

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СТАВРОПОЛЬСКОГО КРАЯ  
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
«ГЕОРГИЕВСКИЙ РЕГИОНАЛЬНЫЙ КОЛЛЕДЖ «ИНТЕГРАЛ»

Отделение: Индустриальных и цифровых технологий

Специальность: 09.02.07 «Информационные системы и программирование»

# КУРСОВОЙ ПРОЕКТ

ПО МДК. 11.01 Технология разработки и защиты баз данных

ПМ.11. Разработка, администрирование и защита баз данных

На тему: «Разработка базы данных "Ежедневник"»

Выполнил(а) студент(ка) 3 курса группы ИСП-\_\_\_

Иванов Иван Иванович

*(ф.и.о. студента)*

Работа допускается к защите \_\_\_\_\_

Петрова И.В.

*Подпись руководителя*

*(ф.и.о.)*

Оценка: \_\_\_\_\_

Дата защиты:

« \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_\_ г.

Подпись

Георгиевск, 2023г.

## СОДЕРЖАНИЕ

|   |    |
|---|----|
| ВВЕДЕНИЕ  | 3  |
| 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ                    | 5  |
| 2. ПОСТРОЕНИЕ ЛОГИЧЕСКОЙ МОДЕЛИ                 | 7  |
| 2.1. ПЕРЕЧЕНЬ СУЩНОСТЕЙ И ИХ АТТРИБУТОВ         | 7  |
| 2.2. ИНФОЛОГИЧЕСКАЯ МОДЕЛЬ («СУЩНОСТЬ – СВЯЗЬ») | 9  |
| 3. РЕЛЯЦИОННАЯ МОДЕЛЬ БД                        | 11 |
| 3.1. СТРУКТУРЫ ТАБЛИЦ ДАТАЛОГИЧЕСКОЙ МОДЕЛИ     | 11 |
| 3.2. ВЫБОР КЛЮЧЕВЫХ ПОЛЕЙ                       | 12 |
| 3.3. НОРМАЛИЗАЦИЯ ОТНОШЕНИЙ                     | 11 |
| 4. РЕАЛИЗАЦИЯ БАЗЫ ДАННЫХ В СУБД                | 15 |
| 4.1. СОЗДАНИЕ ТАБЛИЦ                            | 15 |
| 4.2. СОЗДАНИЕ ЗАПРОСОВ                          | 17 |
| 4.3. СОЗДАНИЕ ФОРМ                              | 22 |
| 4.4. СОЗДАНИЕ ОТЧЁТОВ                           | 26 |
| ЗАКЛЮЧЕНИЕ                                      | 30 |
| СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ                | 31 |

|         |      |              |       |      |  |                                    |      |        |
|---------|------|--------------|-------|------|--|------------------------------------|------|--------|
|         |      |              |       |      | <b>КП – 09.02.07 - 11111 – 2023 – ПЗ</b>       |                                    |      |        |
| Изм     | Лист | № докум.     | Подп. | Дата |  |                                    |      |        |
| Разраб. |      | Иванов И.И.  |       |      | <b>Разработка базы данных<br/>«Ежедневник»</b> | Лит.                               | Лист | Листов |
| Пров.   |      | Петрова И.В. |       |      |  | У                                  | 2    | 31     |
| Н.контр |      |              |       |      |  | ГБПОУ ГРК<br>«Интеграл»<br>ИСП-___ |      |        |
| Утв.    |      |              |       |      |  |                                    |      |        |
|         |      |              |       |      |  |                                    |      |        |

## ВВЕДЕНИЕ

В современном мире во всех приложениях для взаимодействия с пользователями, для хранения данных о пользователе и других данных, необходимо использовать базу данных.

База данных (БД) — это программа, которая позволяет хранить и обрабатывать информацию в структурированном виде. В базе данных можно сохранять любую информацию, чтобы позже получать к ней доступ.

Данные надо не просто как-то хранить, но и иметь возможность их обрабатывать и анализировать. Именно для этих целей и существуют базы данных.

Для управления базами данных используется система управления базами данных (СУБД) – комплекс программ, позволяющих создать базу данных и манипулировать данными. Система обеспечивает безопасность, надёжность хранения и целостность данных, а также представляет средства для администрирования БД. [5]

Ежедневник прекрасно повышает продуктивность, помогает освободить немного времени на отдых, общение с близкими или друзьями. Визуализированные списки дел не кажутся такими большими и невыполнимыми, из головы не вылетит ни одна мелочь. Ежедневник поможет разгрузить голову и не держать в памяти тысячу вещей одновременно.

Разработчики предлагают сотни программ и приложений для организации личного времени и дел. Их плюсы – вся информация в одном месте, функции архивирования и совместное редактирование с другими пользователями. Приложение в любую секунду под рукой, в гаджете, а значит, не занимает пространства в сумке. Даже потеряв телефон, пользователь восстановит данные на другом устройстве.

Таким образом достаточно простое приложение может помочь человеку полностью распределить своё время, стать более пунктуальным и собранным, что

|      |       |          |       |      |                                   |      |
|------|-------|----------|-------|------|-----------------------------------|------|
|      |       |          |       |      | КП - 09.02.07 – 11111 – 2023 – ПЗ | Лист |
| Изм. | Лист. | № докум. | Подп. | Дата |                                   | 3    |

в последствии может изменить впечатление о себе среди окружающих и даже помочь добиться успехов на работе или в любимом деле.

Т.о., тема данного курсового проекта «Разработка базы данных «Ежедневник» является актуальной.

Целью курсового проекта является разработка базы данных «Ежедневник» в выбранной СУБД.

Для достижения поставленной цели требуется выполнить следующие шаги:

- провести анализ предметной области;
- построить логическую модель;
- выбрать одну из требуемых СУБД;
- создать реляционную модель БД;
- реализовать базу данных в выбранной СУБД.

|      |       |          |       |      |                                   |      |
|------|-------|----------|-------|------|-----------------------------------|------|
|      |       |          |       |      | КП - 09.02.07 – 11111 – 2023 – ПЗ | Лист |
| Изм. | Лист. | № докум. | Подп. | Дата |                                   | 4    |

## 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Базы данных нужны для хранения информации. Чтобы получить полное понимание необходимости использования БД в современном программировании, необходимо ответить на три вопроса:

- Какую информацию и зачем мы храним?
- В каком виде и как надо хранить эту информацию?
- Как и каким способом можно получить доступ к этой информации?

Исходя из выше сказанного разрабатываемая база данных должна решать следующие задачи:

- 1) Хранить данные предметной области;
- 2) Иметь интуитивно понятный интерфейс;
- 3) Быстро и удобно отбирать необходимые пользователю данные;
- 4) Выдавать данные пользователю в удобном виде.

Предметной областью данного курсового проекта является процесс организации и распределения событий и задач для любого человека – т.е. организация ежедневника.

Органайзер - изначально небольшая книга, содержащая календарь, адресную книгу и блокнот, служащая для организации информации о личных контактах и событиях. С развитием информационных технологий книга стала заменяться сначала электронными органайзерами, затем карманными персональными компьютерами, компьютерными программами и онлайн-органайзерами, обладающими дополнительными функциями: напоминание о предстоящих событиях, защита и синхронизация информации. [8]

Органайзер является средством управления временем. Предварительное планирование дел помогает повысить плодотворность любой деятельности, как личной, так и профессиональной. Поэтому и человека, который консультирует организации и частных лиц в сфере повышения эффективности управления временем, тоже иногда называют "органайзер".

|      |       |          |       |      |                                   |      |
|------|-------|----------|-------|------|-----------------------------------|------|
|      |       |          |       |      | КП - 09.02.07 – 11111 – 2023 – ПЗ | Лист |
| Изм. | Лист. | № докум. | Подп. | Дата |                                   | 5    |

Следовательно, централизованный органайзер может и должен стать информационным центром, выполняющим свою самостоятельную функцию. В такой центр постоянно обращаются для решения как оперативных, так и более значимых вопросов деятельности.

Программа-органайзер - компьютерная программа, относящаяся к прикладному программному обеспечению и предназначенная для накопления пользовательской информации, а затем оперативного поиска по ней, организации дел и контроля за их выполнением, отслеживания определённых пользователем событий. Является одной из форм персонального органайзера. Их безусловными преимуществами являются большой объем вводимых данных, установка автоматических напоминаний. Формы органайзера могут быть разными.

Функции типичной компьютерной программы-органайзера связаны с обеспечением работы следующих подразделов:

1. Календарь – содержит перечень чисел, дней недели, месяцев, а также список годов;
2. записная книжка и листки-заметки – содержит список дел, комментариев к ним, а также может содержать время, в которое их надо выполнить;
3. события, привязанные к определенной дате и времени – содержат событие, дату и время. (например, праздники или встречи);
4. планировщик задач (заданий) для контроля за их самостоятельным или сторонним выполнением;
5. напоминатели - будильники об определённых пользователем событиях.

|      |       |          |       |      |                                   |      |
|------|-------|----------|-------|------|-----------------------------------|------|
|      |       |          |       |      | КП - 09.02.07 – 11111 – 2023 – ПЗ | Лист |
| Изм. | Лист. | № докум. | Подп. | Дата |                                   | 6    |

## 2. ПОСТРОЕНИЕ ЛОГИЧЕСКОЙ МОДЕЛИ

### 2.1. Перечень сущностей и их атрибутов

Сущность — это объект, информация о котором должна быть представлена в базе данных.

Атрибут - свойство сущности или связи. [3]

Сущности и атрибуты в данном курсовом проекте:

Сущность Задачи. Содержит следующие атрибуты: id Задачи, наименование задачи, описание задачи, статус задачи, дата начала задачи, дата конца задачи, id приоритет, id статус повторения

Сущность Уведомления. Содержит следующие атрибуты: id уведомления, задача id, состояние уведомления, дата напоминания, время напоминания

Сущность Статус повторения задачи. Содержит следующие атрибуты: id повторения, статус повторения задачи.

