МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Череповецкий государственный университет»

**Лабораторная работа № 2**

**«SQL. Язык манипулирования данными (DML)»**

**Выполнил:**

студент гр. 1ИВТпб-01-31оп

Климов А.Г.  
**Проверил:**  преподаватель

Селяничев О.Л.  
Отметка о зачете:

Череповец

2017 год

**Задания**

1. Создайте БД ПАНСИОН (см. электронный учебник Book1.sql) .
2. Создайте в ней таблицы БЛЮДА и ПОСТАВЩИКИ. Задание выполните средствами SQL.
3. Занесите в них записи. (Для ускорения ввода данных можно скопировать их непосредственно в Access’овскую таблицу через clipboard.)
4. Остальные таблицы создавайте по мере необходимости – тоже средствами SQL. И их наполняйте данными из файлов учебника.
5. Ознакомьтесь с теорией в главах 2 и 3 (список необходимых для изучения пунктов – ниже). Каждый встретившийся запрос проверьте практически -воспроизведите запрос в программе DbLoader. Изменяйте всякий запрос по-своему так, чтобы и он давал правильное решение (естественно, другое).
6. В отчет по ЛР должны попасть задания и ответы на них – из учебника и свое. В отчет занесите это в форме, например:

2.2.3. Выборка вычисляемых значений

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Получить значение калорийности всех продуктов (учесть, что при окислении 1 г углеводов или белков в организме освобождается в среднем 4.1 ккал, а при окислении 1 г жиров - 9.3 ккал). |  |
| Решение | SELECT Продукт, ((Белки+Углев)\*4.1+Жиры\*9.3) FROM Продукты; |  |

**Пункты учебника Book1.sql, которые должны быть рассмотрены.**

Глава 2. Запросы с использованием единственной таблицы

2.1. О предложении SELECT

2.2. Выборка без использования фразы WHERE

2.2.1. Простая выборка

2.2.2. Исключение дубликатов

2.2.3. Выборка вычисляемых значений

2.3. Выборка c использованием фразы WHERE

2.3.1. Использование операторов сравнения

2.3.2. Использование BETWEEN

2.3.3. Использование IN

2.3.4. Использование LIKE

2.3.5. Вовлечение неопределенного значения

2.4. Выборка с упорядочением

2.5. Агрегирование данных

2.5.1. SQL-функции

2.5.2. Функции без использования фразы GROUP BY

2.5.3. Фраза GROUP BY

2.5.4. Использование фразы HAVING

Глава 3. Запросы с использованием нескольких таблиц

3.1. О средствах одновременной работы с множеством таблиц

3.2. Запросы, использующие соединения

3.2.1. Декартово произведение таблиц

3.2.2. Эквисоединение таблиц

3.2.3. Естественное соединение таблиц

3.2.4. Композиция таблиц

3.2.5. Тета-соединение таблиц

3.2.6. Соединение таблиц с дополнительным условием

3.2.7. Соединение таблицы со своей копией

**Выполнение заданий**

1. Создаём БД ПАНСИОН.

a) Создаём таблицу “Блюда”:

**CREATE TABLE Блюда(БЛ int, Блюдо varchar(20), В char(1), Основа varchar(8), Выход decimal(5,2), Труд int);**

Создаём таблицу “Поставщики”:

**CREATE TABLE Поставщики(ПС int, Название varchar(16), Статус varchar(16), Город varchar(16), Адрес varchar(16), Телефон int);**

b) Заносим записи в созданные таблицы рис. 1 и рис. 2:

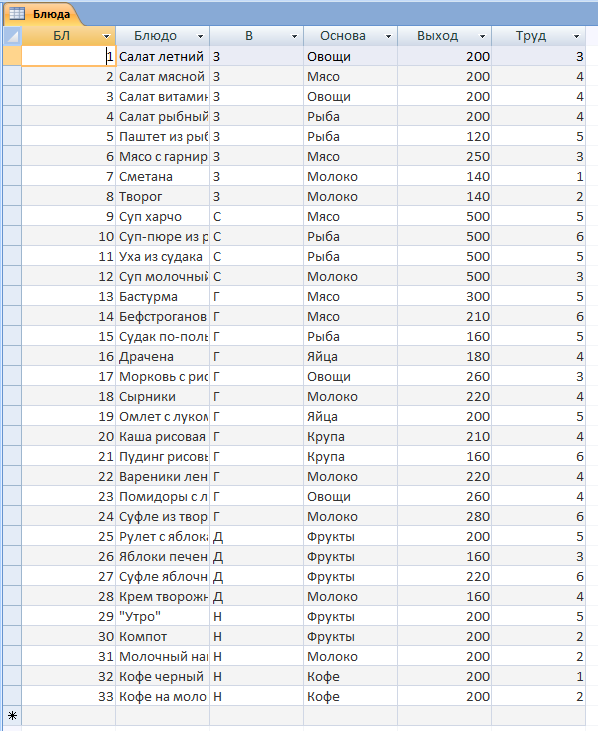


Рис. 1. Записи таблицы “Блюда”

