

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Шебзухова Татьяна Александровна

Должность: Директор Пятигорского института (филиал) Северо-Кавказского
федерального университета

Дата подписания: 13.06.2023 12:07:35

Уникальный программный ключ:

d74ce93cd40e39275c3ba2f58486412a1c8ef966f

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**

**Федеральное государственное автономное образовательное учреждение
высшего образования**

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Пятигорский институт (филиал) СКФУ

Колледж Пятигорского института (филиал) СКФУ

УТВЕРЖДАЮ

**Директор Пятигорского института
(филиал) СКФУ**

Т.А.Шебзухова

**ПМ.07 СОАДМИНИСТРИРОВАНИЕ БАЗ ДАННЫХ И СЕРВЕРОВ
ПМ.07.01 УПРАВЛЕНИЕ И АВТОМАТИЗАЦИЯ БАЗ ДАННЫХ
МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ
РАБОТ**

Специальности СПО

09.02.07 Информационные системы и программирование

Пятигорск 2023

Методические указания для выполнения лабораторных работ по дисциплине ПМ.07.01 Управление и автоматизация баз данных составлены в соответствии с требованиями ФГОС СПО. Предназначены для студентов, обучающихся по специальности 09.02.07 Информационные системы и программирование.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

При изучении предмета следует соблюдать единство терминологии и обозначения в соответствии с действующими стандартами, Международной системой единицы (СИ).

В результате изучения Управления и автоматизация баз данных студенты *должны*:

знать:

модели данных, основные операции и ограничения;
технологии установки и настройки сервера баз данных;
требования к безопасности сервера базы данных;
государственные стандарты и требования к обслуживанию баз данных.

уметь:

проектировать и создавать базы данных;
выполнять запросы по обработке данных на языке SQL;
осуществлять основные функции по администрированию баз данных;
разрабатывать политику безопасности SQL сервера, базы данных и отдельных объектов базы данных;

владеть технологиями проведения сертификации программного средства.

иметь практический опыт в:

участии в соадминистрировании серверов;
разработке политики безопасности SQL сервера, базы данных и отдельных объектов базы данных;
применении законодательства Российской Федерации в области сертификации программных средств информационных технологий.

В результате освоения учебной дисциплины студент должен овладевать:

Общими компетенциями:

ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.

ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.

ОК 09. Использовать информационные технологии в профессиональной деятельности.

ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке.

Профессиональными компетенциями:

ПК 7.1. Выявлять технические проблемы, возникающие в процессе эксплуатации баз данных и серверов.

ПК 7.2. Осуществлять администрирование отдельных компонент серверов.

ПК 7.3. Формировать требования к конфигурации локальных компьютерных сетей и серверного оборудования, необходимые для работы баз данных и серверов.

ПК 7.4. Осуществлять администрирование баз данных в рамках своей компетенции.

ПК 7.5. Проводить аудит систем безопасности баз данных и серверов с использованием регламентов по защите информации.

ОБЩИЕ МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

По дисциплине Управление и автоматизация баз данных лабораторные работы содержат задачи и теоретические вопросы. Варианты для каждого обучающегося - индивидуальные. Задачи и ответы на вопросы, выполненные не по своему варианту, не засчитываются.

Лабораторная работа выполняется в отдельной тетради. Условия задачи и формулировки вопросов переписываются полностью. Формулы, расчеты, ответы на вопросы пишутся ручкой, а чертежи, схемы и рисунки выполняются карандашом, на графиках и диаграммах указывается масштаб. Вначале задача решается в общем виде, затем делаются расчёты по условию задания. Решение задач обязательно ведется в Международной системе единиц (СИ).

При выполнении лабораторной работы необходимо следовать методическим указаниям: повторить краткое содержание теории, запомнить основные формулы и законы, проанализировать пример выполнения аналогичного задания, затем преступить непосредственно к решению задачи. К зачету допускаются студенты, получившие положительные оценки по всем лабораторным работам.

Правила выполнения лабораторных работ.

1. Студент должен прийти на лабораторную работу подготовленным к выполнению лабораторной работы.
2. Каждый студент после проведения работы должен представить отчет о проделанной работе с анализом полученных результатов и выводом по работе.
3. Таблицы и рисунки следует выполнять с помощью чертежных инструментов (линейки, циркуля, и т.д.) карандашом с соблюдением ЕСКД.
4. Расчет следует проводить с точностью до двух значащих цифр.
5. Исправления проводить на обратной стороне листа. При мелких исправлениях неправильное слово (буква, число и т.п.) аккуратно зачеркивается и над ним пишут правильное пропущенное слово (букву, число и т.п.).
6. Вспомогательные расчеты можно выполнять на отдельных листах, а при необходимости на листах отчета.
7. Если студент не выполнит лабораторную работу или часть работы, то он выполнит ее во внеурочное время, согласованное с преподавателем.
8. Оценку по практической работе студент получает с учетом срока выполнения работы, если;
 - расчеты выполнены правильно и в полном объеме;
 - сделан анализ проделанной работы и вывод по результатам работы;
 - студент может пояснить выполнение любого этапа работы;
 - отчет выполнен в соответствии с требованиями к выполнению работы.

Лабоаторная работа №1.

Создание базы данных

Целью практической работы является знакомство с интерфейсом СУБД Access, изучение схемы данных, назначения и видов индексов в таблицах, типов отношений между таблицами, а также приобретение практических навыков по созданию новой многотабличной БД с использованием различных инструментов СУБД Access, созданию индексов для каждой таблицы БД, связыванию таблиц созданной базы данных с помощью схемы данных и реализации работы с данными в таблицах (заполнение, редактирование, сортировка и фильтрация данных). Задание рассчитано на 6 часов аудиторных занятий и 4 часа самостоятельной работы.

Порядок выполнения практической работы.

1. Создать новую базу данных и сохранить в личном каталоге.
2. В БД определить три новые таблицы со структурой полей согласно номеру варианта задания. Для создания двух таблиц использовать конструктор, для третьей — мастер создания таблиц.

Лабоаторная работа №2.

Заполнение таблиц базы данных.

Целью практической работы является знакомство с интерфейсом СУБД Access, изучение схемы данных, назначения и видов индексов в таблицах, типов отношений между таблицами, а также приобретение практических навыков по созданию новой многотабличной БД с использованием различных инструментов СУБД Access, созданию индексов для каждой таблицы БД, связыванию таблиц созданной базы данных с помощью схемы данных и реализации работы с данными в таблицах (заполнение, редактирование, сортировка и фильтрация данных). Задание рассчитано на 6 часов аудиторных занятий и 4 часа самостоятельной работы.

Порядок выполнения практической работы.

1. Внести в таблицы БД по 5 – 10 записей, содержание которых соответствует семантике и типу полей.
2. Выполнить операции удаления, добавления, копирования и редактирования записей таблиц БД.
3. Определить требуемые индексы для таблиц БД (первичный ключ и вторичные ключи).

Лабоаторная работа №3.

Создание связей таблиц базы данных.

Целью практической работы является знакомство с интерфейсом СУБД Access, изучение схемы данных, назначения и видов индексов в таблицах, типов отношений между таблицами, а также приобретение практических навыков по созданию новой многотабличной БД с использованием различных инструментов СУБД Access, созданию индексов для каждой таблицы БД, связыванию таблиц созданной базы данных с помощью схемы данных и реализации работы с данными в таблицах (заполнение, редактирование, сортировка и фильтрация данных). Задание рассчитано на 6 часов аудиторных занятий и 4 часа самостоятельной работы.

Порядок выполнения практической работы

1. Установить связи между таблицами БД.
2. Ответить на один из контрольных вопросов к заданию.

Вариант 1. База данных «Студенческая научная работа»

Кафедры

Название поля	Тип поля	Длина	Длина дробной части
Код кафедры	Числовой	Целое	–
<i>Название кафедры</i>	Текстовый	40	–
<i>Телефон</i>	Числовой	Длинное целое	–

Индекс: *Код кафедры*.

Студенты

Название поля	Тип поля	Длина	Длина дробной части
Номер зачетной книжки	Текстовый	6	–
<i>Фамилия</i>	Текстовый	20	–
<i>Имя</i>	Текстовый	20	–
<i>Отчество</i>	Текстовый	20	–
<i>Группа</i>	Текстовый	3	–
<i>Табельный номер научного руководителя</i>	Числовой	Целое	–
<i>Дата рождения</i>	Дата / Время	Краткий формат	–
<i>Бюджетное / Платное обучение</i>	Логический	–	–

Индексы: *Код научного руководителя, Номер зачетной книжки + Группа*.

