# Введение

Темой дипломной работы является разработка информационно-справочной системы для ДЮСШ №1 г. Алапаевска. Вопрос автоматизации и информационной поддержки учебного процесса в школе до сих пор остается открытым, то есть не используется никакая информационная система. Подобных информационных систем для спортивных школ на рынке ИС найти не удалось.

Информационная система (ИС) должна обеспечивать руководителя, тренеров, методиста оперативной информацией для контроля учебного процесса и анализа его результатов

Информационная система должна содержать базу данных, в которой хранятся данные о: контингенте учащихся, тренеров, проводимых мероприятиях и соревнованиях, позволяющую вести учет выступлений на соревнованиях, выполнение нормативных требований, сохранность контингента учащихся в течение учебного года, уровень состояния здоровья учащихся, успеваемость.

Информационная система должна отвечать следующим требованиям:

– В БД должна быть представлена вся необходимая для работы пользователей информация и организован удобные и быстрый ввод, поиск и просмотр данных

– В ИС должна быть предусмотрена генерация и печать необходимых текущих отчетных, аналитических и др. документов.

– ИС должна обладать удобным и простым для восприятия интерфейсом и справочной системой.

**Постановка задачи**

1. Провести анализ предметной области, выделить категории пользователей и собрать их информационные требования

2. Спроектировать логическую модель данных для РБД.

3. Реализовать проект средствами СУБД Microsoft Access 2003.

4. Спроектировать и реализовать необходимые запросы.

5. Разработать и реализовать простой и удобный пользовательский интерфейс.

# 1. Основные сведения из теории реляционных баз данных

## 

## 1.1 Определение основных понятий

**Сущность** – это любой отличимый объект, информацию о котором мы хотим хранить в БД.

**Тип сущности** – это набор (множество) однородных объектов, т.е. объектов, обладающих определенным набором общих свойств и выступающих как единое целое.

**Экземпляр сущности** – каждый конкретный элемент набора (типа сущности).

**Атрибут** – это поименованная характеристика типа сущности, т.е. свойство, общее для всех экземпляров данного типа.

Атрибут также имеет тип (т.е. множество значений), называемый **доменом,** и конкретный экземпляр атрибута, совпадающий с одним из значений, содержащихся в домене.

**Ключ (или возможный ключ)** типа сущности – это атрибут или множество атрибутов, по значениям которых можно однозначно идентифицировать каждый экземпляр сущности. Из самого определения сущности как отличимого объекта следует, что каждый тип сущности обладает ключом.

Один из возможных ключей выбирается в качестве первичного ключа (**РК – Primary Кеу**)

Из определения РК вытекают следующие его ограничения (свойства):

– **Уникальность**. Это означает, что в произвольный момент времени ни у каких двух экземпляров сущностей не допускается одинаковых значений ключа.

– Не допускается, чтобы какой-либо из атрибутов, входящий в РК, принимал для какого-либо экземпляра сущности **неопределенное (или NULL) значение**, так как последнее означало бы, что такой экземпляр невозможно идентифицировать. Это свойство называют **целостностью по сущностям.**

**Связь** – это связывание между собой двух или более сущностей.

## 1.2 Классификация сущностей и связей по К. Дейту

При проектировании необходимо установить между объектами БД определенные связи, отражающие связи в реальном мире. Для эффективной реализации БД часто оказывается полезной **классификация связей по количественному признаку**, т.е. в соответствии с тем, сколько экземпляров одного типа сущности со сколькими экземплярами другого типа сущности может связываться.

Рассмотрим такую классификацию для связей между двумя типами сущностей.

Пусть А и В-типы сущностей, тогда возможны четыре вида связей:

а) **Один-к-одному** (1:1). Это означает, что в каждый момент времени каждый экземпляр сущности А связывается не более чем с одним экземпляром сущности В. Это самый простой и довольно редкий вид связи.

б) **Один-ко-многим** (1:М), М/0. Связь (1:1) фактически входит сюда как частный случай. Здесь с одним экземпляром сущности А связывается М/0 экземпляров сущности В.

в) Многие-к-одному (N:1), N/1. N экземпляров сущности А связываются с одним экземпляром сущности В

г) Многие-ко-многим (N:M), N/1, M/0. Это наиболее общий вид связи, его обычно называют ассоциацией, а числа M и N степенями связи.

Классификация связей, основанная на принципе так называемого независимого существования связи по отношению к целевой сущности. При определении независимого существования необходимо дать ответ на вопрос: «Прекращает или нет свое существование связь, если предпринимается попытка удаления экземпляра целевой сущности, с которым она связывается?»

Характеристика – это чаще всего связь типа (1:М), либо (1:1), имеющая записимое от цели существование. Это означает, что при удалении экземпляра целевой сущности, на которую ссылается экземпляр характеристики, последний тоже должен прекратить свое существование, т.е. быть удален.

Обозначение – это обычно связь типа (N:1), N/1**,** имеющаянезависимое от цели существование, т.е.фактически должно быть запрещено удаление экземпляра цели, если у него есть связанные записи.

**Ассоциация** – связь типа (N:M), и она может иметь как зависимое, так и независимое от целей существование.

К. Дейт определяет три основные класса сущностей: стержневые, ассоциативные и характеристические, а также подкласс ассоциативных сущностей – обозначения.

Введенный К. Дейтом термин **стержень** (или стержневая сущность) – это сущность, имеющая независимое существование и не являющаяся связью. Стержни отображают основные предметы или понятия той предметной области, для которой проектируется БД. В среде проектировщиков их часто называют **справочниками.**

Для описания инфологической модели используются ER-диаграммы и специальные языки инфологического моделирования – ЯИМ.

Для реализации связей в БД введем понятие **внешнего ключа (FK – Foreigh Кеу)** и остановимся на вопросе выбора внешних ключей. Дадим неформальное, но конструктивное определение внешнего ключа для различных видов связей:

Если сущность А связывает сущности Е1 (с первичным ключом PK1) и Е2 (с первичным ключом PK2) и является ассоциацией, то в состав ее атрибутов должны входить внешние ключи (FК1, FК2), соответствующие первичным ключам целевых сущностей Е1 и Е2. Совокупность внешних ключей должна входить в состав ключа ассоциации.

Если сущность С обозначает или характеризует сущность Е (первичный ключ РК), то в состав ее атрибутов нужно включать внешний ключ (FК), соответствующий РК целевой сущности Е. Причем если сущность С является обозначением, РК не обязательно включается в состав ее первичного ключа.

Следующий важный момент – это спецификация ограничений для каждого внешнего ключа. Ограничения определяются правилами и соглашениями, принятыми в предметной области.

1. неопределенное значение (NULL)

2. действие удаления ограничивается (RESTRUCT);

3. действие удаления распространяется или каскадируется (CASCADE).

## 1.3 Основы теории нормализации Э. Кодда

Рассмотрим общую идею нормализации. Причиной, которая может привести к нарушению целостности данных, является избыточность. Э. Кодд исследовал и установил причины, порождающие избыточность, а именно наличие в таблице нежелательных зависимостей между атрибутами. Он предложил способы для избавления от этих зависимостей и, следовательно, от избыточности данных. Кодд ввел понятия функциональных зависимостей между атрибутами и нормальных форм для реляционных таблиц: 1НФ, 2НФ, 3НФ, 4НФ, 5НФ, НФБК.

**Определение 1.**

Поле В (может быть составным) таблицы **функционально зависит** от поля А (может быть составным) той же таблицы, если в любой заданный момент времени для каждого из различных значений поля А обязательно существует только одно из различных значений поля В. Обозначение А 🡪В

**Определение 2**

Поле В (может быть составным) таблицы находится в **полной ФЗ** от составного поля А той же таблицы, если оно функционально зависит от А (А 🡪 В) и не зависит функционально ни от какого подмножества А. Обозначение: А => В.

**Определение 3**

Если существует ФЗ между не ключевыми атрибутами (F1 🡪 F2), то такая зависимость называется **транзитивной.**

**Определение 4**

Таблица находится в **первой нормальной** **форме** (1НФ) тогда и только тогда, когда в каждой ее строке на пересечении со столбцом может содержаться только атомарное значение и ни одно из ее ключевых полей не имеет неопределенного (NULL) значения, т.о. любая реляционная таблица находится в 1НФ.

**Определение 5**

Таблица находится во **второй нормальной форме** (2НФ) тогда и только тогда, когда она находится в 1НФ и все ее поля, не входящие в РК (не ключевые), связаны полной ФЗ с РК.

**Определение 6**

Таблица находится в **третьей нормальной форме** (ЗНФ) тогда и только тогда, когда она находится в 2НФ и в ней нет транзитивных зависимостей.

**Определение 7**

Таблица находится в **нормальной форме Бойса-Кодда (НФБК)** тогда и только тогда, когда любая ФЗ между ее полями является полной ФЗ от возможного ключа.

**Определение 8**

**Полной декомпозицией** таблицы называют такую совокупность ее проекций, естественное соединение которых дает исходную таблицу, причем сохраняются все существовавшие в ней ФЗ.

**Процесс нормализации таблицы** – это пошаговое разбиение (декомпозиция) таблицы на две (или более) части, которые обладают меньшей избыточностью и, следовательно, лучшими свойствами по отношению к операциям включения, обновления и удаления данных.

## 1.4 Этапы проектирования базовых таблиц РБД

**Проектирование базы данных** – это разработка структуры данных и связей с целью создания системы, удовлетворяющей информационным потребностям всех категорий пользователей, для которых она предназначается.

Методология проектирования логических моделей данных делится на два уровня:

– **инфологическое проектирование** или построение инфологической модели данных;

**Базовые таблицы** – совокупность реляционных таблиц, представляющих все сущности, входящие в инфологическую модель.

– **даталогическое проектирование** или построение даталогиеской модели (концептуальной схемы) для реляционной БД.

**Этапы проектирования.**

**1**. Сбор и анализ информационных требований к БД.

На этом этапе должны быть определены все категории пользователей и собраны все атрибуты необходимых им данных.

**2**. Сбор информации об использовании данных.

На этом этапе должно быть проанализировано, каким образом будут использоваться данные, т.е. какие данные, каким процедурам обработки будут подвергаться и как часто.

Имеет смысл информацию об использовании данных разделить на два вида:

– Информация, связанная с основными производственными функциями.

– Информация, связанная с функциями управления.

**3**. Первоначальное структурирование собранной информации.

На этом этапе заканчивается анализ предметной области и этап предпроектирования, результатами которого должно быть:

– универсальное отношение

– домены всех атрибутов

– ограничения модели по отношению к предметной области

– декларация всех функциональных зависимостей.

**4**. Проектирование инфологической модели.

Инфологическое проектирование – процесс выделения и классификаций сущностей и связей по методологии К. Дейта.

**5**. Проектирование даталогической модели для РБД.

На этом этапе осуществляется проектирование совокупности базовых реляционных таблиц, представляющих все сущности, входящие в инфологическую модель.

Т.о. результатом логического проектирования РБД должно быть описание всей совокупности базовых таблиц на ЯОД и схема данных.

Для привязки даталогической модели к среде хранения используется модель данных физического уровня. Эта модель определяется используемыми запоминающими устройствами, способами физической организации данных в среде хранения

Если структуры таблиц отвечают поставленным требованиям, то можно вводить все данные. Затем можно создавать любые запросы, формы, отчеты, макросы и модули.