Сущность Статус задачи. Содержит следующие атрибуты: id статуса задачи, статус задачи.

Сущность Приоритет. Содержит следующие атрибуты: id приоритета, приоритет.

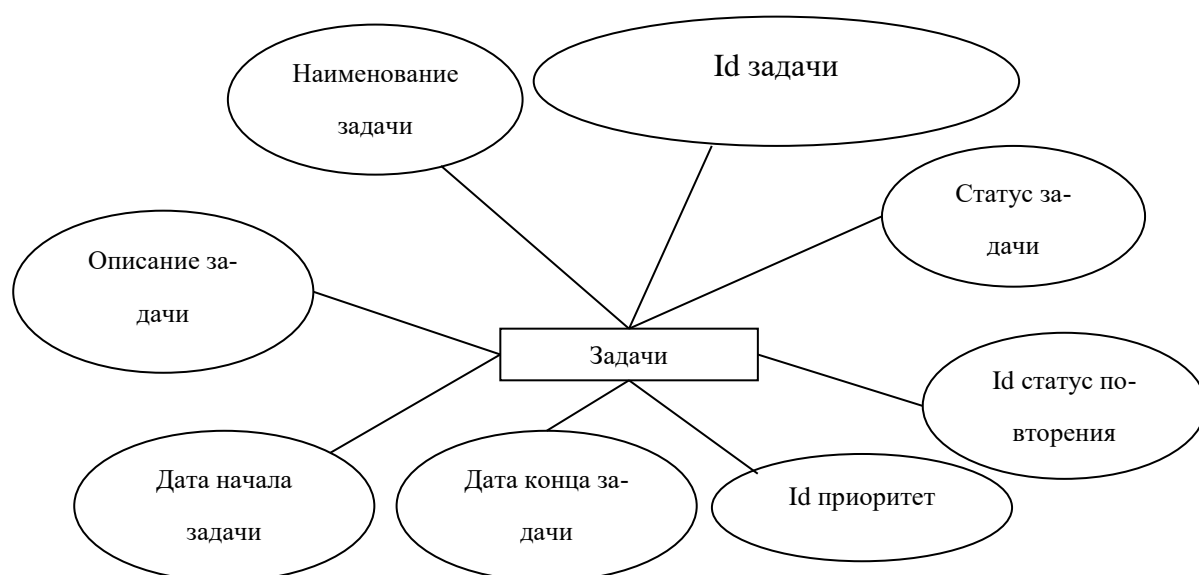


Рис. 2.1.1. Структура сущности «Задачи»

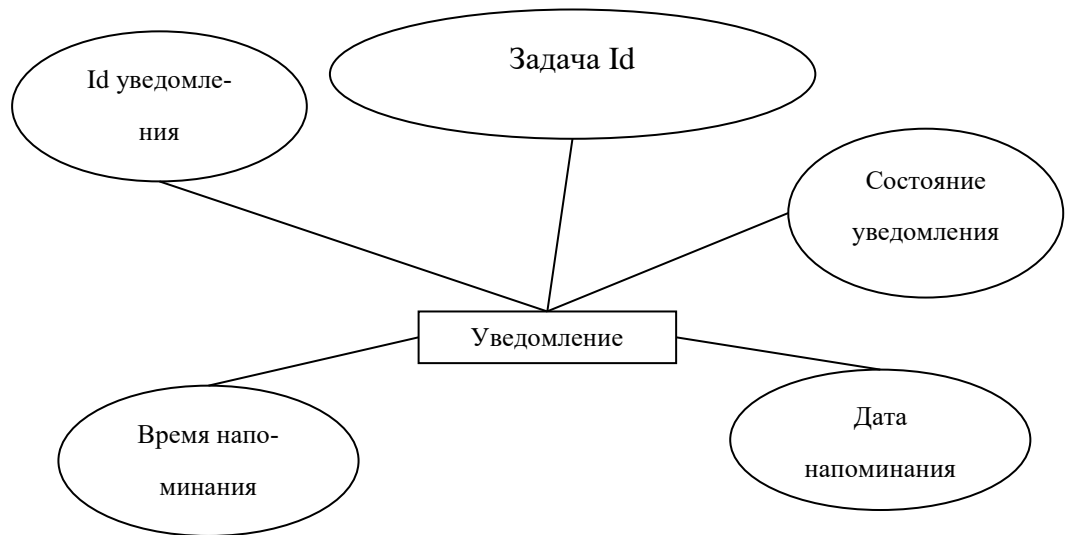


Рис. 2.1.2. Структура сущности «Уведомления»

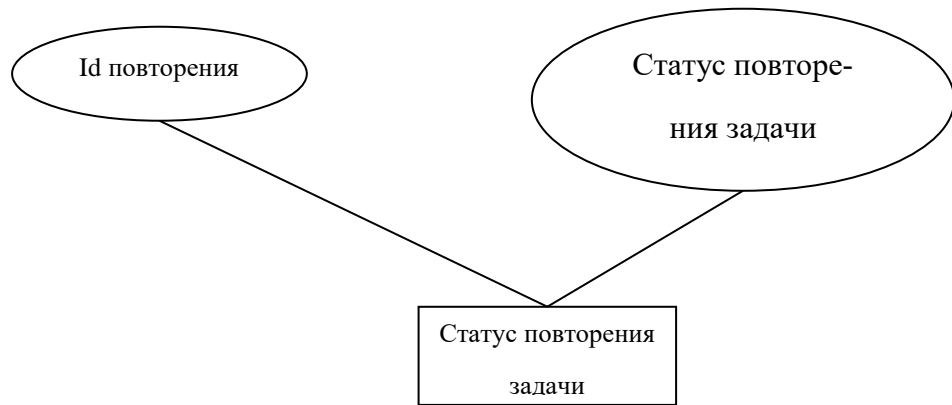


Рис. 2.1.3. Структура сущности «Статус повторения задачи»

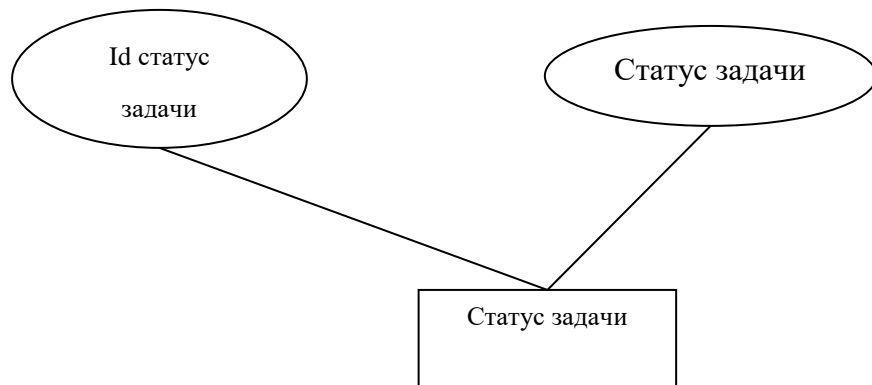


Рис. 2.1.4. Структура сущности «Статус задачи»



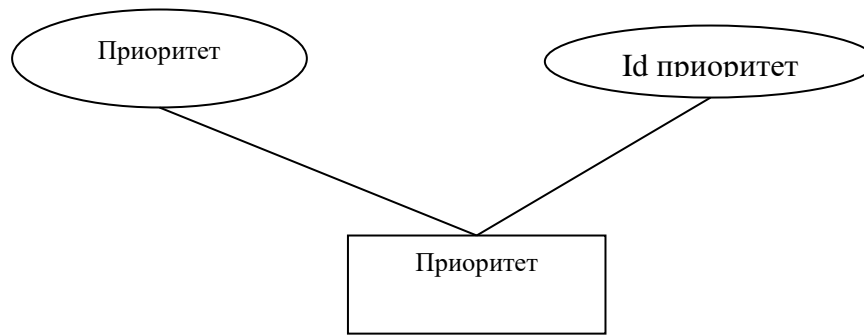


Рис. 2.1.5. Структура сущности «Приоритет»

## 2.2. Инфологическая модель («сущность – связь»)

Сущность — это объект, информация о котором должна быть представлена в базе данных. Экземпляр сущности — это информация о конкретном представителе объекта.

Связь - соединение между двумя и более сущностями. [1]

Между сущностями «Задачи» и «Уведомления» устанавливается связь “один ко многим”

Между сущностями «Приоритет» и «Задачи» устанавливается связь “один ко многим”

Между сущностями «Статус задачи» и «Задачи» устанавливается связь “один ко многим”

Между сущностями «Статус повторения задачи» и «Задачи» устанавливается связь “один ко многим”

### Сущности, представленные в данном курсовом проекте:

-Уведомления (Содержит информацию о статусе уведомления, и дате уведомления),

-Задачи (Содержит информацию о задачах),

-Повторяющиеся задачи (Содержит информацию о повторении задач),

-Статус задачи (содержит информации о статусе задачи),

-Приоритет (Содержит информацию о приоритете задач).

|      |       |          |       |      |                                   |      |
|------|-------|----------|-------|------|-----------------------------------|------|
|      |       |          |       |      | КП - 09.02.07 – 11111 – 2023 – ПЗ | Лист |
| Изм. | Лист. | № докум. | Подп. | Дата |                                   | 9    |