Научные руководители

Название поля	Тип поля	Длина	Длина дробной части
Табельный номер научного руководителя	Числовой	Целое	–

<i>Фамилия</i>	Текстовый	20	–
<i>Имя</i>	Текстовый	20	–
<i>Отчество</i>	Текстовый	20	–
<i>Телефон</i>	Числовой	Длинное целое	–
<i>Домашний адрес</i>	Текстовый	40	–
<i>Код кафедры</i>	Числовой	Целое	–

Индекс: *Табельный номер научного руководителя, Код кафедры, Табельный номер научного руководителя + Код кафедры.*

Вариант 2. База данных «Абоненты телефонной компании»

Абоненты

Название поля	Тип поля	Длина	Длина дробной части
<i>Регистрационный номер абонента</i>	Числовой	Целое	–
<i>Фамилия</i>	Текстовый	20	–
<i>Имя</i>	Текстовый	20	–
<i>Отчество</i>	Текстовый	20	–
<i>Улица</i>	Текстовый	30	–
<i>Дом</i>	Числовой	Целое	–
<i>Квартира</i>	Числовой	Целое	–

Индексы: *Регистрационный номер абонента, Улица+Регистрационный номер абонента.*

Телефоны

Название поля	Тип поля	Длина	Длина дробной части
<i>Номер телефона</i>	Числовой	Длинное целое	–
<i>Регистрационный номер</i>	Числовой	Целое	–

<i>мер абонента</i>			
<i>Тариф оплаты за телефон</i>	Денежный	Основной	2

Индексы: *Номер телефона, Регистрационный номер абонента.*

Оплата

Название поля	Тип поля	Длина	Длина дробной части
<i>Номер телефона</i>	Числовой	Длинное целое	–
<i>Дата оплаты</i>	Дата / Время	Длинный формат	–
<i>Сумма оплаты</i>	Денежный	Основной	2

Индекс: *Номер телефона + Дата оплаты, Номер телефона, Дата оплаты.*

Вариант 3. База данных «Статистика медицинского учреждения»

Диагнозы

Название поля	Тип поля	Длина	Длина дробной части
<i>Код диагноза</i>	Текстовый	6	–
<i>Диагноз</i>	Текстовый	25	–
<i>Нормативное количество дней для стационарного лечения</i>	Числовой	Целое	–

Индекс: *Код диагноза.*

Отделения

Название поля	Тип поля	Длина	Длина дробной части
<i>Код отделения</i>	Числовой	Целое	–

Название отделения	Текстовый	40	–
--------------------	-----------	----	---

Индекс: Код отделения.

Пациенты

Название поля	Тип поля	Длина	Длина дробной части
Номер карты	Числовой	Целое	–
<i>Фамилия</i>	Текстовый	20	–
<i>Имя</i>	Текстовый	20	–
<i>Отчество</i>	Текстовый	20	–
<i>Ребенок / Взрослый</i>	Логический	–	–
<i>Адрес</i>	Текстовый	40	–
<i>Код диагноза</i>	Текстовый	6	–
<i>Дата поступления</i>	Дата / Время	Краткий формат	–
<i>Код отделения</i>	Числовой	Целое	–

Индексы: Номер карты, Код отделения, Код диагноза, Дата поступления + Номер карты, Взрослый / Ребенок + Номер карты.

Вариант 4. База данных «Заработная плата сотрудников»

Персонал

Название поля	Тип поля	Длина	Длина дробной части
Табельный номер сотрудника	Числовой	Целое	–
<i>Фамилия</i>	Текстовый	20	–
<i>Имя</i>	Текстовый	20	–
<i>Отчество</i>	Текстовый	20	–

<i>Пол (Мужской / Женский)</i>	Логический	–	
<i>Код должности</i>	Числовой	Целое	–
<i>Номер отдела</i>	Числовой	Целое	–

Индекс: *Табельный номер сотрудника, Код должности + Табельный номер сотрудника, Номер отдела + Табельный номер сотрудника.*

Должности

Название поля	Тип поля	Длина	Длина дробной части
<i>Код должности</i>	Числовой	Целое	–
<i>Должность</i>	Текстовый	20	–
<i>Номер разряда</i>	Числовой	Байт	–

Индексы: *Код должности, Номер разряда.*

Тарифная сетка

Название поля	Тип поля	Длина	Длина дробной части
<i>Номер разряда</i>	Числовой	Байт	–
<i>Оклад</i>	Денежный	Основной	2

Индекс: *Номер разряда, Оклад.*

Вариант 5. База данных «Учет прихода товара»

Товары

Название поля	Тип поля	Длина	Длина дробной части
<i>Штрих-код</i>	Числовой	Целое	–
<i>Наименование товара</i>	Текстовый	40	–

<i>Номер товарной группы</i>	Числовой	Целое	–
<i>Единица измерения</i>	Текстовый	4	–
<i>Цена товара за единицу измерения</i>	Денежный	Основной	2

Индексы: *Штрих-код, Номер товарной группы + Штрих-код.*

Товарные группы

Название поля	Тип поля	Длина	Длина дробной части
<i>Номер товарной группы</i>	Числовой	Целое	–
<i>Название группы</i>	Текстовый	40	–

Индекс: *Номер товарной группы.*

Приход

Название поля	Тип поля	Длина	Длина дробной части
<i>Дата прихода</i>	Дата / время	Краткий формат	–
<i>Штрих-код</i>	Числовой	Целое	–
<i>Количество</i>	Числовой	Одинарное с плавающей точкой	2
<i>Поставщик</i>	Текстовый	30	–

Индекс: *Дата прихода+Штрих-код, Штрих-код,Поставщик + Штрих-код.*

Вариант 6. База данных «Расписание движения поездов»

Пункты назначения

Название поля	Тип поля	Длина	Длина дробной части
---------------	----------	-------	---------------------

<i>Код пункта назначения</i>	Текстовый	6	–
<i>Название пункта назначения</i>	Текстовый	30	–
<i>Время следования</i>	Числовой	Байт	–
<i>Расстояние</i>	Числовой	Целое	–

Индекс: *Код пункта назначения, Расстояние + Код пункта назначения, Время следования + Код пункта назначения.*

Поезда

Название поля	Тип поля	Длина	Длина дробной части
<i>Номер поезда</i>	Числовой	Целое	–
<i>Нормативное количество посадочных мест</i>	Числовой	Целое	–
<i>Проходящий / Местный</i>	Логический	–	–

Индекс: *Номер поезда, Проходящий/Местный+Номер поезда.*

Расписание

Название поля	Тип поля	Длина	Длина дробной части
<i>Номер поезда</i>	Числовой	Целое	–
<i>Прибытие / Убытие</i>	Логический		–
<i>Код пункта назначения</i>	Текстовый	6	–
<i>Дата прибытия (убытия) + Время прибытия (убытия)</i>	Дата / Время	Полный формат	–

Индексы: *Номер поезда, Код пункта назначения, Прибытие/Убытие+Но-мер поезда.*

Вариант 7. База данных «Кадры»

Отделы предприятия

Название поля	Тип поля	Длина	Длина дробной части
<i>Номер отдела</i>	Числовой	Целое	–
<i>Название отдела</i>	Текстовый	30	–
<i>Фамилия руководителя</i>	Текстовый	20	–

Индекс: *Номер отдела.*

Сотрудники

Название поля	Тип поля	Длина	Длина дробной части
<i>Табельный номер сотрудника</i>	Числовой	Целое	–
<i>Фамилия</i>	Текстовый	20	–
<i>Имя</i>	Текстовый	20	–
<i>Отчество</i>	Текстовый	20	–
<i>Дата рождения</i>	Дата / Время	Краткий формат	–
<i>Пол</i>	Логический	–	–
<i>Адрес</i>	Текстовый	40	–

Индекс: *Табельный номер сотрудника, Пол+Табельный номер сотрудника.*

Поступления и переводы

Название поля	Тип поля	Длина	Длина дробной части
<i>Номер отдела</i>	Числовой	Целое	–

<i>Табельный номер сотрудника</i>	Числовой	Целое	–
<i>Дата поступления / перевода</i>	Дата / Время	Краткий фор- мат	–

Индексы: Номер отдела + Табельный номер сотрудника, Дата поступления/перевода.

Вариант 8. База данных «Учебное расписание»

Учебные дисциплины

Название поля	Тип поля	Длина	Длина дробной части
<i>Код дисциплины</i>	Числовой	Целое	–
<i>Название дисциплины</i>	Текстовый	30	–
<i>Плановое количество часов за семестр</i>	Числовой	Байт	–

Индекс: Код дисциплины.

Преподаватели

Название поля	Тип поля	Длина	Длина дробной части
<i>Табельный номер преподавателя</i>	Числовой	Целое	–
<i>Фамилия</i>	Текстовый	20	–
<i>Имя</i>	Текстовый	20	–
<i>Отчество</i>	Текстовый	20	–
<i>Телефон</i>	Числовой	Длинное целое	–
<i>Адрес</i>	Текстовый	40	–
<i>Название кафедры</i>	Текстовый	30	–

Индексы: Табельный номер преподавателя, Название кафедры + Табельный номер преподавателя.

Расписание занятий

Название поля	Тип поля	Длина	Длина дробной части
<i>День недели</i>	Числовой	Байт	–
<i>Лента</i>	Числовой	Байт	–
<i>Аудитория</i>	Текстовый	6	–
<i>Код дисциплины</i>	Числовой	Целое	–
<i>Табельный номер преподавателя</i>	Числовой	Целое	–

Индексы: *Код дисциплины* + *Табельный номер преподавателя*, *День недели* + *Лента*.

Вариант 9. База данных «Учет заказов покупателей»

Товары

Название поля	Тип поля	Длина	Длина дробной части
<i>Штрих-код</i>	Числовой	Целое	–
<i>Наименование товара</i>	Текстовый	30	–
<i>Продуктовый / Промышленный</i>	Логический	–	–
<i>Номер партии</i>	Числовой	Целое	–
<i>Срок годности</i>	Дата / Время	Краткий формат	–

Индекс: *Штрих-код*, *Продуктовый/Промышленный* + *Штрих-код*.