## 1.5 Общие сведения о языке SQL

**SQL** (Structuret Query Language) – это международный стандарт языка реляционных баз данных, принятый ANSI (American National Standart Institute) и ISO (International Standart Organization)

**SQL** – декларативный, непроцедурный язык, предназначен для:

– создания РБД

– использования данных, хранящихся в РБД, в том числе и при работе в сети

– управления данными и пользователями РБД и обеспечения безопасности данных

– спецификации проектов баз данных (описания даталогической модели данных)

Этот язык содержит очень важные операторы, которые позволяют делать структуру БД динамической.

Все операторы языка в соответсвии с их назначением делятся на нескольно групп или подъязыков: язык определения структуры данных (DDL), язык манипулированиями данными (DML), язык запросов (DQL)

Оператор DDL.

**Оператор** **CREATE TABLE.**

Назначение оператора:

– Описание структуры новой таблицы (имена столбцов, тип данных)

– Спецификация ключей (PK и FK) и ограничение целостности для них

– Спецификация ограничений целостности для столбцов

– Спецификация ограничений целостности для всей таблицы

<определение таблицы>:=CREATE TABLE <имя\_таблицы>

(<определение\_столбца>[,<определение\_столбца>,…]

[<ограничения таблицы>];

<определение\_столбца>:=<имя\_столбца><тип\_данных>

[<ограничения\_столбца>]

<ограничения\_столбца>:=NOT NULL|UNIQUE|PRIMARY KEY

<ограничения таблицы>:=<ограничения\_уникальности>

|<ограничения\_по\_ссылкам>

|<ограничительное\_условие>

<ограничения\_уникальности>:=<спецификация\_уникальности>

(<имя\_столбца>[,<имя\_столбца>,…])

<спецификация\_уникальности>:= UNIQUE|PRIMARY KEY

<ограничения\_по\_ссылкам>:=FOREIGN KEY (<имя\_столбца>

[,<имя\_столбца>,…])

REFERENCES<имя\_целевой\_таблицы>

[<ограничения\_FK>]

<ограничения\_FK>:=ON UPDATE<действие>|ON DELETE<действие>

<действие>:=CASCADE|SET NULL|RESTRICT|SET DEFAULTNO ACTION

<ограничительное\_условие>:=CHECK (<условие>)

Оператор DQL

**Оператор SELECT**

Это центральный и наиболее часто используемый оператор SQL, с его помощью формулируются и выполняются запросы на поиск и выдачу необходимой информации из БД.

Общая структура оператора

|  |  |  |
| --- | --- | --- |
| Ключевое слово предложения | Обязательная часть | Что задается в предложении |
| SELECT | Да | Столбцы, выводимые в результат запроса |
| FROM | Да | Имена таблиц, участвующих в выполнении запроса |
| WHERE | Нет | Условие поиска (выборки) строк |
| GROUP BY | Нет | Столбцы для группировок |
| HAVING | Нет | Условие отбора строк в группу |
| ORDER BY | Нет | Столбцы и вид сортировки |

**Виды запросов:**

– однотабличные запросы

– многотабличные запросы

– составные запросы

– вложенные запросы или подзапросы

2. Описание проекта

## 2.1 Анализ предметной области

**Наименование учреждения в соответствии с Уставом**: муниципальное образовательное учреждение дополнительного образования детей «Детско-юношеская спортивная школа №1».

**Аккредитация**: МОУ ДОД «Детско-юношеская спортивная школа №1» не имеет государственной аккредитации.

**1. Общие сведения о состоянии развития МОУ ДОД «Детско-юношеская спортивная школа №1».**

Детско-юношеская спортивная школа основана в 1947 году. Школа практикует опыт передачи наиболее талантливых учащихся в ШИСПОР г. Екатеринбурга. Ежегодно обучающиеся принимают участие во Всероссийских, областных, городских, районных соревнованиях и праздничных мероприятиях, конкурсах, фестивалях. Школа, является центром методической и спортивно-массовой работы по физической культуре в городе.

Тренеры-преподаватели ДЮСШ №1 проводят спартакиады школьников, участвуют в проведении Олимпиады по физической культуре.

ДЮСШ №1 имеет положительный опыт сотрудничества с образовательными учреждениями г. Алапаевска, СДЮСШОР. Школа поддерживает связь с ВУЗами и профессиональными колледжами Свердловской области, сотрудничает с учреждениями культуры, СМИ, общественными организациями города.

Учебно-воспитательный процесс в ДЮСШ №1 осуществляется шестнадцатью квалифицированными педагогами, имеющими большой опыт и стаж педагогической работы. Средней возраста педколлектива составляет 41 год. 14 педагогов аттестовано (88%), из них:

63% на 1 квалификационную категорию,

25% на 2 квалификационную категорию.

Таким образом, уровень квалификации педагогических и руководящих кадров достаточен для успешного решения стоящих перед ДЮСШ задач.

В ДЮСШ №1 обучается 587 человек, 26,7% от общего количества учащихся из неполных, многодетных, малообеспеченных семей и опекаемых детей.

За 3 года было подготовлено 452 человека, выполнивших юношеский разряд, 204 чел. – взрослый разряд, 5 чел. – КМС и 2 чел. – МС по спортивной гимнастике, легкой атлетике, волейболу.

В течение 3-х лет учащиеся ДЮСШ приняли участие в:

– 16-ти Всероссийских соревнованиях (82 чел.)

– 49-ти областных соревнованиях (384 чел.)

– 85-ти городских и районных соревнованиях (1128 чел.)

– 48-ми внутришкольных соревнованиях (918 чел.).

Подготовлено 132 победителя и призера Всероссийских и областных соревнований, 694 чел. – победителей и призеров городских и районных соревнований, 44 чел. – победителей городских фестивалей Детского творчества. Сборная девушек по волейболу является победителем и призером областных соревнований 2001, 2002, 2003 гг.; сборная юношей по волейболу – 2002, 2003 гг. Это свидетельствует о том, что образовательное учреждение работает результативно.

Административное управление обеспечивает оперативное управление образовательным процессом и реализует основные управленческие функции: анализ, планирование, организационно-содержательную деятельность, организацию контроля, самоконтроля, регулирование деятельности педагогического коллектива, осуществление мотивации.

Административно-управленческая работы школы обеспечивается следующим кадровым составом:

– директор,

– методист,

– заместитель директора по административно-хозяйственной части,

– старшие тренеры-преподаватели отделений.

Администрация школы осуществляет управление всеми структурными звеньями. Основной формой взаимодействия администрации и педагогического коллектива является обмен информацией, индивидуальные и групповые консультации, собеседования, совещания, сбор аналитической информации.

Администрация ДЮСШ №1 поощряет взаимоконтроль, способствует росту профессионального мастерства педагогов. В режиме самоконтроля в школе работают опытные специалисты, способные к объективной оценке своего труда.

**Основные деловые процессы подлежащие автоматизации.**

1. Зачисление учащихся

Входящие документы:

Заявление родителей, справка от врача

Исходящие документы:

Приказ о зачислении

2. Проведение соревнований.

Входящие документы:

положение о соревнованиях, календарь соревнований, приказ

Исходящие документы:

Отчет о соревнованиях (о выполнении разряда)

Приказ о присвоении разряда

Протокол (на каких именно соревнованиях получен разряд)

3. Выполнение нормативов.

Протокол (в нем фиксируются результаты сдачи нормативов)

4. Оздоровление учащихся.

Врач предоставляет данные о состоянии здоровья каждого учащегося по запросу тренера в свободной форме (средний уровень, высокий уровень,…). Тренер следит за данными в течении года и по запросу директора предоставляет отчет (в виде диаграммы) о том, как изменился уровень здоровья детей за определенный промежуток времени (то есть сколько было детей с низкими уровнем здоровья, высоким, и сколько стало, например, за год)

5. Аттестация тренеров

Входящие документы:

График прохождения аттестации, заявление на категорию

Исходящие документы:

Аттестационный лист (Документ по утвержденной категории)

6. Выпуск/отчисление учащихся.

Входящие документы:

Приказ об окончании

Исходящие документы:

Разрядная книжка.

**Пользователи системы:**

Методист. Может вносить, редактировать и удалять данные по всем таблицам. Так же генерирует отчеты

* по численности занимающихся за определенный период
* по характеристике тренерско-преподавательского состава за определенный год
* по списку подготовленных разрядников за определенный период
* по результатам участия учащихся МОУ ДОД ДЮСШ №1 в соревнованиях международного, всероссийского, регионального и областного уровней за определенный год
* о проведенных соревнованиях за определенный год
* о результатах деятельности МОУ ДОД ДЮСШ №1 за определенный год
* по качеству спортивной подготовки

– информационные справки о МОУ ДОД ДЮСШ №1 за определенный период

Директор. Имеет доступ ко всем таблицам. Генерирует и анализирует отчеты перечисленные выше.

Тренер. Просматривает данные по здоровью учащихся по соревнованиям, по аттестации тренеров. Генерирует и анализирует отчеты

* по результатам аттестации тренеров
* отчет по уровню физического здоровья учащихся

После сбора информационного требования было составлено универсальное отношение предметной области.

