

Министерство образования и науки Российской Федерации

Государственное образовательное учреждение
высшего профессионального образования
Санкт - Петербургский государственный технологический институт
(технический университет)

Кафедра систем автоматизированного проектирования и управления

Иванов А.А., Авербух А.Б.

БАЗЫ ДАННЫХ. ЯЗЫКИ ЗАПРОСОВ.

Методические указания к выполнению контрольных работ
для студентов заочной формы обучения
направления подготовки «Информатика и вычислительная техника»

Санкт–Петербург
2011

УДК 681.3.066

Иванов, А.А., Базы данных. Язык запросов: методические указания к выполнению контрольных работ / А.А. Иванов, А.Б. Авербух – СПб.: СПбГТИ(ТУ), 2011. – 28 с.

В методические указания включены задания для выполнения контрольных работ, предназначенных для изучения языков запросов, используемых в большинстве современных систем управления базами данных.

Контрольные работы включают практические задания, при выполнении которых необходимо применить теоретические знания, полученные при изучении курса «Базы данных», в частности раздела «Языки запросов».

Методическое пособие предназначено для студентов 3 курса заочной формы обучения направления подготовки 230100 «Информатика и вычислительная техника» и соответствует рабочей программе дисциплины «Базы данных».

Ил. 1, табл. 3, библиогр. назв. 8

Рецензенты:

Русинов Л.А., зав. кафедрой автоматизации процессов химической промышленности Санкт-Петербургского государственного технического института (технического университета), д-р техн. наук, проф.

Утверждено на заседании учебно-методической комиссии факультета информатики и управления

Рекомендовано к изданию РИСо СПбГТИ(ТУ)

ОГЛАВЛЕНИЕ

Введение.....	4
Контрольная работа №1. Запросы на выборку данных.....	6
Контрольная работа №2. Запросы на модификацию данных.....	20
Приложение 1. Структура учебной базы данных «Борей».....	26
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ.....	27

ЗАОЧНОЕ ОТДЕЛЕНИЕ

Введение

Сегодня существует большое количество различных систем управления базами данных (СУБД). Это обуславливает значительные функциональные различия этих систем.

В данном учебном пособии представлены задания на выполнение контрольных работ по изучению структурированного языка запросов (SQL).

Дисциплина «Базы данных» принадлежит к числу общепрофессиональных (ОПД).

Цель дисциплины – дать представление об основных понятиях и терминах баз и банков данных, а также дать теоретические основы построения, организации и функционирования баз данных под управлением современных СУБД.

Дисциплина «Базы данных» для студентов заочного отделения читается на 3 курсе. Студенты решают две контрольные работы в рамках раздела «Язык запросов», сдают экзамен, выполняют курсовую работу.

В учебном пособии составлены две контрольные работы, включающие в себя содержание работы, варианты заданий.

Каждая контрольная работа содержит три части: теоретическую, анализ готового кода на SQL и разработку собственного текста запроса заданного функционала.

Тестовые задания должны показать качество проработки теоретических вопросов.

Студент самостоятельно выбирает задания контрольной работы. Номер варианта соответствует номеру первой буквы фамилии студента согласно таблице 1.

Таблица 1 - Распределение вариантов заданий

Первая буква фамилии студента	Номер варианта	Первая буква фамилии студента	Номер варианта
А, Б	1	Р, С	9
В, Г	2	Т, У	10
Д, Е	3	Ф, Х	11
Е, Ж	4	Ц, Ч	12
З, И	5	Ш, Щ	13
К, Л	6	Э, Ю	14
М, Н	7	Я	15
О, П	8	Образец	16

Студенту необходимо представить отчёт о выполненных контрольных работах в распечатанном виде и в электронном виде на любом носителе информации. Отчёт должен включать: титульный лист, условие задачи (либо поставленный вопрос) и предложенное решение поставленных задач (либо развернутый ответ на поставленный теоретический вопрос). На титульном листе отчёта о выполнении контрольных работ необходимо указать фамилию, имя и отчество студента, номер учебной группы, номер контрольной работы, номер варианта.

Приступая к выполнению контрольных работ, рекомендуется также ознакомиться с методическими материалами и литературой, предложенной в списке рекомендуемой литературы данного пособия, а также теоретического пособия по курсу "Базы данных".

Контрольная работа №1. Запросы на выборку данных.

Условия задач одинаковы для всех вариантов контрольной работы и приведены ниже. В качестве учебной базы данных выступает база «Борей» (см. прил.1).

Задача 1

Содержат ли тексты запросов ошибку? Если содержат, то в чем она заключается? Если запрос верный, что будет результатом выполнения запроса?

Задача 2

Приведите тексты SQL-запросов, выполняющих заданные действия.

Задача 3

Дайте теоретическую справку по приведенному ниже вопросу.