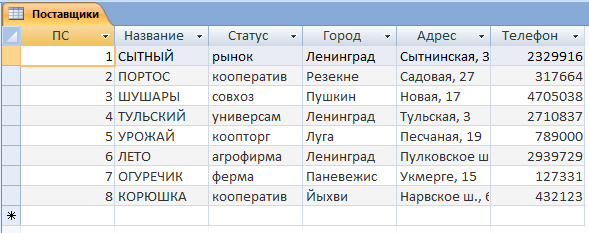


Рис. 2. Записи таблицы “Поставщики”

**2.2. Выборка без использования фразы WHERE**

**2.2.1. Простая выборка**

Таблица 1

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Запрос выдать название, статус и адрес поставщиков | Запрос выдать название, статус, адрес и телефон поставщиков |
| Решение | SELECT Название, Статус, Адрес  FROM Поставщики; | SELECT Название, Статус, Адрес, Телефон FROM Поставщики; |

Результат простой выборки из табл. 1 изображен на Рис. 3.

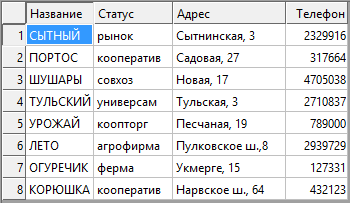


Рис. 3. Выборка для Табл. 1

Таблица 2

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Запрос получить всю информации о поставщиках | Запрос получить всю информации о блюдах |
| Решение 1 | SELECT ПС, Название, Статус, Город, Адрес, Телефон  FROM Поставщики; | SELECT БЛ, Блюдо, В, Основа, Выход, Труд FROM Блюда; |
| Решение 2 | SELECT \*  FROM Поставщики; | SELECT \* FROM Блюда;  **Используется по умолчанию, если вы не указываете ни один из предикатов. Ядро СУБД Microsoft Access выбирает все записи, которые удовлетворяют условиям в инструкции SQL.**  **Синтаксис:** SELECT [ALL | DISTINCT] FROM таблица  **Эти предикаты задают записи, выбираемые с помощью запросов SQL.** |
| Решение 3 |  | **SELECT ALL \* FROM Блюда;**  **Предикаты используются для ограничения числа возвращаемых записей. Если предикат не задан, по умолчанию используется ALL.** |

Результат простой выборки из Табл. 2 изображен на Рис. 3.

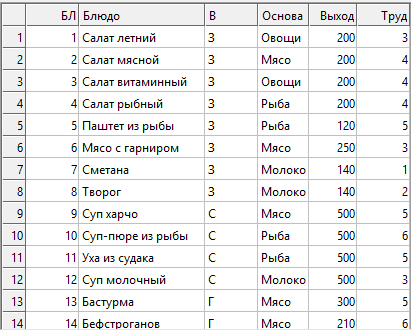


Рис. 3. Выборка для Табл. 2

Таблица 3

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Выдать основу всех блюд | Выдать статус поставщиков |
| Решение | SELECT Основа  FROM Блюда; | SELECT Статус FROM Поставщики; |

Результат простой выборки из Табл. 3 изображен на Рис. 4.

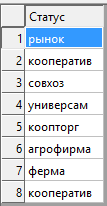


Рис. 4. Выборка для Табл. 3

**2.2.2. Исключение дубликатов**

Таблица 4

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Основные продукты без дубликатов | Статус поставщиков без дубликатов |
| Решение | SELECT DISTINCT Основа  FROM Блюда; | SELECT DISTINCT Статус FROM Поставщики; |

Результат простой выборки из Табл. 4 изображен на Рис. 5.

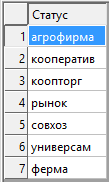


Рис. 5. Выборка для Табл. 4

**2.2.3. Выборка вычисляемых значений**

Создаём таблицу “Продукты”:

**CREATE TABLE Продукты(ПР int, Продукт varchar(16), Белки decimal(5,2), Жиры decimal(5,2), Углев decimal(5,2), K int, Ca int, Na int, B2 decimal(2,1), PP decimal(3,1), C int);**

Заносим записи в созданную таблицу Рис. 6:

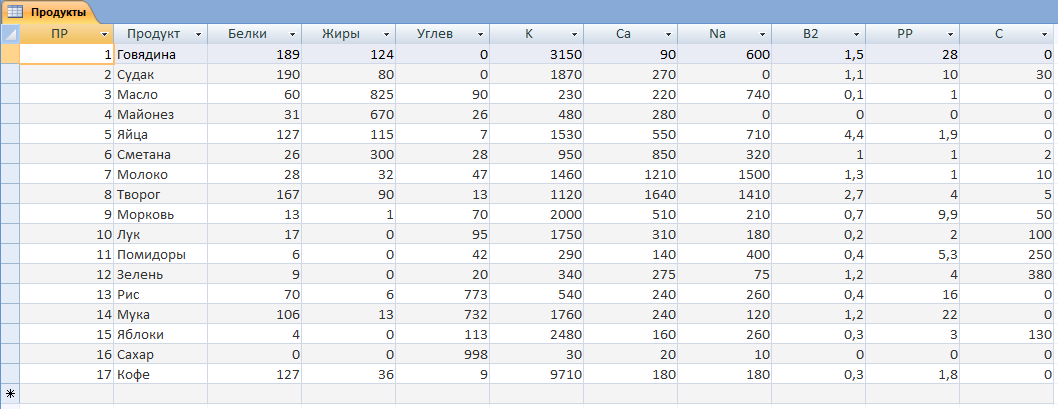


Рис. 6. Записи таблицы “Продукты”