Универсальное отношение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | | **Название атрибута в предметной области** | Название в модели | Домен |
| 1 | | Ф.И.О. сотрудника | Ф.И.О.\_сотр | Текстовый |
| 2 | Дата рождения | | дата\_р\_с | Дата |
| 3 | | Телефон сотрудника домашинй | Телефон\_дом\_с | Числовой |
| 4 | | Паспортные данные | Пасп\_дан | Символьный |
| 5 | | Домашний адрес | Адрес\_с | Символьный |
| 6 | | Дата начала работы | Дата\_нач\_р | Дата |
| 7 | | Дата окончания работы | дата\_оконч\_р | Дата |
| 8 | | Должность | должность | Текстовый перечислимый (тренер, директор, концертмейстер, методист) |
| 9 | | Дата соревнований | Дата\_сор | Дата |
| 10 | | Образование | образование | Текстовый перечислимый (высшее, среднее,…) |
| 11 | | Этап подготовки учащихся | этап | Текстовый перечислимый |
| 12 | | № полиса ОМС | №полиса | Символьный |
| 13 | | Отделение | отделение | Текстовый перечислимый (легкая атлетика, спортивная гимнастика, волейбол) |
| 14 | | Пол | пол | Текстовый перечислимый (м, ж) |
| 15 | | Почетное звание | звание | Тестовый перечислимый (заслуженный работник общего образования,…) |
| 16 | | Ф.И.О. учащегося | Ф.И.О.\_уч | Текстовый |
| 17 | | Дата рождения | Дата\_р\_уч | Дата |
| 18 | | Телефон учащегося домашний | Тел\_дом\_уч | Числовой |
| 19 | | Адрес | Адрес\_уч | Символьный |
| 20 | | Ф.И.О. мамы | Ф.И.О.\_м | Текстовый |
| 21 | | Ф.И.О. папы | Ф.И.О.\_п | Текстовый |
| 22 | | Дата зачисления | Дата\_зач | Дата |
| 23 | | Дата окончания | Дата\_оконч | Дата |
| 24 | | Уровень здоровья | Ур\_здор | Текст перечислимый (высокий, средний, ниже среднего, низкий) |
| 25 | | Разряд | разряд | Текст перечислимый |
| 26 | | Дата присвоения разряда | Дата \_разр | Дата |
| 27 | | Номер приказа о присвоении разряда | Ном\_прик | символьный |
| 28 | | Группа | группа | Символьный перечислимый |
| 29 | | Город соревнований | город | Текст перечислимый (Екб, Тагил, Алапаевск, Челябинск) |
| 30 | | Ранг соревнований | ранг | Текстовый |
| 31 | | Программа соревнований | Прогр\_сор | Текстовый перечислимый (общая, произвольная) |
| 32 | | Результат выступлений | место | Числовой |
| 33 | | Награда за соревнования | награда | Символьный перечислимый (диплом 1 степени,…) |
| 34 | | Дата аттестации тренера | Дата\_атт\_трен | Дата |
| 35 | | Заявленная категория | Кат\_заяв | числовой |
| 36 | | Утвержденная категория | Утв\_кат | числовой |
| 37 | | Месяц проведения аттестации | месяц | Дата |
| 38 | | Количество баллов, полученных на соревновниях | Кол\_балл | Числовой (Вычисляется судьями на соревнованиях) |
| 39 | | № приказа о зачислении | №прик\_зач | символьный |
| 40 | | № приказа об отчислении | №прик\_отч | символьный |
| 41 | | № приказа о выдачи разрядной книжки (документ об окончании) | №прик\_разр\_к | символьный |
| 42 | | Уровень соревнований | Ур\_сор | Текстовый перечислимый (международные, всероссийские, областные,…) |
| 43 | | Наименование мероприятий | меропр | Текстовый перечислимый (Открыт уч-трен занятие по спорт гимнастике,…) |
| 44 | | Этап подготовки | этап | Текстовый перечислимый (спорт – оздор, уч-трен,…) |
| 45 | | Штатный сотрудник | Штат\_сотр | логический |
| 46 | | Соц. статус семей учащихся | Соц\_стат | Текстовый, перечислимый (многодет,…) |
| 47 | | Характер образовательной программы | Хар\_прог | Текстовый перечислимый (общеразвив, глубок изуч предм) |
| 48 | | Учебная нагрузка (ставка) | ставка | Числовой |
| 49 | | Лучший учащийся | Лучший\_уч | Логический- |
| 50 | | Финансирование соревнований | Фин\_сор | Логический |
| 51 | | Образовательные учереждения | Обр\_уч | Символьный перечислимый (МОУ СОШ №1,…)  (нужен для сведений об учащихся) |
| 52 | | Табельный номер сотрудника | Таб\_ном | числовой |
| 53 | | Категория | категория | Текстовый, перечислимый |
| 54 | | Дата зачисления на должность | дата\_зач\_должн | дата |
| 55 | | Год в котором учащийся был лучшим | год\_л\_уч | дата |
| 56 | | Ф.И.О. экспертов | Ф.И.О.\_экспертов | Текстовый |
| 57 | | Вид упражнения на соревнованиях | Вид | Текстовый перечислимый |
| 58 | | Дата получения показателей здоровья | Дата\_з | дата |
| 59 | | Участие спортсменов в судейской коллегии | судейство | логический |
| 60 | | Участие спортсменов в показательных выступлениях | Показ\_выст | логический |
| 61 | | Участие спортсменов в фестивалях | фестиваль | логический |
| 62 | | Участие спортсменов в конкурсах | конкурс | логический |
| 63 | | Публикации спортсменов в СМИ | СМИ | логический |
| 64 | | Телефон сотрудника сотовый | Тел\_сот\_с | числовой |
| 65 | | Телефон учащегося сотовый | Тел\_сот\_уч | числовой |
| 66 | | Месяц | Месяц | Текстовый, перечислимый |
| 67 | | Дата проведения соревнования | Дата\_сор | дата |

**Ограничения предметной области:**

1. У сотрудника может быть только один адрес и один телефон, телефон может принимать NULL-значение

2. Номер страхового полиса уникален, у одного сотрудника может быть только один полис.

3. Паспортные данные уникальны, у сотрудника может быть только один паспорт

4. Сотрудник может совмещать несколько должностей.

5. Каждый тренер ведет занятия только в одном отделении.

6. Один учащийся в заданный момент времени занимается только в одном отделении и одной группе.

7. В одном городе может проходить несколько соревнований

8. У тренера может быть несколько званий

9. В один день может быть несколько соревнований только у разных отделений.

10. Одним приказом можно присвоить только массовые разряды группе учащихся, занимающихся у одного тренера

11. Некоторые из сотрудников могут быть экспертами.

12. В одной группе может вести занятия только один тренер

13. Один тренер может вести занятия в нескольких группах.

14. У одной семьи может быть несколько социальных статусов

15. Учащийся может быть отчислен несколько раз

16. Учащийся может быть зачислен несколько раз

17. У одного соревнования может быть несколько программ

## Далее была спроектирована и описана на ЯИМ инфологическая модель данных.

## 2.2 Проектирование инфологической модели

**Справочники (стержни):**

СПИСОК СОТРУДНИКОВ (таб\_ном, Ф.И.О.\_сотр, дата\_р\_с, телефон\_с, пасп\_дан, адрес\_с, дата\_нач\_р, дата\_оконч\_р, №полиса, пол, ставка, штат\_сотр);

СПИСОК УЧАЩИХСЯ (ном\_уч, Ф.И.О.\_уч, дата\_р\_уч, тел\_уч, адрес\_уч, Ф.И.О.\_м, Ф.И.О.\_п, пол);

СОЦ СТАТУС СЕМЬИ (код\_стат, соц\_стат);

ЛУЧШИЕ УЧАЩИЕСЯ (ном\_уч, год\_л\_уч, лучший\_уч) [список учащихся];

ОБРАЗОВАТЕЛЬНЫЕ УЧЕРЕЖДЕНИЯ (код\_обр\_уч, обр\_уч);

ОТДЕЛЕНИЯ (код\_отделения, отделение);

ПОЧЕТНЫЕ ЗВАНИЯ (Код\_звания, звание);

ДОЛЖНОСТИ (код\_долж, должность);

ГОРОДА (код\_города, город);

УРОВЕНЬ СОРЕВНОВАНИЙ (код\_ур, ур\_сор);

РАЗРЯД (код\_разр, разряд);

УРОЕНЬ ЗДОРОВЬЯ (код\_ур\_з, уровень здор);

КАТЕГОРИИ (код\_кат, категория);

ПРОГРАММА СОРЕВНОВАНИЙ (код\_прогр\_сор, прогр, сор);

ОБРАЗОВАНИЕ (код\_образ, образование);

**Обозначения:**

СПИСОК ГРУПП (код\_группы, группа, код\_отделения,) [ОТДЕЛЕНИЯ];

ОБРАЗОВАНИЕ СОТРУДНИКОВ (таб\_ном, код\_образ) [СПИСОК СОТРУДНИКОВ] [ОБРАЗОВАНИЕ];

ЗАЧИСЛЕНИЕ УЧАЩИХСЯ (код\_зач, ном\_уч, дата\_зач, №прик\_зач) [СПИСОК УЧАЩИХСЯ];

ВЫПУСК УЧАЩИХСЯ (код\_зач, дата\_оконч, №прик\_отч, №прик\_разр\_к) [СПИСОК УЧАЩИХСЯ];

СЕМЬЯ УЧАЩЕГОСЯ (ном\_уч, код\_стат) [СПИСОК УЧАЩИХСЯ] [СОЦ СТАТУС СЕМЬИ];

АТТЕСТАЦИЯ ТРЕНЕРОВ (таб\_ном, код\_эксп, код\_заяв\_кат, код\_утв\_кат, дата\_атт\_трен) [СПИСОК СОТРУДНИКОВ] [КАТЕГОРИИ];

СОРЕВНОВАНИЯ (Код\_сор, код\_города, ранг, код\_ур, код\_прогр\_сор, фин\_сор, дата\_сор) [ГОРОДА] [УРОВЕНЬ СОРЕВНОВАНИЙ] [ПРОГРАММА СОРЕВНОВАНИЙ];

КОМПЛЕКТОВАНИЕ УЧ (код\_группы, ном\_уч) [СПИСОК ГРУПП] [СПИСОК УЧАЩИХСЯ];

РЕЗУЛЬТАТ ВЫСТУПЛЕНИЙ НА СОРЕВНОВАНИЯХ (ном\_уч, код\_сор, кол\_бал, таб\_ном) [СПИСОК УЧАЩИХСЯ] [СПИСОК СОТРУДНИКОВ] [СОРЕВНОВАНИЯ];

СПОРТСМЕНЫ – РАЗРЯДНИКИ (код\_с\_р, ном\_уч, код\_разр, дата\_разр, таб\_ном, ном\_прик) [СПИСОК УЧАЩИХСЯ] [СПИСОК СОТРУДНИКОВ] [РАЗРЯД];

ОЗДОРОВЛЕНИЕ УЧАЩИХСЯ (ном\_уч, дата\_з, код\_ур\_з) [СПИСОК УЧАЩИХСЯ] [УРОВЕНЬ ЗДОРОВЬЯ];

ЗАСЛУЖЕННЫЕ РАБОТНИКИ (таб\_ном, код\_звания,) [СПИСОК СОТРУДНИКОВ] [ПОЧЕТНЫЕ ЗВАНИЯ];

ДОЛЖНОСТИ СОТРУДНИКОВ (таб\_ном, код\_долж, дата\_зач\_должн) [СПИСОК СОТРУДНИКОВ] [ДОЛЖНОСТИ];

Далее было проектирование базовых реляционных таблиц.

## 2.3 Проектирование даталогической модели

**Описание таблиц на SQL.**

CREATE TABLE СПИСОК СОТРУДНИКОВ

(Таб\_ном INTEGER PRIMARY KEY,

Ф.И.О.\_сотр TEXT(30) NOT NULL,

Дата\_р\_с DATE NOT NULL,

Телефон\_с CHAR(10),

Пасп\_дан CHAR (100) UNIQUE NOT NULL,

Адрес\_с CHAR(100) NOT NULL,

Дата\_нач\_р DATE NOT NULL,

Дата\_оконч\_р DATE,

№полиса CHAR (10) UNIQUE,

ставка

штат\_сотр

пол TEXT (1) NOT NULL,

CHECK (пол IN («м», «ж»)));

CREATE TABLE СПИСОК УЧАЩИХСЯ

(ном\_уч INTEGER PRIMARY KEY,

Ф.И.О.\_уч TEXT(30) NOT NULL,

дата\_р\_уч DATE NOT NULL,

тел\_уч CHAR(10),

адрес\_уч CHAR(100) NOT NULL,

Ф.И.О.\_м TEXT(30),

Ф.И.О.\_п TEXT(30),

пол TEXT (1) NOT NULL,

CHECK (пол IN («м», «ж»)));

CREATE TABLE СОЦ СТАТУС СЕМЬИ

(код\_стат INTEGER PRIMARY KEY,

соц\_стат TEXT(20);

CHECK (соц\_стат IN («многодетная семья», «неполная семья», «опекаемые дети», «состоит на учете в ОППН», «малоимущая семья», «неблагополучная семья»)));

CREATE TABLE ОБРАЗОВАТЕЛЬНЫЕ УЧЕРЕЖДЕНИЯ

(код\_обр\_уч INTEGER PRIMARY KEY,

обр\_уч CHAR(10)

CHECK (обр\_уч IN («МОУ СОШ №1»,» МОУ СОШ №2»,» МОУ СОШ №3»,» МОУ СОШ №4»,» МОУ СОШ №5»,» МОУ СОШ №6»,» МОУ СОШ №8»,» МОУ СОШ №10», «МОУ СОШ №12», «МОУ СОШ №15», «Профессиональные и высшие учебные заведения»,» МДОУ»)));