Покупатели

Название поля	Тип поля	Длина	Длина дробной части
<i>Регистрационный номер покупателя</i>	Числовой	Целое	–
<i>Название покупателя</i>	Текстовый	30	–
<i>Адрес</i>	Текстовый	40	–
<i>Оптовый / Розничный</i>	Логический		–

--	--	--	--

Индекс: *Регистрационный номер покупателя, Оптовый/Розничный+Регистрационный номер покупателя.*

Заказы

Название поля	Тип поля	Длина	Длина дробной части
<i>Номер заказа</i>	Числовой	Длинное целое	–
<i>Дата заказа</i>	Дата / Время	Краткий формат	–
<i>Штрих-код</i>	Числовой	Целое	–
<i>Количество</i>	Числовой	Одинарное с плавающей точкой	2
<i>Регистрационный номер покупателя</i>	Числовой	Целое	–

Индексы: *Номер заказа, Штрих-код, Регистрационный номер покупателя, Дата заказа + Номер заказа, Регистрационный номер покупателя + Номер заказа.*

Вариант 10. База данных «Биржа труда»

Безработные

Название поля	Тип поля	Длина	Длина дробной части
<i>Регистрационный номер безработного</i>	Числовой	Целое	–
<i>Фамилия</i>	Текстовый	20	–
<i>Имя</i>	Текстовый	20	–
<i>Отчество</i>	Текстовый	20	–
<i>Адрес</i>	Текстовый	40	–
<i>Телефон</i>	Числовой	Длинное целое	–
<i>Образование</i>	Текстовый	30	–
<i>Код профессии</i>	Числовой	Целое	–
<i>Стаж</i>	Числовой	Байт	–

--	--	--	--

Индексы: Регистрационный номер безработного, Код профессии, Код профессии + Регистрационный номер безработного.

Профессии

Название поля	Тип поля	Длина	Длина дробной части
Код профессии	Числовой	Целое	–
<i>Название профессии</i>	Текстовый	30	–

Индексы: Код профессии.

Вакансии

Название поля	Тип поля	Длина	Длина дробной части
<i>Наименование предприятия</i>	Текстовый	30	–
<i>Адрес</i>	Текстовый	40	–
<i>Сфера деятельности (отрасль)</i>	Текстовый	30	–
<i>Государственное / Частное</i>	Логическое	–	–
Код предприятия	Числовой	Целое	–
Код профессии	Числовой	Целое	–
<i>Оклад</i>	Денежный	Основной	2

Индекс: Код предприятия + Код профессии, Код профессии, Государственное / Частное + Код предприятия.

Контрольные вопросы.

1. Перечислите основные инструменты СУБД Access, предназначенные для создания таблиц. Охарактеризуйте достоинства и недостатки каждого из них.
2. Перечислите типы данных для полей таблиц, поддерживаемые Access. В каком режиме можно самостоятельно задавать или изменять типы данных для полей?
3. Опишите, как осуществляется выбор подтипа (размера) поля.
4. Какие режимы существуют для работы с таблицей? Для чего каждый из них предназначен?

5. Каким способом производится сортировка данных в таблице? Какие виды сортировки используются в Access?
6. Что представляют собой фильтры и для чего они предназначены? Как осуществить фильтрацию данных в режиме таблицы?
7. Приведите определение и назначение индексов. Как происходит индексирование данных в таблице?
8. Перечислите виды ключевых полей таблиц, существующих в СУБД Access. Опишите их свойства.
9. В каком режиме работы с таблицей можно создавать, удалять различные индексы или изменять их состав? Продемонстрируйте порядок создания в таблице простого и составного ключа.
10. Для чего предназначен первичный ключ таблицы? Какими способами в Access можно создать первичный ключ таблицы?
11. Укажите назначение вторичного ключа таблицы. Как связаны между собой понятия вторичного и внешнего ключа?
12. В чем сходство и различие первичного и альтернативного ключа?
13. Что такое отношение между таблицами? Перечислите виды отношений между таблицами. Приведите примеры для каждого вида.
14. Для чего предназначена схема данных? Опишите последовательность действий для работы в схеме.
15. Дайте определение родительской и дочерней таблицы. Как в Access создается связь (отношение) между ними?
16. Каким образом может производиться объединение записей связанных таблиц? Для чего оно используется? Перечислите типы объединения записей, используемые в Access.
17. Что представляют собой целостность данных и ссылочная целостность? Опишите механизм каскадных изменений и его назначение.

Лабораторная работа №4. Простые запросы.

Цель работы: изучение понятия запроса к БД, выборки данных, классификации запросов, назначения и синтаксиса языков запросов: языка структурированных запросов SQL и языка запросов по образцу QBE, а также практика создания запросов различной сложности в режимах мастера запросов или конструктора с использованием условий отбора записей, сортировки, группировки данных и вычисляемых полей. Задание рассчитано на 6 часов аудиторных занятий и 4 часа самостоятельной работы.

Порядок выполнения практической работы.

1. Произвести выборку данных из одной таблицы. Запрос создать с помощью конструктора запросов. Использовать условие отбора.
2. Построить сложный запрос, работающий с данными из нескольких таблиц. При выборке данных использовать вычисляемые поля. Для создания вычисляемых полей (или условий) использовать построитель выражений.

Лабораторная работа №5. Сложные запросы

Цель работы: изучение понятия запроса к БД, выборки данных, классификации запросов, назначения и синтаксиса языков запросов: языка структурированных запросов SQL и языка запросов по образцу QBE, а также практика создания запросов различной сложности в режимах мастера запросов или конструктора с использованием условий

отбора записей, сортировки, группировки данных и вычисляемых полей. Задание рассчитано на 6 часов аудиторных занятий и 4 часа самостоятельной работы.

Порядок выполнения практической работы.

1. Создать сложный запрос с вычисляемыми полями с помощью мастера. При необходимости использовать группировку данных.
2. Создать сложный запрос с вычисляемыми полями и группировкой данных с помощью конструктора запросов.

Лабоаторная работа №6. Запросы в режиме SQL.

Цель: изучение понятия запроса к БД, выборки данных, классификации запросов, назначения и синтаксиса языков запросов: языка структурированных запросов SQL и языка запросов по образцу QBE, а также практика создания запросов различной сложности в режимах мастера запросов или конструктора с использованием условий отбора записей, сортировки, группировки данных и вычисляемых полей. Задание рассчитано на 6 часов аудиторных занятий и 4 часа самостоятельной работы.

Порядок выполнения практической работы.

1. Просмотреть все запросы в режиме SQL.
2. Ответить на один из контрольных вопросов к заданию.

Вариант 1. Запросы к базе данных «Студенческая научная работа»

1. Из таблицы **Студенты** выбрать студентов группы М31.
2. Выдать список студентов, специализирующихся на кафедре прикладной математики.
3. Вычислить количество студентов каждого курса (номер курса – второй символ значения поля *Группа*).
4. В таблицу **Студенты** добавить новое поле *Стипендия* (тип – денежный, основной, длина дробной части равна 2). Заполнить значения этого поля для всех записей таблицы, полагая 0 при отсутствии у студента стипендии. Для каждой студенческой группы определить итоговую суммарную стипендию по всей группе в целом.

Вариант 2. Запросы к базе данных «Абоненты телефонной компании»

1. Выбрать из таблицы **Абоненты** всех лиц, проживающих на одной заданной улице (любой на выбор).
2. Вывести список абонентов, внесших оплату за телефон до 15 числа текущего месяца.
3. Вывести список всех улиц с указанием количества абонентов, проживающих на соответствующих улицах.
4. Посчитать и вывести объем недоплаты или переплаты в текущем месяце для каждого номера телефона. Значение тарифа взять из таблицы **Теле-фоны**, а сумму оплаты – из таблицы **Оплата**.

Вариант 3. Запросы к базе данных

«Статистика медицинского учреждения»

1. Выбрать из таблицы **Пациенты** медицинские карты детей.
2. Для пациента с заданным номером карты найти дату предполагаемой выписки. Использовать значения полей: *Дата поступления* в медицинское учреждение из таблицы **Пациенты**, *Нормативное количество дней для стационарного лечения* из таблицы **Диагнозы**.
3. Определить количество пациентов каждого отделения.
4. Вычислить оценку затрат на лечение больного в каждом отделении. Эта оценка равна среднему арифметическому числу дней, затраченному на лечение одного пациента в отделении.

Вариант 4. Запросы к базе данных «Заработная плата сотрудников»

1. Выбрать из таблицы **Персонал** всех сотрудников, которые работают отделе с номером 2.
2. Определить список сотрудников, получающих зарплату меньше 3000 рублей. Использовать таблицы **Персонал**, **Должности** и **Тарифная сетка**. Найти всех сотрудников, работающих на одной и той же должности (любой на выбор). Для каждого из них вывести значения полей: *Табельный номер сотрудника*, *Фамилия*, *Имя*, *Отчество*, *Номер отдела*.
3. Сформировать список должностей с указанием количества человек, работающих на каждой из них.

Вариант 5. Запросы к базе данных «Учет прихода товара»

1. Из таблицы **Товары** выбрать товары, цена которых превышает 50 рублей. Для каждого товара указать значения полей *Штрих-код*, *Наименование товара*, *Название группы*, *Цена товара за единицу измерения*.
2. Вывести список товарных групп со всеми входящими в них штрих-кодами товаров. Использовать таблицы **Товары**, **Товарные группы**.
3. Сформировать полную таблицу прихода, в которую включить следующие поля: *Дата прихода*, *Штрих-код*, *Наименование товара*, *Цена товара за единицу измерения*, *Количество*, *Сумма оплаты*. Значения поля *Сумма оплаты* вычислить по формуле:
$$\text{Сумма оплаты} = \text{Цена товара за единицу измерения} \cdot \text{Количество}.$$
4. Вычислить суммарную стоимость всех товаров, поступивших от заданного поставщика в текущем месяце.