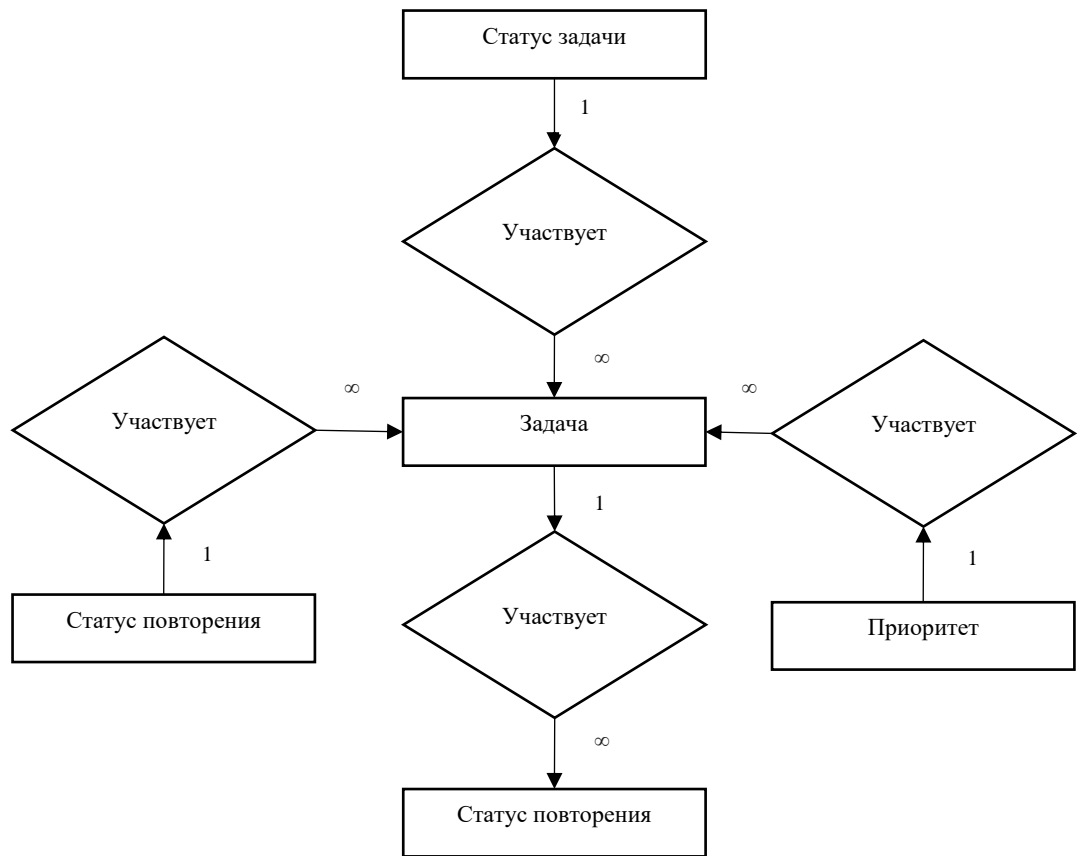


Рис. 2.2.1. Структура сущность-связь

|      |       |          |       |      |
|------|-------|----------|-------|------|
|      |       |          |       |      |
| Изм. | Лист. | № докум. | Подп. | Дата |

### 3. РЕЛЯЦИОННАЯ МОДЕЛЬ БД

#### 3.1. Нормализация отношений

Нормализация — это процесс организации данных в базе данных. Это включает создание таблиц и установку отношений между этими таблицами в соответствии с правилами, предназначенными для защиты данных и обеспечения большей гибкости базы данных за счет исключения избыточности и несогласованности зависимости. [2]

Существует несколько правил нормализации баз данных. Каждое правило называется «нормальной формой». Если выполняется первое правило, говорят, что база данных представлена в «первой нормальной форме». Если выполняются три первых правила, считается, что база данных представлена в «третьей нормальной форме». Есть и другие уровни нормализации, однако для большинства приложений достаточно нормализовать базы данных до третьей нормальной формы.

#### Приведение отношений к первой нормальной форме

Таблица считается приведенной к первой нормальной форме (1НФ) если выполняются следующие условия:

- все значения таблицы являются атомарными (неделимыми). Это означает, что ячейки таблицы должны содержать одиночные значения и все записи в отдельном столбце таблицы (атрибуте) должны иметь один и тот же тип,
- в качестве значений атрибута таблицы не допускаются группы или массивы повторяющихся данных.

Все отношения, полученные при отображении концептуальной инфологической модели данных на реляционную, атомарные, т.е. все значения атрибутов не являются множеством или повторяющейся группой. Следовательно, все отношения находятся в 1НФ.

#### Приведение отношений ко второй нормальной форме

|      |       |          |       |      |                                   |      |
|------|-------|----------|-------|------|-----------------------------------|------|
|      |       |          |       |      | КП - 09.02.07 – 11111 – 2023 – ПЗ | Лист |
| Изм. | Лист. | № докум. | Подп. | Дата |                                   | 11   |

Чтобы база данных находилась во второй нормальной форме (2NF), необходимо чтобы ее таблицы удовлетворяли следующим требованиям:

- Таблица должна находиться в первой нормальной форме
- Таблица должна иметь ключ
- Все не ключевые столбцы таблицы должны зависеть от полного ключа (в случае если он составной)

Отношения отвечают требованиям 2NF, т.к. они соответствуют 1NF, и каждый не ключевой атрибут в этих отношениях полностью зависит от ключа.

Приведение отношений к третьей нормальной форме

- Устраните поля, не зависящие от ключа.

Все полученные нами на предыдущем этапе отношения находятся во второй нормальной форме, и каждый не ключевой атрибут не транзитивно зависит от ключа. Так как между атрибутами оставшихся отношений нет транзитивной зависимости, то, следовательно, эти отношения соответствуют требованиям 3NF, и дальнейшей нормализации не требуется.

### 3.2. Выбор ключевых полей

Первичный ключ — это одно или несколько полей, комбинация значений которых однозначно определяет каждую запись в таблице. Если для таблицы определены ключевые поля, то Microsoft Access предотвращает дублирование или ввод пустых значений в ключевое поле.

Внешний (вторичный) ключ — это одно или несколько полей в таблице, содержащих ссылку на поле или поля первичного ключа в другой таблице. [1]

Ключи в данном курсовом проекте:

Сущность Задачи. Содержит следующие Ключи: id Задачи

Сущность Уведомления. Содержит следующие Ключи: id уведомления

Сущность Повторяющиеся задачи. Содержит следующие Ключи: id повторения

Сущность Статус задачи. Содержит следующие Ключи: id Статус Задачи

Сущность Приоритет. Содержит следующие Ключи: id Приоритет

|      |       |          |       |      |                                   |      |
|------|-------|----------|-------|------|-----------------------------------|------|
|      |       |          |       |      | КП - 09.02.07 – 11111 – 2023 – ПЗ | Лист |
| Изм. | Лист. | № докум. | Подп. | Дата |                                   | 12   |

### 3.3. Структуры таблиц даталогической модели

Даталогическая модель описывает объекты и связи предметной области на формальном уровне. Ее разработка основывается на инфологической модели. В процессе разработки осуществляется выбор модели данных, и определяются ее элементы. [6]

Таблица 3.2.1 - Структура таблицы "Задачи"

| Имя поля             | Тип данных     | Обязательное поле | Индексированное поле           |
|----------------------|----------------|-------------------|--------------------------------|
| Id задачи            | Счетчик        | Да                | Да (совпадения не допускаются) |
| Наименование задачи  | Короткий текст | Да                | Нет                            |
| Описание задачи      | Короткий текст | Нет               | Нет                            |
| Статус задачи        | Числовой       | Да                | Нет                            |
| Дата начала задачи   | Дата и время   | Да                | Нет                            |
| Дата конца задачи    | Дата и время   | Нет               | Нет                            |
| Id приоритет         | Числовой       | Нет               | Нет                            |
| Id статус повторения | Числовой       | Нет               | Нет                            |

Таблица 3.2.2 - Структура таблицы "Уведомления"

| Имя поля              | Тип данных | Обязательное поле | Индексированное поле           |
|-----------------------|------------|-------------------|--------------------------------|
| Id уведомления        | Счетчик    | Да                | Да (совпадения не допускаются) |
| Id задачи             | Числовой   | Да                | Нет                            |
| Состояние уведомления | Логический | Да                | Нет                            |
| Дата напоминания      | Дата Время | Да                | Нет                            |
| Время напоминания     | Дата Время | Да                | Нет                            |

Таблица 3.2.3 – Структура таблицы "Повторяющиеся задачи"

| Имя поля                 | Тип данных     | Обязательное поле | Индексированное поле           |
|--------------------------|----------------|-------------------|--------------------------------|
| Id повторения            | Счетчик        | Да                | Да (совпадения не допускаются) |
| Статус Повторения задачи | Короткий текст | Да                | Нет                            |

Таблица 3.2.4 - Структура таблицы "Статус Задачи"

| Имя поля         | Тип данных     | Обязательное поле | Индексированное поле           |
|------------------|----------------|-------------------|--------------------------------|
| Id статус задачи | Счетчик        | Да                | Да (совпадения не допускаются) |
| Статус задачи    | Короткий текст | Да                | Нет                            |

Таблица 3.2.5 – Структура таблицы "Приоритет"

| Имя поля      | Тип данных     | Обязательное поле | Индексированное поле           |
|---------------|----------------|-------------------|--------------------------------|
| Id приоритета | Счетчик        | Да                | Да (совпадения не допускаются) |
| Приоритет     | Короткий текст | Да                | Нет                            |

## 4. РЕАЛИЗАЦИЯ БАЗЫ ДАННЫХ В СУБД

### 4.1. Создание таблиц

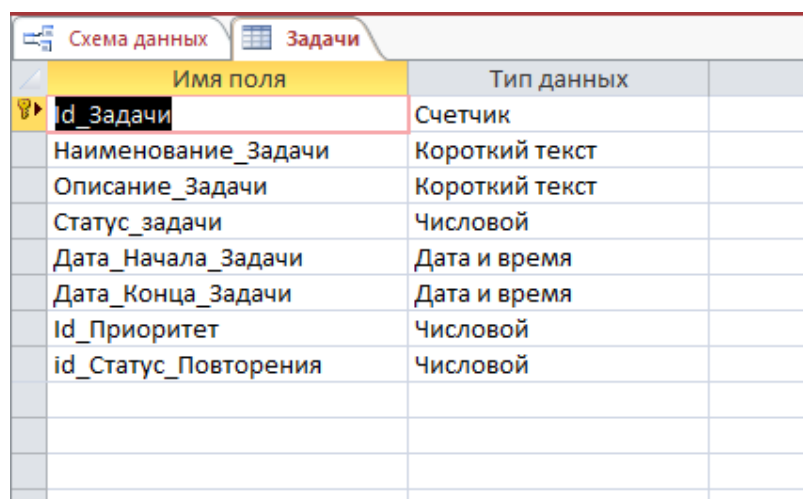
Большинство баз данных имеет табличную структуру. Таблицы являются основой, на которой строится все дальнейшее создание базы данных. В программе «Access» предусмотрено несколько вариантов построения таблиц, их можно увидеть при выборе в области объектов раздела Таблицы.