CREATE TABLE ОТДЕЛЕНИЯ

(код\_отделения INTEGER PRIMARY KEY,

отделение TEXT(20);

CHECK (отделение IN («спортивная гимнастика»,» лёгкая атлетика»,» волейбол»)));

CREATE TABLE ПОЧЕТНЫЕ ЗВАНИЯ

(Код\_звания INTEGER PRIMARY KEY,

звание TEXT(30)

CHECK (звание IN («Заслуженный тренер России»,» Почётный работник общего образования»,» Отличник физической культуры и спорта РФ»,» Мастер спорта международного класса»,» Мастер спорта РФ»,» Знак «80-лет Министерству по физической культуре, спорту и туризму Свердловской области»)));

CREATE TABLE ДОЛЖНОСТИ

(код\_долж INTEGER PRIMARY KEY,

должность TEXT(15)

CHECK (должность IN («тренер», «методист», «директор»)));

CREATE TABLE ГОРОДА

(код\_города INTEGER PRIMARY KEY,

города TEXT(20)

CHECK (города IN («Алапаевск», «Екатеринбург», «Челябинск», «Нижний Тагил»)));

CREATE TABLE УРОВЕНЬ СОРЕВНОВАНИЙ

(код\_ур INTEGER PRIMARY KEY,

ур\_сор TEXT(20)

CHECK (ур\_сор IN («международные», «всероссийские», «зональные», «областные», «районные», «городские»)));

CREATE TABLE РАЗРЯД

(код\_разд INTEGER PRIMARY KEY,

разряд TEXT(20)

CHECK (разряд IN («первый юношеский», «второй юношеский», «третий юношеский», «первый взрослый», «второй взрослый», «третий взрослый», «четвертый взрослый», «пятый взрослый», «кандидат в мастера спорта», «мастер спорта России», «мастер спорта России международного класса»)));

CREATE TABLE УРОВЕНЬ ЗДОРОВЬЯ

(код\_ур\_з INTEGER PRIMARY KEY,

Ур\_здор TEXT

CHECK (группа IN («низкий», «ниже среднего», «средний», «выше среднего», «высокий»)));

CREATE TABLE КАТЕГОРИИ

(код\_кат INTEGER PRIMARY KEY,

категория CHAR(10)

CHECK (группа IN («первая», «вторая», «высшая»)));

CREATE TABLE ПРОГРАММА СОРЕВНОВАНИЙ

(код\_прогр\_сор INTEGER PRIMARY KEY,

прогр\_сор TEXT

CHECK (прогр\_сор IN («произвольная», «общая»);

ОБРАЗОВАНИЕ СОТРУДНИКОВ (таб\_ном, код\_образ) [список сотрудников] [образование];

CREATE TABLE ОБРАЗОВАНИЕ СОТРУДНИКОВ

(таб\_ном INTEGER

код\_образ INTEGER)

PRIMARY KEY (таб\_ном, код\_образ),

FOREIN KEY (таб\_ном) REFERENCES СПИСОК СОТРУДНИКОВ

ON DELETE CASCADE

ON UPDATE CASCADE

FOREIN KEY (код\_образ) REFERENCES ОБРАЗОВАНИЕ

ON DELETE RESTRICT

ON UPDATE CASCADE

CREATE TABLE СПИСОК ГРУПП

(код\_группы INTEGER PRIMARY KEY,

код\_отделения INTEGER PRIMARY KEY,

группа CHAR(10))

PRIMARY KEY (код\_группы),

FOREIN KEY (код\_отделения) REFERENCES ОТДЕЛЕНИЯ

ON DELETE CASCADE

ON UPDATE CASCADE

CHECK (группа IN («СОГ», «Н/П-1г», «Н/П-1в», «Н/П-1а», «Н/П-2г», «Н/П-2в», «Н/П-2а», «Н/П-3г», «Н/П-3в», «Н/П-3а», «УТГ-1г», «УТГ-1в», «УТГ-1а», «УТГ-2г», «УТГ-2в», «УТГ-2а», «УТГ-3г», «УТГ-3в», «УТГ-3а», «УТГ-4г», «УТГ-4в», «УТГ-4а», «УТГ-5г», «УТГ-5в», «УТГ-5а», «С/С-1г», «С/С-1в», «С/С-1а», «С/С-2г», «С/С-2в», «С/С-2а», «ВСМ»)));

CREATE TABLE ВЫПУСК УЧАЩИХСЯ

(ном\_учINTEGER NOT NULL,

Дата\_окончDATA

№прик\_отч TEXT

№прик\_разр\_к TEXT),

PRIMARY KEY (ном\_уч),

FOREIN KEY (ном\_уч) REFERENCES СПИСОК УЧАЩИХСЯ

ON DELETE RESTRICT

ON UPDATE CASCADE

CREATE TABLE ЗАЧИСЛЕНИЕ УЧАЩИХСЯ

(ном\_учINTEGER NOT NULL,

код\_обр\_уч INTEGER NOT NULL,

Дата\_зач DATA

№прик\_зач TEXT),

PRIMARY KEY (ном\_уч)

FOREIN KEY (ном\_уч) REFERENCES СПИСОК УЧАЩИХСЯ

ON DELETE RESTRICT

ON UPDATE CASCADE

FOREIN KEY (код\_обр\_уч) REFERENCES ОБРАЗОВАТЕЛЬНЫЕ УЧЕРЕЖДЕНИЯ

ON DELETE RESTRICT

ON UPDATE CASCADE

CREATE TABLE СЕМЬЯ УЧАЩЕГОСЯ

(ном\_уч INTEGER NOT NULL

код\_стат INTEGER NOT NULL)

PRIMARY KEY (ном\_уч),

FOREIN KEY (ном\_уч) REFERENCES СПИСОК УЧАЩИХСЯ

ON DELETE RESTRICT

ON UPDATE CASCADE

FOREIN KEY (код\_стат) REFERENCES СОЦ СТАТУС СЕМЬИ

ON DELETE RESTRICT

ON UPDATE CASCADE;

CREATE TABLE АТТЕСТАЦИЯ ТРЕНЕРОВ

(таб\_ном INTEGER NOT NULL

код\_меропр INTEGER NOT NULL

месяц DATA

кат\_заяв INTEGER

утв\_кат INTEGER

дата\_атт\_трен DATA),

PRIMARY KEY (таб\_ном, код\_меропр),

FOREIN KEY (таб\_ном) REFERENCES СПИСОК СОТРУДНИКОВ,

ON DELETE CASCADE

ON UPDATE CASCADE

FOREIN KEY (кат\_заяв) REFERENCES КАТЕГОРИИ,

ON DELETE RESTRICT

ON UPDATE CASCADE

FOREIN KEY (утв\_кат) REFERENCES КАТЕГОРИИ,

ON DELETE RESTRICT

ON UPDATE CASCADE

CREATE TABLE СОРЕВНОВАНИЯ

(код\_сор INTEGER NOT NULL,

код\_города INTEGER NOT NULL

ранг TEXT

код\_ур INTEGER NOT NULL

прогр\_сор TEXT

фин\_сор LOGICAL

Дата\_сор DATE),

PRIMARY KEY (код\_сор),

FOREIN KEY (код\_гор) REFERENCES ГОРОДА

ON DELETE RESTRICT

ON UPDATE CASCADE

FOREIN KEY (код\_ур) REFERENCES УРОВЕНЬ СОРЕВНОВАНИЙ;

ON DELETE RESTRICT

ON UPDATE CASCADE

FOREIN KEY (код\_прогр\_сор) REFERENCES ПРОГРАММА СОРЕВНОВАНИЙ;

ON DELETE RESTRICT

ON UPDATE CASCADE

CREATE TABLE КОМПЛЕКТОВАНИЕ УЧ

(код\_группы INTEGER NOT NULL

ном\_уч INTEGER NOT NULL),

PRIMARY KEY (код\_группы, ном\_уч),

FOREIN KEY (ном\_уч) REFERENCES СПИСОК УЧАЩИХСЯ,

ON DELETE RESTRICT

ON UPDATE CASCADE

FOREIN KEY (код\_группы) REFERENCES СПИСОК ГРУПП;

ON DELETE RESTRICT

ON UPDATE CASCADE

CREATE TABLE РЕЗУЛЬТАТ ВЫСТУПЛЕНИЙ НА СОРЕВНОВАНИЯХ

(ном\_уч INTEGER NOT NULL

код\_сор INTEGER NOT NULL

кол\_бал INTEGER

таб\_ном INTEGER NOT NULL),

PRIMARY KEY (ном\_уч, код\_сор),

FOREIN KEY (таб\_ном) REFERENCES СПИСОК СОТРУДНИКОВ,

ON DELETE RESTRICT

ON UPDATE CASCADE

FOREIN KEY (код\_сор) REFERENCES СОРЕВНОВАНИЯ,

ON DELETE RESTRICT

ON UPDATE CASCADE

FOREIN KEY (ном\_уч) REFERENCES СПИСОК УЧАЩИХСЯ;

ON DELETE RESTRICT

ON UPDATE CASCADE

CREATE TABLE СПОРТСМЕНЫ – РАЗРЯДНИКИ

(код\_с\_р INTEGER NOT NULL

ном\_уч INTEGER NOT NULL

ном\_п\_п INTEGER NOT NULL

разряд TEXT

код\_отделения INTEGER NOT NULL

дата\_разр DATA

таб\_ном INTEGER NOT NULL

ном\_прик INTEGER),

PRIMARY KEY (код\_с\_р),

FOREIN KEY (таб\_ном) REFERENCES СПИСОК СОТРУДНИКОВ,

ON DELETE RESTRICT

ON UPDATE CASCADE

FOREIN KEY (ном\_уч) REFERENCES СПИСОК УЧАЩИХСЯ

ON DELETE RESTRICT

ON UPDATE CASCADE

FOREIN KEY (код\_отделения) REFERENCES ОТДЕЛЕНИЯ

ON DELETE RESTRICT

ON UPDATE CASCADE

CREATE TABLE ОЗДОРОВЛЕНИЕ УЧАЩИХСЯ

(ном\_уч INTEGER NOT NULL

дата\_з DATA

код\_ур\_здор TEXT),

PRIMARY KEY (ном\_уч, дата\_з),

FOREIN KEY (ном\_уч) REFERENCES СПИСОК УЧАЩИХСЯ

ON DELETE CASCADE

ON UPDATE CASCADE

FOREIN KEY (код\_ур\_з) REFERENCES СПИСОК УЧАЩИХСЯ

ON DELETE RESTRICT

ON UPDATE RESTRICT

CREATE TABLE ЗАСЛУЖЕННЫЕ РАБОТНИКИ

(таб\_ном INTEGER NOT NULL

код\_звания INTEGER NOT NULL),

PRIMARY KEY (таб\_ном, код\_звания

FOREIN KEY (таб\_ном) REFERENCES СПИСОК СОТРУДНИКОВ

ON DELETE CASCADE

ON UPDATE CASCADE

FOREIN KEY (код\_звания) REFERENCES ЗВАНИЯ

ON DELETE RESTRICT

ON UPDATE CASCADE

CREATE TABLE ДОЛЖНОСТИ СОТРУДНИКОВ

(таб\_ном INTEGER NOT NULL

Ф.И.О.\_сотр TEXT

Код\_долж INTEGER NOT NULL

Должность TEXT

Дата\_зач\_должн DATA),

PRIMARY KEY (таб\_ном, код\_долж

FOREIN KEY (таб\_ном) REFERENCES СПИСОК СОТРУДНИКОВ

ON DELETE CASCADE

ON UPDATE CASCADE

FOREIN KEY (код\_долж) REFERENCES ДОЛЖНОСТИ

ON DELETE RESTRICT

ON UPDATE CASCADE

## 2.4 Проверка нормализации спроектированных таблиц

1. Таблицы, состоящие из 2х атрибутов находятся в НФБК.