Варианты заданий контрольной работы №1 приведены в таблице 2.

Таблица 2. Варианты заданий контрольной работы №1.

№ Варианта	Задача 1	Задача 2	Задача 3
1.	<p>SELECT Клиенты.Название, Заказы.КодЗаказа, Доставка.Название, Заказы.ДатаРазмещения, Заказы.ДатаРазмещения FROM Клиенты LEFT JOIN (Доставка RIGHT JOIN Заказы ON Доставка.КодДоставки = Заказы.Доставка) ON Клиенты.КодКлиента = Заказы.КодКлиента ORDER BY Клиенты. Название, Заказы.КодЗаказа;</p> <p>SELECT Типы.Категория, Count(Товары.КодТовара) AS [Count-КодТовара], Max(Товары.Цена) AS [Max-Цена], Типы.Описание, Min(Товары.МинимальныйЗапас) AS [Min-МинимальныйЗапас] FROM Типы LEFT JOIN Товары ON Типы.КодТипа = Товары.КодТипа GROUP BY Типы.Категория, Типы.Описание;</p>	<p>Вывести список сотрудников вида Фамилия, Имя, Дата найма, количество заказов данного сотрудника. Список отсортировать по дате найма по количеству заказов сотрудника по убыванию и по дате найма по возрастанию.</p>	<p>Реализация языка запросов в СУБД, разработанных в России.</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
2.	<p>SELECT Товары.Марка, Типы.Категория, Поставщики.Название, Товары.НаСкладе, MAX (Товары.МинимальныйЗапас) FROM Поставщики RIGHT JOIN (Типы RIGHT JOIN Товары ON Типы.КодТипа = Товары.КодТипа) ON Поставщики.КодПоставщика = Товары.КодПоставщика WHERE (((Товары.Цена)<5000))</p> <p>SELECT Доставка.Название, Доставка.Телефон, Заказы.КодЗаказа, Заказано.Цена, Заказано.Скидка, [Заказано]![Цена]*(1-[Заказано]![Скидка]) AS Итоговая_цена FROM (Доставка INNER JOIN Заказы ON Доставка.КодДоставки = Заказы.Доставка) INNER JOIN Заказано;</p>	<p>Вывести список следующего вида Код Заказа, Категория товара, Марка товара, стоимость данного товара в заказе. В список включить товары, поставки которых не прекращены и цена складского запаса которых меньше 15000. Список отсортировать по категории товара по возрастанию и по марке товара по убыванию.</p>	<p>Общий формат оператора выборки в SQL.</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
3.	<p>SELECT Товары.Марка, Типы.Категория, Поставщики.Название, Товары.НаСкладе, Товары.МинимальныйЗапас FROM Поставщики RIGHT JOIN (Типы RIGHT JOIN Товары ON Типы.КодТипа = Товары.КодТипа) ON Поставщики.КодПоставщика = Товары.КодПоставщика WHERE (((Товары.Цена)<15000)) ORDER BY Товары.НаСкладе DESC , Товары.МинимальныйЗапас;</p> <p>SELECT Типы.Категория, Count(Товары.КодТовара) AS [Count-КодТовара], Max(Товары.Цена) AS [Max-Цена], Типы.Описание, Min(Товары.МинимальныйЗапас) AS [Min-МинимальныйЗапас] FROM Типы LEFT JOIN Товары ON Типы.КодТипа = Товары.КодТипа GROUP BY Типы.Категория, Типы.Описание;</p>	<p>Вывести список заказов с минимальной датой размещения. Список следующего вида Код заказа, Адрес получателя, Название службы доставки, Стоимость доставки, Стоимость доставки с НДС (18%). Для заказов у которых не определена служба доставки вывести в Название службы доставки null.</p>	<p>Скалярные функции СУБД ЛИНТЕР. Запросы без таблиц.</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
4.	<p>SELECT Товары.Марка, Типы.Категория, Поставщики.Название, Товары.НаСкладе, MAX (Товары.МинимальныйЗапас) FROM Поставщики RIGHT JOIN (Типы RIGHT JOIN Товары ON Типы.КодТипа = Товары.КодТипа) ON Поставщики.КодПоставщика = Товары.КодПоставщика GROUP BY Товары.Марка, Типы.Категория, Поставщики.Название</p> <p>SELECT Товары.Марка, Товары.Цена, Типы.Категория, Товары.ЕдиницаИзмерения, Товары.НаСкладе, Заказано.КодЗаказа, Заказано.Количество, Заказано.Цена FROM (Типы RIGHT JOIN Товары ON Типы.КодТипа = Товары.КодТипа) LEFT JOIN Заказано ON Товары.КодТовара = Заказано.КодТовара ORDER BY 2 DESC;</p>	<p>Сформировать список клиентов, имеющих не более двух заказов (включая клиентов не имеющих заказов), следующего вида Название, Город, Телефон, Число заказов клиента. Список отсортировать по названию клиентов по убыванию.</p>	<p>Запрос. Языки запросов. Характеристика языка QBE-</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
5.	<p>SELECT Доставка.КодДоставки, Доставка.Название, Sum(Заказы.КодКлиента) AS Sum-КодКлиента FROM (Доставка INNER JOIN Заказы ON Доставка.КодДоставки = Заказы.Доставка) INNER JOIN Заказано ON Заказы.КодЗаказа = Заказано.КодЗаказа GROUP BY Доставка.КодДоставки, Доставка.Название;</p> <p>SELECT Клиенты.КодКлиента, Клиенты.Название, Sum(Заказы.СтоимостьДоставки) AS Sum-СтоимостьДоставки, Min(Заказы.ДатаИсполнения) AS Min-ДатаИсполнения, Max(Заказано.Цена) AS Max-Цена FROM Клиенты LEFT JOIN (Заказы LEFT JOIN Заказано ON Заказы.КодЗаказа = Заказано.КодЗаказа) ON Клиенты.КодКлиента = Заказы.КодКлиента GROUP BY Клиенты.КодКлиента, Клиенты.Название DESC ORDER BY Клиенты.КодКлиента DESC , Sum(Заказы.СтоимостьДоставки);</p>	<p>Сформировать список заказов, в которых цена товара в заказе ниже цены, указанной в таблице «Товары». Список имеет следующий вид: Номер заказа, Наименование товара, Цена товара в заказе, Цена товара в таблице «Товары».</p>	<p>Коррелированные вложенные и иерархические подзапросы.</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
6.	<p>SELECT Доставка.КодДоставки, Доставка.Название, Заказы.КодКлиента, Заказано.Количество FROM Доставка, Заказано, Заказы;</p> <p>SELECT Доставка.КодДоставки, Доставка.Название, Sum(Заказы.КодКлиента) AS Sum-КодКлиента FROM (Доставка INNER JOIN Заказы ON Доставка.КодДоставки = Заказы.Доставка) INNER JOIN Заказано ON Заказы.КодЗаказа = Заказано.КодЗаказа GROUP BY Доставка.КодДоставки;</p>	<p>Сформировать список заказов, в котором присутствует товар с минимальной ценой следующего вида: Код заказа, Дата исполнения, Стоимость доставки. Список отсортировать по Стоимости доставки по возрастанию.</p>	<p>Использование агрегирующих функций без фразы GROUP BY.</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