Таблицы, как и любой другой объект базы данных, имеют два основных режима создания: [7]

- режим Конструктора;
- режим Таблицы.

Процесс создания таблицы «Задачи»

- 1) В Access переходим в раздел «создание»
- 2) Нажимаем на кнопку «Конструктор Таблиц»
- 3) Вводим имя поля, далее тип данных этого поля
- 4) Задаём первичным ключи и задаём ему тип данных «Счётчик»
- 5) Заполнив все поля, сохраняем таблицу и даём ей название.



| Имя поля             | Тип данных     |
|----------------------|----------------|
| Id_Задачи            | Счетчик        |
| Наименование_Задачи  | Короткий текст |
| Описание_Задачи      | Короткий текст |
| Статус_задачи        | Числовой       |
| Дата_Начала_Задачи   | Дата и время   |
| Дата_Конца_Задачи    | Дата и время   |
| Id_Приоритет         | Числовой       |
| id_Статус_Повторения | Числовой       |

Рис. 4.1.1. Структура таблицы «Задачи» в режиме конструктора

Аналогично созданы таблицы «Уведомления», «Статус задачи», «Статус повторения задачи», «Приоритет». (рис. 4.1.2-4.1.6).

| Приоритет     |                |
|---------------|----------------|
| Имя поля      | Тип данных     |
| Id_Приоритета | Числовой       |
| Приоритет     | Короткий текст |

Рис. 4.1.2. Структура таблицы «Приоритет» в режиме конструктора

| Статус_задачи    |                |
|------------------|----------------|
| Имя поля         | Тип данных     |
| Id_Статус_Задачи | Счетчик        |
| Статус_задачи    | Короткий текст |

Рис. 4.1.3. Структура таблицы «Статус задачи» в режиме конструктора

| Статус_повторения_задачи |                |
|--------------------------|----------------|
| Имя поля                 | Тип данных     |
| Id_Повторения            | Счетчик        |
| Статус Повторения Задачи | Короткий текст |

Рис. 4.1.4. Структура таблицы «Статус повторения задачи» в режиме конструктора

| Уведомления           |              |
|-----------------------|--------------|
| Имя поля              | Тип данных   |
| Id_уведомления        | Счетчик      |
| задача_id             | Числовой     |
| Состояние_Уведомления | Логический   |
| Дата_Напоминания      | Дата и время |
| Время_Напоминания     | Дата и время |

Рис. 4.1.5. Структура таблицы «Уведомления» в режиме конструктора

После создания таблиц и связи их между собой через внешние ключевые поля была получена схема данных, представленная на рисунке 4.1.6.



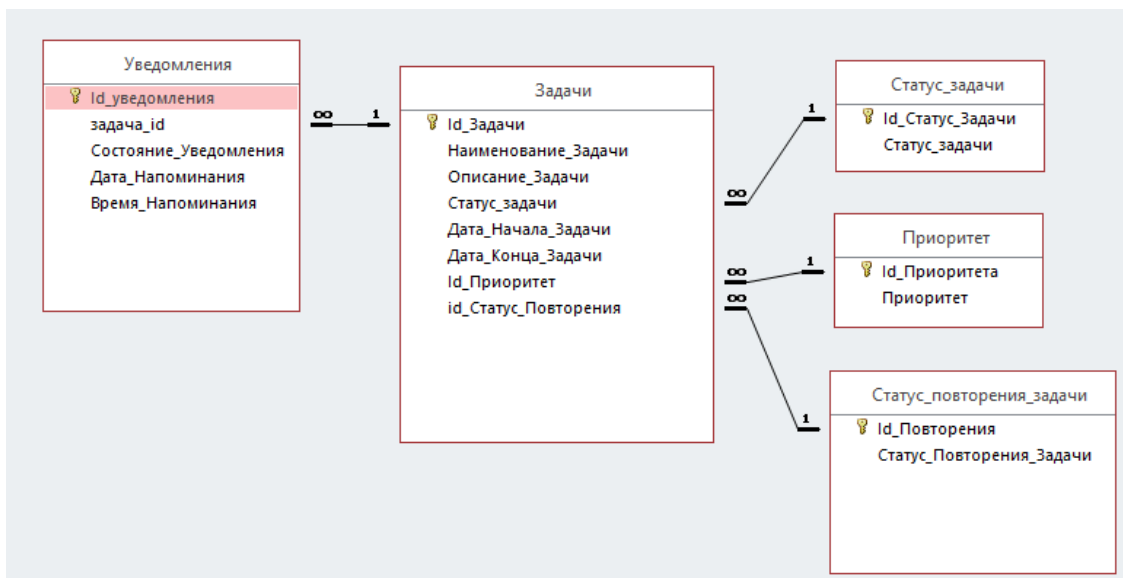


Рис. 4.1.6. Схема данных

## 4.2. Создание запросов

Запросы используются для просмотра, анализа и изменения данных в одной или нескольких таблицах. Например, можно использовать запрос для отображения данных из одной или нескольких таблиц и отсортировать их в определенном порядке, выполнить вычисления над группой записей, осуществить выборку из таблицы по определенным условиям. Запросы могут служить источником данных для форм и отчетов Microsoft Access. Сам запрос не содержит данных, но позволяет выбирать данные из таблиц и выполнять над ними ряд операций.

Выделяют следующие виды запросов:

Запрос на выборку, при котором происходит извлечение данных по указанному условию. В рамках данного запроса происходит группировка записей, а выполненные в полях таблицы вычисления представляются.

Запрос на изменение, при котором изменяются данные в первоначальных таблицах. С помощью них данные могут подвергаться корректировке, а также могут создаваться новые таблицы.

Запрос с параметрами, при котором вводятся определенные условия или данные.

Перекрестные запросы, предназначенные для расчетов и предоставления данных, как правило, в форме электронных таблиц с целью облегчения анализа.  
[6]

Процесс создания запроса «Активные задачи»:

- 1) В Access переходим в раздел «Создания»
- 2) Выбираем «Конструктор запросов»
- 3) В открывшемся окне выбрать таблицы, которые будут участвовать в запросе
- 4) Добавляем поля, которые будут участвовать в запросе
- 5) Введём условие отбора нужного нам поля
- 6) Сохраняем запрос и вводим название

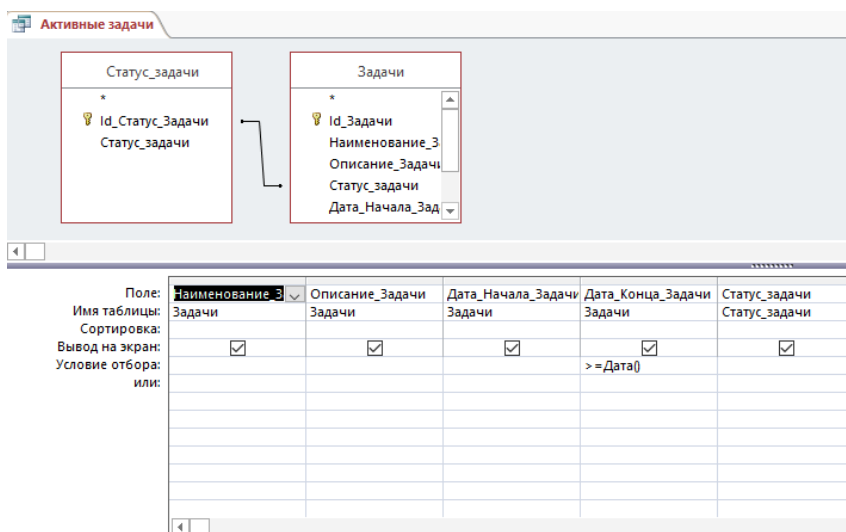


Рис. 4.2.1. Структура запроса «Активные задачи» в режиме конструктора

Аналогично созданы запросы «Время уведомления», «Все задачи», «Выполненные задачи», «Задачи по приоритету», «Не выполненные задачи», «Повторяющиеся задачи», «Поиск по дате» (рис. 4.2.2-4.2.8).