Таблица 5

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Получить значение калорийности всех продуктов | Получить значение жиров и прибавить 50 |
| Решение 1 | SELECT Продукт, ((Белки+Углев)\*4.1+Жиры\*9.3)  FROM Продукты; | SELECT Продукт, (Жиры+50) FROM Продукты; |
| Решение 2 | SELECT Продукт, 'Калорий =',  ((Белки+Углев)\*4.1+Жиры \*9.3)  FROM Продукты; | SELECT Продукт, ‘Жиры+50=’, (Жиры+50) FROM Продукты; |

Результат выборки вычисляемых выражений из Табл. 5 изображен на Рис. 7 и Рис. 8.



Рис. 7. Выборка для Табл. 5



Рис. 8. Выборка для Табл. 5

Создаём таблицу “Поставки”:

**CREATE TABLE Поставки(ПС int, ПР int, Цена decimal(3,2), К\_во int);**

**Заносим записи в созданную таблицу Рис. 9:**

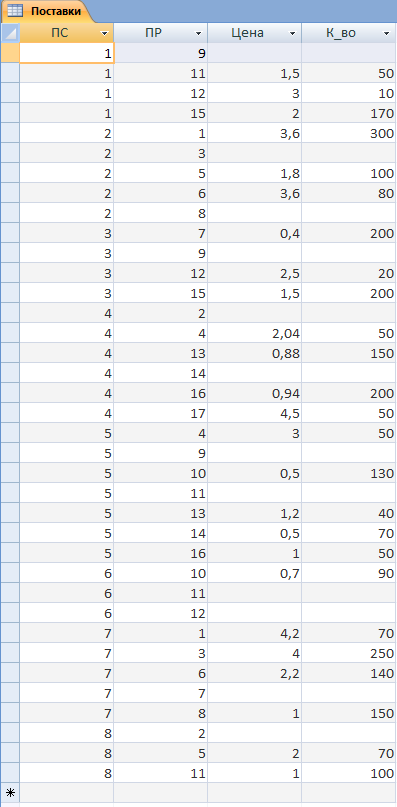


Рис. 9. Записи таблицы “Поставки”

Таблица 6

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Получить значения ПР, Цены, Количества и Цены\*Количество | Получить значения Цены, Количества и Цены\*Количество |
| Решение | SELECT ПР, Цена, К\_во, (Цена \* К\_во)  FROM Поставки; | SELECT Цена, К\_во, (Цена \* К\_во)  FROM Поставки; |

Результат выборки вычисляемых выражений из Табл. 6 изображен на Рис. 10.

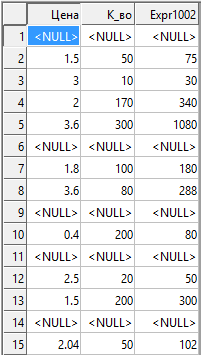


Рис. 10. Выборка для Табл. 6

**2.3. Выборка c использованием фразы WHERE**

**2.3.1. Использование операторов сравнения**

Таблица 7

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Получить перечень продуктов, практически не содержащих углеводов | Получить перечень продуктов, практически не содержащих жиров |
| Решение | SELECT Продукт, Белки, Жиры, Углев, K, Ca, Na, B2, PP, C FROM Продукты WHERE Углев = 0; | SELECT Продукт, Белки, Жиры, Углев, K, Ca, Na, B2, PP, C FROM Продукты WHERE Жиры = 0; |

Результат выборки с использованием операторов сравнения из Табл. 7 изображен на Рис. 11.

****

Рис. 11. Выборка для Табл. 7

Таблица 8

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Получить перечень продуктов, практически не содержащих углеводов и натрия | Получить перечень продуктов, практически не содержащих жиров или белков |
| Решение | SELECT Продукт, Белки, Жиры, Углев, K, Ca, Na, B2, PP, C FROM Продукты WHERE Углев = 0 AND Na = 0; | SELECT Продукт, Белки, Жиры, Углев, K, Ca, Na, B2, PP, C FROM Продукты WHERE Жиры = 0 OR Белки = 0; |

Результат выборки с использованием операторов сравнения из Табл. 8 изображен на Рис. 12.

****

Рис. 12. Выборка для Табл. 8

**2.3.2. Использование BETWEEN**

Таблица 9

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Получить перечень продуктов, в которых значение содержания белка находится в диапазоне от 10 до 50 | Получить перечень продуктов, в которых значение содержания жиров находится в диапазоне от 20 до 40 |
| Решение | SELECT Продукт, Белки FROM Продукты WHERE Белки BETWEEN 10 AND 50; | SELECT Продукт, Жиры FROM Продукты WHERE Жиры BETWEEN 20 AND 40; **>= AND <=.** |

Результат выборки в заданном диапазоне из Табл. 9 изображен на Рис. 13.

