рождения; сведения об оконченном среднем учебном заведении (название, номер, город, дата окончания); наличие диплома с отличием (техникум) или золотой/серебряной медали (школа); адрес (индекс, территория, населенный пункт, улица, номер дома, телефон); выбранную специальность. В вузе определен список специальностей для приема, который может изменяться ежегодно. По каждой специальности вуза определен список предметов, сдаваемых абитуриентами, например: математика (письменно), математика (устно), русский язык и т. д. Обеспечить выполнение следующих функций: ввод и коррекцию анкетных данных абитуриентов; просмотр анкетных данных по специальностям в алфавитном порядке; ввод, коррекцию и просмотр специальностей и сдаваемых предметов; вывод на экран анкетных данных абитуриентов, имеющих красный диплом или медаль.

Разработать: меню приложения и средства диалога, формы ввода и изменения данных, запросы (если они нужны), отчеты для вывода на печать.

Целью работы является разработка приложения для БД средней сложности с типовым пользовательским интерфейсом.

Для разработки автоматизированной системы использовались следующие программные средства: Microsoft Office Access.

**Содержание**

Введение

1. Анализ предметной области

1.1 Описание предметной области

1.2 Построение инфологической модели на языке «Таблицы-связи»

2. Разработка приложения

2.1 Построение реляционной модели данных

2.2 Разработка физической структуры при помощи СУБД MS Access

2.2.1 Построение таблиц

2.2.2 Формы

2.2.3 Запросы

2.2.4 Отчеты

3. Работа с программой

3.1 Макет таблиц

3.2 Анкета абитуриента

3.3 Специальности

3.4 Сдаваемые предметы по специальности

3.5 Поиск абитуриентов по выбранной специальности

3.6 Студенты с з/с медалями, дипломом с отличием

3.7 Отчеты

3.7.1 Анкеты об абитуриентах

3.7.2 Список специальностей

3.7.3 Студенты с золотом, серебром, дипломом с отличием

3.7.4 Факультеты

3.8 Выход

Заключение

Список использованных источников

**Введение**

В современных условиях динамично развивается рынок комплексных интегрированных систем автоматизации предприятий и учреждений самого различного профиля (финансовых, промышленных, офисных, учебных) и самых различных размеров с разнообразными схемами иерархии, начиная от малых предприятий с численностью в несколько десятков человек и заканчивая крупными корпорациями численностью в десятки тысяч сотрудников. Такие системы предназначены для решения задач как предприятия в целом (управление финансовыми ресурсами, управление запасами, планирование и производство, сбыт и снабжение и т.д.), так и уровня его производственных подразделений, цехов и участков.

Главная особенность индустрии систем автоматизации различных предприятий и учреждений, характеризующихся широкой номенклатурой входных данных с различными маршрутами обработки этих данных, состоит в концентрации сложности на начальных этапах анализа требований и проектирования спецификаций системы при относительно невысокой сложности и трудоемкости последующих этапов. Фактически здесь и происходит понимание того, что будет делать будущая система, и каким образом она будет работать, чтобы удовлетворить предъявленные к ней требования. А именно нечеткость и неполнота системных требований, нерешенные вопросы и ошибки, допущенные на этапах анализа и проектирования, порождают на последующих этапах трудные, часто неразрешимые проблемы и, в конечном счете, приводят к неуспеху всей работы в целом.

С другой стороны, не существует двух одинаковых организаций. А следовательно, простое тиражирование даже очень хорошей системы управления предприятием никогда не устроит заказчика полностью, поскольку не может учесть его специфики. Более того, в данном случае возникает проблема выбора именно той системы, которая наиболее подходит для конкретного предприятия.

Фактически проблема комплексной автоматизации стала актуальной для каждого предприятия.

Современные информационные системы управления базируются на комплексных системах обработки информации, на современных информационных технологиях.

Важной категорией являются системы обработки информации, от которых во многом зависит эффективность работы любого предприятия ли учреждения. Такая система должна:

* обеспечивать получение общих и/или детализированных отчетов по итогам работы;
* позволять легко определять тенденции изменения важнейших показателей;
* получение информации без задержек по времени;
* выполнять точный и полный анализ данных.

Базы данных являются эффективным средством представления структур данных и манипулирования ими. Концепция баз данных предполагает использование интегрированных средств хранения информации, позволяющих обеспечить централизованное управление данными и обслуживание ими многих пользователей.

Целью курсовой работы является разработка для ЭВМ базы данных средней сложности с типовым пользовательским интерфейсом, а в частности, разработка базы данных АБИТУРИЕНТЫ на основе реляционной системы управления базами данных Microsoft Access, входящей в комплект Microsoft Office.