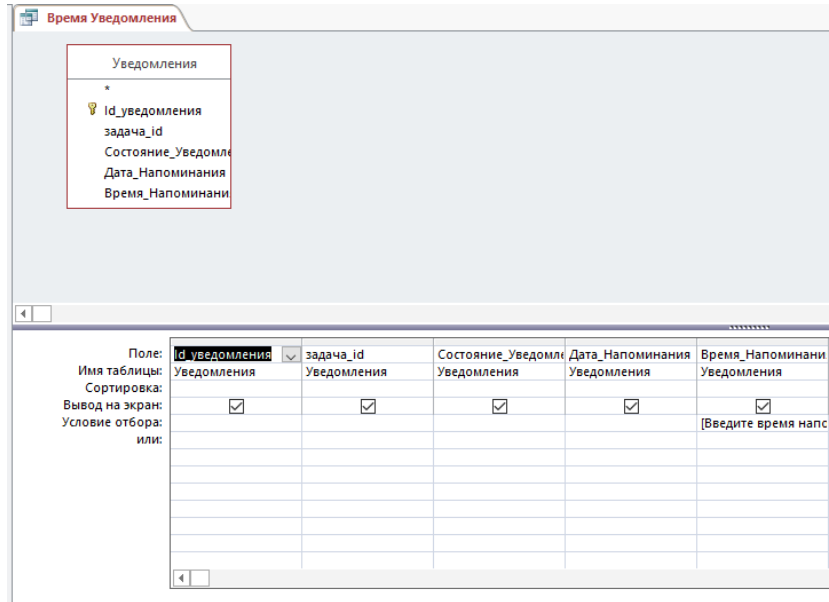


Рис. 4.2.2. Структура запроса «Время уведомления» в режиме конструктора

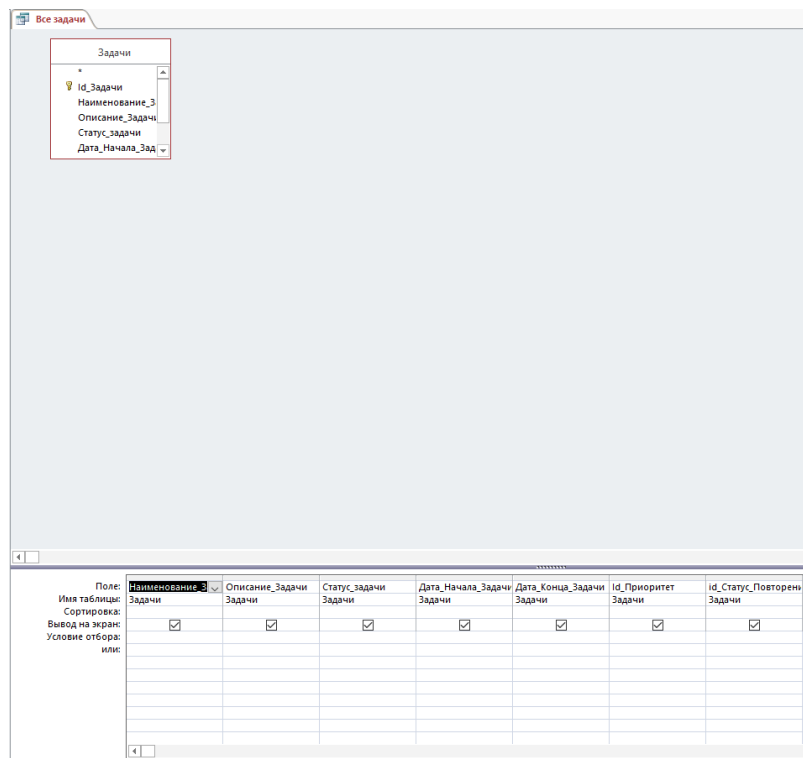


Рис. 4.2.3. Структура запроса «Все задачи» в режиме конструктора

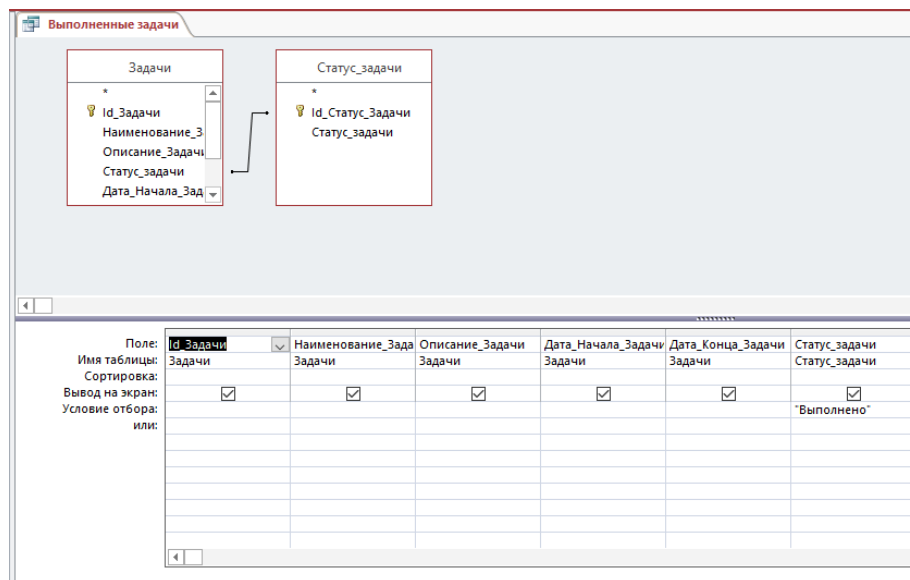


Рис. 4.2.4. Структура запроса «Выполняющие задачи» в режиме конструктора

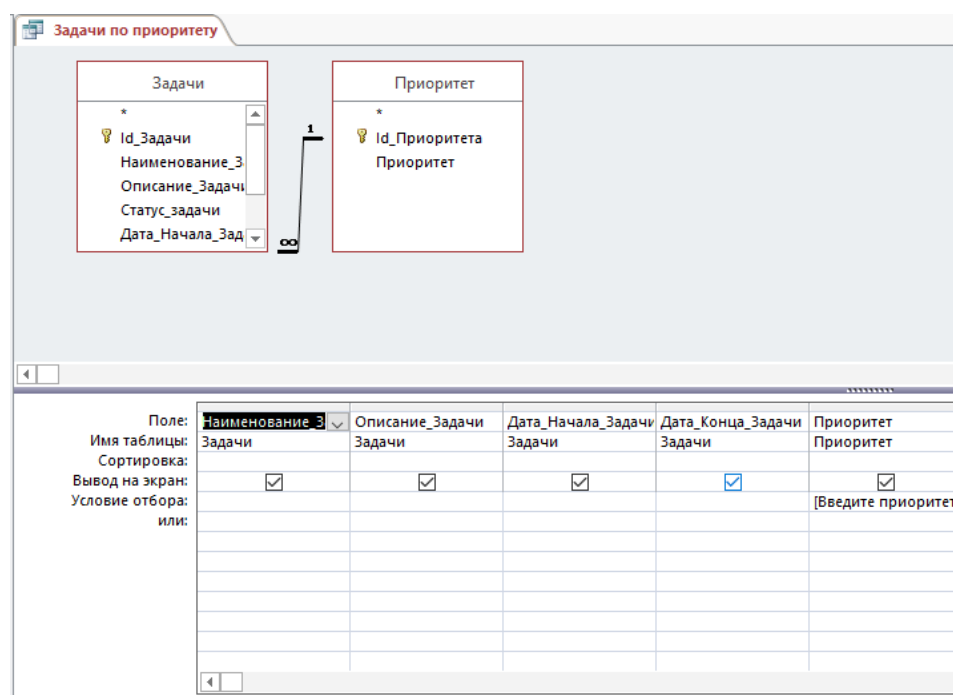


Рис. 4.2.5. Структура запроса «Задачи по приоритету» в режиме конструктора

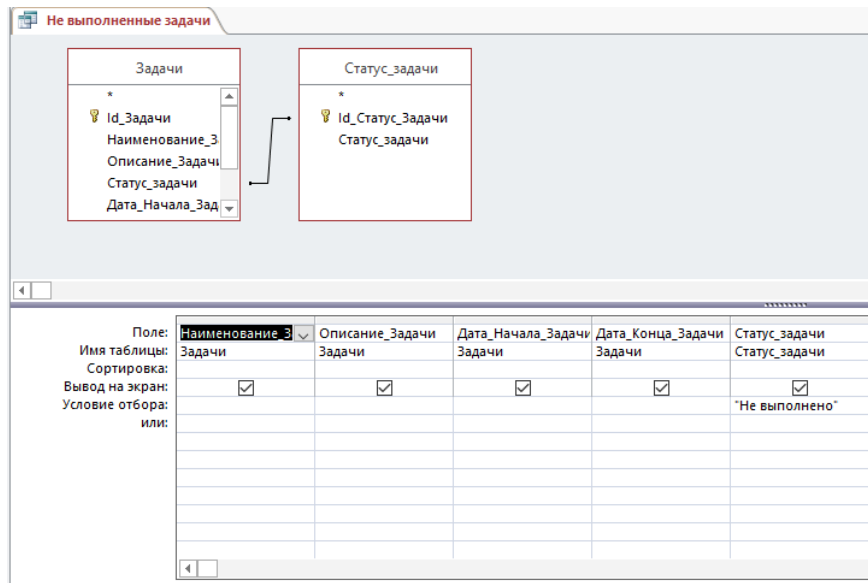


Рис. 4.2.6. Структура запроса «Не выполненные задачи»  
в режиме конструктора

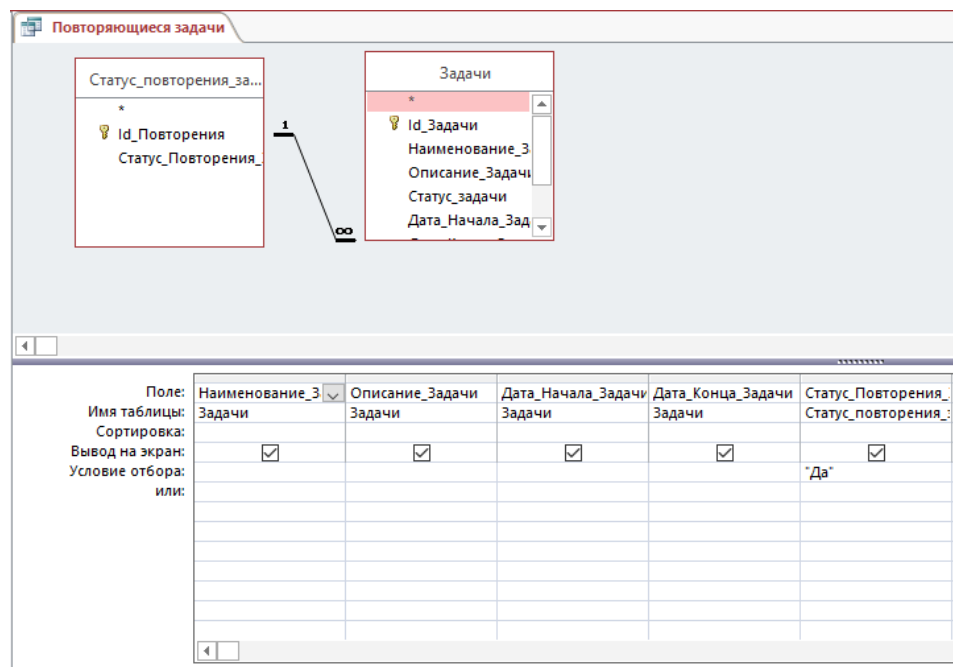


Рис. 4.2.7. Структура запроса «Повторяющиеся задачи» в  
режиме конструктора

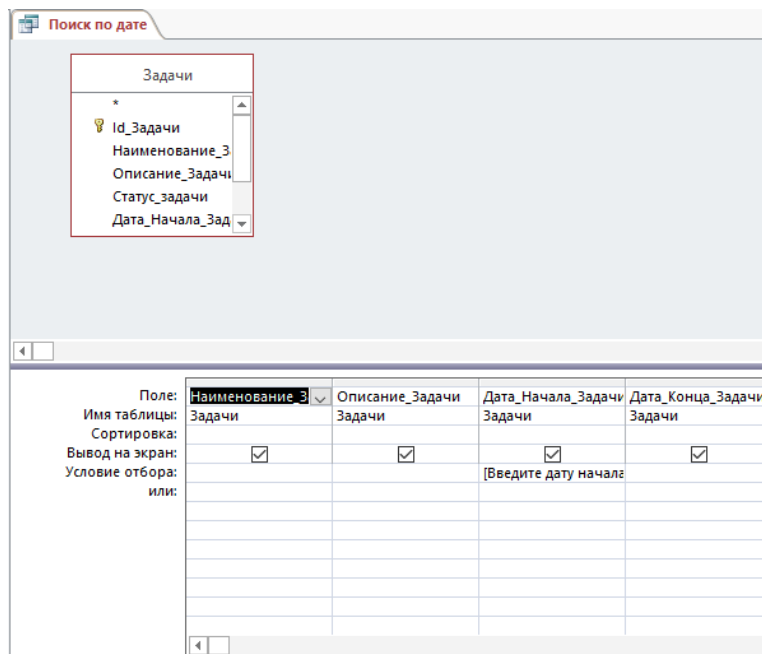


Рис. 4.2.8. Структура запроса «Поиск по дате» в режиме Конструктора

### 4.3. Создание отчётов

Отчеты позволяют выбрать из базы данных требуемую пользователем информацию и оформить ее в виде документов, которые можно просмотреть и напечатать. Источником данных для отчета может служить таблица или запрос. [3]

Процесс создания отчёта «Активные задачи»:

- 1) Выделяем запрос, для которого создаём отчёт
- 2) Выбираем в Access раздел «Отчёт»
- 3) В созданном отчёте располагаем элементы так, чтобы отчёт поместился на одном листе
- 4) Добавляем кнопку «Печать» и кнопку «Вернуться»
- 5) Сохраняем и вводим название

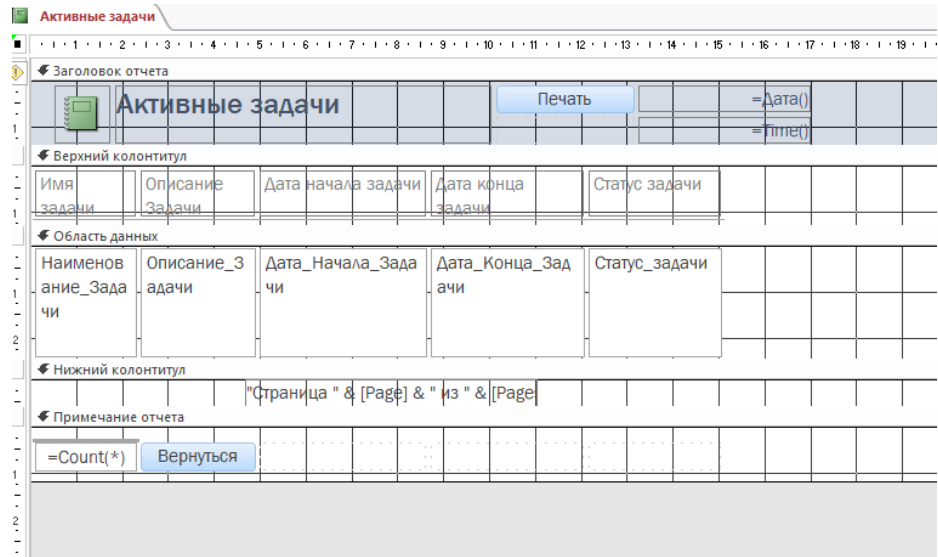


Рис. 4.3.1. Структура отчёта «Активные задачи» в режиме конструктора

Аналогично созданы отчёты «Время уведомления», «Все задачи», «Выполненные задачи», «Задачи по приоритету», «Не выполненные задачи», «Повторяющиеся задачи», «Поиск по дате» рис (4.3.2-4.3.8).

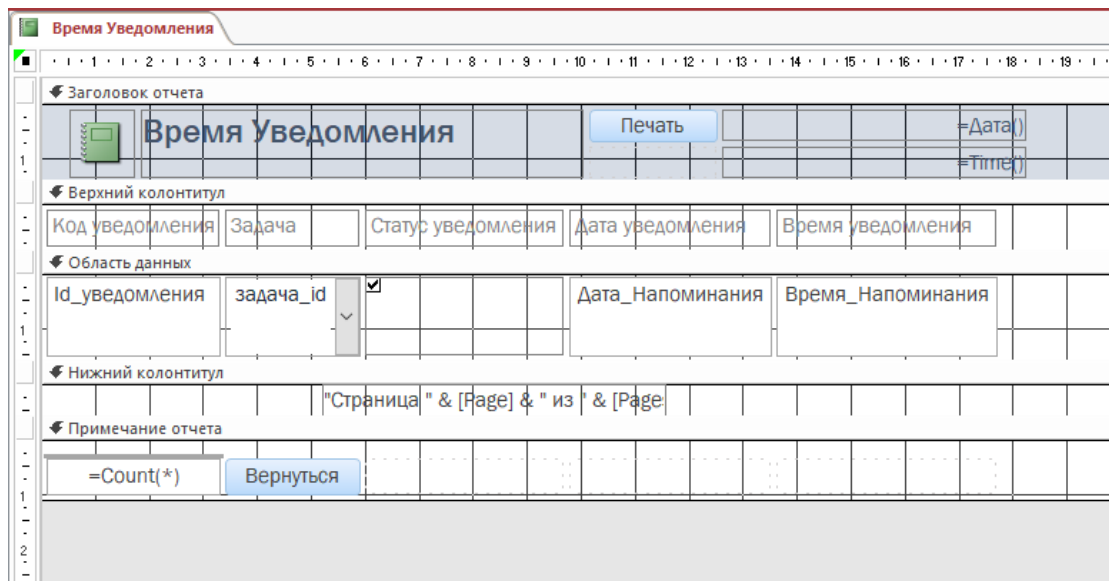


Рис. 4.3.2. Структура отчёта «Время уведомления» в режиме конструктора

| Все задачи                             |                  |               |                    |                   |              |                      |         |  |  |
|--|------------------|---------------|--------------------|-------------------|--------------|----------------------|---------|--|--|
| Заголовок отчета                       |                  |               |                    |                   |              |                      |         |  |  |
| Все задачи                             |                  |               |                    |                   | Печать       |                      | =Дата() |  |  |
|  |                  |               |                    |                   |              |                      | =Time() |  |  |
| Верхний колонтитул                     |                  |               |                    |                   |              |                      |         |  |  |
| Наименование задачи                    | Описание задачи  | Статус задачи | Дата начала задачи | Дата конца задачи | Приоритет    | Статус выполнения    |         |  |  |