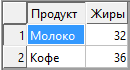


Рис. 13. Выборка для Табл. 9

Таблица 10

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Получить перечень продуктов, в которых значение содержания белка находится не в диапазоне от 10 до 50 и жиры > 100 | Получить перечень продуктов, в которых значение содержания жиров находится не в диапазоне от 20 до 40 а также белки > 90 |
| Решение | SELECT Продукт, Белки, Жиры FROM Продукты WHERE Белки NOT BETWEEN 10 AND 50 AND Жиры > 100; | SELECT Продукт, Жиры FROM Продукты WHERE Белки > 90 AND Жиры NOT BETWEEN 20 AND 40; |

Результат выборки в заданном диапазоне из Табл. 9 изображен на Рис. 14.

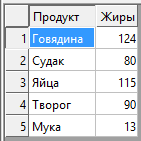


Рис. 14. Выборка для Табл. 10

Создаём таблицу “Минимальные оклады”:

**CREATE TABLE Миноклады(Миноклад int, Начало date, Конец date);**

Заносим записи в созданную таблицу Рис. 9:

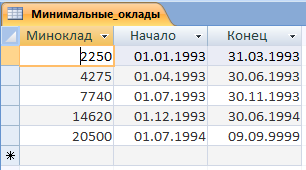


Рис. 15. Записи таблицы “Минимальные\_оклады”

Таблица 11

|  |  |  |
| --- | --- | --- |
|  | Из учебника | Мое |
| Задание | Узнать, какие изменения минимальных окладов производились в 1993/94 учебном году | Узнать, какие изменения минимальных окладов производились в 1994/99 учебном году |
| Решение | SELECT Начало, Миноклад FROM Миноклады WHERE Начало BETWEEN '1-9-1993' AND '31-8-1994' | SELECT Конец, Миноклад FROM Миноклады WHERE Конец BETWEEN #1-9-1994# AND #31-8-1999# |

Таблица 12

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выявить значения минимальных окладов, которые существовали в 1993/94 учебном году | Выявить значения минимальных окладов, которые существовали в 1993/94 учебном году |
| SELECT \* FROM Миноклады WHERE Начало BETWEEN '1-9-1993' AND '31-8-1994' OR Конец BETWEEN '1-9-1993' AND '31-8-1994' | SELECT \* FROM Миноклады WHERE Начало  BETWEEN #1-9-1993# AND #31-8-1994#  OR Конец  BETWEEN #1-9-1993# AND #31-8-1994# |

Таблица 13

|  |  |
| --- | --- |
| Из учебника | Мое |
| Получить минимальный оклад на 15-5-1994 | Получить минимальный оклад на 6-4-1993 |
| SELECT Миноклад FROM Миноклады WHERE '15-05-1994' BETWEEN Начало AND Конец | SELECT Миноклад FROM Миноклады WHERE #06-04-1993# BETWEEN Начало AND Конец |

**2.3.3. Использование IN**

Таблица 14

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выдать сведения о блюдах на основе яиц, крупы и овощей | Выдать сведения о блюдах на основе овощей, мяса и яиц |
| SELECT \* FROM Блюда WHERE Основа IN (‘Яйца’, ‘Крупа’, ‘Овощи’); | SELECT \* FROM Блюда WHERE Основа IN (‘Овощи’, ‘Мясо’, ‘Яйца’); |

Таблица 15

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выдать сведения о блюдах на основе яиц, крупы и овощей | Выдать сведения о блюдах на основе овощей, мяса и яиц |
| SELECT \* FROM Блюда WHERE Основа=’Яйца’ OR Основа=’Крупа’ OR Основа=’Овощи’; | SELECT \* FROM Блюда WHERE Основа=’Овощи’ OR Основа=’Мясо’ OR Основа=’Яйца’; |

**2.3.4. Использование LIKE**

Таблица 15

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выдать перечень салатов | Выдать перечень супов |
| SELECT Блюдо FROM Блюда WHERE Блюдо LIKE 'Салат%'; | SELECT Блюдо FROM Блюда WHERE Блюдо LIKE '%Суп%'; |

Символы этой константы интерпретируются следующим образом:

* символ \_ (подчеркивание) – заменяет любой одиночный символ,
* символ % (процент) – заменяет любую последовательность из N символов (где N может быть нулем),
* все другие символы означают просто сами себя.