**1. Анализ предметной области**

* 1. **Описание предметной области**

В базе данных "Абитуриенты" отражаются личные данные абитуриентов в которых хранятся данные об аттестате, об абитуриенте, сведения о сдаче вступительных экзаменах, форма обучения а также некоторые дополнительные данные (информация о поданных заявлениях, сведения о родителях, воинский учет, увлечения). Формируются списки на зачисление согласно результатам вступительных экзаменов.

Широкое применение компьютерных технологий в учебном процессе выдвинуло перед работниками приемной комиссии вуза задачу автоматизировать работу приемной комиссии от момента заполнения личной карточки абитуриента до выполнения различного рода отчетов.

Автоматизируем Базу данных «Абитуриенты» согласно поставленной задачи.

База данных содержит данные о абитуриентах поступающих в ВУЗ, специальности ВУЗа и вступительные экзамены по специальностям при поступлении в ВУЗ, так же выводит отчеты, а также осуществляется вывод на печать (на принтер). Конечными пользователями базы данных являются работники приемной комиссии вуза. Согласно поставленной задаче необходимо разработать информационную модель.

Разрабатывая информационную модель, необходимо выполнить логическое проектирование. В результате концептуального проектирования получаем описание информационной модели. Отражаем инфологическую модель предметной области на реляционную модель данных.

Результатом работы базы данных «Абитуриенты» являются отчеты, запросы.

**1.2 Построение инфологической модели на языке «Таблицы-Связи»**

Определим входные данные:

- Регистрационный номер;

- Фамилия;

- Имя;

- Отчество;

- Дата рождения;

- Сведения об оконченном среднем учебном заведении

(название, номер, город);

- Награды (золотая медаль, серебряная медаль, наличие диплома с отличием);

- Адрес (индекс, территория, населенный пункт, улица, номер дома, телефон);

- Код выбранной специальности;

- Название специальности;

- Вступительные экзамены.

Проведем нормализацию логической модели.

Нормализация - процесс проверки и реорганизации сущностей и атрибутов с целью удовлетворения требований к реляционной модели данных. Нормализация позволяет быть уверенным, что каждый атрибут определен для своей сущности, значительно сократить объем памяти для хранения информации и устранить аномалии в организации хранения данных. В результате проведения нормализации должна быть создана структура данных, при которой информация о каждом факте хранится только в одном месте. Процесс нормализации сводится к последовательному приведению структуры данных к нормальным формам - формализованным требованиям к организации данных.

1 НФ – когда просматривается ключ, поля и нет повторяющихся групп. Этому условию наше отношение соответствует. Приведем к 2 НФ - отношение находиться в 1 НФ и все атрибуты функционально зависят от ключа.

Выделю сущность «Специальность», т.к. специальностей существует много, то целесообразно выделить «Специальность» в отдельную сущность. Связь сущности «Персона» и «Специальность» один ко многим.

Выделю сущность «Сведения об оконченном среднем учебном заведении», т.к. средних учебных заведений существует много, то целесообразно выделить «Сведения об оконченном среднем учебном заведении» в отдельную сущность. Связь сущности «Персона» и «Сведения об оконченном среднем учебном заведении» один ко многим. Связь сущности «Персона» с сущностями «Награды» и «Адрес Персоны» один к одному. Для удобства выделим их в отдельные сущности. Т.к. наград существует много, то целесообразно выделить «Вид Награды» в отдельную сущность. Связь сущности «Награды» и «Вид Награды» один ко многим. Т.к. на одном факультете может быть много кафедр, то целесообразно выделить «Кафедры» в отдельную сущность. Т.к. на одной кафедре может быть несколько специальностей, то целесообразно выделить «Специальность» в отдельную сущность. Связь «Факультеты» и «Кафедры» один ко многим, «Кафедры» и «Специальность» один ко многим. Выделим сущность «Предметы» в отдельную, т. к существует много предметов сдаваемых при поступлении на одну специальность. Сущность «Специальность-Предмет» является связующей сущностью сущностей «Специальность» и «Предмет».

Выявим сущности предметной области «Абитуриенты».

Основываясь на анализе предметной области можно выявить такие основные сущности: Персона, Адрес Персоны, Награды, Вид Награды, Сведения об оконченном среднем учебном заведении, Предмет, Специальность - Предмет, Кафедры, Специальность, Факультеты.

Основные предметно – значимые атрибуты сущностей следующие.