7.	<p>SELECT COUNT(Доставка.КодДоставки), COUNT(Доставка.Название), MIN (Заказы.КодКлиента), Заказано.Количество FROM Доставка, Заказано, Заказы;</p> <p>SELECT Доставка.КодДоставки, Доставка.Название, Sum(Заказы.КодКлиента) AS Sum-КодКлиента FROM (Доставка INNER JOIN Заказы ON Доставка.КодДоставки = Заказы.Доставка) INNER JOIN Заказано ON Заказы.КодЗаказа = Заказано.КодЗаказа GROUP BY Доставка.КодДоставки, Доставка.Название;</p>	<p>Сформировать список всех городов, в которых расположены клиенты, исключив дубликаты. Итоговый список отсортировать в алфавитном порядке. Кроме того, посчитать и вывести количество различных городов (то есть число записей в списке).</p>	<p>Неопределенные значения. Вовлечение неопределенного значения в SQL-запросы</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
8.	<p>SELECT Клиенты.Название, Заказы.ДатаРазмещения, Заказы.ДатаНазначения, Поставщики.Название, Товары.Марка, Заказано.Количество FROM Поставщики LEFT JOIN (Товары INNER JOIN ((Клиенты RIGHT JOIN Заказы ON Клиенты.КодКлиента = Заказы.КодКлиента) INNER JOIN Заказано ON Заказы.КодЗаказа = Заказано.КодЗаказа) ON Товары.КодТовара = Заказано.КодТовара) ON Поставщики.КодПоставщика = Товары.КодПоставщика ORDER BY Клиенты.Название, Заказы.ДатаРазмещения DESC , Поставщики.Название DESC , Товары.Марка;</p> <p>SELECT Типы.Категория, Count(Товары.КодТовара) AS [Count-КодТовара], Max(Товары.Цена) AS [Max-Цена], Типы.Описание, Min(Товары.МинимальныйЗапас) AS [Min-МинимальныйЗапас] FROM Типы LEFT JOIN Товары ON Типы.КодТипа = Товары.КодТипа GROUP BY Типы.Категория, Типы.Описание; ORDER BY Типы.Категория DESC;</p>	<p>Сформировать список товаров (Марка, Категория, Минимальный Запас, стоимость остатков на складе), поставки которых прекращены. Список отсортировать по величине минимального запаса по возрастанию.</p>	<p>Предложение INSERT. Форматы. Вставка множества записей.</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
9.	<p>SELECT Клиенты.КодКлиента, Клиенты.Название, Sum(Заказы.СтоимостьДоставки) AS Sum-СтоимостьДоставки, Min(Заказы.ДатаИсполнения) AS Min-ДатаИсполнения, Max(Заказано.Цена) AS Max-Цена FROM Клиенты LEFT JOIN (Заказы LEFT JOIN Заказано ON Заказы.КодЗаказа = Заказано.КодЗаказа) ON Клиенты.КодКлиента = Заказы.КодКлиента GROUP BY Клиенты.Название ORDER BY Клиенты.КодКлиента DESC , Sum(Заказы.СтоимостьДоставки);</p> <p>SELECT Доставка.Название, Доставка.Телефон, Заказы.КодЗаказа, Заказано.Цена, Заказано.Скидка, Заказано.Цена*(1-Заказано.Скидка) AS Итоговая_цена FROM (Доставка INNER JOIN Заказы ON Доставка.КодДоставки = Заказы.Доставка) INNER JOIN Заказано ON Заказы.КодЗаказа = Заказано.КодЗаказа;</p>	<p>Сформировать список товаров (Марка, Цена, Ожидается, На складе), которые вошли в заказ с минимальным кодом. Список отсортировать по цене по убыванию</p>	<p>Реализация реляционных операторов с помощью средств языка SQL.</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
10.	<p>SELECT Доставка.КодДоставки, Доставка.Название, Sum(Заказы.КодКлиента) AS Sum-КодКлиента FROM (Доставка INNER JOIN Заказы ON Доставка.КодДоставки = Заказы.Доставка) INNER JOIN Заказано ON Заказы.КодЗаказа = Заказано.КодЗаказа</p> <p>SELECT Клиенты.КодКлиента, Клиенты.Название, Sum(Заказы.СтоимостьДоставки) AS Sum-СтоимостьДоставки, Min(Заказы.ДатаИсполнения) AS Min-ДатаИсполнения, Max(Заказано.Цена) AS Max-Цена FROM Клиенты LEFT JOIN (Заказы LEFT JOIN Заказано ON Заказы.КодЗаказа = Заказано.КодЗаказа) ON Клиенты.КодКлиента = Заказы.КодКлиента GROUP BY Клиенты.КодКлиента, Клиенты.Название;</p>	<p>Сформировать список поставщиков (Наименование, адрес, телефон), чьи товары поставлялись клиенту разметившему заказ с минимальным кодом.</p>	<p>Запрос. Языки запросов. Характеристика языка SQL</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
11.	<p>SELECT Клиенты.Название, Заказы.ДатаРазмещения, Заказы.ДатаНазначения, Поставщики.Название, Товары.Марка, Заказано.Количество FROM Поставщики INNER JOIN (Товары INNER JOIN ((Клиенты INNER JOIN Заказы ON Клиенты.КодКлиента = Заказы.КодКлиента) INNER JOIN Заказано ON Заказы.КодЗаказа = Заказано.КодЗаказа) ON Товары.КодТовара = Заказано.КодТовара) ON Поставщики.КодПоставщика = Товары.КодПоставщика ORDER BY Клиенты.Название, Заказы.ДатаРазмещения DESC , Поставщики.Название DESC , Товары.Марка;</p> <p>SELECT Товары.Марка, Товары.Цена, Типы.Категория, Товары.ЕдиницаИзмерения, Товары.НаСкладе, Заказано.КодЗаказа, Заказано.Количество, Заказано.Цена FROM (Типы RIGHT JOIN Товары ON Типы.КодТипа = Товары.КодТипа) LEFT JOIN Заказано ON Товары.КодТовара = Заказано.КодТовара ORDER BY Товары.Марка DESC , Товары.Цена, Типы.Категория;</p>	<p>Сформировать список следующего вида: Фамилия сотрудника, Общее количество обслуженных им заказов, Последняя дата исполненного заказа, Максимальная стоимость доставки среди всех заказов, обслуженных сотрудником. При формировании учесть сотрудников, не имеющих обслуженных заказов, а также однофамильцев.</p>	<p>Конструкция GROUP BY. Агрегирующие функции.</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
12.	<p>SELECT Товары.Марка, Товары.Цена, Типы.Категория, Товары.ЕдиницаИзмерения, Товары.НаСкладе, Заказано.КодЗаказа, Заказано.Количество, Заказано.Цена FROM (Типы RIGHT JOIN Товары) LEFT JOIN Заказано ON Товары.КодТовара = Заказано.КодТовара ORDER BY 2 DESC;</p> <p>SELECT 2, 3, 4 From Типы Order by 2,3;</p>	Сформировать список заказов (код заказа, код клиента, дата размещения заказа, суммарная стоимость заказа).	Коррелированные вложенные подзапросы.
13.	<p>SELECT Типы.Категория, Count(Товары.КодТовара) AS [Count-КодТовара], Max(Товары.Цена) AS [Max-Цена], Типы.Описание, Min(Товары.МинимальныйЗапас) AS [Min-МинимальныйЗапас] FROM Типы LEFT JOIN Товары ON Типы.КодТипа = Товары.КодТипа GROUP BY Типы.Категория, Типы.Описание; ORDER BY 3 DESC, 1;</p> <p>SELECT Типы.Категория, Товары.Марка, Товары.ЕдиницаИзмерения, Товары.НаСкладе, Товары.ПоставкиПрекращены FROM Типы INNER JOIN Товары ON Типы.КодТипа=Товары.КодТипа WHERE (((Товары.ПоставкиПрекращены)<>Yes)) ORDER BY Типы.Категория, Товары.Марка;</p>	Сформировать список поставщиков тех товаров, поставки которых прекращены. Список должен содержать все атрибуты поставщика, а также не содержать дубликатов. Список отсортировать по названию поставщика в порядке, обратном алфавитному.	Запросы, использующие EXISTS.