Вариант 6. Запросы к базе данных «Расписание движения поездов»

1. Вывести из таблицы **Пункты назначения** коды и названия станций, время следования до которых превышает 14 часов.
2. Найти номера поездов, прибывших сегодня после 12.00 часов. Также вывести информацию: является поезд проходящим или местным. Использовать таблицы **Расписание** и **Поезда**.
3. В таблицу **Расписание** добавить поле *Наполняемость* (тип числовой, одинарное с плавающей точкой, длина дробной части равна 2). Это поле отражает реальный процент занятости мест в поезде. Заполнить значения этого поля для всех записей таблицы. Вывести номера убывших поездов со 100 % наполняемостью за текущую дату.

4. Вычислить общее число человек, воспользовавшихся услугами местных составов за текущий месяц.

Вариант 7. Запросы к базе данных «Кадры»

1. Из таблицы **Сотрудники** выбрать всех молодых специалистов пред-приятия, т.е. сотрудников моложе 30 лет.

2. Для заданного номера отдела сформировать список новых сотрудни-ков, т.е. сотрудников, поступивших или переведенных туда в текущем году. Для каждого из них указать значения полей: *Табельный номер сотрудника, Фамилия, Имя, Отчество, Дата поступления / перевода*. Использовать таблицы **Поступления и переводы** и **Сотрудники**.

3. Посчитать количество сотрудников в каждом отделе.

4. В таблицу **Поступления и переводы** добавить поле *Оклад* (тип денежный, основной, длина дробной части равна 2). Это поле определяет размер оклада сотрудника. Заполнить значения этого поля для всех записей таблицы **Поступления и переводы**. Посчитать сумму окладов всех сотруд-ников каждого отдела.

Вариант 8. Запросы к базе данных «Учебное расписание»

1. Из таблицы **Преподаватели** выбрать преподавателей кафедры прикладной математики.

2. На каждый день недели сформировать список дисциплин, по кото-рым занятия ведутся в этот день. Для каждой дисциплины указать значения полей: *Код дисциплины, Название дисциплины, Лента, Аудитория*.

3. Посчитать недельную рабочую нагрузку для каждого преподавателя, т.е. количество рабочих академических часов в неделю. Считать, что одна лента состоит из двух академических часов.

4. Для заданной кафедры найти список учебных дисциплин, занятия по которым ведут преподаватели данной кафедры.

Вариант 9. Запросы к базе данных «Учет заказов покупателей»

1. Вывести список всех промышленных товаров, у которых не истек срок годности. Указать значения всех полей таблицы **Товары**.

2. Вывести список всех оптовых покупателей, которые делали в текущем месяце крупные заказы – заказы, включающие товары в количестве больше 200 единиц измерения. Для каждого из них указать значения полей: *Регистрационный номер покупателя, Название покупателя, Адрес*.

3. В таблицу **Товары** добавить новое поле *Отпускная цена* (тип денежный, основной, длина дробной части равна 2). Заполнить значения этого поля для всех записей таблицы. Сформировать таблицу заказов, в которую включить: *Номер заказа, Дата заказа, Наименование товара, Стоимостьзаказа*. Значения поля *Стоимость заказа* вычислить по формуле:

$$\text{Стоимость заказа} = \text{Отпускная цена} \cdot \text{Количество}.$$

4. Посчитать и вывести торговый оборот на текущую дату, т.е. итоговую стоимость всех сделанных в этот день заказов.

Вариант 10. Запросы к базе данных «Биржа труда»

1. Выбрать из таблицы **Безработные** всех лиц с высшим образованием. Для каждого из них выдать следующие поля: *Регистрационный номер безработного, Фамилия, Имя, Отчество, Название профессии, Стаж.*

2. Вывести список безработных по заданному названию профессии, имеющих стаж работы более 5 лет. Указать значения всех полей таблицы **Безработные**.

3. Вывести все сведения по имеющимся вакансиям частных предприятий, подходящих под требования безработных.

4. В таблицу **Безработные** добавить новое поле *Желаемая оплата* (тип денежный, основной, длина дробной части равна 2). Это размер оклада, который человек желает получать в случае трудоустройства. Заполнить значения этого поля для всех записей таблицы. Определить среднее арифметическое величины желаемой оплаты для каждой профессии и найти отклонение этого среднего от оклада, предлагаемого предприятиями.

Контрольные вопросы

1. Дайте определения запроса и выборки. Как запрос и выборка связаны между собой?
2. Что представляет собой результатная таблица?
3. Перечислите все существующие типы запросов и опишите назначение каждого типа.
4. Для чего предназначены языки запросов SQL и QBE? Опишите представление запроса на обоих языках.
5. Что представляет собой бланк запросов по образцу? Существует ли в Access режим, реализующий работу с подобным бланком?
6. Перечислите способы создания запросов в Access. Охарактеризуйте достоинства и недостатки каждого из них.
7. Существует ли в СУБД Access режим, позволяющий создавать и выполнять запрос на языке SQL?
8. Какие запросы позволяет создавать мастер создания простых запросов?
9. Какими способами можно добавлять поля в список выводимых полей в режиме конструктора?
10. Приведите последовательность действий, необходимых для организации сортировки данных запроса в конструкторе.
11. Что представляет собой условие отбора записей в выборку данных? Из каких частей оно может состоять? Каким образом задается условие отбора записей?
12. Может ли проводиться отбор записей в выборку одновременно по нескольким полям? Что означает выражение с ключевым словом OR, находящееся в следующей строке за условием отбора?
13. Дайте определение вычисляемых полей. Для чего они предназначены? Приведите способы добавления вычисляемых полей в запрос.
14. Опишите назначение, свойства и структуру построителя выражений. Для чего используются встроенные функции Access? Перечислите ряд функций из числа встроенных.
15. Что такое группировка данных? Опишите область применения группирования и механизм группировки.
16. Для каких целей в запросах используются статистические (итоговые) функции? Приведите список статистических функций. Как осуществляется работа с подобными функциями в режиме конструктора?
17. Как в режиме конструктора осуществляется создание сложных запросов, основанных на данных из нескольких таблиц?

Лабоаторная работа №7. Формы запросов и отчетов.

1. Добавить к последней форме кнопку **Выполнить запрос** с подключением любого запроса и кнопку **Выдача отчета** с подключением любого отчета.
2. Ответить на один из контрольных вопросов к заданию.

Вариант 1. Формы к базе данных «Студенческая научная работа»

Выполнить построение формы **Кафедры** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Код кафедры*. Выполнить поиск записей по условию: поле *Название кафедры* имеет значение “прикладная математика”;

формы **Студенты** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Номер зачетной книжки*. Выполнить поиск записей по условию: студент родился в 1985 году (воспользоваться полем *Дата рождения*);

формы **Научные руководители** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Табельный номер научного руководителя*. Выполнить поиск записей по условию: преподаватель работает на кафедре прикладной математики (воспользоваться полем *Код кафедры*).

2. Создать форму **Специализация**, работающую с данными трех таблиц: **Студенты**, **Кафедры**, **Научные руководители**. На форме должны находиться следующие поля и соответствующие надписи к ним: *Номер зачетной книжки*, *Фамилия*, *Имя*, *Отчество*, *Группа студента*, *Фамилия*, *Имя*, *Отчество*, *Телефон*, *Домашний адрес* научного руководителя, *Название кафедры*. Форма должна содержать те же кнопки, что и первые три формы, кроме кнопки **Добавить новую запись**.
3. Проверить возможность подключения к созданной форме **Специализация** любого ранее созданного запроса или отчета через кнопки **Выполнить запрос**, **Выдача отчета** соответственно.

Вариант 2. Формы к базе данных «Абоненты телефонной компании»

1. Выполнить построение формы **Абоненты** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Регистрационный номер абонента*. Выполнить поиск записей по условию: абонент живет на проспекте Мира; формы **Телефоны** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Номер телефона*.

2. Выполнить поиск записей по условию: тариф оплаты за телефон составляет двести рублей; формы **Оплата** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полям *Номер телефона*, *Дата оплаты*. Выполнить поиск записей по условию: оплата произведена сегодня.

3. Создать форму **Текущая оплата**, работающую с данными трех таблиц: **Абоненты**, **Телефоны**, **Оплата**. На форме должны находиться следующие поля и соответствующие надписи к ним: *Номер телефона*, *Регистрационный номер абонента*, *Фамилия*, *Имя*, *Отчество*, *Категория абонента*, *Та-риф оплаты за телефон*, *Дата оплаты*, *Сумма оплаты*. Форма должна содержать те же кнопки, что первые три формы, кроме кнопки **Добавить новую запись**.

4. Проверить возможность подключения к созданной форме **Текущая оплата** любого ранее созданного запроса или отчета через кнопки **Выполнить запрос, Выдача отчета** соответственно.

Вариант 3. Формы к базе данных «Статистика медицинского учреждения»

1. Выполнить построение формы **Диагнозы** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Код диагноза*. Выполнить поиск записей по условию: количество дней для стационарного лечения равно 10; формы **Отделения** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Код отделения*. Выполнить поиск записей по условию: поле *Название отделения* имеет значение “терапия”; формы **Пациенты** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Номер карты*. Выполнить поиск записей по условию: пациент поступил в текущем месяце (воспользоваться полем *Дата поступления*).