**2.3.5. Вовлечение неопределенного значения (NULL-значения)**

Создаём таблицу “Наличие”:

**CREATE TABLE Наличие(ПР int, К\_во int, Стоим decimal(5,2));**

Заносим записи в созданную таблицу Рис. 16:

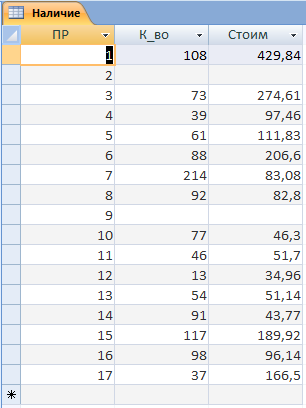


Рис. 16. Записи таблицы “Наличие”

Таблица 16

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выявить названия продуктов, отсутствующие в кладовой | Выявить названия продуктов, которые не поставляются |
| SELECT DISTINCT ПР FROM Наличие WHERE К\_во IS NULL; | SELECT DISTINCT ПР FROM Поставки WHERE К\_во IS NULL; |

Таблица 17

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выявить продукты, существующие в кладовой | Выявить названия продуктов, которые поставляются |
| SELECT DISTINCT ПР FROM Наличие WHERE К\_во IS NOT NULL; | SELECT DISTINCT ПР FROM Поставки WHERE К\_во IS NOT NULL OR К\_во <> 0; |

**2.4. Выборка с упорядочением**

Таблица 18

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выдать перечень продуктов и содержание в них основных веществ в порядке убывания содержания белка | Выдать перечень продуктов и содержание в них основных веществ в порядке убывания содержания углеводов |
| SELECT Продукт, Белки, Жиры, Углев FROM Продукты ORDER BY Белки DESC; | SELECT Продукт, Углев, Белки, Жиры FROM Продукты ORDER BY Углев DESC |

Таблица 19

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выдать содержимое таблицы Блюда, отсортировав ее строки по видам блюд и основе | Выдать содержимое таблицы Поставки, отсортировав ее строки по цене и количеству |
| SELECT \* FROM Блюда ORDER BY В Основа; | SELECT \* FROM Поставки ORDER BY Цена, К\_во; |

Таблица 20

|  |  |
| --- | --- |
| Из учебника | Мое |
| Получить список продуктов, переупорядоченный по возрастанию значений калорийности | Получить переупорядоченные по возрастанию значения жиров+50 |
| SELECT Продукт, ((Белки+Углев)\*4.1+Жиры\*9.3) FROM Продукты ORDER BY 2; | SELECT Продукт, (Жиры+50) FROM Продукты ORDER BY 2;  **Выборка с упорядочением.**  **В список ORDER BY можно включать не только имя столбца, а его порядковую позицию в перечне SELECT. Благодаря этому возможно упорядочение результатов на основе вычисляемых столбцов, не имеющих имен** |

**2.5. Агрегирование данных**

**2.5.1 SQL-функции**

В SQL существует ряд специальных стандартных функций (SQL-функций). Кроме специального случая COUNT(\*) каждая из этих функций оперирует совокупностью значений столбца некоторой таблицы и создает единственное значение, определяемое так:

COUNT - число значений в столбце,

SUM - сумма значений в столбце,

AVG - среднее значение в столбце,

MAX - самое большое значение в столбце,

MIN - самое малое значение в столбце.

Для функций SUM и AVG рассматриваемый столбец должен содержать числовые значения.

Следует отметить, что здесь столбец - это столбец виртуальной таблицы, в которой могут содержаться данные не только из столбца базовой таблицы, но и данные, полученные путем функционального преобразования и (или) связывания символами арифметических операций значений из одного или нескольких столбцов. При этом выражение, определяющее столбец такой таблицы, может быть сколь угодно сложным, но не должно содержать SQL-функций (вложенность SQL-функций не допускается). Однако из SQL-функций можно составлять любые выражения.

Аргументу всех функций, кроме COUNT(\*), может предшествовать ключевое слово DISTINCT (различный), указывающее, что избыточные дублирующие значения должны быть исключены перед тем, как будет применяться функция. Специальная же функция COUNT(\*) служит для подсчета всех без исключения строк в таблице (включая дубликаты).

**2.5.2. Функции без использования фразы GROUP BY**

Таблица 21

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выдать данные о массе лука (ПР=10), проданного поставщиками, и указать количество этих поставщиков | Выдать данные о цене помидоров (ПР=11), массе, проданных поставщиками, и указать количество этих поставщиков |
| SELECT SUM(К\_во),COUNT(К\_во) FROM Поставки WHERE ПР = 10; | SELECT SUM(Цена), SUM(К\_во), COUNT(К\_во) FROM Поставки WHERE ПР = 11; |

Таблица 22

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выдать данные о массе лука (ПР=10), проданного поставщиками, и указать количество этих поставщиков с номером продукта | Выдать данные о цене помидоров (ПР=11), массе, проданных поставщиками, и указать количество этих поставщиков с номером продукта |
| SELECT ПР,SUM(К\_во),COUNT(К\_во) FROM Поставки WHERE ПР = 10; | SELECT ПР, SUM(Цена), SUM(К\_во), COUNT(К\_во) FROM Поставки WHERE ПР = 11; |

Таблица 23

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выдать данные о массе лука (ПР=10), проданного поставщиками, и указать количество этих поставщиков | Выдать данные о цене помидоров (ПР=11), массе, проданных поставщиками, и указать количество этих поставщиков |
| SELECT 'Кол-во лука =',SUM(К\_во),COUNT(К\_во) FROM Поставки WHERE ПР = 10; | SELECT ‘Цена помидоров=’, SUM(Цена), SUM(К\_во), COUNT(К\_во) FROM Поставки WHERE ПР = 11; |

