**ТЕМА № 2. ОСНОВЫ ЯЗЫКА SQL**

**ОПЕРАТОРЫ МАНИПУЛИРОВАНИЯ ДАННЫМИ**

**Цели занятия:**

1. Развивать практические умения в написании SQL-команд для вставки данных в таблицы базы данных.
2. Развивать практические умения в написании SQL-команд для обновления и удаления данных из таблиц базы данных.

**УЧЕБНЫЕ ВОПРОСЫ**

**1. ВСТАВКА ДАННЫХ В ТАБЛИЦУ**

**Задание 1.1:** Написать SQL-команду для создания в БД «*test№\_2*» *(№ - цифра, указанная в конце имени вашего логина)* **(т.е., если у вас цифра 5 в конце логина, то имя вашей БД будет *test5\_2*)** таблицы с названием *table*\_1 и столбцами (ограничениями), представленными в таблице 1.

*Таблица* 1 – Столбцы (ограничения) таблицы *table*\_1

| **Название** | **Тип данных** | **Дополнительные характеристики** |
| --- | --- | --- |
| *fio* | символьный, размер до 20 символов | не может принимать пустые значения, составная часть первичного ключа (название ограничения *pk*\_1) |
| *N\_group* | целочисленный | не может принимать пустые значения, составная часть первичного ключа (название ограничения *pk*\_1), значение по умолчанию «123» (название ограничения *d*\_1) |
| *rang* | символьный, размер до 15 символов | значение по умолчанию «курсант» (название ограничения *d*\_2) |
| *salary* | численный, с плавающей запятой | значение по умолчанию «100,25» (название ограничения *d*\_3) |
| *date\_b* | дата | вводимая дата должна быть в диапазоне от «01.01.1990» до «31.12.1999» (название ограничения *ck*\_1) |

Вставить в таблицу *table*\_1 данные (табл. 2).

*Таблица* 2 – Данные таблицы *table*\_1

| **fio** | **N\_group** | **rang** | **salary** | **date\_b** |
| --- | --- | --- | --- | --- |
| *Антонов А.А.* | 123 | *курсант* | 100,25 | 31.01.1995 |
| *Борисов Б.Б.* | 123 | *командир* | 125,75 | 22.02.1995 |
| *Иванов И.И.* | 456 | *курсант* | 100,25 | 13.03.1995 |
| *Петров П.П.* | 123 | *курсант* | 100,25 | *NULL* |
| *Сидоров С.С.* | 123 | *NULL* | *NULL* | *NULL* |

Выполнить попытку вставки в таблицу *table*\_1 данных (табл. 3).

*Таблица* 3 – Данные

| **fio** | **N\_group** | **rang** | **salary** | **date\_b** |
| --- | --- | --- | --- | --- |
| *Сергеев С.С.* | 123 | *курсант* | 100,25 | 04.04.1985 |
| *NULL* | 123 | *курсант* | 100,25 | 15.05.1995 |
| *Петров П.П.* | 123 | *курсант* | 100,25 | 26.06.1995 |

Проанализировать выданные СУБД сообщения об ошибках и установить причину того, почему данные (табл. 3) не были вставлены в таблицу *table*\_1.

***Решение:*** Запустить веб-браузер и зайти на веб-сервис (см. Инструкция подключения к MS SQL Server). Создать БД с именем «*test№\_2*» в СУБД Microsoft SQL Server 2019. Для создания БД с кириллической кодировкой необходимо воспользоваться следующей командой:

**create database test0**

**collate Cyrillic\_General\_CI\_AS**

Для создания таблицы необходимо ввести SQL-команду вида:

**create table table\_1 (**

**fio varchar (20),**

**n\_group int constraint d\_1 default 123,**

**rang varchar (15) constraint d\_2 default 'курсант' null,**

**salary float constraint d\_3 default 100.25 null,**

**date\_b date null,**

**constraint pk\_1 primary key (fio, n\_group),**

**constraint ck\_1 check (date\_b>='1990-01-01' and date\_b<='1999-12-31'));**

**Следует обратить внимание**, что время указано в формате ГГГГ-ММ-ДД, а не ДД.ММ.ГГГГ, как это задано в таблицах 2 и 3 из описания задания. Такое форматирование даты необходимо потому, что тип данных времени *date* в MS SQL Server 2019 имеет строгую стандартизацию формата написания даты.