| Область данных                         |                  |               |                    |                   |              |                      |         |  |  |
| Наименование_Задачи                    | Описание_Задач и | Статус_задачи | Дата_Начала_Задачи | Дата_Конца_Задачи | Id_Приоритет | id_Статус_Повторения |         |  |  |
| Нижний колонтитул                      |                  |               |                    |                   |              |                      |         |  |  |
| "Страница " & [Page] & " из " & [Page] |                  |               |                    |                   |              |                      |         |  |  |
| Примечание отчета                      |                  |               |                    |                   |              |                      |         |  |  |
| =Count(*)                              |                  | Вернуться     |                    |                   |              |                      |         |  |  |

Рис. 4.3.3. Структура отчёта «Все задачи» в режиме конструктора

| Выполненные задачи                     |               |                 |                    |                   |               |  |         |  |  |
|--|---------------|-----------------|--------------------|-------------------|---------------|--|---------|--|--|
| Заголовок отчета                       |               |                 |                    |                   |               |  |         |  |  |
| Выполненные задачи                     |               |                 |                    |                   | Печать        |  | =Дата() |  |  |
|  |               |                 |                    |                   |               |  | =Time() |  |  |
| Верхний колонтитул                     |               |                 |                    |                   |               |  |         |  |  |
| Код задачи                             | Имя задачи    | Описание задачи | Дата начала задачи | Дата конца задачи | Статус задачи |  |         |  |  |
| Область данных                         |               |                 |                    |                   |               |  |         |  |  |
| Id_Задачи                              | Наименование_ | Описание_Задачи | Дата_Начала_Задач  | Дата_Конца_Задачи | Статус_задач  |  |         |  |  |
| Нижний колонтитул                      |               |                 |                    |                   |               |  |         |  |  |
| "Страница " & [Page] & " из " & [Page] |               |                 |                    |                   |               |  |         |  |  |
| Примечание отчета                      |               |                 |                    |                   |               |  |         |  |  |
| =Count(*)                              |               | Вернуться       |                    |                   |               |  |         |  |  |