Это таблицы:

СОЦ СТАТУС СЕМЬИ (код\_стат, соц\_стат);

ОБРАЗОВАТЕЛЬНЫЕ УЧЕРЕЖДЕНИЯ (код\_обр\_уч, обр\_уч);

ОТДЕЛЕНИЯ (код\_отделения, отделение);

ПОЧЕТНЫЕ ЗВАНИЯ (Код\_звания, звание);

ДОЛЖНОСТИ (код\_долж, должность);

ГОРОДА (код\_города, город);

УРОВЕНЬ СОРЕВНОВАНИЙ (код\_ур, ур\_сор);

РАЗРЯД (код\_разр, разряд);

УРОЕНЬ ЗДОРОВЬЯ (код\_ур\_з, уровень здор);

КАТЕГОРИИ (код\_кат, категория);

ПРОГРАММА СОРЕВНОВАНИЙ (код\_прогр\_сор, прогр, сор);

ОБРАЗОВАНИЕ (код\_образ, образование);

КОМПЛЕКТОВАНИЕ ТРЕНЕР (таб\_ном, код\_группы) [список сотрудников] [список групп];

КОМПЛЕКТОВАНИЕ УЧ (код\_группы, ном\_уч) [список групп] [список учащихся];

ОБРАЗОВАНИЕ СОТРУДНИКОВ (Ф.И.О.\_сотр, образование) [список сотрудников] [образование];

СЕМЬЯ УЧАЩЕГОСЯ (ном\_уч, код\_стат) [список учащихся] [Соц статус семьи];

ЗАСЛУЖЕННЫЕ РАБОТНИКИ (таб\_ном, код\_звания,) [список сотрудников] [почетные звания];

2. Все таблицы, имеющие несоставные PK, находятся во 2НФ. Проверим, нет ли в них транзитивных зависимостей.

СПИСОК СОТРУДНИКОВ (таб\_ном, Ф.И.О.\_сотр, дата\_р\_с, телефон\_с, пасп\_дан, адрес\_с, дата\_нач\_р, дата\_оконч\_р, №полиса, пол, ставка, штат\_сотр);

СПИСОК УЧАЩИХСЯ (ном\_уч, Ф.И.О.\_уч, дата\_р\_уч, тел\_уч, адрес\_уч, Ф.И.О.\_м, Ф.И.О.\_п, пол);

Очевидно, что таблицы СПИСОК СОТРУДНИКОВ и СПИСОК УЧАЩИХСЯ находятся в НФБК.

СПИСОК ГРУПП (группа, отделениие) [отделения];

ФЗ: отделение 🡪 группа. Отделение – возможный ключ. Значит таблица находится в НФБК

СОРЕВНОВАНИЯ (город, ранг, ур\_сор, прогр\_сор, фин\_сор, дата\_сор) [города] [уровень соревнований] [программа соревнований];

ФЗ: (город, ур\_сор, прогр\_сор, дата\_сор)🡪ранг, (город, ур\_сор, прогр\_сор, дата\_сор)🡪фин\_сор. Но (город, ур\_сор, прогр\_сор, дата\_сор) – возможный ключ. Значит все ФЗ в таблице сводятся к полной ФЗ от возможного ключа. Т.о. таблица СОРЕВНОВАНИЯ находится в НФБК.

СПОРТСМЕНЫ – РАЗРЯДНИКИ (Ф.И.О.\_уч, разряд, дата\_разр, Ф.И.О.\_сот, ном\_прик) [список учащихся] [список сотрудников] [разряд];

ФЗ: (Ф.И.О.\_уч, разряд, дата\_разр) 🡪Ф.И.О.\_сот, (Ф.И.О.\_уч, разряд, дата\_разр) 🡪ном\_прик. Но (Ф.И.О.\_уч, разряд, дата\_разр) – возможный ключ. Значит все ФЗ в таблице сводятся к полной ФЗ от возможного ключа. Т.о. таблица СПОРТСМЕНЫ – РАЗРЯДНИКИ находится в НФБК.

3. Таблицы с составными PK.

ЛУЧШИЕ УЧАЩИЕСЯ (ном\_уч, год\_л\_уч, лучший\_уч) [список учащихся];

ФЗ: (ном\_уч, год\_л\_уч) 🡪лучший учащийся.

(ном\_уч, год\_л\_уч) – составной ключ. Таблица находится в НФБК.

ЗАЧИСЛЕНИЕ УЧАЩИХСЯ (ном\_уч, дата\_зач, №прик\_зач) [список учащихся];

ФЗ: (ном\_уч, дата\_зач) 🡪№прик\_зач

(ном\_уч, дата\_зач) – составной ключ. Таблица находится в НФБК.

ВЫПУСК УЧАЩИХСЯ (ном\_уч, дата\_зач, дата\_оконч, №прик\_отч, №прик\_разр\_к) [список учащихся];

ФЗ: (ном\_уч, дата\_зач) 🡪дата\_оконч, (ном\_уч, дата\_зач) 🡪№прик\_отч, (ном\_уч, дата\_зач) 🡪№прик\_разр\_к

(ном\_уч, дата\_зач) – возможный составной ключ. Таблица находится в НФБК.

АТТЕСТАЦИЯ ТРЕНЕРОВ (Ф.И.О.\_сотр, Ф.И.О.\_эксп, заяв\_кат, утв\_кат, дата\_атт\_трен) [список сотрудников] [категории];

ФЗ: (Ф.И.О.\_сотр, дата\_атт\_трен) 🡪Ф.И.О.\_эксп, (Ф.И.О.\_сотр, дата\_атт\_трен) 🡪заяв\_кат, (Ф.И.О.\_сотр, дата\_атт\_трен) 🡪утв\_кат.

(Ф.И.О.\_сотр, дата\_атт\_трен) – возможный составной ключ. Таблица находится в НФБК.

РЕЗУЛЬТАТ ВЫСТУПЛЕНИЙ НА СОРЕВНОВАНИЯХ (Ф.И.О.\_уч, код\_сор, кол\_бал, Ф.И.О.\_трен) [список учащихся] [список сотрудников] [соревнования];

ФЗ: (Ф.И.О.\_уч, код\_сор) 🡪кол\_бал, (Ф.И.О.\_уч, код\_сор) 🡪Ф.И.О.\_трен.

(Ф.И.О.\_уч, код\_сор) – возможный составной ключ. Таблица находится в НФБК.

СПОРТСМЕНЫ – РАЗРЯДНИКИ (Ф.И.О.\_уч, разряд, дата\_разр, Ф.И.О.\_сот, ном\_прик) [список учащихся] [список сотрудников] [разряд];

ФЗ: (Ф.И.О.\_уч, разряд, дата\_разр) 🡪Ф.И.О.\_сот, (Ф.И.О.\_уч, разряд, дата\_разр) 🡪ном\_прик

(Ф.И.О.\_уч, разряд, дата\_разр) – возможный составной ключ. Таблица находится в НФБК.

ОЗДОРОВЛЕНИЕ УЧАЩИХСЯ (Ф.И.О.\_уч, дата\_з, ур\_здор) [список учащихся] [уровень здоровья];

ФЗ: (Ф.И.О.\_уч, дата\_з) 🡪ур\_з.

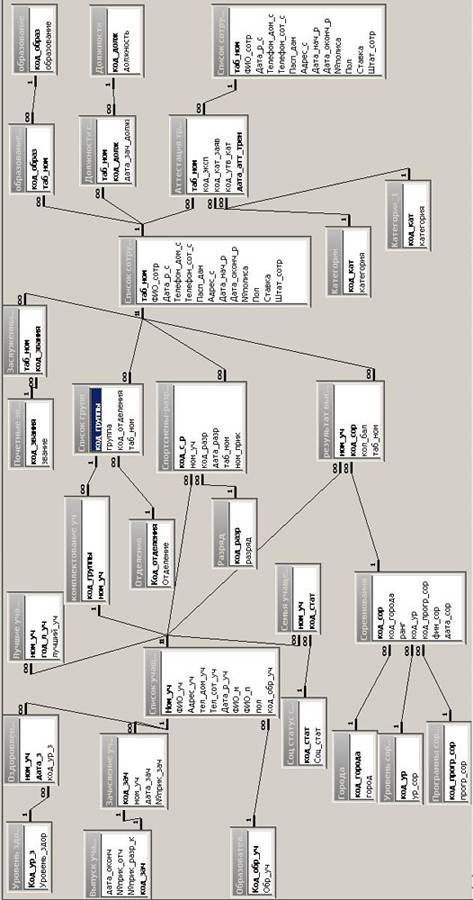
Таблица находится в НФБК.

ДОЛЖНОСТИ СОТРУДНИКОВ (Ф.И.О.\_сот должность, дата\_зач\_должн) [список сотрудников] [должности];

ФЗ: (Ф.И.О.\_сот, должность) 🡪дата\_зач\_должн.

Таблица находится в НФБК.

Схема данных



## 2.4 Описание реализации

### 

### 2.4.1 Обоснование выбора СУБД

Microsoft Access 2003 представляет собой СУБД которая предназначена для создания настольных приложений и приложений клиент / сервер, работающих под управлением Windows 2000 и XP.

1. Access предоставляет разработчикам средства, необходимые для создания развитых сложных баз данных, легко интегрирующихся со структурой данных предприятия, обеспечивая при этом прямую и обратную совместимость с существующими и новыми решениями. Access 2003 предоставляет средства для создания решений, интегрирующих и использующих преимущества Internet

2. Способы анализа данных пользователями с помощью динамических сводных таблиц и сводных диаграмм,

3. Страницы доступа к данным, позволяющие пользователям распространять приложения корпоративных баз данных в Internet.

4. Система поддерживает все средства для создания, введения и защиты от несанкционированного доступа к данным

5. Система позволяет импортировать и экспортировать файлы многих известных форматов

6. Одна из важных возможностей access – создание запросов

Таким образом, средств Access 2003 достаточно для реализации моего проекта. Кроме того, в организации приобретение лицензии другой СУБД в ближайшее время не планируется.

### 2.4.2 Описание интерфейса

Для информационная система для поддержки учебной деятельности ДЮСШ №1 г. Алапаевск был спроектирован многостраничный интерфейс в соответствии с категориями пользователей. Такой интерфейс позволяет просто переводить информационную систему в многопользовательский вариант технологии клиент-сервер.