***Вставку данных можно выполнить с помощью инструкции INSERT несколькими способами:***

1) с использованием полного списка столбцов:

– столбцы расположены по порядку:

**insert table\_1 (fio, n\_group, rang, salary, date\_b)**

**values ('Антонов А.А.', 123, 'курсант', 100.25, '1995-01-31');**

– столбцы расположены в произвольном порядке:

**insert table\_1 (date\_b, salary, rang, n\_group, fio)**

**values ('1995-01-31', 100.25, 'курсант', 123,**

**'Антонов А.А.');**

2) с использованием усеченного списка столбцов:

**insert table\_1 (fio, date\_b)**

**values ('Антонов А.А.', '1995-01-31');**

3) без использования списка столбцов:

**insert table\_1 values ('Антонов А.А.', 123, 'курсант', 100.25, '1995-01-31');**

4) без использования списка столбцов и с использованием предложения DEFAULT:

**insert table\_1 values ('Антонов А.А.', default, default, default, '1995-01-31');**

При использовании любого способа СУБД проинформирует о вставке данных в таблицу *table*\_1 (рис. 1).

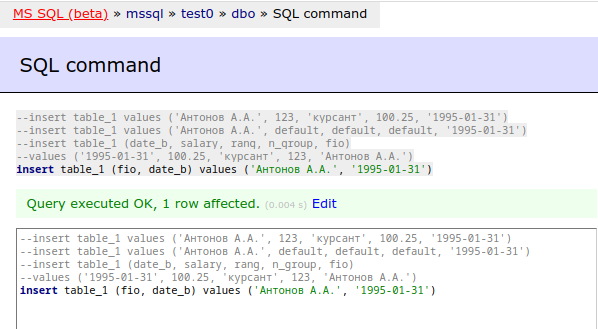


Рисунок 1

Просмотреть данные таблицы *table*\_1 можно, выполнив запрос:

**select \* from table\_1;**

При этом СУБД выдаст результаты выполнения запроса (рис. 2):

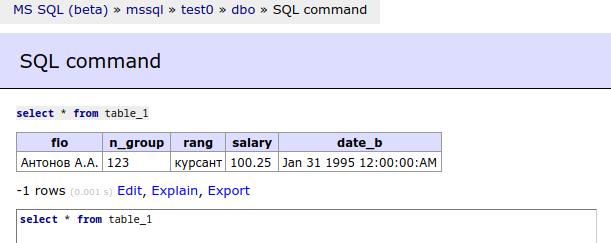


Рисунок 2

Для вставки остальных данных (табл. 2) необходимо вести SQL-команды вида:

**insert table\_1 (fio, rang, salary, date\_b) values ('Борисов Б.Б.', 'командир', 125.75, '1995-02-22');**

**insert table\_1 (fio, n\_group, date\_b) values ('Иванов И.И.', 456, '1995-03-13');**

**insert table\_1 (fio) values ('Петров П.П.');**

или явно задать пустое значение

**insert table\_1 (fio, date\_b)**

**values ('Петров П.П.', Null);**

**insert table\_1 (fio, rang, salary, date\_b)**

**values ('Сидоров С.С.', NULL, NULL, NULL);**

В результате таблица *table*\_1 будет содержать данные (рис. 3).

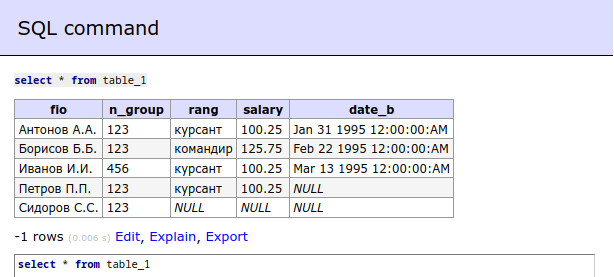


Рисунок 3

Добиться правильного написания SQL-команд для вставки требуемых данных в таблицу *table*\_1. В случае возникновения ошибки при выполнении SQL-команды проанализировать выданное СУБД сообщение об ошибке и внести необходимые коррективы в ее код.

***При выполнении SQL-команды вида:***

**insert table\_1 (fio, date\_b)**

**values ('Сергеев С.С.', '1985-04-04');**

СУБД прервет выполнение инструкции INSERT и проинформирует о возникшем конфликте с ограничением CHECK «*ck*\_1». Этот конфликт возник вследствие того, что вводимое значение в столбец «*date*\_*b*» в инструкции INSERT не принадлежит диапазону, заданному правилом «*ck*\_1» в инструкции CREATE TABLE для таблицы *table*\_1.