Рис. 4.3.4. Структура отчёта «Выполненные задачи» в режиме конструктора

| Задачи по приоритету                   |                 |                    |                   |           |        |  |         |  |  |
|--|-----------------|--------------------|-------------------|-----------|--------|--|---------|--|--|
| Заголовок отчета                       |                 |                    |                   |           |        |  |         |  |  |
| Задачи по приоритету                   |                 |                    |                   |           | Печать |  | =Дата() |  |  |
|  |                 |                    |                   |           |        |  | =Time() |  |  |
| Верхний колонтитул                     |                 |                    |                   |           |        |  |         |  |  |
| Имя задачи                             | Описание задачи | Дата начала задачи | Дата конца задачи | Приоритет |        |  |         |  |  |
| Область данных                         |                 |                    |                   |           |        |  |         |  |  |
| Наименование_Задачи                    | Описание_Задачи | Дата_Начала_Задачи | Дата_Конца_Задачи | Приоритет |        |  |         |  |  |
| Нижний колонтитул                      |                 |                    |                   |           |        |  |         |  |  |
| "Страница " & [Page] & " из " & [Page] |                 |                    |                   |           |        |  |         |  |  |
| Примечание отчета                      |                 |                    |                   |           |        |  |         |  |  |
| =Count(*)                              |                 | Вернуться          |                   |           |        |  |         |  |  |

Рис. 4.3.5. Структура отчёта «Задачи по приоритету» в режиме конструктора



| Не выполненные задачи                  |  |                 |  |                    |  |                   |  |               |  |
|--|--|-----------------|--|--------------------|--|-------------------|--|---------------|--|
| Заголовок отчета                       |  |                 |  |                    |  |                   |  |               |  |
| Не выполненные задачи                  |  |                 |  |                    |  |                   |  |               |  |
| Печать                                 |  |                 |  |                    |  |                   |  |               |  |
| =Дата()                                |  |                 |  |                    |  |                   |  |               |  |
| =Типте()                               |  |                 |  |                    |  |                   |  |               |  |
| Верхний колонтитул                     |  |                 |  |                    |  |                   |  |               |  |
| Имя задачи                             |  | Описание задачи |  | Дата начала задачи |  | Дата конца задачи |  | Статус задачи |  |
| Область данных                         |  |                 |  |                    |  |                   |  |               |  |
| Наименование                           |  | Описание_Задачи |  | Дата_Начала_Задачи |  | Дата_Конца_Задачи |  | Статус_задачи |  |
| Нижний колонтитул                      |  |                 |  |                    |  |                   |  |               |  |
| "Страница " & [Page] & " из " & [Page] |  |                 |  |                    |  |                   |  |               |  |
| Примечание отчета                      |  |                 |  |                    |  |                   |  |               |  |
| =Count(*)                              |  | Вернуться       |  |                    |  |                   |  |               |  |

Рис. 4.3.6. Структура отчёта «Не выполненные задачи» в режиме конструктора

| Повторяющиеся задачи                   |  |                 |  |                    |  |                   |  |                          |  |
|--|--|-----------------|--|--------------------|--|-------------------|--|--------------------------|--|
| Заголовок отчета                       |  |                 |  |                    |  |                   |  |                          |  |
| Повторяющиеся задачи                   |  |                 |  |                    |  |                   |  |                          |  |
| Печать                                 |  |                 |  |                    |  |                   |  |                          |  |
| =Дата()                                |  |                 |  |                    |  |                   |  |                          |  |
| =Типте()                               |  |                 |  |                    |  |                   |  |                          |  |
| Верхний колонтитул                     |  |                 |  |                    |  |                   |  |                          |  |
| Имя задачи                             |  | Описание задачи |  | Дата начала задачи |  | Дата конца задачи |  | Статус повторения задачи |  |
| Область данных                         |  |                 |  |                    |  |                   |  |                          |  |
| Наименование_За                        |  | Описание_Задачи |  | Дата_Начала_Зада   |  | Дата_Конца_Зада   |  | Статус_Повторения_Зада   |  |
| Нижний колонтитул                      |  |                 |  |                    |  |                   |  |                          |  |
| "Страница " & [Page] & " из " & [Page] |  |                 |  |                    |  |                   |  |                          |  |
| Примечание отчета                      |  |                 |  |                    |  |                   |  |                          |  |
| =Count(*)                              |  | Вернуться       |  |                    |  |                   |  |                          |  |

Рис. 4.3.7. Структура отчёта «Повторяющиеся задачи» в режиме конструктора

| Поиск по дате                          |  |                 |  |                    |  |                   |  |  |  |
|--|--|-----------------|--|--------------------|--|-------------------|--|--|--|
| Заголовок отчета                       |  |                 |  |                    |  |                   |  |  |  |
| Поиск по дате                          |  |                 |  |                    |  |                   |  |  |  |
| Печать                                 |  |                 |  |                    |  |                   |  |  |  |
| =Дата()                                |  |                 |  |                    |  |                   |  |  |  |
| =Типте()                               |  |                 |  |                    |  |                   |  |  |  |
| Верхний колонтитул                     |  |                 |  |                    |  |                   |  |  |  |
| Имя задачи                             |  | Описание задачи |  | Дата начала задачи |  | Дата конца задачи |  |  |  |
| Область данных                         |  |                 |  |                    |  |                   |  |  |  |
| Наименование_                          |  | Описание_Задачи |  | Дата_Начала_Задачи |  | Дата_Конца_Задачи |  |  |  |
| Нижний колонтитул                      |  |                 |  |                    |  |                   |  |  |  |
| "Страница " & [Page] & " из " & [Page] |  |                 |  |                    |  |                   |  |  |  |
| Примечание отчета                      |  |                 |  |                    |  |                   |  |  |  |
| =Count(*)                              |  | Вернуться       |  |                    |  |                   |  |  |  |

Рис. 4.3.8. Структура отчёта «Поиск по дате» в режиме конструктора

#### 4.4. Создание форм

Форма — это объект Access, без которого можно обойтись. Он создан специально для пользователя. Экран принимает удобный для обработки информации вид с учетом того, что пользователь может работать в *БД* несколько часов. Оттого, каким будет экран компьютера все это время, зависит настроение и здоровье. Часто формы приближают к виду уже известных бумажных документов. Такой подход позволяет специалистам других отраслей легко перейти на электронные документы. [3]

Процесс создания формы «Main»:

- 1) Раздел создания
- 2) Нажимаем Конструктор формы
- 3) Добавляем объект «Вкладка»
- 4) Добавляем в заголовок формы кнопку выхода из формы
- 5) Во вкладке «Задачи» добавляем поля «Наименование задачи», «Описание задачи», «Статус задачи», «Дата начала задачи», «Дата конца задачи»
- 6) На этой же вкладке добавляем кнопки «настроить уведомление», «не выполненные», «выполненные», «активные задачи», «поиск по дате», «все задачи», «добавить», «сохранить», «удалить» и кнопки навигации
- 7) Во вкладке «Приоритет» добавляем поля «Наименование задачи», «Описание задачи», «Id Приоритет» и кнопки «задачи по приоритету», «добавить», «сохранить», «удалить» и кнопки навигации
- 8) Во вкладке «Статус Повторения» добавим поля «Наименование задачи», «Описание задачи», «Id Статус Повторения» и кнопки «повторяющиеся задачи», «добавить», «сохранить», «удалить» и кнопки навигации
- 9) Во вкладке «Список Уведомления» добавим таблицу «Уведомление» и кнопки навигации

|      |       |          |       |      |                                   |      |
|------|-------|----------|-------|------|-----------------------------------|------|
|      |       |          |       |      | КП - 09.02.07 – 11111 – 2023 – ПЗ | Лист |
| Изм. | Лист. | № докум. | Подп. | Дата |                                   | 26   |