Таблица 24

|  |  |
| --- | --- |
| Из учебника | Мое |
| Получить суммы цен, средней цены, количества поставляемых продуктов и количество разных цен продуктов, проданных коопторгом УРОЖАЙ (ПС=5), а также получить количество продуктов, которые могут поставляться этим коопторгом |  |
| SELECT SUM(Цена), AVG(Цена), COUNT(Цена), COUNT(DISTINCT Цена),COUNT(\*) FROM Поставки WHERE ПС = 5; |  |

Таблица 25

|  |  |
| --- | --- |
| Из учебника | Мое |
| Сколько поставлено моркови и сколько поставщиков ее поставляют? | Сколько поставлено помидоров и сколько поставщиков их поставляют? |
| SELECT SUM(К\_во),COUNT(К\_во) FROM Поставки WHERE ПР = 2; | SELECT SUM(К\_во),COUNT(К\_во) FROM Поставки WHERE ПР = 11; |

Таблица 26

|  |  |
| --- | --- |
| Из учебника | Мое |
| Получить сумму массы поставленного лука с его средней ценой ("Сапоги с яичницей") | Получить сумму массы поставленных помидоров с их средней ценой |
| SELECT (SUM(К\_во)+AVG(Цена)) FROM Поставки WHERE ПР = 10; | SELECT ‘Сумма массы поставленных помидоров и цены=’, (SUM(К\_во)+AVG(Цена)) FROM Поставки WHERE ПР = 11; |

**2.5.3. Фраза GROUP BY**

Таблица 26

|  |  |
| --- | --- |
| Из учебника | Мое |
| Вычислить общую массу каждого из продуктов, поставляемых в настоящее время поставщиками | Вычислить общую цену каждого из продуктов, поставляемых в настоящее время поставщиками |
| SELECT ПР, SUM(К\_во) FROM Поставки GROUP BY ПР; | SELECT ПР, SUM(Цена) FROM Поставки GROUP BY ПР; |

Таблица 27

|  |  |
| --- | --- |
| Из учебника | Мое |
| Вычислить общую массу каждого из продуктов, поставляемых в настоящее время поставщиками и упорядочить по ПР | Вычислить общую цену каждого из продуктов, поставляемых в настоящее время поставщиками и упорядочить по ПР |
| SELECT ПР, SUM(К\_во) FROM Поставки GROUP BY ПР ORDER BY ПР; | SELECT ПР, SUM(Цена) FROM Поставки GROUP BY ПР ORDER BY ПР; |

Таблица 28

|  |  |
| --- | --- |
| Из учебника | Мое |
| Узнать коды и количество порций блюд, заказанных отдыхающими пансионата (32 человека) на каждую из трапез следующего дня |  |
| SELECTТ, БЛ, COUNT(БЛ) FROM Заказ GROUP BY Т, БЛ; |  |

Таблица 29

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выдать для каждого продукта его код и общий объем возможных поставок, учитывая временную недееспособность поставщика с ПС=2 | Выдать для каждого продукта его код и общий объем возможных поставок, учитывая временную недееспособность поставщиков: ПС=3, ПС=5 и ПС=7 |
| SELECT ПР, SUM(К\_во) FROM Поставки WHERE ПС <> 2 GROUP BY ПР; | SELECT ПР, SUM(К\_во) FROM Поставки WHERE ПС <> 3 AND ПС <> 5 AND ПС <> 7 GROUP BY ПР; |

**2.5.4. Использование фразы HAVING**

Таблица 30

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выдать коды продуктов, поставляемых более чем двумя поставщиками |  |
| SELECT FROM Поставки GROUP BY ПС HAVING COUNT(\*) > 2; |  |

**Глава 3. Запросы с использованием нескольких таблиц**

**3.1. О средствах одновременной работы с множеством таблиц**

Создаём таблицу “ Состав”:

**CREATE TABLE Состав(БЛ int, ПР int, Вес int);**

Заносим записи в созданную таблицу Рис. 17:

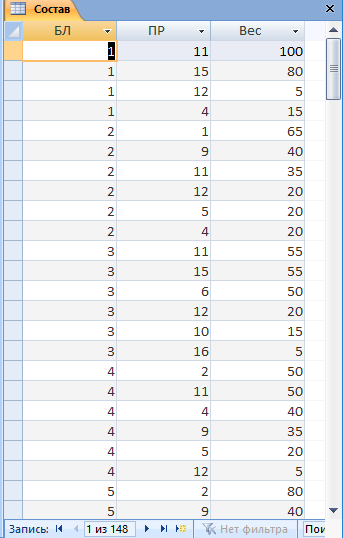


Рис. 17. Записи таблицы “ Состав”