2. Создать форму **Карты пациентов**, работающую с данными трех таблиц: **Диагнозы, Отделения, Пациенты**. На форме должны находиться следующие поля и соответствующие надписи к ним: *Номер карты, Фамилия, Имя, Отчество, Ребенок / Взрослый, Диагноз, Нормативное количество дней для стационарного лечения, Дата поступления, Название отделения*. Форма должна содержать те же кнопки, что первые три формы, кроме кнопки **Добавить новую запись**.

3. Проверить возможность подключения к созданной форме **Карты пациентов** любого ранее созданного запроса или отчета через кнопки **Выполнить запрос, Выдача отчета** соответственно.

Вариант 4. Формы к базе данных «Заработная плата сотрудников»

1. Выполнить построение формы **Персонал** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Табельный номер сотрудника*. Выполнить поиск записей по условию: сотрудник работает во втором отделе; формы **Должности** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Код должности*. Выполнить поиск записей по условию: поле *Должность* имеет значение “директор”; формы **Тарифная сетка** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Номер разряда*. Выполнить поиск записей по условию: размер оклада составляет 10000 рублей.

2. Создать форму **Личные счета сотрудников**, работающую с данными трех таблиц: **Персонал, Должности, Тарифная сетка**. На форме должны находиться следующие поля и соответствующие надписи к ним: *Табельный номер сотрудника, Фамилия, Имя, Отчество, Должность, Номер разряда, Оклад*. Форма должна содержать те же кнопки, что первые три формы, кроме кнопки **Добавить новую запись**.

3. Проверить возможность подключения к созданной форме **Личные счета сотрудников** любого ранее созданного запроса или отчета через кнопки **Выполнить запрос, Выдача отчета** соответственно.

Вариант 5. Формы к базе данных «Учет прихода товара»

1. Выполнить построение формы **Товары** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Штрих-код*. Выполнить поиск записей по условию: цена товара равна 500 рублей; формы **Товарные группы** и связать ее с одноименной

таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Номер товарной группы*. Выполнить поиск записей по условию: поле *Товарные группы* имеет значение “хлебобулочные изделия”; формы **Приход** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полям *Дата прихода*, *Штрих-код*. Выполнить поиск записей по условию: сегодняшняя дата прихода.

2. Создать форму **Учет прихода**, работающую с данными трех таблиц: **Товары**, **Товарные группы**, **Приход Студенты**. На форме должны находиться следующие поля и соответствующие надписи к ним: *Дата прихода*, *Штрих-код*, *Наименование товара*, *Название группы*, *Единица измерения*, *Цена товара за единицу измерения*, *Количество*, *Поставщик*. Форма должна содержать те же кнопки, что первые три формы, кроме кнопки **Добавить новую запись**.

3. Проверить возможность подключения к созданной форме **Учет прихода** любого ранее созданного запроса или отчета через кнопки **Выполнить запрос**, **Выдача отчета** соответственно.

Вариант 6. Формы к базе данных «Расписание движения поездов»

1. Выполнить построение формы **Пункты назначения** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Код пункта назначения*. Выполнить поиск записей по условию: время следования равно 24 часам; формы **Поезда** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Номер поезда*. Выполнить поиск записей по условию: поезд является местным; формы **Расписание** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полям *Номер поезда*. Выполнить поиск записей по условию: поезд прибывает или убывает сегодня.

2. Создать форму **Статистика**, работающую с данными трех таблиц: **Пункты назначения**, **Поезда**, **Расписание**. На форме должны находиться следующие поля и соответствующие надписи к ним: *Дата прибытия (убытия) + Время прибытия (убытия)*, *Номер поезда*, *Проходящий / Местный*, *Название пункта назначения*, *Нормативное количество посадочных мест*, *Время следования*, *Наполняемость*. Форма должна содержать те же кнопки, что первые три формы, кроме кнопки **Добавить новую запись**.

3. Проверить возможность подключения к созданной форме **Статистика** любого ранее созданного запроса или отчета через кнопки **Выполнить запрос**, **Выдача отчета** соответственно.

Вариант 7. Формы к базе данных «Кадры»

1. Выполнить построение формы **Отделы предприятия** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Номер отдела*. Выполнить поиск записей по условию: поле *Название отдела* имеет значение “бухгалтерия”; формы **Сотрудники** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Табельный номер сотрудника*. Выполнить поиск записей по условию: сотрудник родился в 1970 году (воспользоваться полем *Дата рождения*); формы **Поступления и переводы** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полям *Номер отдела*, *Табельный номер сотрудника*. Выполнить поиск записей по условию: сотрудник поступил на работу в текущем году.

2. Создать форму **Передвижения сотрудников**, работающую с данными трех таблиц: **Отделы предприятия, Сотрудники, Поступления и переводы**. На форме должны находиться следующие поля и соответствующие надписи к ним: *Табельный номер сотрудника, Фамилия, Имя, Отчество, На-звание отдела, Дата поступления/перевода*. Форма должна содержать те же кнопки, что первые три формы, кроме кнопки **Добавить новую запись**.

3. Проверить возможность подключения к созданной форме **Передвижения сотрудников** любого ранее созданного запроса или отчета через кнопки **Выполнить запрос, Выдача отчета** соответственно.

Вариант 8. Формы к базе данных «Учебное расписание»

1. Выполнить построение формы **Учебные дисциплины** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Код дисциплины*. Выполнить поиск записей по условию: количество часов за семестр составляет 36 часов; формы **Преподаватели** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Табельный номер преподавателя*. Выполнить поиск записей по условию: преподаватель работает на кафедре прикладной математики; формы **Расписание занятий** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полям *Код дисциплины, Табельный номер преподавателя*. Выполнить поиск записей по условию: занятие проводится в понедельник.

2. Создать форму **Аудиторная нагрузка**, работающую с данными трех таблиц: **Учебные дисциплины, Преподаватели, Расписание занятий**. На форме должны находиться следующие поля и соответствующие надписи к ним: *День недели, Лента, Аудитория, Название дисциплины, Категория дисциплины (обязательная, альтернативная, спецкурс), Фамилия, Имя, Отчество преподавателя, Название кафедры*. Форма должна содержать те же кнопки, что первые три формы, кроме кнопки **Добавить новую запись**.

3. Проверить возможность подключения к созданной форме **Аудиторная нагрузка** любого ранее созданного запроса или отчета через кнопки **Выполнить запрос, Выдача отчета** соответственно.

Вариант 9. Формы к базе данных «Учет заказов покупателей»

1. Выполнить построение формы **Товары** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Штрих-код*. Выполнить поиск записей по условию: срок годности товара истекает в текущем году; формы **Покупатели** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Регистрационный номер покупателя*. Выполнить поиск записей по условию: покупатель является оптовым; формы **Заказы** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полям *Номер заказа*. Выполнить поиск записей по условию: количество заказанных товаров равно 20.

2. Создать форму **Портфель заказов**, работающую с данными трех таблиц: **Товары, Покупатели, Заказы**. На форме должны находиться следующие поля и соответствующие надписи к ним: *Номер заказа, Дата заказа, Штрих-код, Наименование товара, Отпускная цена, Количество заказанного товара, Название, Адрес покупателя*.

Оптовый / Розничный. Форма должна содержать те же кнопки, что первые три формы, кроме кнопки **Добавить новую запись**.

3. Проверить возможность подключения к созданной форме **Портфель заказов** любого ранее созданного запроса или отчета через кнопки **Выполнить запрос**, **Выдача отчета** соответственно.

Вариант 10. Формы к базе данных «Биржа труда»

1. Выполнить построение формы **Профессии** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Код профессии*. Выполнить поиск записей по условию: поле *Название профессии* имеет значение “бухгалтер”; формы **Безработные** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полю *Регистрационный номер безработного*. Выполнить поиск записей по условию: безработный имеет высшее образование; формы **Вакансии** и связать ее с одноименной таблицей. Форма должна содержать все поля таблицы и соответствующие надписи к ним. Записи должны быть упорядочены по полям *Код предприятия*, *Код профессии*. Выполнить поиск записей по условию: вакансия предоставлена частным предприятием.

2. Создать форму **Список безработных**, работающую с данными трех таблиц: **Профессии**, **Безработные**. На форме должны находиться следующие поля и соответствующие надписи к ним: *Название профессии*, *Фамилия*, *Имя*, *Отчество*, *Адрес*, *Телефон*, *Образование*, *Стаж* безработного. Форма должна содержать те же кнопки, что первые три формы, кроме кнопки **Добавить новую запись**.

3. Проверить возможность подключения к созданной форме **Список безработных** любого ранее созданного запроса или отчета через кнопки **Выполнить запрос**, **Выдача отчета** соответственно.

Контрольные вопросы

1. Дайте понятие формы и опишите ее назначение. В чем формы схожи с отчетами, а в чем различаются?
2. Для кого предназначены формы? Укажите, в чем заключается основное удобство работы с формами.
3. Перечислите основные инструменты создания форм в СУБД Access. Охарактеризуйте особенности каждого из них.
4. Какие режимы существуют для работы с формами? Для чего каждый из них предназначен?
5. В каком режиме работы с формой можно редактировать данные, создавать, удалять записи? В чем заключается сходство и различие в отображении и работе с данными в таблицах и формах? Продемонстрируйте вышеуказанные действия на примере одной из созданных форм.
6. Опишите последовательность действий, необходимых для создания в мастере формы, отображающей данные из одной таблицы. Изменяются ли действия в случае создания формы, основанной на запросе? Чем отличаются эти формы?
7. Что представляют собой главная форма и подчиненная к ней? Как в мастере форм создается сложная форма, состоящая из главной и подчиненной формы?
8. Существуют ли какие-то отличия в отображении и работе с данными между простой и сложной формой?
9. Опишите назначение и свойства режима конструктора форм. Перечислите основные окна и инструменты данного режима.