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
14.	<p>SELECT Заказано.КодЗаказа, Заказано.КодТовара, Товары.Марка, Заказано.Цена, Заказано.Количество, Заказано.Скидка, (Заказано.Цена*[Количество]*(1-[Скидка])/100)*100 AS ОтпускнаяЦена FROM Товары INNER JOIN Заказано ON Товары.КодТовара=Заказано.КодТовара ORDER BY Заказано.КодЗаказа;</p> <p>SELECT Типы.КодТипа, Типы.Категория, Товары.Марка, Sum([Сведения о заказах].ОтпускнаяЦена) AS ПродажиТовара FROM Типы INNER JOIN (Товары INNER JOIN (Заказы INNER JOIN [Сведения о заказах] ON Заказы.КодЗаказа=[Сведения о заказах].КодЗаказа) ON Товары.КодТовара=[Сведения о заказах].КодТовара) ON Типы.КодТипа=Товары.КодТипа WHERE (((Заказы.ДатаРазмещения) Between '1/1/1997' And '12/31/1997')) GROUP BY Типы.КодТипа, Типы.Категория, Товары.Марка ORDER BY Товары.Марка;</p>	<p>SELECT COUNT(Доставка.КодДоставки), COUNT(Доставка.Название), MIN (Заказы.КодКлиента), Заказано.Количество FROM Доставка, Заказано, Заказы WHERE Клиенты.КодКлиента < > 10000;</p>	<p>Соединение таблицы со своей копией.</p>