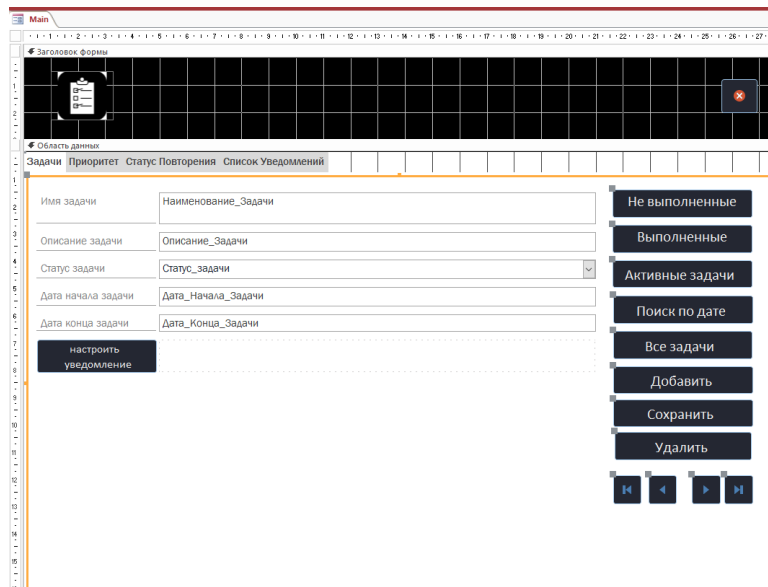


Рис. 4.4.1. Структура формы «Main» во вкладке «Задачи» в режиме конструктора

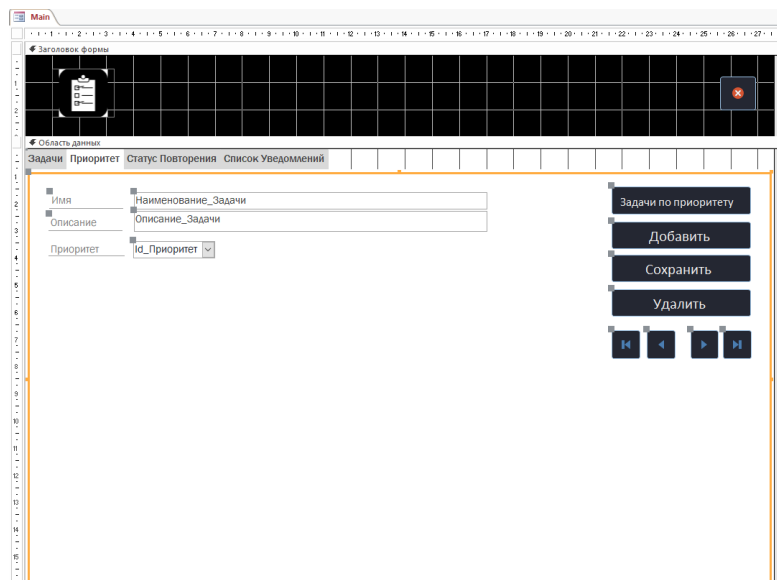


Рис. 4.4.2. Структура формы «Main» во вкладке «Приоритет» в режиме конструктора

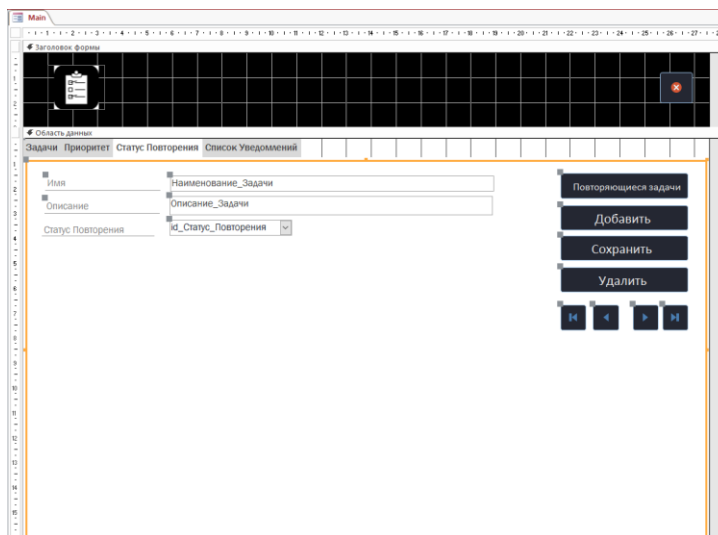


Рис. 4.4.3. Структура формы «Main» во вкладке «Статус Повторения» в режиме конструктора

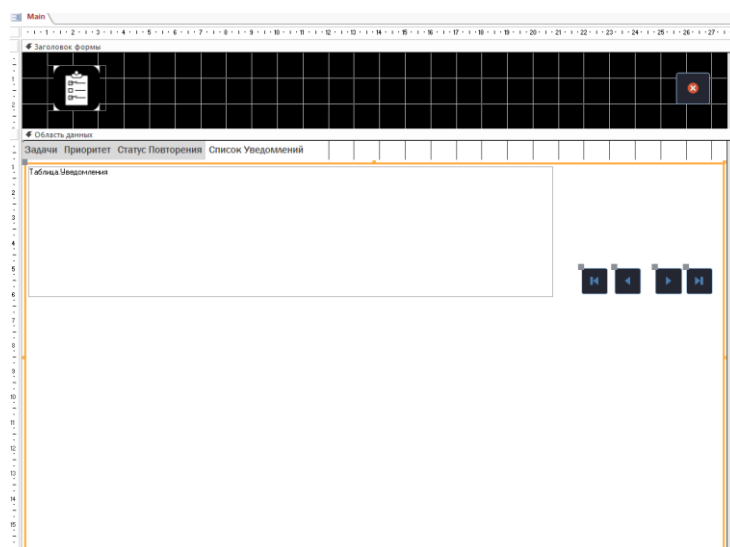


Рис. 4.4.4. Структура формы «Main» во вкладке «Список уведомлений» в режиме конструктора

Процесс создания формы «Уведомления»:

- 1) Раздел создания
- 2) Нажимаем Конструктор формы
- 3) Добавляем объект «Вкладка»
- 4) Добавляем в заголовок формы кнопку выхода из формы
- 5) Добавляем в заголовок формы кнопку возвращения на форму «Main»

- б) Добавляем поля «задача id», «состояние уведомления», «Дата напоминания», «Время напоминания», а также кнопки «время уведомления», «добавить», «сохранить», «удалить» и кнопки навигации

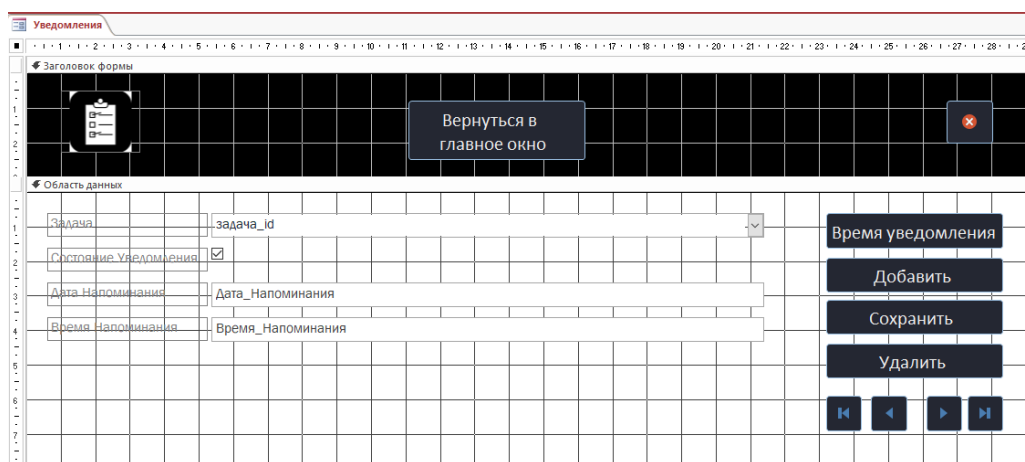


Рис. 4.4.5. Структура формы «Уведомления» в режиме конструктора

## ЗАКЛЮЧЕНИЕ

В соответствии с заданием на курсовой проект была разработана база данных «Ежедневник», которая помогает пользователю составить список своих задач на день, что позволяет эффективнее использовать свое время.

При запуске программы открывается форма «Main», содержащая основные функции ежедневника. Через эту форму можно зайти на форму «Уведомление» и просмотреть все отчёты.

При выполнении курсового проекта были выполнены следующие задачи:

- проведён анализ предметной области;
- разработана логическая модель базы данных;
- разработана реляционная модель базы данных:
  - структуры таблиц базы данных;
  - выбраны ключевые поля;
  - выполнена нормализация отношений;
  - реализована база данных в СУБД;
  - созданы таблицы и связи между ними;
  - созданы запросы к базе данных;
  - созданы формы;
  - созданы отчёты.

|      |       |          |       |      |                                   |      |
|------|-------|----------|-------|------|-----------------------------------|------|
|      |       |          |       |      | КП - 09.02.07 – 11111 – 2023 – ПЗ | Лист |
| Изм. | Лист. | № докум. | Подп. | Дата |                                   | 30   |

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

### Литература

1. Семакин И.Г. Основы программирования и баз данных: учеб. для студ. учрежд. СПО – 2-е издание, стер. – М.: ИЦ «Академия», 2017 – 224с.
2. Фёдорова Г.Н. Основы проектирования баз данных: учебник для студ. учрежд. СПО / Г.Н.Фёдорова – М.: ИЦ «Академия», 2017 – 224с.
3. Фёдорова Г.Н. Разработка, администрирование и защита баз данных: учебник для студ. учрежд. СПО / Г.Н.Фёдорова – М.: ИЦ «Академия», 2018 – 288с.
4. Федорова Г.Н. Сопровождение информационных систем: учеб. для студ. учреждений сред. проф. образования/Г.Н. Федорова. - М.: Издательский центр "Академия", 2018. - 320 с.

### Электронные издания (электронные ресурсы)

5. Баженова, И. Ю. Основы проектирования приложений баз данных: учебное пособие для СПО / И. Ю. Баженова. — Саратов : Профобразование, 2019. — 325 с. — ISBN 978-5-4488-0361-1. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование : [сайт]. — URL: <https://profspo.ru/books/86200>
6. Разработка и защита баз данных в Microsoft SQL Server 2005 : учебное пособие для СПО / . — Саратов : Профобразование, 2019. — 148 с. — ISBN 978-5-4488-0366-6. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование : [сайт]. — URL: <https://profspo.ru/books/86207>
7. Швецов, В. И. Базы данных : учебное пособие для СПО / В. И. Швецов. — Саратов : Профобразование, 2019. — 219 с. — ISBN 978-5-4488-0357-4. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование : [сайт]. — URL: <https://profspo.ru/books/86192>
8. Общая характеристика и функции органайзера: [сайт]. — URL: [https://studwood.net/1589650/informatika/analiz\\_predmetnoy\\_oblasti](https://studwood.net/1589650/informatika/analiz_predmetnoy_oblasti)

|      |       |          |       |      |                                   |      |
|------|-------|----------|-------|------|-----------------------------------|------|
|      |       |          |       |      | КП - 09.02.07 – 11111 – 2023 – ПЗ | Лист |
| Изм. | Лист. | № докум. | Подп. | Дата |                                   | 31   |