10. Дайте определение раздела формы. Из каких разделов состоит форма? Что может содержать каждый из перечисленных разделов?
11. Что представляет собой элемент управления? Перечислите элементы управления, используемые только для форм, и опишите их назначение. Для элементов управления **Поле** и **Флажок** расшифруйте их свойства, отвечающие за различие в работе с отчетами и формами.
12. Опишите назначение окна **Свойства** и его вкладок. Перечислите основные свойства формы.
13. Для чего предназначена функция “Автоформат”? Опишите ее составные части. Продемонстрируйте применение данной функции к любой из созданных форм.
14. Что представляет собой источник данных для формы? Какие объекты БД могут выступать в этой роли? Укажите способ задания источника данных в режиме конструктора.
15. Опишите назначение и свойства списка и раскрывающегося списка. Приведите последовательность действий для создания списков с помощью специального мастера или самостоятельного создания. Продемонстрируйте эти действия на примере одной из форм. Чем различаются эти два вида списков?
16. Что в современных СУБД и языках программирования понимается под событием и обработчиком события? Каково назначение командных кнопок?
17. Перечислите категории командных кнопок и их состав. Продемонстрируйте добавление и работу с командной кнопкой на любой форме.

Лабоаторная работа №8. Приемы защиты баз данных.

1. Реализация трех следующих приемов защиты:
 - использование параметров запуска базы данных;
 - отображение и скрытие объектов в окне базы данных
 - применение пароля для открытия базы данных.
2. Ответить на один из контрольных вопросов к заданию.

Контрольные вопросы

1. Перечислите средства защиты данных, используемые в MSAccess. Ка-кие средства из перечисленных являются наиболее надежными?
2. Какие возможности для контроля целостности данных существуют в Access?
3. Для чего предназначено кодирование и декодирование данных?
4. Опишите способ защиты данных путем отображения и скрытия объектов в окне базы данных.
5. Какие настройки базы данных можно изменять с помощью параметров запуска?
6. Укажите последовательность действий, которые необходимо выполнить для задания пароля базы данных. Опишите достоинства и недостатки использования пароля при открытии базы данных.
7. Что представляет собой репликация данных? Какую цель преследует запрещение репликации данных?
8. В чем заключается способ защиты на уровне пользователя? Продемонстрируйте его реализацию для конкретной базы данных.
9. Опишите два способа защиты программы на языке VisualBasic. Для каких приложений они предназначены?
10. Приведите определение триггера. Опишите назначение триггеров и преимущества их использования.

Лабоаторная работа №9. Заполнение базы данных в среде MicrosoftSQLServer

Созданная база данных пуста, т. е. не содержит ни одной таблицы. Поэтому следующей задачей является создание таблиц, структура которых аналогична таблицам из базы данных Access. При создании таблиц необходимо обратить внимание на соотношения типов Access и SQL Server, представленные в таблице 1.

Таблица 1. Соответствие типов данных Microsoft Access и Microsoft SQL

№	Тип данных Microsoft Access	Тип данных Microsoft SQL	Описание типа данных Microsoft SQL
1	Текстовый	nvarchar	Тип данных для хранения текста до 4000 символов
2	Поле МЕМО	ntext	Тип данных для хранения символов в кодировке Unicode до 1 073 741 823 символов
3	Числовой	int	Численные значения (целые) в диапазоне от -2 147 483 648 до +2 147 483 647
4	Дата/время	smalldatetime	Дата и время от 1 января 1900 г. до 6 июня 2079 года с точностью до одной минуты
5	Денежный	money	Денежный тип данных, значения которого лежат в диапазоне от -922 337 203 685 477.5808 до +922 337 203

			685 477.5807, с точностью до одной десятичной
6	Счетчик	int	См. пункт 3
7	Логический	bit	Переменная, способная принимать только два значения - 0 или 1
8	Поле объекта OLE	image	Переменная для хранения массива байтов от 0 до 2 147 483 647 байт
9	Гиперссылка	ntext	См. пункт 2
10	Мастер подстановок	nvarchar	См. пункт 1

Для создания таблиц необходимо выбрать в контекстном меню ветки «Таблицы» пункт «Создать таблицу». Среда Management Studio принимает следующий вид (рис. 4).

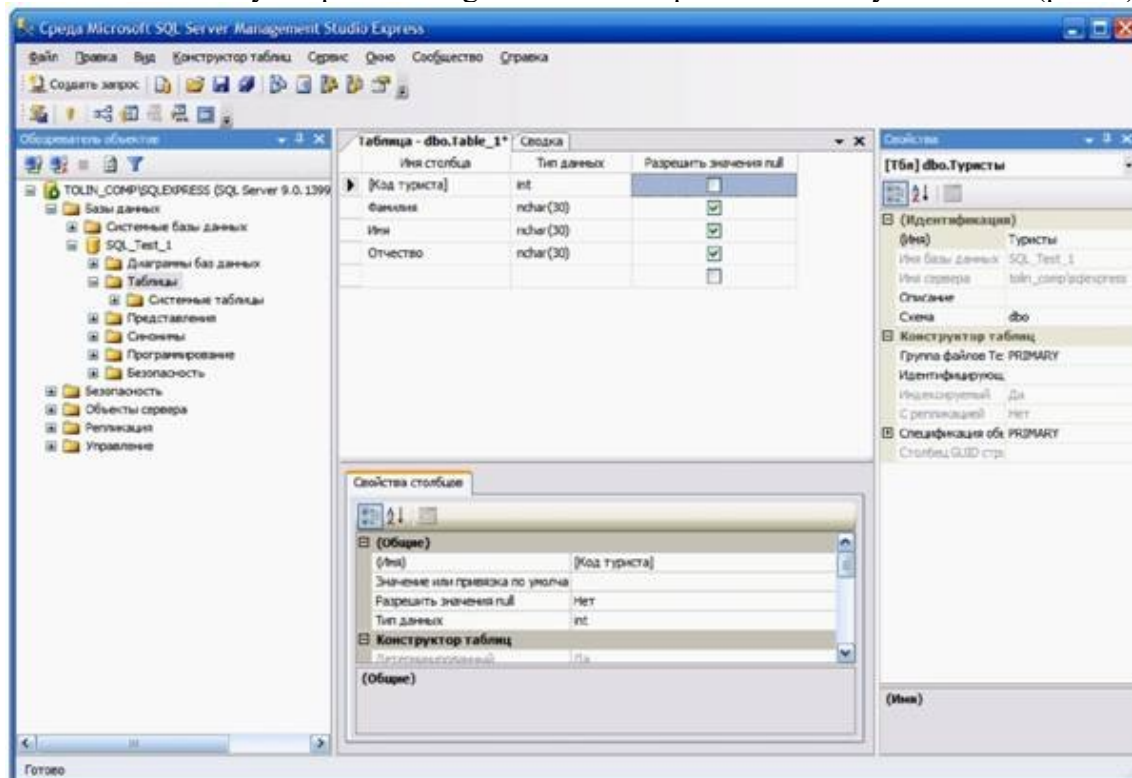


Рис. 4. Среда Management Studio в режиме создания таблицы

Для определения связей между таблицами необходимо задать первичные ключи таблиц. Для этого в контекстном меню соответствующего поля выбрать пункт «Задать первичный ключ» (рис. 5).

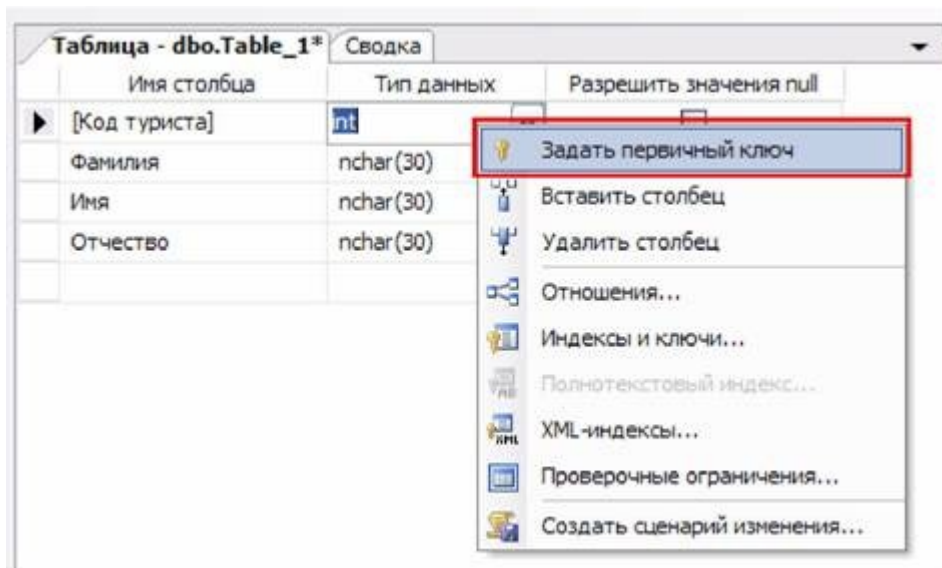


Рис. 5. Задание первичного ключа

Для создания связей между таблицами и схемы базы данных необходимо создать новую диаграмму базы данных, выбрав соответствующий пункт в контекстном меню ветви «Диаграммы баз данных». Добавив в появившемся окне необходимые таблицы в диаграмму, получаем следующий вид среды Management Studio (рис. 6).