№ Варианта	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
15.	<p>SELECT Типы.КодТипа, Типы.Категория, Товары.Марка, Sum([Сведения о заказах].ОтпускнаяЦена) AS ПродажиТовара FROM Типы INNER JOIN (Товары INNER JOIN (Заказы INNER JOIN [Сведения о заказах] ON Заказы.КодЗаказа=[Сведения о заказах].КодЗаказа) ON Товары.КодТовара=[Сведения о заказах].КодТовара) ON Типы.КодТипа=Товары.КодТипа WHERE (((Заказы.ДатаРазмещения) Between '1/1/1997' And '12/31/1997')) GROUP BY Типы.КодТипа, Типы.Категория, Товары.Марка ORDER BY 3;</p>	<p>Сформировать список клиентов (Название, ОбращатьсяК, Должность, Адрес, Город, Количество сотрудников, обслуживавших заказы этого клиента). Список отсортировать по названию города в алфавитном порядке, и по названию клиента в обратном алфавитном порядке.</p>	<p>Вложенные подзапросы.</p>

Контрольная работа №2. Запросы на модификацию данных.

Условия задач одинаковы для всех вариантов контрольной работы и приведены ниже. В качестве учебной базы данных выступает база «Борей» (см. прил.1).

Задача 1

Содержит ли текст запроса ошибку? Если содержит, то в чем она заключается? Если запрос верный, что произойдет в результате выполнения запроса?

Задача 2

Приведите тексты SQL-запросов, выполняющих заданные действия.

Задача 3

Дайте теоретическую справку по приведенному ниже вопросу.

Варианты заданий контрольной работы №1 приведены в таблице 2.

Таблица 2. Варианты заданий контрольной работы №2.