***При выполнении SQL-команды вида:***

**insert table\_1 (fio, date\_b)**

**values (NULL, '1995-05-15');**

СУБД прервет выполнение инструкции INSERT и проинформирует о том, что не удалось вставить пустое значение *NULL* в столбец «*fio*» таблицы *table*\_1. Этот конфликт возник вследствие того, что в столбце «*fio*» таблицы *table*\_1 запрещены пустые значения *NULL* (см. инструкцию CREATE TABLE).

***При выполнении SQL-команды вида:***

**insert table\_1 (fio, date\_b)**

**values ('Петров П.П.', '1995-06-26');**

СУБД прервет выполнение инструкции INSERT и проинформирует о нарушении ограничения PRIMARY KEY «*pk*\_1» и невозможности вставки повторяющегося ключа в таблицу *table*\_1. Этот конфликт возник вследствие того, что вводимые значения в столбцы «*fio*» и «*n*\_*group*», которые выступают в роли составного первичного ключа и предназначены для уникальной идентификации каждой строки таблицы, уже имеются в таблице *table*\_1, что приводит, в свою очередь, к нарушению уникальности.

**Задание 1.2:** Написать SQL-команду для создания в БД «*test№\_2*» таблицы *table*\_2 и столбцами (ограничениями), представленными в таблице 4.

*Таблица* 4 – Столбцы (ограничения) таблицы *table*\_2

| **Название** | **Тип данных** | **Дополнительные характеристики** |
| --- | --- | --- |
| *id* | целочисленный | не может принимать пустые значения, первичный ключ (название ограничения *pk*\_2), автоматически заполняемое |
| *fio* | символьный, размер до 20 символов |  |
| *N\_group* | целочисленный | значение по умолчанию «789» (название ограничения *d*\_4) |
| *rang* | символьный, размер до 15 символов | значение по умолчанию «курсант» (название ограничения *d*\_5) |

Вставить в таблицу *table*\_2 данные (табл. 5).

*Таблица* 5 – Данные таблицы *table*\_2

| **fio** | **N\_group** | **rang** |
| --- | --- | --- |
| *Иванов И.И.* | 789 | *курсант* |
| *Петров П.П.* | 789 | *курсант* |
| *Сидоров С.С.* | 789 | *курсант* |

Выполнить попытку вставки в таблицу *table*\_2 данных (табл. 6).

*Таблица* 6 – Данные

| **fio** | **N\_group** | **rang** |
| --- | --- | --- |
| *Сергеев С.С.* | *С*89 | *курсант* |
| *Сергеев С.С.* | 789 | *курсант* |

Проанализировать работу автоинкрементного заполнения, т.е. ответить на вопрос почему у записи, соответствующей Сергееву С.С., значение поля *id* равно 5 (или более), а не 4.

***Решение:*** Для создания таблицы необходимо вести SQL-команду вида:

**create table table\_2 (**

**id INT NOT NULL IDENTITY,**

**fio varchar (20),**

**n\_group int constraint d\_4 default 789,**

**rang varchar (15) constraint d\_5 default 'курсант',**

**constraint pk\_2 primary key (id));**

Во избежание повторного выполнения неиспользуемых SQL-команд необходимо создать новую вкладку для ввода запроса или закомментировать неиспользуемые SQL-команды.

Для вставки строк в таблицу необходимо вести SQL-команды вида:

**insert table\_2 (fio) values ('Иванов И.И.');**

**insert table\_2 (fio) values ('Петров П.П.');**

**insert table\_2 (fio) values ('Сидоров С.С.');**

***При выполнении SQL-команды вида:***

**insert table\_2 (fio, n\_group)**

**values ('Сергеев С.С.', 'С12');**

СУБД выдаст ошибку преобразования значения типа данных VARCHAR «С12» в тип данных INT. Этот конфликт возник вследствие того, что вводимое значение «С12» в столбец «*n\_group*» в инструкции INSERT относится к типу данных VARCHAR, в то время как для столбеца «*n\_group*» задан инструкцией CREATE TABLE тип данных INT.

Здесь ошибка преобразования типов была вызвана не случайно. Следует обратить внимание на то, что хотя попытка вставки данных закончилась неудачно, но при этом автоинкрементному полю было присвоено следующее значение: 4. Это ***обстоятельство следует иметь в виду*** при использовании полей с автоинкрементным заполнением.

Теперь при выполнении SQL-команды вида:

**insert table\_2 (fio) values ('Сергеев С.С.');**

Таблица *table*\_2 будет содержать следующие данные (рис. 4).