Таблица 29

|  |  |
| --- | --- |
| Из учебника | Мое |
| Получить перечень поставщиков продуктов, необходимых для приготовления Сырников | Получить перечень поставщиков продуктов, необходимых для приготовления Кофе |
| SELECT Продукт, Цена, Название, Статус  FROM Продукты, Состав, Блюда, Поставки, Поставщики  WHERE Продукты.ПР = Состав.ПР  AND Состав.БЛ = Блюда.БЛ  AND Поставки.ПР = Состав.ПР  AND Поставки.ПС = Поставщики.ПС  AND Блюдо = 'Сырники'  AND Цена IS NOT NULL; | SELECT Продукт, Цена, Название, Статус, Город  FROM Продукты, Состав, Блюда, Поставки, Поставщики  WHERE Продукты.ПР = Состав.ПР  AND Состав.БЛ = Блюда.БЛ  AND Поставки.ПР = Состав.ПР  AND Поставки.ПС = Поставщики.ПС  AND Блюдо = 'Кофе черный'  AND Цена IS NOT NULL; |

Таблица 30

|  |  |
| --- | --- |
| Из учебника | Мое |
| Получить перечень тех поставщиков продуктов для Сырников, которые поставляют нужные продукты за минимальную цену | Получить перечень тех поставщиков продуктов для Салата летнего, которые поставляют нужные продукты за минимальную цену |
| SELECT Продукт, Цена, Название, Статус  FROM Продукты, Состав, Блюда, Поставки, Поставщики  WHERE Продукты.ПР = Состав.ПР  AND Состав.БЛ = Блюда.БЛ  AND Поставки.ПР = Состав.ПР  AND Поставки.ПС = Поставщики.ПС  AND Блюдо = 'Сырники'  AND Цена = ( SELECT MIN(Цена)  FROM Поставки X  WHERE X.ПР = Поставки.ПР ); | SELECT Продукт, Цена, Название, Статус, Город  FROM Продукты, Состав, Блюда, Поставки, Поставщики  WHERE Продукты.ПР = Состав.ПР  AND Состав.БЛ = Блюда.БЛ  AND Поставки.ПР = Состав.ПР  AND Поставки.ПС = Поставщики.ПС  AND Блюдо = 'Салат летний'  AND Цена = ( SELECT MIN(Цена)  FROM Поставки X  WHERE X.ПР = Поставки.ПР ); |

**3.2. Запросы, использующие соединения**

**3.2.1. Декартово произведение таблиц**

Создаём таблицу “ Вид\_блюд”:

**CREATE TABLE Вид\_блюд (В char(1), Вид varchar(10));**

Заносим записи в созданную таблицу Рис. 18:

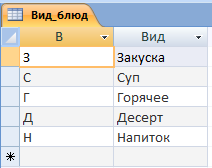


Рис. 18. Записи таблицы “ Вид\_блюд”

Создаём таблицу “ Трапезы”:

**CREATE TABLE Трапезы(Т int, Трапеза varchar(10));**

Заносим записи в созданную таблицу Рис. 19:

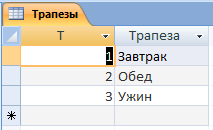


Рис. 19. Записи таблицы “ Трапезы”

Таблица 31

|  |  |
| --- | --- |
| Из учебника | Мое |
| Получить декартово произведение Вид\_блюд и Трапезы | Получить декартово произведение Вид\_блюд и Блюда |
| SELECT Вид\_блюд.\*, Трапезы.\*  FROM Вид\_блюд, Трапезы; | SELECT Вид\_блюд.\*, Блюда.\*  FROM Вид\_блюд, Блюда; |

Создаём таблицу “ Меню”:

**CREATE TABLE Меню(Т int, В char(1), БЛ int);**

Заносим записи в созданную таблицу Рис. 20:

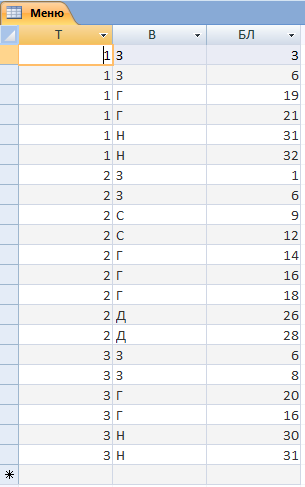


Рис. 20. Записи таблицы “ Меню”

Таблица 32

|  |  |
| --- | --- |
| Из учебника | Мое |
| Перемножить таблицы Меню, Трапезы, Вид\_блюд, Блюда | Перемножить таблицы Блюда, Состав, Продукты |
| SELECT Меню.\*, Трапезы.\*, Вид\_блюд.\*, Блюда.\*  FROM Меню, Трапезы, Вид\_блюд, Блюда; | SELECT Блюда.\*, Состав.\*, Продукты.\*  FROM Блюда, Состав, Продукты; |