№ Ва- ри- ан- та	Задача 1	Задача 2	Задача 3
1.	INSERT INTO Доставка (КодДоставки, Название, Телефон) SELECT Доставка.КодДоставки, Доставка.КодДоставки, Дос- тавка.Телефон FROM Доп_Доставка WHERE (((Достав- ка.КодДоставки)>10000) AND ((Доставка.КодДоставки) Is Not Null));	Увеличить на 50% цену всех товаров, постав- ляемых не российскими поставщиками, в случае если стоимость склад- ского остатка меньше 10000.	Фраза WHERE в SQL-запросах на модифика- цию данных. Использование фразы WHERE.
2.	DELETE FROM Заказано WHERE (((Заказано.Скидка)>10) AND ((Заказано.Количество)<1500)); GROUP BY Заказано.Скидка	Сформировать список (Название, Город, Ад- рес, Телефон) россий- ских поставщиков то- варов с наибольшим значением Минималь- ного запаса. Список от- сортировать по Назва- нию. Список записать в новую таблицу с име- нем «Т1»	Предложение INSERT. Фор- маты. Особен- ности примене- ния

№ Ва- ри- ан- та	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
3.	UPDATE Товары SET Товары.Цена = Цена*0,98 WHERE (((Товары.МинимальныйЗапас < 1500 Or Товары.НаСкладе = 1000) And Това- ры.ЕдиницаИзмерения<>'Л'));	Сформировать список заказов, доставкой ко- торых занята фирма «Фирма1». Список должен иметь вид: код заказа, дата размеще- ния, код сотрудника, код клиента. Список отсортировать по коду заказа по убыванию. Список записать в таб- лицу с именем «Т1», предполагая, что такая таблица существует	Типы команд языков запро- сов.
4.	INSERT INTO Доставка SELECT Доставка.КодДоставки, Доставка.КодДоставки, Доста- вка.Телефон FROM Доставка WHERE (((Достав- ка.КодДоставки)>10000) AND ((Доставка.КодДоставки) Is Not Null));	Увеличить вдвое цену всех товаров, у которых название категории на- чинается со слова «Винт», либо поставки которых прекращены.	Вложенные подзапросы. Виды. Особен- ности использо- вания вложен- ных подзапро- сов в предложе- ниях модифика- ции данных.
5.	UPDATE Сотрудники SET Сотруд- ники.Обращение = "Sr" WHERE (((Сотрудники.Должность) Like "Директор%") AND ((Сотруд- ники.Страна)="Испания")) OR (((Со- трудники.Страна)="Мексика")) GROUP BY Сотрудники.Должность;	Вывести всю информа- цию о заказах (то есть все атрибуты заказа), которые обслуживают- ся сотрудниками, ро- дившимися в период с 01/05/1972 по 31/12/1990	Предложение UPDATE. Фор- маты. Особен- ности примене- ния

№ Ва- ри- ан- та	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
6.	UPDATE Сотрудники SET Сотрудники.Обращение = "Sr" WHERE (((Сотрудники.Должность) Like "Директор%") AND ((Сотруд- ники.Страна)="Испания")) OR (((Со- трудники.Страна)="Мексика"));	Вывести всю информа- цию о товарах, с мак- симальной величиной ожидаемой поставки (независимо от единиц измерения), а также о товарах, относящихся к категориям «Продукты питания», «Бытовая химия» и «Автозапча- сти». Информацию за- писать в таблицу «Т1», предполагая, что такая таблица существует	Фраза HAVING. Особенности. Применение.
7.	DELETE * FROM Заказано WHERE (((Заказано.Скидка)>10) AND ((Заказано.Количество)<1500)); GROUP BY Заказано.Скидка	Сформировать список поставщиков товаров (все атрибуты постав- щиков), в описании ти- пов которых отсутству- ет изображение. В спи- ске исключить дубли- каты. Информацию за- писать в таблицу «Т1», предполагая, что такая таблица существует	Типы команд языков запро- сов.
8.	DELETE Заказано.Цена, Заказано.Скидка FROM Заказано WHERE (((Заказано.Цена)>100) AND ((Заказано.Скидка)<5));	Сформировать список заказов, которые об- служивали самые мо- лодые сотрудники (с учетом того, что могут быть сотрудники, ро- дившиеся в один день). Список записать в но- вую таблицу с именем «Т1»	Создание и уничтожение базовых таблиц.

№ Ва- ри- ан- та	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
9.	DELETE Заказано.Цена, Заказано.Скидка FROM Заказано WHERE (((Заказано.Цена)>100) AND ((Заказано.Скидка)<5)); ORDER BY Цена	Сформировать список всех товаров, в названии которых встречается слово «гайка». Список включает в себя Марку, Описание категории товара, На Складе, Минимальный Запас. Список отсортировать по цене по убыванию. Список записать в новую таблицу с именем «Т1»	Конструкции IN, BETWEEN, LIKE в запросах на модификацию данных.
10.	DELETE FROM Заказано WHERE (((Заказано.Скидка)>10) AND ((Заказано.Количество)<1500));	Исправить дату исполнения заказов с 05/05/2007 и 10/10/2007 на 01/02/2008.	Предложение INSERT. Форматы. Особенности применения
11.	UPDATE Товары SET Товары.Цена = Цена*0,98 WHERE (((Товары.МинимальныйЗапас < 1500 Or Товары.НаСкладе = 1000) And Товара.ЕдиницаИзмерения<>'Л')) GROUP BY Товара.МинимальныйЗапас;	Сформировать единый список сотрудников и клиентов следующего вида: Фамилия (или ОбращатьсяК для клиентов), страна, город, адрес. Список записать в новую таблицу с именем «Т2»	Предложение DELETE. Форматы. Особенности применения