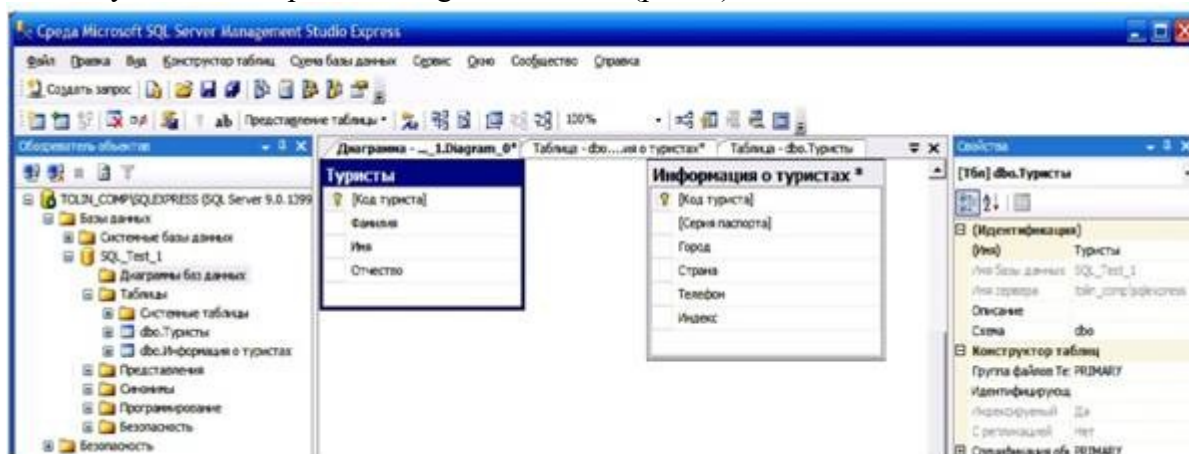


Рис. 6. Начало построения диаграммы БД

Лабораторная работа №10. Базовая конструкция SQL-запроса.

Основной операцией для описания запроса к БД в языке SQL является конструкция вида:

Select <список атрибутов> From <список отношений> Where <условие>

Эта операция представляет собой композицию реляционных операторов проекции, соединения и выбора. Проекция берется для указанного списка атрибутов, соединение выполняется для указанного списка отношений, выбор определяется условием отбора записей where.

В результате выполнения операции соединения данные из указанных в списке отношений представляются одной таблицей. В этой таблице из всех имеющихся столбцов исходных отношений списка отношений остаются только те столбцы, которые указаны в списке атрибутов, и только те строки, которые удовлетворяют условию where.

Итак, напомним первый запрос и нажмем клавишу F5 (пункт меню Запрос - Выполнить):

`select * from Туристы;`

В результате возвращаются все записи из таблицы «Туристы» базы данных BDTur_firmSQL.

Главное окно программы принимает вид (рис. 12).

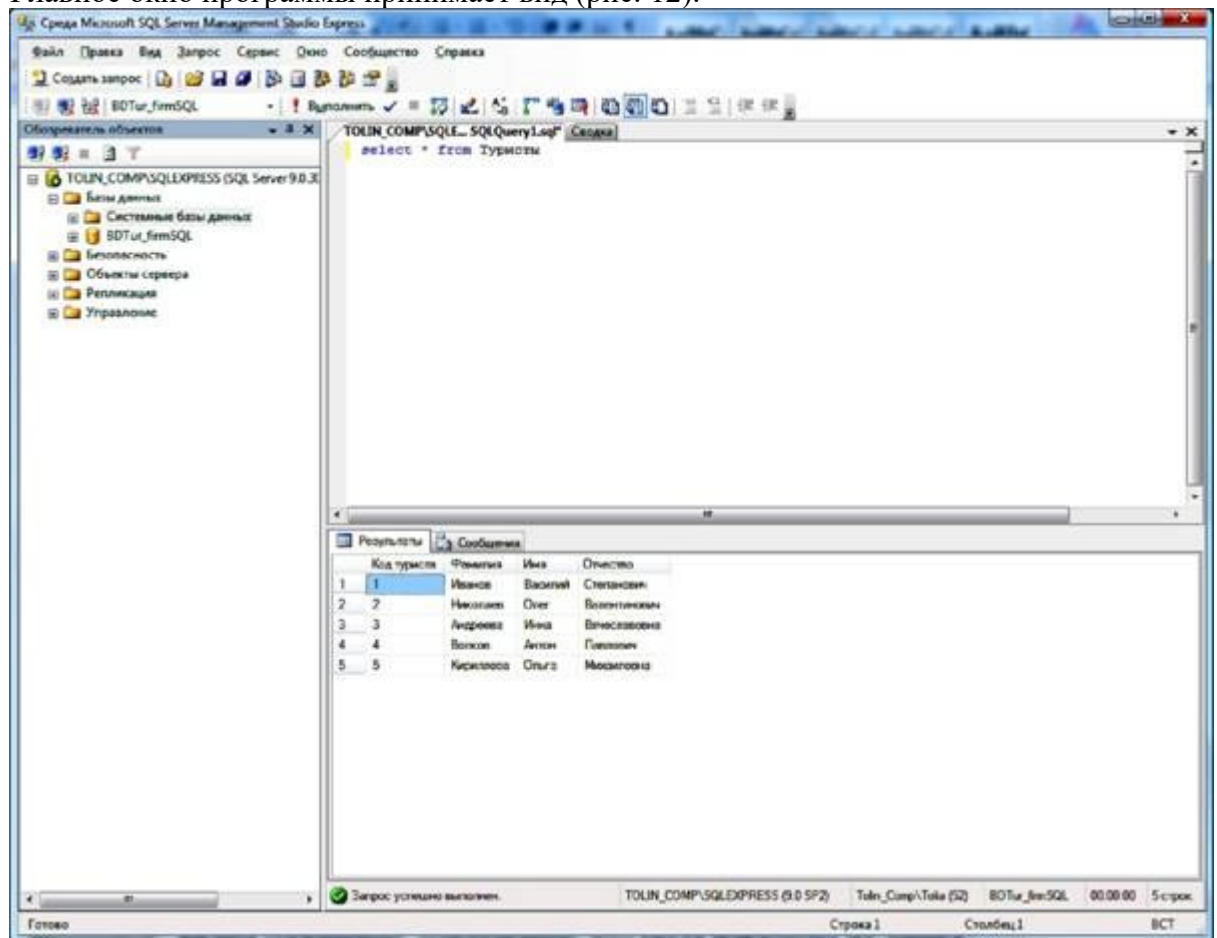


Рис. 12. Извлечение строк из таблицы «Туристы»

Данный запрос извлекал все столбцы таблицы. Если необходимо выбрать только столбец «Фамилия», запрос необходимо модифицировать следующим образом:

`select Фамилия from Туристы;`

Результат выполнения данного запроса представлен на рисунке 13.

Для вывода определенного количества записей используем следующий запрос (рис. 14):

`select top 3 Фамилия from Туристы;`

The screenshot shows the 'Results' tab of the SQL query window. It displays a table with 5 rows and 1 column: 'Фамилия'. The first three rows are highlighted, corresponding to the 'top 3' specified in the query.

	Фамилия
1	Иванов
2	Николаев
3	Андреева
4	Волков
5	Кириллова

Рис. 13. Извлечение столбца «Фамилия»

	Фамилия
1	Иванов
2	Николаев
3	Андреева

Рис. 14. Извлечение заданного количества записей

Извлекаются первые три записи поля «Фамилия», расположенные в самой таблице «Туристы». Обратим внимание на то, что фамилии расположены не в алфавитном порядке, а в порядке, в котором они были сохранены в базе данных.

Добиться алфавитного порядка можно с помощью предложения `order by`, содержащего список атрибутов, после каждого из которых стоит либо ключевое слово `asc` (сортировка по возрастанию), либо ключевое слово `desc` (сортировка по убыванию). Теперь предыдущий запрос может выглядеть так:

```
select top 3 Фамилия from Туристы order by Фамилия asc;
```

Вводя оператор `percent`, можем получить указанный [процент](#) записей от общего числа:

```
select top 25 percent Фамилия from Туристы;
```

Результат выполнения запроса представлен на рисунке 15.

	Фамилия
1	Иванов
2	Николаев

Рис. 15. Извлечение нескольких записей

Для отбора записей, отвечающих заданному условию, используем оператор `where`:

```
select * from Туры where Цена > 27000;
```

Этот запрос возвращает все записи из таблицы «Туры», в которых поле «Цена» имеет значение, большее 27000 (рис. 16).