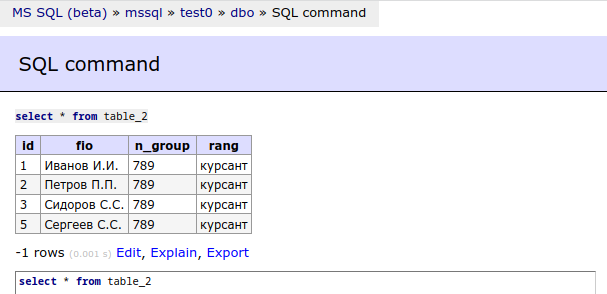


Рисунок 4

Добиться правильного написания SQL-команд для вставки требуемых данных в таблицу *table*\_2. В случае возникновения ошибки при выполнении SQL-команды проанализировать выданное СУБД сообщение об ошибке и внести необходимые коррективы в ее код.

**Задание 1.3:** Написать SQL-команду для создания в БД «*test№\_2*» таблицы *table*\_3 и столбцами (ограничениями), представленными в таблице 7.

*Таблица* 7 – Столбцы (ограничения) таблицы *table*\_3

| **Название** | **Тип данных** | **Дополнительные характеристики** |
| --- | --- | --- |
| *id* | целочисленный | не может принимать пустые значения, первичный ключ (название ограничения *pk*\_3) |
| *fio* | строковый, переменной длины, размер до 20 символов |  |
| *N\_group* | целочисленный |  |
| *rang* | строковый, переменной длины, размер до 15 символов |  |

Вставить в таблицу *table*\_3 данные (табл. 8).

*Таблица* 8 – Данные таблицы *table*\_3

| **id** | **fio** | **N\_group** | **rang** |
| --- | --- | --- | --- |
| 1 | *Иванов И.И.* | 123 | *курсант* |
| 2 | *Петров П.П.* | 123 | *курсант* |
| 3 | *Сидоров С.С.* | 123 | *курсант* |

Написать SQL-команду для создания в БД «*test№\_2*» дочерней таблицы *table*\_4, ссылающейся на родительскую таблицу *table*\_3 и имеющей столбцы (ограничения), представленные в таблице 9.

*Таблица* 9 – Столбцы (ограничения) таблицы *table*\_4

| **Название** | **Тип данных** | **Дополнительные характеристики** |
| --- | --- | --- |
| *id* | целочисленный | не может принимать пустые значения, первичный ключ (название ограничения *pk*\_4), внешний ключ, ссылающийся на столбец *id* таблицы *table*\_3 (название ограничения *fk*\_1) |
| *disc* | строковый, переменной длинны, размер до 25 символов | не может принимать пустые значения, первичный ключ (название ограничения *pk*\_4) |
| *date\_m* | дата | не может принимать пустые значения, первичный ключ (название ограничения *pk*\_4) |
| *mark* | целочисленный | вводимое значение должна принадлежать множеству [2, 3, 4, 5] (название ограничения *ck*\_2) |

Вставить в таблицу *table*\_4 данные (табл. 10).

*Таблица* 10 – Данные таблиц *table*\_4

| **id** | **disc** | **date\_m** | **mark** |
| --- | --- | --- | --- |
| 1 | *История* | 15.09.2015 | 5 |
| 2 | *Математика* | 17.09.2015 | 4 |
| 3 | *Физика* | 19.09.2015 | 3 |
| 1 | *Физика* | 15.09.2015 | 2 |

Выполнить попытку вставки в таблицу *table*\_4 данных (табл. 11).

*Таблица* 11 – Данные

| **id** | **disc** | **date\_m** | **mark** |
| --- | --- | --- | --- |
| 4 | *История* | 17.09.2015 | 5 |
| 3 | *Физика* | 19.09.2015 | 5 |
| 2 | *Математика* | 15.09.2015 | 1 |

Проанализировать выданные СУБД сообщения об ошибках и установить причину того, почему данные (табл. 11) не были вставлены в таблицу *table*\_4.

Написать SQL-команду для создания в БД «*test№\_2*» дочерней таблицы *table*\_5, ссылающейся на родительскую таблицу *table*\_3 и имеющей столбцы (ограничения), представленные в таблице 12.

*Таблица* 12 – Столбцы (ограничения) таблицы *table*\_5

| **Название** | **Тип данных** | **Дополнительные характеристики** |
| --- | --- | --- |
| id | целочисленный | не может принимать пустые значения, первичный ключ (название ограничения *pk*\_5), внешний ключ, ссылающийся на столбец *id* таблицы *table*\_3 (название ограничения *fk*\_2) **с правилом CASCADE для обновления и удаления** |
| disc | строковый, переменной длинны, размер до 25 символов | не может принимать пустые значения, первичный ключ (название ограничения *pk*\_5) |
| date\_m | дата | не может принимать пустые значения, первичный ключ (название ограничения *pk*\_5) |
| mark | целочисленный | вводимое значение должна принадлежать множеству [2, 3, 4, 5] (название ограничения *ck*\_3) |

Вставить в таблицу *table*\_5 данные из таблицы *table*\_4.