№ Ва- ри- ан- та	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
12.	<p>SELECT Типы.КодТипа, Ти- пы.Категория, Товары.Марка, Sum([Сведения о заказах].ОтпускнаяЦена) AS ПродажиТовара INTO T21 FROM Типы INNER JOIN (Товары INNER JOIN (Заказы INNER JOIN [Сведения о заказах] ON Зака- зы.КодЗаказа=[Сведения о зака- зах].КодЗаказа) ON Това- ры.КодТовара=[Сведения о зака- зах].КодТовара) ON Ти- пы.КодТипа=Товары.КодТипа WHERE (((Заказы.ДатаРазмещения) Between '1/1/1997' And '12/31/1997')) GROUP BY Типы.КодТипа, Ти- пы.Категория, Товары.Марка ORDER BY 3;</p>	<p>Вывести в новую таб- лицу с именем «Т1» на- звания и телефоны фирм-доставщиков, доставлявших заказы клиентов из Москвы, Пскова и Смоленска. Список не должен со- держат дубликатов и должен быть отсорти- рован по названиям фирм в алфавитном по- рядке.</p>	<p>Предложение UPDATE. Фор- маты. Особен- ности примене- ния</p>
13.	<p>SELECT Заказы.КодКлиента, Зака- зы.КодСотрудника, Заказы.ДатаРазмещения, Заказано.КодТовара, Заказано.Цена, Заказано.Количество INTO T1 FROM Заказы INNER JOIN Заказано ON Заказы.КодЗаказа = Заказано.КодЗаказа;</p>	<p>Сформировать список сотрудников, не обслу- живавших ни один за- каз. Вид списка: Фами- лия, Должность, Стра- на, Город. Список от- сортировать по дате рождения по убыва- нию.</p>	<p>Вложенные подзапросы. Использование вложенных под- запросов в предложениях модификации данных</p>

№ Ва- ри- ан- та	<i>Задача 1</i>	<i>Задача 2</i>	<i>Задача 3</i>
14.	DELETE FROM Сотрудники WHERE (((Сотрудни- ки.КодСотрудника)>100000) AND ((Сотрудники.Город)="Москва") AND ((Сотрудники.Примечания) Is Not Null)) OR (((Сотрудни- ки.КодСотрудника)<150) AND ((Со- трудники.Город)="Псков"));	Установить величину скидки 32% на все то- вары, заказанные (по дате размещения) меж- ду 01/01/2003 и 05/07/2005	Модификация базовых таблиц с помощью язы- ков запросов.
15.	DELETE FROM Сотрудники WHERE (((Сотрудни- ки.КодСотрудника)>100000) AND ((Сотрудники.Город)="Москва") AND ((Сотрудники.Примечания) Is Not Null)) OR (((Сотрудни- ки.КодСотрудника)<150) AND ((Со- трудники.Город)="Псков")) ORDER BY 1, 3;	Составить SQL-запрос, увеличивающий на 50% цену всех товаров, по- ставляемых не россий- скими поставщиками, в случае если стоимость складского остатка меньше 10000	Создание и уничтожение базовых таблиц.

Приложение 1. Структура учебной базы данных «Борей».