Для сущности Персона определяем такие основные атрибуты:

- Регистрационный номер;

- Фамилия;

- Имя;

- Отчество;

- Дата рождения;

- Код специальности;

- Код среднего учебного заведения;

Для сущности Адрес Персоны следующие атрибуты:

- Код магазина;

- Индекс;

- Территория;

- Населенный пункт;

- Улица;

- Номер дома;

- Телефон

Для сущности Награды следующие атрибуты:

- Регистрационный номер;

- Вид награды

Для сущности Вид награды следующие атрибуты:

- Вид награды;

- Наименование

Для сущности Сведения об оконченном среднем учебном заведении следующие атрибуты:

- Код среднего учебного заведения;

- Название;

- Номер;

- Город

Для сущности Предмет следующие атрибуты:

- Код предмета;

- Название

Для сущности Специальность- Предмет следующие атрибуты:

- Код специальности;

- Код предмета

Для сущности Факультеты следующие атрибуты:

- Код факультета;

- Название

Для сущности Кафедры следующие атрибуты:

- Код кафедры;

- Код факультета;

- Название

Для сущности Специальность следующие атрибуты:

- Код специальность;

- Код кафедры;

- Название

На основе выше перечисленного создам логическую модель данных проектируемой информационной системы, образец представлен в Приложении Б.

Построенную на языке «сущность-связь» модель предметной области легко отразить в реляционной базу данных. Так каждая сущность будет определена как отдельная таблица, а каждый атрибут сущности – как отдельное свойство (поле), имеющее свои значения.

**2. Разработка приложения**

**2.1 Построение реляционной модели данных**

Реляционная модель данных – модель данных, которая описывает:

* + структуры данных в виде (изменяющихся во времени) наборов отношений;
	+ теоретико-множественные операции над данными: объединение, пересечение, разность и декартово произведение;
	+ специальные реляционные операции: селекция, проекция, соединение и деление;
	+ специальные правила, обеспечивающие целостность данных.

В реляционной модели данных объекты и взаимосвязи между ними представляются с помощью таблиц.

Взаимосвязи также рассматриваются в качестве объектов. Каждая таблица представляет один объект и состоит из строк и столбцов. В реляционной базе данных каждая таблица должна иметь первичный ключ (ключевой элемент) – поле или комбинацию полей, которые единственным образом идентифицируют каждую строку в таблице.

Основными достоинствами реляционной модели данных являются:

* + простота и доступность;
	+ независимость данных;
	+ гибкость;
	+ возможность непроцедурных запросов;
	+ хорошее теоретическое обоснование.

Недостатки данной модели:

* + низкая производительность;
	+ необходимость глубокого рассмотрения отношений (нормализация), в том числе отношений М:М;
	+ возможность логических ошибок и необходимость осторожной работы с моделью;
	+ линейность структуры таблиц.

Реляционную модель данных я выбрала, потому что она наиболее удовлетворяет требованиям построенной информационно-логической модели.

Определяем каждую сущность как отдельную таблицу.

Следовательно, имеем 10 таблиц:

- Адрес персоны;

- Вид награды;

- Кафедры;

- Награды;

- Персона;

- Предметы;

- Сведения об оконченном среднем учебном заведении;

- Специальность;

- Специальность – Предмет;

- Факультеты.

Реляционная модель Приложение В.

**2.2 Разработка физической структуры при помощи СУБД MS Access**

СУБД MS Access - это комплекс программ, позволяющий хранить большие массивы данных в определенном формате, обрабатывать их и представлять в удобном для пользователя виде. MS Access дает также возможность автоматизировать часто выполняемые операции.

Кроме того, MS Access является СУБД реляционного типа, в которой разумно сбалансированы все средства и возможности, типичных для современных СУБД. Реляционная база упрощает поиск, анализ, поддержку и защиту данных, поскольку они сохраняются в одном месте.

При этом каждая реляционная таблица становится таблицей базы данных, каждый столбец реляционной таблицы становится полем таблицы базы данных, и каждая строка реляционной таблицы становится записью таблицы базы данных. Для автоматизации проектируемой информационной системы я выбрала СУБД MS Access, потому что разработанная мной модель данных является реляционной, следовательно, реализовать ее целесообразней на СУБД реляционного типа. На основе построенной модели данных создадим физическую модель данных (схему данных).

**2.2.1 Построение таблиц**

Таблица представляет собой набор данных, упорядоченных в виде строк и столбцов.

Каждая таблица предназначена для определенного набора данных.

Использование отдельной таблицы для определенного набора данных позволяет исключить дублирование данных, хранить данные более эффективно и уменьшить количество ошибок при вводе данных.

Для создания таблиц необходимо присвоить имена таблицам и их полям, определить параметры полей таблиц. Пример Таблица с именем Персона.