Главная форма:



Рис. 2.1

Директор:



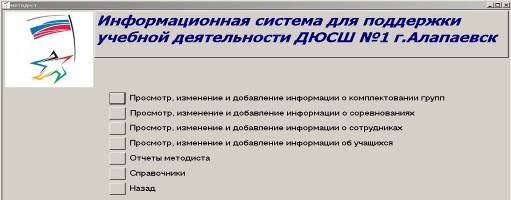
Автоматизированное рабочее место директора, рис 2.2

Просмотр и печать отчетов директора:



Форма для просмотра и печати директоров, рис 2.3

Методист:



Автоматизированное рабочее место методиста, рис 2.4

Отчеты методиста:



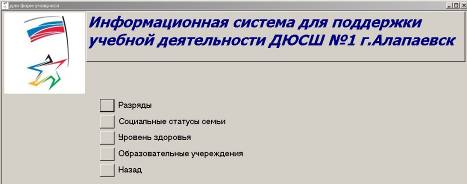
Отчеты методиста"рис 2.5

Форма «Справочники»:



Справочники, рис 2.6

Форма «Для форм учащихся»:



Для форм учащихся, рис 2.7

Форма «Для форм сотрудников»:



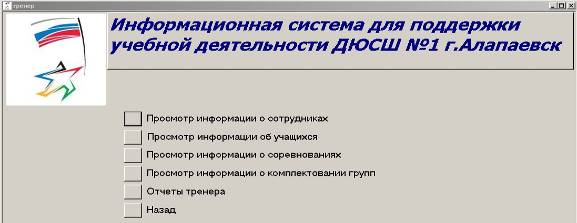
Для форм сотрудников, рис 2.8

Форма «Для форм соревнований»:



Для форм соревнований, рис 2.9

Тренер:



Автоматизированное рабочее место тренера, рис 2.4

Форма «Просмотр и печать отчетов тренера»:



Отчеты тренера, рис 2.10

**2.4.3 Проектирование запросов**

1. Запрос «аттестация тренеров запрос». Для отчета «Аттестация тренеров».

SELECT [Аттестация тренеров]. таб\_ном, [Список сотрудников]. Ф.И.О.\_сотр, [Аттестация тренеров]. дата\_атт\_трен, [Аттестация тренеров]. код\_эксп, [Список сотрудников\_1].Ф.И.О.\_сотр, [Аттестация тренеров]. код\_кат\_заяв, Категории.категория, [Аттестация тренеров]. код\_утв\_кат, Категории\_1.категория

FROM [Список сотрудников] AS [Список сотрудников\_1] INNER JOIN (Категории AS Категории\_1 INNER JOIN ([Список сотрудников] INNER JOIN (Категории INNER JOIN [Аттестация тренеров] ON Категории.код\_кат = [Аттестация тренеров]. код\_кат\_заяв) ON [Список сотрудников]. таб\_ном = [Аттестация тренеров]. таб\_ном) ON Категории\_1.код\_кат = [Аттестация тренеров]. код\_утв\_кат) ON [Список сотрудников\_1].таб\_ном = [Аттестация тренеров]. код\_эксп;

2. «Запрос должности сотрудников». Для связанной формы «должности сотрудников».

SELECT [Должности сотрудников]. таб\_ном, [Список сотрудников]. Ф.И.О.\_сотр, [Должности сотрудников]. код\_долж, Должности.должность, [Должности сотрудников]. дата\_зач\_должн

FROM [Список сотрудников] INNER JOIN (Должности INNER JOIN [Должности сотрудников] ON Должности.код\_долж = [Должности сотрудников]. код\_долж) ON [Список сотрудников]. таб\_ном = [Должности сотрудников]. таб\_ном;

3. Запрос «Вспомог запрос» для запроса «Список учащихся за определенный период».

SELECT [Список учащихся]. Ф.И.О.\_уч, [Зачисление учащихся]. дата\_зач, [Выпуск учащихся]. дата\_оконч

FROM [Список учащихся] INNER JOIN ([Зачисление учащихся] LEFT JOIN [Выпуск учащихся] ON [Зачисление учащихся]. код\_зач = [Выпуск учащихся]. код\_зач) ON [Список учащихся]. Ном\_уч = [Зачисление учащихся]. ном\_уч;

4. «Заслуженные работники запрос». Для связанной формы «заслуженные работники».

SELECT [Список сотрудников]. Ф.И.О.\_сотр, [Почетные звания]. звание

FROM [Список сотрудников] INNER JOIN ([Почетные звания] INNER JOIN [Заслуженные работники] ON [Почетные звания]. код\_звания = [Заслуженные работники]. код\_звания) ON [Список сотрудников]. таб\_ном = [Заслуженные работники]. таб\_ном;

5. Запрос «затч-отч». Для отчета «учет поступлений и выпуска учащихся».

SELECT [Список учащихся]. Ном\_уч, [Список учащихся]. Ф.И.О.\_уч, [Зачисление учащихся]. дата\_зач, [Зачисление учащихся]. №прик\_зач, [Выпуск учащихся]. дата\_оконч, [Выпуск учащихся]. №прик\_отч

FROM [Список учащихся] INNER JOIN ([Зачисление учащихся] LEFT JOIN [Выпуск учащихся] ON [Зачисление учащихся]. код\_зач = [Выпуск учащихся]. код\_зач) ON [Список учащихся]. Ном\_уч = [Зачисление учащихся]. ном\_уч;

6. Запрос «Список группы» для формы «Список группы».

SELECT [комплектование уч]. код\_группы, [Список групп]. группа, [Список групп]. код\_отделения, [Список групп]. таб\_ном, [комплектование уч]. ном\_уч, [Список учащихся]. Ф.И.О.\_уч

FROM [Список учащихся] INNER JOIN ([Список групп] INNER JOIN [комплектование уч] ON [Список групп]. код\_группы = [комплектование уч]. код\_группы) ON [Список учащихся]. Ном\_уч = [комплектование уч]. ном\_уч;

7. Запрос «КАЧЕСТВО СПОРТИВНОЙ ПОДГОТОВКИ УЧАЩИХСЯ МОУ ДОД ДЮСШ №1» для сводной таблицы, сводной диаграммы, отчета «КАЧЕСТВО СПОРТИВНОЙ ПОДГОТОВКИ УЧАЩИХСЯ МОУ ДОД ДЮСШ №1»

SELECT [Спортсмены-разрядники]. ном\_уч, [Спортсмены-разрядники]. код\_разр, Разряд.разряд, [Спортсмены-разрядники]. дата\_разр

FROM Разряд INNER JOIN [Спортсмены-разрядники] ON Разряд.код\_разр = [Спортсмены-разрядники]. код\_разр;

8,9,10. Запросы «кол групп волейбол», «кол групп легкой атлетики», «кол групп спорт гимнастики» для отчета «Справка о МОУ».

SELECT Count([Список групп]. код\_группы) AS [Count-код\_группы]

FROM [Список групп]

WHERE ((([Список групп]. код\_отделения)=2));

11,12,13,14,15. Запросы «кол засл тренер россии», «кол знак 80 л мин физ к», «кол мастер спорта», «кол отлич физ культ и спорта рф», «кол почет работ общего образ» для отчета «Справка о МОУ часть3».

SELECT Count([Заслуженные работники]. таб\_ном) AS [Count-таб\_ном]

FROM [Заслуженные работники]

WHERE ((([Заслуженные работники]. код\_звания)=2));

16,17,18,19,20,21. Запросы «кол малоимущих семей», «кол многодет семей», «кол неблагополуч семей», «кол неполных семей», «кол опекаемых детей», «кол состоят на учете в ОППН» для отчета «Справка о МОУ часть4»

SELECT Count([Семья учащегося]. ном\_уч) AS [Count-ном\_уч]

FROM [Семья учащегося]

WHERE ((([Семья учащегося]. код\_стат)=4));

22,23,24 Запросы «кол сотр вторая кат», «кол сотр высшей кат», «кол сотр первой кат» для отчета «Справка о МОУ часть 2»

SELECT Count([Аттестация тренеров]. таб\_ном) AS [Count-таб\_ном]

FROM [Аттестация тренеров]

WHERE ((([Аттестация тренеров]. код\_утв\_кат)=1));

25,26,27,28,29. Запросы «кол сотр с высшим обр», «кол сотр с высшим физкульт образ», «кол сотр со средним образ», «кол сотр со средним спец образ», «кол сотр со средним физкульт образ» для отчета «Справка о МОУ часть 2»

SELECT Count([образование сотрудников]. таб\_ном) AS [Count-таб\_ном]

FROM [образование сотрудников]

WHERE ((([образование сотрудников]. код\_образ)=2));

30. Запрос «кол сотр-м» для отчета «Справка о МОУ часть 2».

SELECT Count([Список сотрудников]. таб\_ном) AS [Count-таб\_ном]

FROM [Список сотрудников]

WHERE ((([Список сотрудников]. Пол)= «м»));

31,32,33. Запросы «кол уч волейбол»,» кол уч легкая атлетика»,» кол уч спорт гимпнастика» для отчета «Справка о МОУ».

SELECT Count([комплектование уч]. ном\_уч) AS [Count-ном\_уч]

FROM [Список групп] INNER JOIN [комплектование уч] ON [Список групп]. код\_группы=[комплектование уч]. код\_группы

WHERE ((([Список групп]. код\_отделения)=2));

34,35,36,37,38. Запросы «кол уч на всеросс сор за период», «кол уч на город сор за период»,» кол уч на междунар сор за период»,» кол уч на област сор за период», «кол уч на школьн сор за период» для отчета» участие в сор за опр период»

SELECT Count([Список учащихся]. Ном\_уч) AS [Count-Ном\_уч]

FROM [Список учащихся] INNER JOIN (Соревнования INNER JOIN [результат выступлений на соревнованиях] ON Соревнования.код\_сор = [результат выступлений на соревнованиях]. код\_сор) ON [Список учащихся]. Ном\_уч = [результат выступлений на соревнованиях]. ном\_уч

WHERE (((Соревнования.дата\_сор)>[введите начальну дату] And (Соревнования.дата\_сор)<[введите конечную дату]))

GROUP BY Соревнования.код\_ур

HAVING (((Соревнования.код\_ур)=2));

39. Запрос «количество групп» для отчета «Справка о МОУ».

SELECT Count([Список групп]. код\_группы) AS [Count-код\_группы]

FROM [Список групп];

40. Запрос «количество учащихся» для отчета «Справка о МОУ».

SELECT Count([Список учащихся]. Ном\_уч) AS [Count-Ном\_уч]

FROM [Список учащихся];

41. Запрос «количество штатных сотрудников» «Справка о МОУ часть 2».

SELECT Count([Список сотрудников]. таб\_ном) AS [Count-таб\_ном]

FROM [Список сотрудников]

WHERE ((([Список сотрудников]. Штат\_сотр)=Yes));

42. «лучшие учащиеся запрос» для связанной формы «лучший учащийся».

SELECT [Лучшие учащиеся]. ном\_уч, [Список учащихся]. Ф.И.О.\_уч, [Лучшие учащиеся]. год\_л\_уч, [Лучшие учащиеся]. лучший\_уч

FROM [Список учащихся] INNER JOIN [Лучшие учащиеся] ON [Список учащихся]. Ном\_уч = [Лучшие учащиеся]. ном\_уч;

43,44,45,46. Запросы «мастер спорта», «мастер спорта международного класса», «отлич физ к и спорта рф», «почет раб общего образ» для личного пользования методиста.

SELECT [Список сотрудников]. Ф.И.О.\_сотр

FROM [Список сотрудников] INNER JOIN ([Почетные звания] INNER JOIN [Заслуженные работники] ON [Почетные звания]. код\_звания = [Заслуженные работники]. код\_звания) ON [Список сотрудников]. таб\_ном = [Заслуженные работники]. таб\_ном

WHERE ((([Заслуженные работники]. код\_звания)=2));

47. Запрос «Показатель здоровья» для личного пользования методиста, сводной таблицы, диаграммы, отчета «показатель здоровья».