***Решение:*** Для создания родительской таблицы *table*\_3 необходимо вести SQL-команду вида:

**create table table\_3 (**

**id INT NOT NULL,**

**fio varchar (20),**

**n\_group int,**

**rang varchar (15),**

**constraint pk\_3 primary key (id));**

Для вставки данных в таблицу *table*\_3 необходимо вести SQL-команды:

**insert table\_3**

**values (1, 'Иванов И.И.', 123, 'курсант');**

**insert table\_3**

**values (2, 'Петров П.П.', 123, 'курсант');**

**insert table\_3**

**values (3, 'Сидоров С.С.', 123, 'курсант');**

Для создания дочерней таблицы *table*\_4 необходимо вести SQL-команду:

**create table table\_4 (**

**id int not null,**

**disc varchar (25) not null,**

**date\_m date not null,**

**mark tinyint,**

**constraint pk\_4 primary key(id, disc, date\_m),**

**constraint ck\_2 check (mark in (2, 3, 4, 5)),**

**constraint fk\_1 foreign key (id)**

**references table\_3 (id));**

Для вставки данных в таблицу *table*\_4 необходимо вести SQL-команды:

**insert table\_4**

**values (1, 'История', '2015-09-15', 5);**

**insert table\_4**

**values (2, 'Математика', '2015-09-17', 4);**

**insert table\_4 values (3, 'Физика', '2015-09-19', 3);**

**insert table\_4 values (1, 'Физика', '2015-09-15', 2);**

В результате таблица *table*\_4 будет содержать данные (рис. 5).

***Следует отметить,*** что столбцы (или столбец) родительской таблицы, указанные в ограничении FOREIGN KEY, должны быть объявлены в родительской таблице первичным или потенциальным ключом. В данном случае столбец *id* в таблице *table*\_3 является первичным ключом.

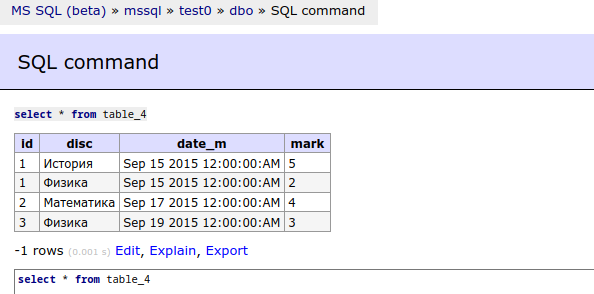


Рисунок 5

Также следует отметить, что данные сначала необходимо вставить в родительскую таблицу, и только затем вставлять в дочернюю таблицу.

***При выполнении SQL-команды вида:***

**insert table\_4**

**values (4, 'История', '2015-09-17', 5);**

СУБД прервет выполнение инструкции INSERT и проинформирует о возникшем конфликте с ограничением FOREIGN KEY «*fk*\_1». Этот конфликт возник вследствие того, что вводимое значение «4» в столбец «*id*» таблицы *table*\_4 отсутствует в столбце «*id*» родительской таблицы *table*\_3.

***При выполнении SQL-команды вида:***

**insert table\_4**

**values (3, 'Физика', '2015-09-19', 5);**

СУБД прервет выполнение инструкции INSERT и проинформирует о нарушении ограничения PRIMARY KEY «*pk*\_4» и невозможности вставки повторяющегося ключа в таблицу *table*\_4. Этот конфликт возник вследствие того, что вводимые значения в столбцы «*id*», «*disc*» и «*date\_m*», которые выступают в роли составного первичного ключа и предназначены для уникальной идентификации каждой строки таблицы, уже имеются в таблице *table*\_4.

***При выполнении SQL-команды вида:***

**insert table\_4**

**values (2, 'Математика', '2015-09-15', 1);**

СУБД прервет выполнение инструкции INSERT и проинформирует о возникшем конфликте с ограничением CHECK «*ck*\_2». Этот конфликт возник вследствие того, что вводимое значение в столбец «*mark*» в инструкции INSERT не принадлежит диапазону, заданному правилом «*ck*\_2» в инструкции CREATE TABLE для таблицы *table*\_4.