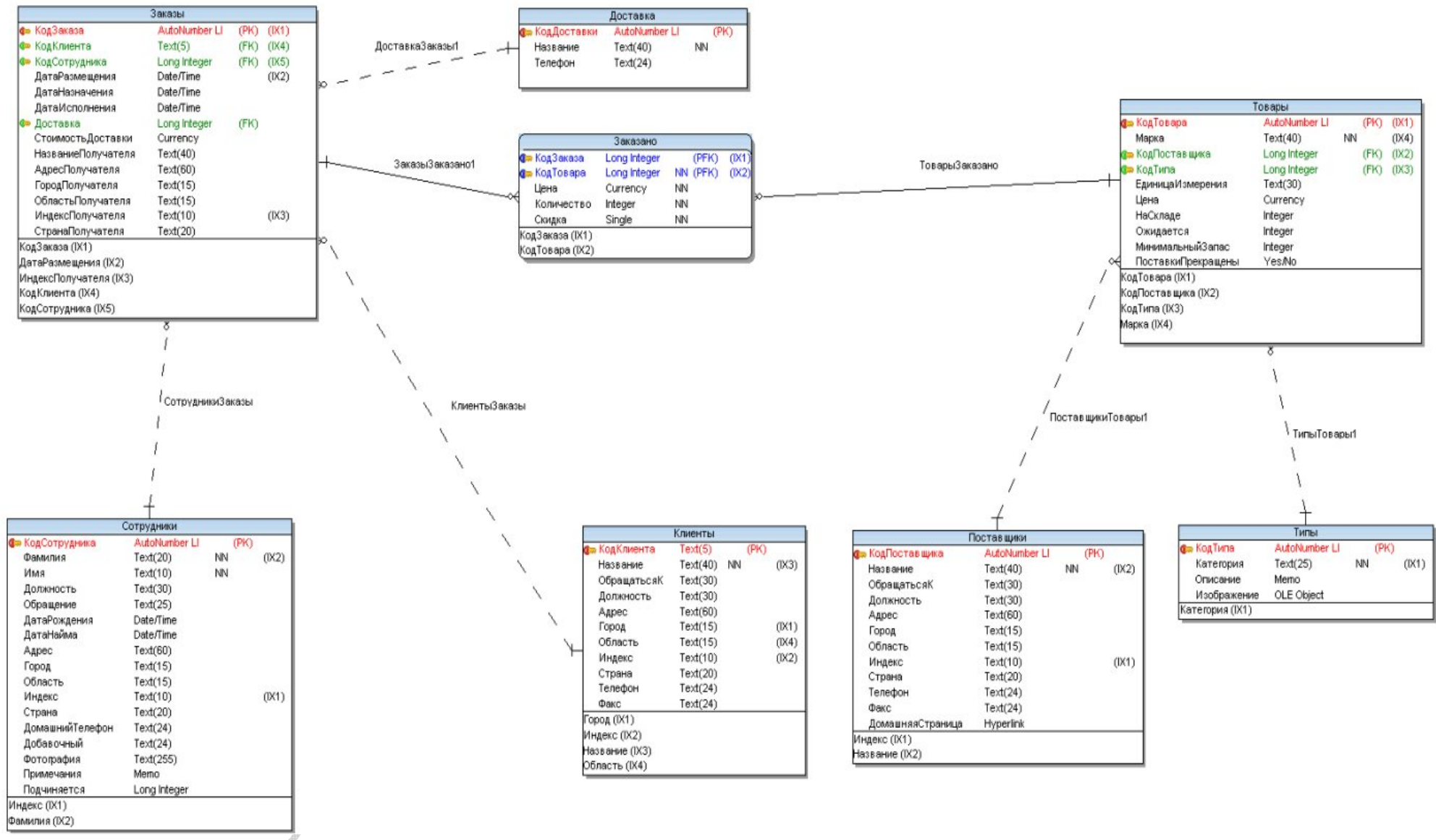


Рисунок 1 – Структура учебной базы данных «Борей».

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

- 1 Иванов А.А., Авербух А.Б. Базы данных. Языки запросов. Базовый курс [Текст]: учебное пособие для студентов заочной формы обучения. – СПб.: СПбГТИ(ТУ), 2011. –45 с.
- 2 Вейскас Дж. Эффективная работа: Microsoft Access 2003 – СПб.: Питер, 2003. – 1163 с.
- 3 Макдональд К. Oracle PL/SQL для профессионалов. Практические решения – М.: Apress, 2005 – 560с.
- 4 Молинаро Э. SQL Сборник рецептов. – М.: Apress, 2009 – 672с.
- 5 Грофф Дж., Вайнберг П SQL: полное руководство – К.: Издательская группа BHV, 2000. – 608 с
- 6 Грабер М SQL. Справочное руководство – М: Лори, 1997. – 291с
- 7 Мамаев Е Microsoft SQL Server 2000 в подлиннике – СПб.: Изд-во BHV, 2001 – 1280с.
- 8 Астахова И.Ф., Толстобров А.П., Мельников В.М. SQL в примерах и задачах. – М.: Новое знание, 2002 – 456 с.

Кафедра систем автоматизированного проектирования и управления

Методические указания к выполнению контрольных работ
для студентов заочной формы обучения
направления подготовки «Информатика и вычислительная техника»

Базы данных. Языки запросов.

Александр Александрович Иванов
Александр Борисович Авербух

Отпечатано с оригинал-макета. Формат 60 x 90 1/16
Печ. л. 1.75 Тираж 100 экз. Заказ №

Государственное образовательное учреждение
высшего профессионального образования
Санкт-Петербургский государственный технологический институт
(технический университет)

190013, Типография издательства СПбГТИ(ТУ), тел. 49-49-365
Санкт-Петербург, Московский пр., 26