SELECT [Оздоровление учащихся]. ном\_уч, [Оздоровление учащихся]. дата\_з, [Оздоровление учащихся]. код\_ур\_з, [Уровень здоровья]. Уровень\_здор

FROM [Уровень здоровья] INNER JOIN [Оздоровление учащихся] ON [Уровень здоровья]. Код\_ур\_з = [Оздоровление учащихся]. код\_ур\_з;

48. Запрос «результат выступлений на соревнованиях Запрос» для связанной формы «результат выступлений на соревнованиях»

SELECT [результат выступлений на соревнованиях]. ном\_уч, [Список учащихся]. Ф.И.О.\_уч, [результат выступлений на соревнованиях]. код\_сор, [результат выступлений на соревнованиях]. кол\_бал, [результат выступлений на соревнованиях]. таб\_ном, [Список сотрудников]. Ф.И.О.\_сотр

FROM [Список учащихся] INNER JOIN ([Список сотрудников] INNER JOIN (Соревнования INNER JOIN [результат выступлений на соревнованиях] ON Соревнования.код\_сор = [результат выступлений на соревнованиях]. код\_сор) ON [Список сотрудников]. таб\_ном = [результат выступлений на соревнованиях]. таб\_ном) ON [Список учащихся]. Ном\_уч = [результат выступлений на соревнованиях]. ном\_уч;

49. Запрос «сведения о количестве, принявших участие в сор» для личного пользования методиста, сводной таблицы, диаграммы, отчета «сведения о количестве принявших участие в соревнованиях».

SELECT [результат выступлений на соревнованиях]. код\_сор, [результат выступлений на соревнованиях]. ном\_уч, [Список учащихся]. Ф.И.О.\_уч, Соревнования.код\_ур, [Уровень соревнований]. ур\_сор, Соревнования.дата\_сор

FROM [Уровень соревнований] INNER JOIN ([Список учащихся] INNER JOIN (Соревнования INNER JOIN [результат выступлений на соревнованиях] ON Соревнования.код\_сор = [результат выступлений на соревнованиях]. код\_сор) ON [Список учащихся]. Ном\_уч = [результат выступлений на соревнованиях]. ном\_уч) ON [Уровень соревнований]. код\_ур = Соревнования.код\_ур;

50. Запрос «сведения об учащихся образовательных учреждений» для личного пользования методиста, сводной таблицы, диаграммы, отчета «сведения об учащихся образовательных учреждений».

SELECT [Список учащихся]. Ном\_уч, [Список учащихся]. код\_обр\_уч, [Образовательные учереждения]. Обр\_уч

FROM [Образовательные учереждения] INNER JOIN [Список учащихся] ON [Образовательные учереждения]. Код\_обр\_уч = [Список учащихся]. код\_обр\_уч;

51. «семьи учащихся запрос» для формы «семья учащегося».

SELECT [Семья учащегося]. ном\_уч, [Список учащихся]. Ф.И.О.\_уч, [Семья учащегося]. код\_стат, [Соц статус семьи]. Соц\_стат

FROM [Список учащихся] INNER JOIN ([Соц статус семьи] INNER JOIN [Семья учащегося] ON [Соц статус семьи]. код\_стат = [Семья учащегося]. код\_стат) ON [Список учащихся]. Ном\_уч = [Семья учащегося]. ном\_уч;

52. «Соревнования Запрос1» для формы «соревнования».

SELECT Соревнования.код\_сор, Соревнования.код\_города, Города.город, Соревнования.ранг, Соревнования.код\_ур, [Уровень соревнований]. ур\_сор, Соревнования.код\_прогр\_сор, [Программы соревнований]. прогр\_сор, Соревнования.фин\_сор, Соревнования.дата\_сор

FROM [Уровень соревнований] INNER JOIN ([Программы соревнований] INNER JOIN (Города INNER JOIN Соревнования ON Города.код\_города = Соревнования.код\_города) ON [Программы соревнований]. код\_прогр\_сор = Соревнования.код\_прогр\_сор) ON [Уровень соревнований]. код\_ур = Соревнования.код\_ур;

53. Запрос «Комплектование групп» для формы «Комплектование групп».

SELECT [Список групп]. код\_группы, [Список групп]. группа, [Список групп]. код\_отделения, Отделения. Отделение, [Список групп]. таб\_ном, [Список сотрудников]. Ф.И.О.\_сотр

FROM [Список сотрудников] INNER JOIN (Отделения INNER JOIN [Список групп] ON Отделения. Код\_отделения = [Список групп]. код\_отделения) ON [Список сотрудников]. таб\_ном = [Список групп]. таб\_ном;

54. Запрос «Список учащихся за определенный период» для личного пользования методиста.

SELECT [Список учащихся]. Ф.И.О.\_уч, [Зачисление учащихся]. дата\_зач, [Выпуск учащихся]. дата\_оконч

FROM [Список учащихся] INNER JOIN ([Зачисление учащихся] LEFT JOIN [Выпуск учащихся] ON [Зачисление учащихся]. код\_зач = [Выпуск учащихся]. код\_зач) ON [Список учащихся]. Ном\_уч = [Зачисление учащихся]. ном\_уч

WHERE ((([Зачисление учащихся]. дата\_зач)>[введите дату начала] And ([Зачисление учащихся]. дата\_зач)<[введите дату конца])) OR ((([Выпуск учащихся]. дата\_оконч)>[введите дату начала] And ([Выпуск учащихся]. дата\_оконч)<[введите дату конца])) OR ((([Зачисление учащихся]. дата\_зач)<[введите дату начала]) AND (([Выпуск учащихся]. дата\_оконч)>[введите дату конца]));

55. Запрос «список учащихся запрос» для формы «список учащихся».

SELECT [Список учащихся]. Ном\_уч, [Список учащихся]. Ф.И.О.\_уч, [Список учащихся]. Адрес\_уч, [Список учащихся]. тел\_дом\_уч, [Список учащихся]. Тел\_сот\_уч, [Список учащихся]. Дата\_р\_уч, [Список учащихся]. пол, [Список учащихся]. Ф.И.О.\_м, [Список учащихся]. Ф.И.О.\_п, [Список учащихся]. код\_обр\_уч, [Образовательные учереждения]. Обр\_уч

FROM [Образовательные учереждения] INNER JOIN [Список учащихся] ON [Образовательные учереждения]. Код\_обр\_уч = [Список учащихся]. код\_обр\_уч;

56. Запрос «спортсмены-разрядники запрос» для формы «Спортсмены-разрядники».

SELECT [Спортсмены-разрядники]. ном\_уч, [Список учащихся]. Ф.И.О.\_уч, [Спортсмены-разрядники]. код\_разр, Разряд.разряд, [Спортсмены-разрядники]. дата\_разр, [Спортсмены-разрядники]. таб\_ном, [Список сотрудников]. Ф.И.О.\_сотр, [Спортсмены-разрядники]. ном\_прик

FROM [Список учащихся] INNER JOIN ([Список сотрудников] INNER JOIN (Разряд INNER JOIN [Спортсмены-разрядники] ON Разряд.код\_разр = [Спортсмены-разрядники]. код\_разр) ON [Список сотрудников]. таб\_ном = [Спортсмены-разрядники]. таб\_ном) ON [Список учащихся]. Ном\_уч = [Спортсмены-разрядники]. ном\_уч;

57. Запрос «таблица результатов выступлений на соревнованиях» для формы» таблица результатов выступлений на соревнованиях».

SELECT [Список учащихся]. Ф.И.О.\_уч, Соревнования.дата\_сор, Города.город, Соревнования.ранг, [Программы соревнований]. прогр\_сор, [Уровень соревнований]. ур\_сор, Соревнования.фин\_сор, [результат выступлений на соревнованиях]. кол\_бал, [Список сотрудников]. Ф.И.О.\_сотр

FROM [Уровень соревнований] INNER JOIN ([Список учащихся] INNER JOIN ([Список сотрудников] INNER JOIN (([Программы соревнований] INNER JOIN (Города INNER JOIN Соревнования ON Города.код\_города = Соревнования.код\_города) ON [Программы соревнований]. код\_прогр\_сор = Соревнования.код\_прогр\_сор) INNER JOIN [результат выступлений на соревнованиях] ON Соревнования.код\_сор = [результат выступлений на соревнованиях]. код\_сор) ON [Список сотрудников]. таб\_ном = [результат выступлений на соревнованиях]. таб\_ном) ON [Список учащихся]. Ном\_уч = [результат выступлений на соревнованиях]. ном\_уч) ON [Уровень соревнований]. код\_ур = Соревнования.код\_ур;

58. Запрос «участие в сор за опр период» для формы «участие в сор за опр период».

SELECT [кол уч на всеросс сор за период]. [Count-Ном\_уч], [кол уч на город сор за период]. [Count-Ном\_уч], [кол уч на междунар сор за период]. [Count-Ном\_уч], [кол уч на област сор за период]. [Count-Ном\_уч], [кол уч на школьн сор за период]. [Count-Ном\_уч]

FROM [кол уч на всеросс сор за период], [кол уч на город сор за период], [кол уч на междунар сор за период], [кол уч на област сор за период], [кол уч на школьн сор за период];

59. Запрос «школы учащихся» для формы «школы учащихся».

SELECT [Список учащихся]. Ном\_уч, [Список учащихся]. Ф.И.О.\_уч, [Список учащихся]. код\_обр\_уч, [Образовательные учереждения]. Обр\_уч

FROM [Образовательные учереждения] INNER JOIN [Список учащихся] ON [Образовательные учереждения]. Код\_обр\_уч = [Список учащихся]. код\_обр\_уч;

60. Запрос «зач-отч учащихся» для формы «зач-отч учащихся».

SELECT [Зачисление учащихся]. код\_зач, [Зачисление учащихся]. ном\_уч, [Зачисление учащихся]. дата\_зач, [Зачисление учащихся]. №прик\_зач, [Выпуск учащихся]. дата\_оконч, [Выпуск учащихся]. №прик\_отч, [Выпуск учащихся]. №прик\_разр\_к

FROM [Зачисление учащихся] LEFT JOIN [Выпуск учащихся] ON [Зачисление учащихся]. код\_зач=[Выпуск учащихся]. код\_зач;

61. Запрос «Список учащихся за определенный период» для отчета «Список учащихся за определенный период».

SELECT [вспомог запрос]. Ф.И.О.\_уч, [вспомог запрос]. дата\_зач, [вспомог запрос]. дата\_оконч

FROM [вспомог запрос]

WHERE ((([вспомог запрос]. дата\_оконч)>[введите дату начала] And ([вспомог запрос]. дата\_оконч)<[введитие дату конца])) OR ((([вспомог запрос]. дата\_оконч) Is Null));

### 2.4.4 Проектирование форм для ввода, редактирования и просмотра

1. Форма «Список сотрудников» доступна для ввода, изменения и удаления данных о сотрудниках директору и методисту. Тренеру доступна только для просмотра. Разработка основана на таблице «Список сотрудников»

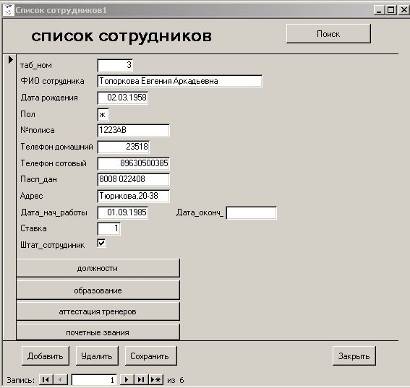


рис 2.11.

2. Форма «Список учащихся» доступна для ввода, изменения и удаления данных об учащихся директору и методисту. Тренеру доступна только для просмотра. Разработка основана на запросе» список учащихся запрос».