Для создания дочерней таблицы *table*\_5 необходимо вести SQL-команду:

**create table table\_5 (**

**id int not null,**

**disc varchar (25) not null,**

**date\_m date not null,**

**mark tinyint,**

**constraint pk\_5 primary key(id, disc, date\_m),**

**constraint fk\_2 foreign key (id)**

**references table\_3 (id) on delete cascade**

**on update cascade);**

Для вставки в таблицу *table*\_5 данных из таблицы *table*\_4 необходимо вести SQL-команду вида:

**insert table\_5 select \* from table\_4;**

В результате таблица *table*\_5 будет содержат те же данные, что и таблица *table*\_4 (рис. 6).

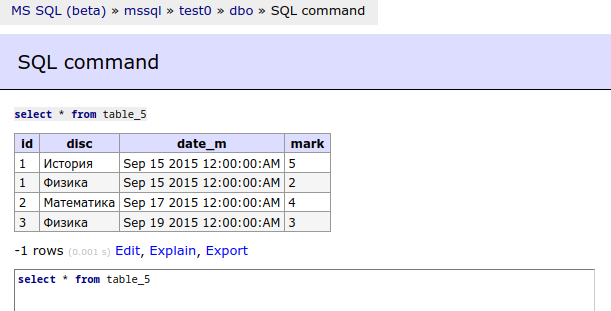


Рисунок 6

Добиться правильного написания SQL-команды для создания заданных таблиц и вставки в них требуемых данных. В случае возникновения ошибки при выполнении SQL-команды проанализировать выданное СУБД сообщение об ошибке и внести необходимые коррективы в ее код.

**2. ИЗМЕНЕНИЕ ДАННЫХ В ТАБЛИЦЕ**

**Задание 2.1:** Написать SQL-команду для изменения в таблице *table*\_4 оценки (столбец *mark*) с «2» на «5» у курсанта с идентификатором (столбец *id*) «1» по дисциплине (столбец *disc*) «Физика» за дату (столбец *date\_m*) «15.09.2015».

***Решение:*** Для изменения в таблице *table*\_4 оценки с «2» на «5» у курсанта с идентификатором «1» по дисциплине «Физика» за «15.09.2015» необходимо вести SQL-команду вида:

**update table\_4 set mark = 5**

**where id = 1 and disc = 'Физика'**

**and date\_m = '2015-09-15';**

В результате таблица *table*\_4 будет содержат данные (рис. 8).

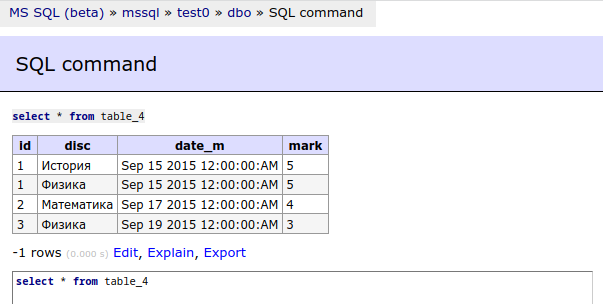


Рисунок 8

Добиться правильного написания SQL-команды для реализации заданного изменения данных в таблице *table*\_4. В случае возникновения ошибки при выполнении SQL-команды проанализировать выданное СУБД сообщение об ошибке и внести необходимые коррективы в ее код.

**Задание 2.2:** Написать SQL-команду для изменения в таблице *table*\_1 должности (столбец *rang*) на «командир» и должностного оклада (столбец *salary*) на «125,75» у военнослужащего (столбец *fio*) «Сидорова С.С.» из «123» группы (столбец *n\_group*).

***Решение:*** Для изменения в таблице *table*\_1 должности на «командир» и должностного оклада на «125,75» у курсанта «Сидорова С.С.» из «123» группы необходимо вести SQL-команду вида:

**update table\_1 set rang = 'командир', salary = 125.75**

**where fio = 'Сидоров С.С.' and n\_group = 123;**

В результате таблица *table*\_1 будет содержат данные (рис. 9).

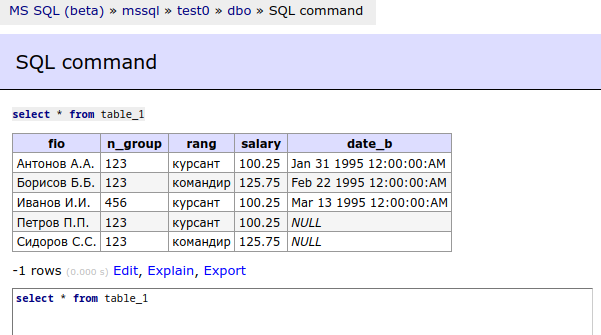


Рисунок 9

Добиться правильного написания SQL-команды для реализации заданного изменения данных в таблице *table*\_1. В случае возникновения ошибки при выполнении SQL-команды проанализировать выданное СУБД сообщение об ошибке и внести необходимые коррективы в ее код.