	Код тура	Назван...	Цена	Информация
1	2	Греция	32000.00	В августе и сентябре действуют специальные скидки
2	3	Таиланд	30000.00	Не включая стоимость авиабилета

Рис. 16. Отбор записей со всеми полями по заданному значению

Оператор `where` поддерживает работу со знаками сравнения `<`, `>`, `>=`, `<=`. Точную выборку только из заданного множества значений осуществляет оператор `in`, в следующем примере извлекаются лишь те записи, в которых значение поля «Цена» в точности равно либо 10 000, либо 20 000, либо 30 000 (рис. 17):

```
select * from Туры where Цена in (10000, 20000, 30000);
```

	Код тура	Назван...	Цена	Информация
1	3	Таиланд	30000.00	Не включая стоимость авиабилета

Рис. 17. Отбор записей по точному совпадению значений поля Цена

Выборка значений, лежащих в указанном интервале, осуществляется оператором `between` `<первое_значение>` and `<второе_значение>` (рис. 18):

```
Select * from Туры where Цена between 10000 and 30000;
```

Результаты		Сообщения		
	Код тура	Назван...	Цена	Информация
1	1	Кипр	25000,00	В стоимость двух взрослых путевок входит цена одн...
2	3	Таиланд	30000,00	Не включая стоимость авиабилета
3	4	Италия	26000,00	Завтрак в отеле включен в стоимость путевки
4	5	Франция	27000,00	Дополнительные экскурсии не входят в стоимость п...

Рис. 18. Отбор записей по значениям в указанном интервале поля Цена

Лабоаторная работа №11. Простые запросы к БД.

Перейдем к созданию своих собственных процедур. Создадим новый бланк запросов и введем следующий запрос:

create procedure proc1 asselect [Код туриста], Фамилия, Имя, Отчество from Туристы

Здесь createprocedure - оператор, указывающий на создание хранимой процедуры, proc1 - ее название, далее после оператора as следует обычный SQL-запрос. Квадратные скобки необходимы для указания поля таблицы, в названии которого содержится пробел. После выполнения запроса появится сообщение:

Выполнение команд успешно завершено.

TheCOMMAND(s) completedsuccessfully.

Данное сообщение означает, что все сделано правильно и команда создала процедуру proc1. Убедиться в этом можно, развернув ветку «Программирование - Хранимые процедуры» в среде ManagementExpress (рис. 21).

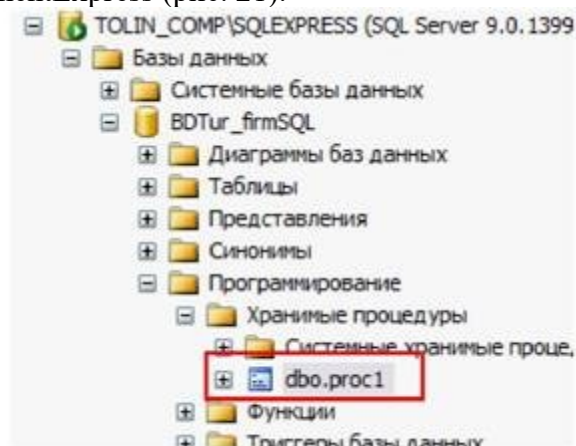


Рис. 21. Созданная хранимая процедура

Для просмотра результата вызываем ее:

execproc1

Появляется уже знакомое извлечение всех записей таблицы «Туристы» со всеми записями (рис. 22).

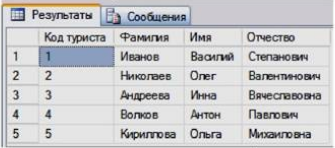
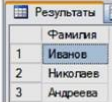
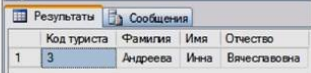
Из полученного результата видно, что создание содержимого хранимой процедуры не отличается ничем от создания обычного SQL-запроса.

В таблице 2 приведены примеры хранимых процедур.

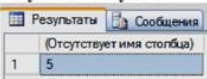
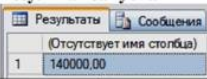
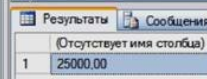
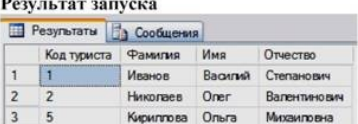
Результаты		Сообщения		
	Код туриста	Фамилия	Имя	Отчество
1	1	Иванов	Василий	Степанович
2	2	Николаев	Олег	Валентинович
3	3	Андреева	Инна	Вячеславовна
4	4	Волков	Антон	Павлович
5	5	Кириллова	Ольга	Михайловна

Рис. 22. Результат запуска процедуры proc1

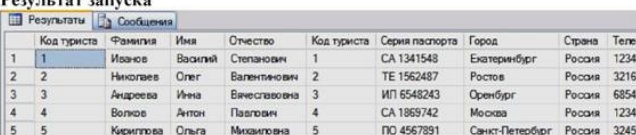
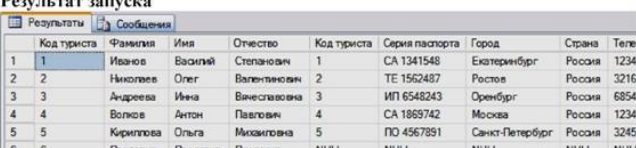
Таблица 2. Примеры хранимых процедур

№	SQL-конструкция для создания	Команда для извлечения	Описание
1	create procedure proc1 as select [Код туриста], Фамилия, Имя, Отчество from Туристы	exec proc1	Вывод всех записей таблицы «Туристы»
Результат запуска 			
2	create procedure proc2 as select top 3 Фамилия from туристы	exec proc2	Вывод первых трех значений поля «Фамилия» таблицы «Туристы»
Результат запуска 			
3	create procedure proc3 as select * from туристы where Фамилия = 'Андреева'	exec proc3	Вывод всех полей таблицы «Туристы», содержащих в поле «Фамилия» значение «Андреева»
Результат запуска 			

Продолжение таблицы 2

№	SQL-конструкция для создания	Команда для извлечения	Описание
4	create procedure proc4 as select count (*) from Туристы	exec proc4	Подсчет числа записей таблицы «Туристы»
Результат запуска 			
5	create procedure proc5 as select sum(Сумма) from Оплата	exec proc5	Подсчет значений поля Сумма таблицы Оплата
Результат запуска 			
6	create procedure proc6 as select max(Цена) from Туры	exec proc6	Вывод максимального значения поля Цена таблицы Туры
Результат запуска 			
7	create procedure proc7 as select min(Цена) from Туры	exec proc7	Вывод минимального значения поля Цена таблицы Туры
Результат запуска 			
8	create procedure proc8 as select * from Туристы where Фамилия like '%и%'	exec proc8	Вывод всех записей таблицы Туристы, содержащих в значении поля Фамилия букву «и» (в любой части слова)
Результат запуска 			

Окончание таблицы 2

№	SQL-конструкция для создания	Команда для извлечения	Описание
9	create procedure proc9 as select * from Туристы inner join [Информация о туристах] on Туристы.[Код Туриста]= [Информация о туристах].[Код Туриста]	exec proc9	Соединение двух таблиц «Туристы» и «Информация о туристах» по полю «Код туриста» и вывод полной связанной информации из обеих таблиц
Результат запуска 			
10	create procedure proc10 as select * from Туристы left join [Информация о туристах] on Туристы.[Код Туриста]= [Информация о туристах].[Код Туриста]	exec proc10	Прежде чем создать эту процедуру и затем ее выполнить, добавим в таблицу «Туристы» базы данных BDTur_fitm новую строку с произвольными значениями. В результате в таблице «Туристы» у нас получится 6 записей, а в связанной с ней таблице «Информация о туристах» – 5. Создаем хранимую процедуру и запускаем ее
Результат запуска 			

Операция innerjoin объединяет записи из двух таблиц, если поле (поля), по которому связаны эти таблицы, содержат одинаковые значения. Общий синтаксис выглядит следующим образом:

from таблица1**innerjoin** таблица2 **on** таблица1.поле1< оператор сравнения>
таблица2.поле2

Операция **leftjoin** используется для создания так называемого левого внешнего соединения. С помощью этой операции выбираются все записи первой (левой) таблицы, даже если они не соответствуют записям во второй (правой) таблице. Общий синтаксис имеет вид:

from таблица1**leftjoin** таблица2 **on** таблица1.поле1< оператор сравнения>
таблица2.поле2.

Основная литература:

1. Нестеров, С.А. Интеллектуальный анализ данных средствами MS SQLServer 2008 / С.А. Нестеров. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 338 с. : ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429083>
2. Сирант, О.В. Работа с базами данных / О.В. Сирант, Т.А. Коваленко. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 150 с. : схем., ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=428978>
3. Кумскова, И. А. Базы данных: учебник для СПО / И. А. Кумскова.- М.: КНОРУС, 2016.- 488 с.
4. Лазицкас, Е.А. Базы данных и системы управления базами данных : учебное пособие / Е.А. Лазицкас, И.Н. Загумённикова, П.Г. Гилевский. - Минск : РИПО, 2016. - 267 с. : ил. - Библиогр. в кн. - ISBN 978-985-503-558-0; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=463305>

Дополнительная литература:

1. Кузнецов, С. Введение в модель данных SQL : курс / С. Кузнецов. - 2-е изд., исправ. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 351 с. : ил. - (Основы информационных технологий). - Библиогр. в кн. - ISBN 5-9556-00028-0; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429087>
2. Кузнецов, С. Введение в реляционные базы данных / С. Кузнецов. - 2-е изд., исправ. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 248 с. : ил. - (Основы информационных технологий). - Библиогр. в кн. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429088>
3. Швецов В.И. Базы данных [Электронный ресурс]/ Швецов В.И.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 218 с.— Режим доступа: <http://www.iprbookshop.ru/52139.html>.— ЭБС «IPRbooks»