Таблица 1. Персона

|  |  |  |
| --- | --- | --- |
| **Название поля** | **Тип данных** | **размер** |
| РегНомер | числовой | длинное целое |
| Фамилия | текстовый | 255 |
|  Имя  | текстовый | 255 |
| Отчество | текстовый | 255 |
| Дата рождения | дата/время |  |
| КодСпец | числовой | длинное целое |
|  КодСрУчЗав | числовой | длинное целое |

**2.2.2 Формы**

Формы являются основным средством организации интерфейса пользователя в СУБД MS Access. Формы могут применяться для различных целей:

* + Вывод и редактирование данных;
	+ Управление ходом выполнения приложения;
	+ Вывод сообщений;
	+ Печать информации.

Виды форм:

Простые формы. В простой форме показывается только одна запись таблицы, причем все необходимые поля отображаются на одной странице.

Многостраничные формы. При необходимости вывести на экран записи, каждая из которых содержит большой объем информации, можно создать многостраничную форму.

Ленточные формы. В отличие от простой формы, которая показывает только одну запись, в ленточной форме отформатированные записи выводятся на экран одна за другой, как в таблице.

Подчиненные формы. Подчиненные формы используются для представления данных, находящихся на стороне «многие» отношения 1:М.

Пример. Создадим в конструкторе форм форму Персона



Рисунок 2.2.2.1 Форма в режиме конструктора

**2.2.3 Запросы**

Запрос представляет собой обращение к данным для получения информации и выполнения действий с данными. Запрос можно использовать для получения ответа на простой вопрос, выполнения расчетов, объединения данных из разных таблиц или даже добавления, изменения или удаления данных в таблице. Запросы, используемые для извлечения данных из таблицы или выполнения расчетов, называются запросами на выборку. Запросы, используемые для добавления, изменения или удаления данных, называются запросами на изменение.

Запросы бывают:

* + простыми – это запросы на выборку из определенных полей;
	+ перекрестными – это вид итоговых запросов, который позволяет увидеть вычисляемые значения в виде перекрестной таблицы, напоминающей электронную таблицу.

Пример запроса в режиме конструктора



Рисунок 2.2.3.1 Запрос на выборку в режиме конструктора

**2.2.4 Отчеты**

Отчеты представляют собой наилучшее средство для представления информации из базы данных в виде печатного документа. По сравнению с другими средствами вывода данных на печать отчеты обладают двумя принципиальными преимуществами:

* + предоставляют широкие возможности для группировки и вычисления промежуточных и общих итогов для больших наборов данных.
	+ могут быть использованы для получения красиво оформленных счетов, заказов на покупку, почтовых наклеек, материалов для презентаций и других документов, которые могут понадобиться для успешного ведения бизнеса.

В отчете каждая группа данных и итоги по ней могут быть представлены отдельно. В отчетах имеются следующие возможности:

* + для обеспечения иерархического представления данных можно определить до 10 условий группировки.
	+ для каждой из групп можно задать отдельные заголовки и примечания.
	+ можно производить сложные вычисления.
	+ в дополнении к верхнему и нижнему колонтитулу можно задать заголовок и примечание для всего отчета в целом.

Окно отчета в режиме Конструктора идентично окну формы в этом режиме, т.е. присутствует такая же Панель элементов, окно со списком полей и окно со списком свойств. Пустой макет отчета содержит разделы Верхний колонтитул, Нижний колонтитул, между которыми находится Область данных. Расположение данных на странице по вертикали определяется тем, как вы зададите верхний и нижний колонтитулы и соответствующие поля страницы. Как и при работе с формами, можно изменить размер любого раздела, перетащив его границу. Ширина всех разделов должна быть одной и той же, поэтому при изменении ширины одного из разделов Ассеss автоматически настраивает ширину всех остальных.

Верхний и нижний колонтитулы будут печататься соответственно вверху и внизу каждой страницы отчета. Добавить или удалить верхний и нижний колонтитулы можно с помощью команды Вид → Колонтитулы. Можно задать заголовок отчета, который будет печататься только в самом начале отчета, и примечание отчета, печатающееся в конце отчета. Чтобы добавить в отчет эти разделы, выберите Формат → Заголовок/примечание отчета.

Пример отчета «Анкета об абитуриенте» в режиме конструктора.



Рисунок 2.2.4.1 Отчет «Анкета об абитуриенте»

**3. Работа с программой**

база данные реляционный

Данное приложение рассчитано на работу в среде Windows под управлением СУБД Microsoft Access. При открытии файла «Абитуриенты Попова С.В.mdb» появляется главная форма приложения.