**Задание 2.3:** Написать SQL-команду для увеличения в таблице *table*\_1 должностного оклада (столбец *salary*) в полтора раза для всех военнослужащих.

***Решение:*** Для увеличения в таблице *table*\_1 должностного оклада для всех курсантов в полтора раза необходимо вести SQL-команду вида:

**update table\_1 set salary=salary\*1.5;**

В результате таблица *table*\_1 будет содержат данные (рис. 10).

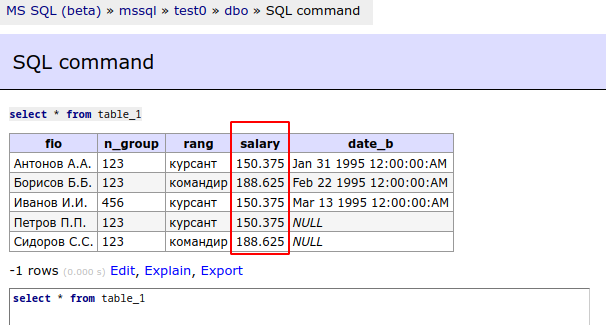


Рисунок 10

Добиться правильного написания SQL-команды для реализации заданного изменения данных в таблице *table*\_1. В случае возникновения ошибки при выполнении SQL-команды проанализировать выданное СУБД сообщение об ошибке и внести необходимые коррективы в ее код.

**Задание 2.4:** Написать SQL-команду для увеличения в таблице *table*\_1 должностного оклада (столбец *salary*) военнослужащим в должности «командир» на 15 у.е.

***Решение:*** Для увеличения в таблице *table*\_1 должностного оклада военнослужащим в должности «командир» на 15 у.е. необходимо вести SQL-команду вида:

**update table\_1 set salary=salary+15**

**where rang='командир';**

В результате таблица *table*\_1 будет содержат данные (рис. 11).

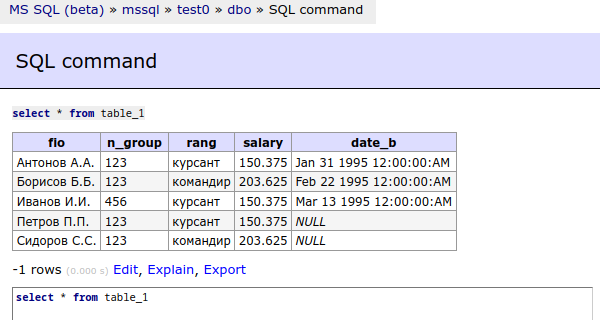


Рисунок 11

Добиться правильного написания SQL-команды для реализации заданного изменения данных в таблице *table*\_1. В случае возникновения ошибки при выполнении SQL-команды проанализировать выданное СУБД сообщение об ошибке и внести необходимые коррективы в ее код.

**3. УДАЛЕНИЕ ДАННЫХ ИЗ ТАБЛИЦЫ**

**Задание 3.1:** Написать SQL-команду для удаления из таблицы *table*\_2 строки с информацией о курсанте Сергееве С.С.

***Решение:*** Для удаления из таблицы *table*\_2 строки с информацией о курсанте Сергееве С.С. необходимо вести SQL-команду вида:

**delete table\_2 where fio='Сергеев С.С.';**

В результате таблица *table*\_2 будет содержат данные (рис. 13).

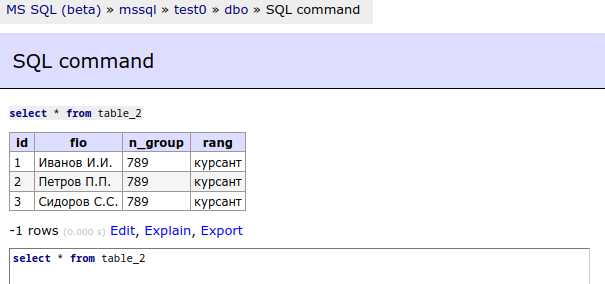


Рисунок 13

Добиться правильного написания SQL-команды для реализации удаления заданных строк из таблицы *table*\_2. В случае возникновения ошибки при выполнении SQL-команды проанализировать выданное СУБД сообщение об ошибке и внести необходимые коррективы в ее код.