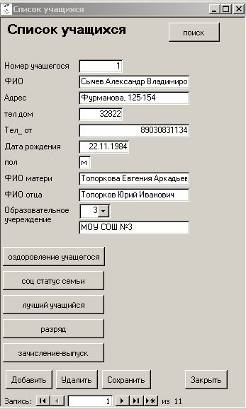


рис 2.12

3. Форма «Соревнования» доступна для ввода, изменения и удаления данных о соревнованиях директору и методисту. Тренеру доступна только для просмотра. Разработка основана на запросе» Соревнования Запрос1».

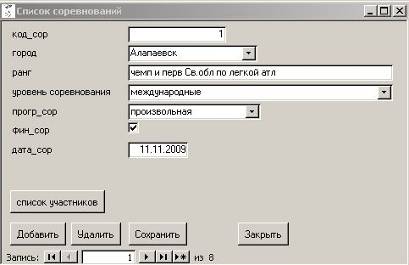


рис 2.13

4. Форма «Комплектование групп» доступна для ввода, изменения и удаления данных о комплектовании групп директору и методисту. Тренеру доступна только для просмотра. Разработка основана на запросе «Комплектование групп».

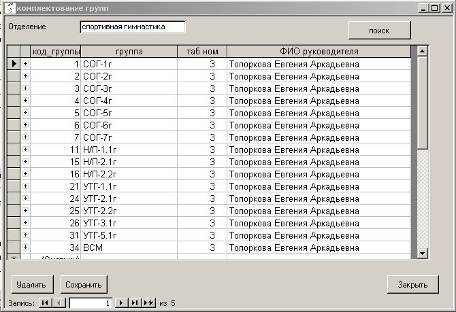


рис 2.14

5. Связанная форма «Должности сотрудников» для формы «Список сотрудников»

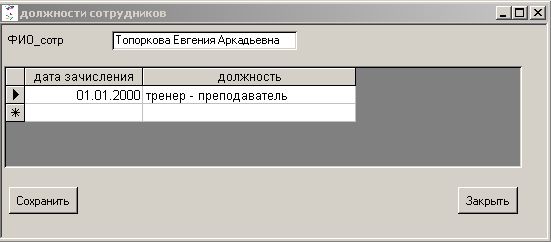


рис 2.15

6. Связанная форма «Образование сотрудников» для формы «Список сотрудников»

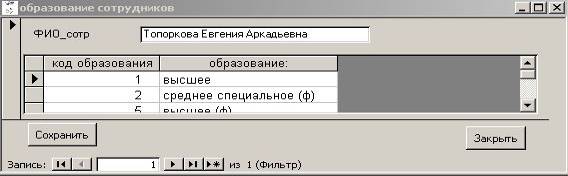


рис 2.16

7. Связанная форма «Аттестация тренеров» для формы «Список сотрудников»

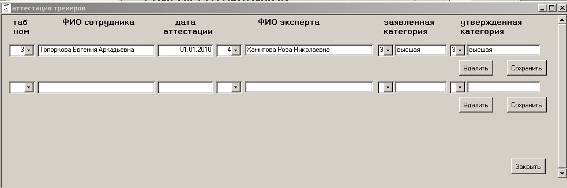


рис 2.17

8. «Должности сотрудников подчиненная форма» для формы «Должности сотрудников»

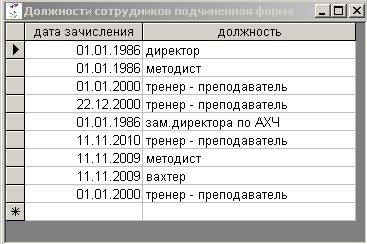


рис 2.18

9. «Образование сотрудников подчиненная форма» для формы «Образование сотрудников»

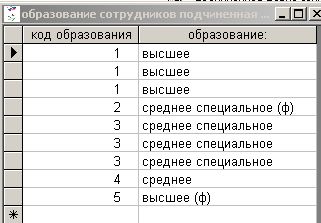


рис 2.19

10. Связанная форма «Оздоровление учащихся» для формы «Список учащихся»

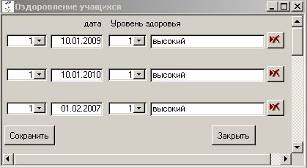


рис 2.20

11. Связанная форма «Семья учащегося» для формы «Список учащихся»

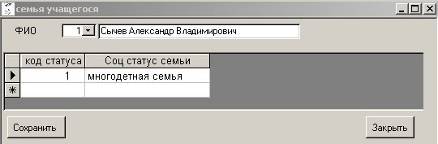


рис 2.21

12. Связанная форма «Лучший учащийся» для формы «Список учащихся»

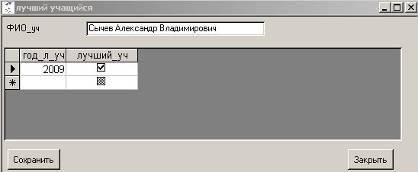


рис 2.22

13. Связанная форма «Спортсмены – разрядники» для формы «Список учащихся»

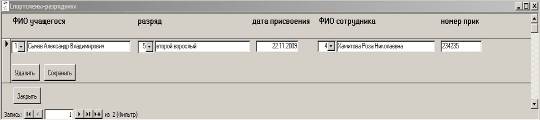


рис 2.23

14. Связанная форма «Зач-отч учащихся» для формы «Список учащихся»

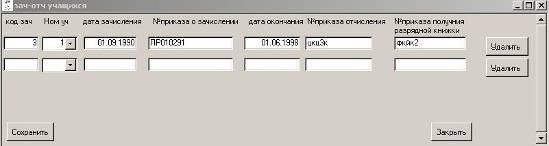


рис 2.24

15. «Семья учащегося подчиненная форма» для формы «Семья учащегося»



рис 2.25

16. «Лучший учащийся подчиненная форма» для формы «Лучший учащийся»



рис 2.26

17. Подчиненная форма «Список групп» для формы «Комплектование групп»

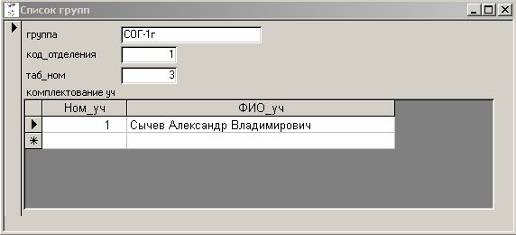


рис 2.27

18. «Список групп подчиненная форма» для формы «Список групп»

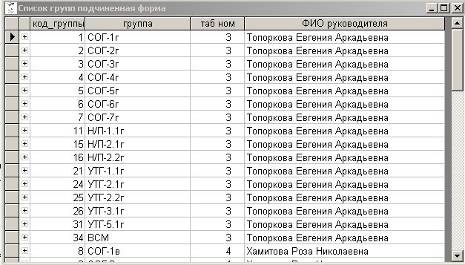


рис 2.28

19. Подчиненная форма «Комплектование уч» для формы «Список групп подчиненная форма»



рис. 2.29

20. Связанная форма «Результаты выступлений на соревнованиях» для формы «Соревнования»



рис 2.30

21. Связанная форма «Заслуженные работники» для формы «Список сотрудников»

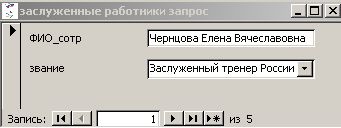


рис 2.31

22, 23. Формы «Показатель здоровья» и» Показатель здоровья диаграмма» для анализа данных методистом и директором о состоянии здоровья учащихся.

24, 25 Формы «сведения об учащихся образовательных учреждений» «сведения об учащихся образовательных учреждений диаграмма» для личного пользования методиста и директора.

26. Форма «Справочники» для общего пользования

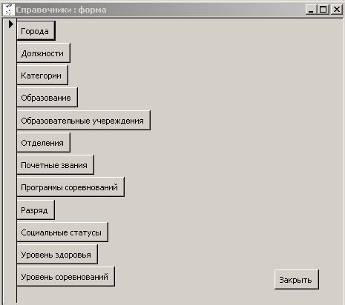


рис 2.32

### 2.4.5 Проектирование отчетов

1. Отчет по аттестации тренеров. Просматривают и печатают все пользователи. Сгенерирован на основе запроса «Аттестация тренеров» – см. Приложение 1.

2. Отчет по качеству спортивно подготовки учащихся. Сгенерирован на основе запроса «Качество спортивной подготовки учащихся МОУ ДОД ДЮСШ №1» – см. Приложение 2.

3. Отчет о количестве учащихся, принявших участие в различных соревнованиях. Сгенерирован на основе запроса «сведения о количестве, принявших участие в сор» – см. Приложение 3.

4. Отчет «Список учащихся за определенный период». Сгенерирован на основе запроса «список учащихся за определенный период» – см. Приложение 4.

5. Отчет «Спортсмены – разрядники». Сгенерирован на основе запроса» спортсмены-разрядники запрос « – см. Приложение 5.

6. Отчет «Справка о МОУ». Сгенерирован на основе запросов» часть1», «часть2», «часть3», «часть4» – см. Приложение 6.

7. Отчет «Учет поступлений и выпуска учащихся». Сгенерирован на основе запроса» зач-отч»

# Заключение

Дипломная работа представляет собой разработку ИС для муниципального образовательного учреждения дополнительного образования детей «Детско-юношеская спортивная школа №1» г. Алапаевск.

В результате проделанной работы:

– спроектирована и реализована база данных (всего 29 таблиц);

– Спроектированы и реализованы необходимые для всех категорий пользователей запросы (всего 61);

– На основе запросов разработан и реализован пользовательский интерфейс, состоящий из форм (всего 45), отчётов (всего 15) и макросов (всего 10).

Разработанная ИС обеспечит автоматизацию следующих процессов:

– учет численности учащихся;

– получение разрядов учащимися;

– участие в соревнованиях;

– процесс аттестации тренеров;

– получение званий тренеров;

– Составление отчетной информации по соревнованиям

– аттестации тренеров;

– оздоровлению учащихся;

– учету зачисления и выпуска учащихся.

В настоящее время разработка системы в основном закончена, она продемонстрирована пользователям, которые будут проводить ее тестирование.

Дальнейшее развитие проекта:

– Для защиты данных базы необходимо провести разделение пользователей на группы с присвоением каждому имени и пароля;

– Расширить атрибутику с тем, чтобы:

– сделать вычисляемым поле «лучший учащийся»;

– разработать и реализовать процесс аттестации учащихся;

– Доработать интерфейс с целью улучшить возможности удобного поиска информации;

– В случае расширения организации необходимо перенести базу данных на Microsoft SQL Server, в многопользовательский вариант с архитектурой «клиент-сервер».

**Список используемой литературы**

1. Дейт К. Руководство по реляционной СУБД DB2.М.: Финансы и статистика, 1988, 320 с.
2. Дейт К.Дж. Введение в систем баз данных. М.: Изд. Дом «Вильямс», 2001. 1072 с.
3. Бекаревич Ю.Б., Пушкина Н.В. Microsoft Access 2002. СПб.: БХВ-Петербург, 2003. 718 с.
4. Гринченко Н.Н., Гусев Е.В., Макаров Н.П. Проектироване баз данных СУБД Microsoft ACCESS: Учеб. Пособие. М.: Горячая линия – Телеком, 2004. 240 с.
5. Шарипова Н.Н. Теория и практика проектирования баз данных: Учеб. Пособие. Екатеринбург: Издательство Уральского университета, 2006. 380 с.