**3.2.2. Эквисоединение таблиц**

Таблица 33

|  |  |
| --- | --- |
| Из учебника | Мое |
| Эквисоединение таблиц Меню, Трапезы, Вид\_блюд и Блюда | Эквисоединение таблиц Меню, Трапезы, Вид\_блюд и Блюда |
| SELECT Меню.\*, Трапезы.\*, Вид\_блюд.\*, Блюда.\*  FROM Меню, Трапезы, Вид\_блюд, Блюда  WHERE Меню.Т = Трапезы.Т  AND Меню.В = Вид\_блюд.В  AND Меню.БЛ = Блюда.БЛ; | SELECT Блюда.\*, Состав.\*, Продукты.\*  FROM Блюда, Состав, Продукты  WHERE Блюда.БЛ = Состав.БЛ  AND Состав.ПР = Продукты.ПР; |

**3.2.3. Естественное соединение таблиц**

Таблица 34

|  |  |
| --- | --- |
| Из учебника | Мое |
| Реализовать естественное соединение таблиц | Реализовать естественное соединение таблиц |
| SELECT Т, В, БЛ, Трапеза, Вид, Блюдо, Основа,  Выход, Труд  FROM Меню, Трапезы, Вид\_блюд, Блюда  WHERE Меню.Т = Трапезы.Т  AND Меню.В = Вид\_блюд.В  AND Меню.БЛ = Блюда.БЛ; | SELECT Трапеза, Вид, Блюдо, Основа, Выход, Труд  FROM Меню, Трапезы, Вид\_блюд, Блюда  WHERE Меню.Т = Трапезы.Т  AND Меню.В = Вид\_блюд.В  AND Меню.БЛ = Блюда.БЛ;    Ссылка на поле 'БЛ' может относиться к полям нескольких таблиц, перечисленных в предложении FROM |

**3.2.4. Композиция таблиц**

Таблица 35

|  |  |
| --- | --- |
| Из учебника | Мое |
| Для исключения всех столбцов, по которым проводится соединение таблиц, создать композицию |  |
| SELECT Трапеза, Вид, Блюдо, Основа, Выход, Труд  FROM Меню, Трапезы, Вид\_блюд, Блюда  WHERE Меню.Т = Трапезы.Т  AND Меню.В = Вид\_блюд.В  AND Меню.БЛ = Блюда.БЛ; | SELECT Трапеза, Вид, Блюдо, Основа  FROM Меню, Трапезы, Вид\_блюд, Блюда  WHERE Меню.Т = Трапезы.Т  AND Меню.В = Вид\_блюд.В  AND Меню.БЛ = Блюда.БЛ; |

**3.2.5. Тета-соединение таблиц**

Таблица 36

|  |  |
| --- | --- |
| Из учебника | Мое |
| Выбрать из полученного в п.3.2.1 декартова произведения таблиц Вид\_блюд и Трапезы лишь те строки, в которых значение трапезы "меньше" (по алфавиту) значения вида блюда | Выбрать из полученного в п.3.2.1 декартова произведения таблиц Вид\_блюд и Трапезы лишь те строки, в которых значение трапезы "больше" (по алфавиту) значения вида блюда |
| SELECT Вид\_блюд.\*, Трапезы.\*  FROM Вид\_блюд, Трапезы  WHERE Вид > Трапеза; | SELECT Вид\_блюд.\*, Трапезы.\*  FROM Вид\_блюд, Трапезы  WHERE Вид < Трапеза; |

**3.2.6. Соединение таблиц с дополнительным условием**

Таблица 37

|  |  |
| --- | --- |
| Из учебника | Мое |
| Получить перечень блюд, предлагаемых в меню на завтрак |  |
| SELECT Вид, Блюдо, Основа, Выход, 'Номер -', БЛ  FROM Меню, Трапезы, Вид\_блюд, Блюда  WHERE Меню.Т = Трапезы.Т  AND Меню.В = Вид\_блюд.В  AND Меню.БЛ = Блюда.БЛ  AND Трапеза = ’Завтрак’; |  |

**3.2.7. Соединение таблицы со своей копией**

Таблица 38

|  |  |
| --- | --- |
| Из учебника | Мое |
| Сформировать временную копию таблицы |  |
| FROM Блюда X, Блюда Y, Блюда Z |  |

Таблица 39

|  |  |
| --- | --- |
| Из учебника | Мое |
| Сформировать запрос соединения таблицы с ней самой на вывод таких пар блюд таблицы Блюда, в которых совпадает основа, а название первого блюда пары меньше (по алфавиту). Создать запрос с одной копией таблицы Блюда (Копия) |  |
| SELECT Блюдо, Копия.Блюдо, Основа  FROM Блюда, Блюда Копия  WHERE Основа = Копия.Основа  AND Блюдо < Копия.Блюдо; |  |

Таблица 40

|  |  |
| --- | --- |
| Из учебника | Мое |
| Сформировать запрос соединения таблицы с ней самой на вывод таких пар блюд таблицы Блюда, в которых совпадает основа, а название первого блюда пары меньше (по алфавиту). Создать запрос с двумя ее копиями (Первая и Вторая) |  |
| SELECT Первая.Блюдо, Вторая.Блюдо, Основа  FROM Блюда Первая, Блюда Вторая  WHERE Первая.Основа = Вторая.Основа  AND Первая.Блюдо < Вторая.Блюдо; |  |