Рисунок 3.1 – Главная форма приложения «Абитуриент»

Главная форма включает следующие пункты (кнопки):

1. Анкета абитуриента
2. Специальности
3. Сдаваемые предметы по специальности
4. Поиск абитуриентов по выбранной специальности
5. Студенты с з/с медалями, дипломом с отличием
6. Вход из приложения
7. Отчеты

**3.1 Макет таблиц**



Рисунок 3.1.1 – Главное меню объекта таблиц

Рисунок 3.1.2 – Таблица «Персона» в конструкторском режиме



Рисунок 3.1.3 – Таблица «Персона» в нормальном виде

**3.2 Анкета абитуриента**

При нажатии данной кнопки вызывается форма «Персона» рисунок 3.2.1.



Рисунок 3.2.1 – Форма «Персона»

Назначение данной формы – добавить в базу данных сведения об абитуриентах поступающих в ВУЗ. Все измененные данные студентов при этом заносятся в таблицу «Персона».

**3.3 Специальности**

При нажатии данной кнопки вызывается форма «Специальность» рисунок 3.3.1.



Рисунок 3.3.1 – Форма «Специальность»

Назначение данной формы – внести в базу данных сведения о специальностях ВУЗа. Введенные данные о специальностях при этом заносятся в таблицу «Специальность».

**3.4 Сдаваемые предметы по специальности**

При нажатии данной кнопки вызывается форма «Специальность-Предмет» рисунок 3.4.1.



Рисунок 3.4.1 – Форма «Специальность- Предмет»

Назначение данной формы – внести в базу данных названия сдаваемых предметов по заданной специальности.

**3.5 Поиск абитуриентов по выбранной специальности**

При нажатии данной кнопки вызывается запрос «Абитуриенты по номеру специальности» рисунок 3.5.1.



Рисунок 3.5.1 – Запрос «Абитуриенты по номеру специальности»

**3.6 Студенты с з/с медалями, дипломом с отличием**

При нажатии данной кнопки вызывается запрос «Запрос, красный диплом или медаль» рисунок 3.6.1.



Рисунок 3.6.1 – Запрос «Запрос, красный диплом или медаль»

**3.7 Отчеты**

При нажатии данной кнопки вызывается форма «Главная отчеты» рисунок 3.7.1.



Рисунок 3.7.1 – Форма «Главная отчеты»

Эта кнопочная форма содержит 5 пунктов меню:

1. Анкеты об абитуриентах
2. Список специальностей
3. Студенты с золотом, серебром, дипломом с отличием
4. Факультеты
5. Выход

Назначение данной формы – сформировать отчеты на основе существующей базы данных.

**3.7.1 Анкеты об абитуриентах**

При нажатии данной кнопки вызывается отчет «Анкета абитуриента» рисунок 3.7.1.1.



Рисунок 3.7.1.1 – Отчет «Анкета абитуриента»

Назначение данного отчета – показать анкетные данные студентов.

**3.7.2 Список специальностей**

При нажатии данной кнопки вызывается отчет “ Список специальностей ” рисунок 3.7.2.1.



Рисунок 3.7.2.1 – Отчет «Список специальностей»

Назначение данного отчета – показать код, название специальности, а также сдаваемые по данной специальности предметы.

**3.7.3 Студенты с золотом, серебром, дипломом с отличием**

При нажатии данной кнопки вызывается отчет «Студенты с золотом, серебром, дипломом с отличием» рисунок 3.7.3.1.



Рисунок 3.7.3.1 – «Студенты с золотом, серебром, диплом с отличием**»**

**3.7.4 Факультеты**

При нажатии данной кнопки вызывается отчет «Факультеты»

Назначение данного отчета – показать названия факультетов относящихся к ним кафедр и специальности, относящиеся к каждой кафедре.

**3.8 Выход**

При нажатии данной кнопки приложение закрывается и осуществляется выход в среду Windows.

**Заключение**

Использование баз данных и информационных систем становится неотъемлемой составляющей деловой деятельности современного человека и функционирования преуспевающих организаций. В связи с этим большую актуальность приобретает освоение принципа построения и эффективного применения соответствующих технологий и программных продуктов.

На первом этапе работ была изучена и проанализирована предметная область заданного процесса.

На втором этапе – по проведенному анализу определенных задач были выделены сущности проектируемой информационной системы и построена ее информационно-логическая модель.

Таким образом, в ходе работы были изучены основы проектирования информационных систем.

В курсовой работе была спроектирована, разработана и отлажена база данных на приложении Microsoft Access из пакета Microsoft Office. Эта база данных имеет удобный интерфейс, главное меню, а также обеспечивает вывод на печать различных сведений из базы