**Задание 3.2:** Написать SQL-команду для удаления из таблицы *table*\_2 строки с информацией о курсанте с идентификатором «6».

***Решение:*** Для удаления из таблицы *table*\_2 строки с информацией о курсанте с идентификатором «6» необходимо вести SQL-команду вида:

**delete table\_2 where id=6;**

Добиться правильного написания SQL-команды для реализации удаления заданных строк в таблице *table*\_2. В случае возникновения ошибки при выполнении SQL-команды проанализировать выданное СУБД сообщение об ошибке и внести необходимые коррективы в ее код.

**Задание 3.3:** Написать SQL-команду для удаления всех строк из таблицы *table*\_2.

***Решение:*** Для удаления всех строк из таблицы *table*\_2 необходимо вести SQL-команду вида:

**delete table\_2;**

В результате таблица *table*\_2 будет содержат данные (рис. 14).

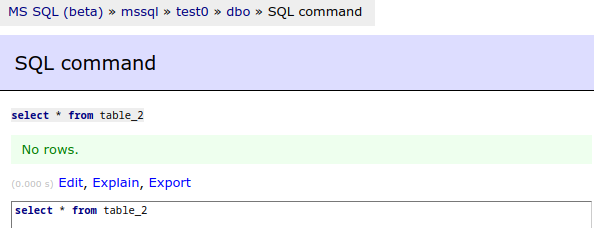


Рисунок 14

**Следует обратить внимание** на то, что при выполнении инструкции DELETE без предложения WHERE из таблицы удаляются все строки, но сама таблица остается в БД.

Добиться правильного написания SQL-команды для реализации удаления всех строк из таблицы *table*\_2. В случае возникновения ошибки при выполнении SQL-команды проанализировать выданное СУБД сообщение об ошибке и внести необходимые коррективы в ее код.

**Задание 3.4:** Написать SQL-команду для удаления из таблицы *table*\_5 строк с оценками по «Физике».

***Решение:*** Для удаления из таблицы *table*\_5 строк с оценками по «Физике» необходимо вести SQL-команду вида:

**delete table\_5 where disc = 'Физика';**

В результате таблица *table*\_5 будет содержат данные (рис. 15).

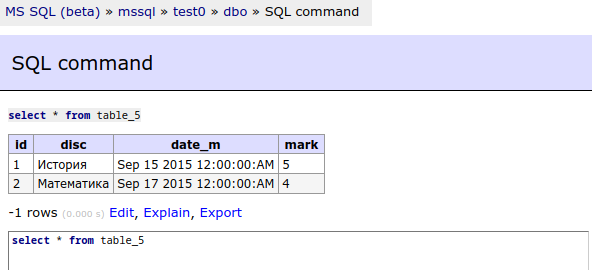


Рисунок 15

Добиться правильного написания SQL-команды для реализации удаления заданных строк из таблицы *table*\_5. В случае возникновения ошибки при выполнении SQL-команды проанализировать выданное СУБД сообщение об ошибке и внести необходимые коррективы в ее код.

**ЗАДАНИЕ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ**

***Задание 1.*** Написать SQL-команду для вставки в таблицу *table*\_2 строки со значениями по умолчанию.

***Задание 2.*** Написать SQL-команду для изменения в таблице *table*\_3 идентификатора (столбец *id*) со значением «1» на « ***x*** ».

***Задание 3.*** Написать SQL-команду для удаления из таблицы *table*\_3 строки о курсанте с идентификатором, равным « ***y*** ».

Варианты по списку:

|  | ***x*** | ***y*** |  | ***x*** | ***y*** |
| --- | --- | --- | --- | --- | --- |
| **1** | 101 | 3 | **16** | 117 | 1 |
| **2** | 102 | 2 | **17** | 118 | 2 |
| **3** | 103 | 1 | **18** | 119 | 3 |
| **4** | 104 | 1 | **19** | 121 | 3 |
| **5** | 105 | 2 | **20** | 122 | 2 |
| **6** | 106 | 3 | **21** | 123 | 1 |
| **7** | 107 | 3 | **22** | 124 | 1 |
| **8** | 108 | 2 | **23** | 125 | 2 |
| **9** | 109 | 1 | **24** | 126 | 3 |
| **10** | 111 | 1 | **25** | 127 | 3 |
| **11** | 112 | 2 | **26** | 128 | 2 |
| **12** | 113 | 3 | **27** | 129 | 1 |
| **13** | 114 | 3 | **28** | 131 | 1 |
| **14** | 115 | 2 | **29** | 132 | 2 |
| **15** | 116 | 1 | **30** | 133 | 3 |