

Базы данных и SQL. Обучение в записи

Урок 4. Семинар: SQL – создание объектов, простые запросы выборки

ЧАСТЬ I.

Домашнее задание

1. Используя операторы языка SQL, создайте таблицу “sales”. Заполните ее данными. Справа располагается рисунок к первому заданию.

id	order_date	count_product
1	2022-01-01	156
2	2022-01-02	180
3	2022-01-03	21
4	2022-01-04	124
5	2022-01-05	341

2. Для данных таблицы “sales” укажите тип заказа в зависимости от кол-ва :
меньше 100 - Маленький заказ
от 100 до 300 - Средний заказ
больше 300 - Большой заказ

id заказа	Тип заказа
1	Средний заказ
2	Средний заказ
3	Маленький заказ
4	Средний заказ
5	Большой заказ

Домашнее задание

3. Создайте таблицу “orders”, заполните ее значениями

id	employee_id	amount	order_status
1	e03	15.00	OPEN
2	e01	25.50	OPEN
3	e05	100.70	CLOSED
4	e02	22.18	OPEN
5	e04	9.50	CANCELLED

Выберите все заказы. В зависимости от поля order_status выведите столбец full_order_status: OPEN – «Order is in open state»; CLOSED - «Order is closed»; CANCELLED - «Order is cancelled»

4. Чем 0 отличается от NULL?

Напишите ответ в комментарии к домашнему заданию на платформе



1. Ход выполнения задания 1.

Используя операторы языка SQL, создать таблицу «sales». Заполнить ее данными.

```
1 • use seminar_2;
2
3 • CREATE TABLE sales
4 (
5     id INT PRIMARY KEY NOT NULL AUTO_INCREMENT,
6     order_date DATE NOT NULL,
7     count_product INT NOT NULL
8 );
9
10 • INSERT INTO sales(order_date, count_product)
11 VALUES
12 ('2022-01-01', 156),
13 ('2022-01-02', 180),
14 ('2022-01-03', 21),
15 ('2022-01-04', 124),
16 ('2022-01-05', 341);
```

	id	order_date	count_product
▶	1	2022-01-01	156
	2	2022-01-02	180
	3	2022-01-03	21
	4	2022-01-04	124
	5	2022-01-05	341

2. Ход выполнения задания 2.

Для данных таблицы «sales» указать тип заказа в зависимости от количества:

- меньше 100 – Маленький заказ;
- от 100 до 300 – Средний заказ;
- больше 300 – Большой заказ.

```
21 • SELECT id AS 'id_заказа',
22 CASE
23     WHEN count_product < 100 THEN 'Маленький заказ'
24     WHEN count_product > 100 AND count_product < 300 THEN 'Средний заказ'
25     WHEN count_product > 300 THEN 'Большой заказ'
26 END AS 'Тип заказа'
27 FROM sales;
28
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

id_заказа	Тип заказа
1	Средний заказ
2	Средний заказ
3	Маленький заказ
4	Средний заказ
5	Большой заказ

3. Ход выполнения задания 3.

Создать таблицу «orders» и заполнить ее значениями.

```
30 • CREATE TABLE orders
31 (
32     id INT PRIMARY KEY NOT NULL AUTO_INCREMENT,
33     employee_id VARCHAR(20) NOT NULL,
34     amount DECIMAL(10,2) NOT NULL,
35     order_status VARCHAR(45) NOT NULL
36 );
37
38 • INSERT INTO orders(employee_id, amount, order_status)
39 VALUES
40 ('e03', 15, 'OPEN'),
41 ('e01', 25.50, 'OPEN'),
42 ('e05', 100.70, 'CLOSED'),
43 ('e02', 22.18, 'OPEN'),
44 ('e04', 9.50, 'CANCELLED');
```

Выбрать все заказы. В зависимости от поля ORDER_STATUS столбец FULL_ORDER_STATUS:

- OPEN – “ORDER IS IN OPEN STATE”;
- CLOSED – “ORDER IS CLOSED”;
- CANCELLED – “ORDER IS CANCELLED”

```
48 • SELECT *,
49 CASE
50     WHEN order_status = 'OPEN'
51     THEN 'ORDER IS IN OPEN STATE'
52     WHEN order_status = 'CLOSED'
53     THEN 'ORDER IS CLOSED'
54     WHEN order_status = 'CANCELLED'
55     THEN 'ORDER IS CANCELLED'
56 END AS 'full_order_status'
57 FROM orders;
```

id	employee_id	amount	order_status	full_order_status
1	e03	15.00	OPEN	ORDER IS IN OPEN STATE
2	e01	25.50	OPEN	ORDER IS IN OPEN STATE
3	e05	100.70	CLOSED	ORDER IS CLOSED
4	e02	22.18	OPEN	ORDER IS IN OPEN STATE
5	e04	9.50	CANCELLED	ORDER IS CANCELLED

4. Ход выполнения задания 4.

Чем 0 отличается от NULL:

Ноль – числовой эквивалент (отсутствие чего-либо) и нужен для арифметических или логических операций.

В арифметических операциях ноль можно вычитать и складывать с другими числами — значение этих чисел при этом не изменится. Также можно ноль умножить или разделить любое число, но в результате это даст ноль.

В логических операциях ноль обозначает ложное значение — `false`. При этом любое другое число будет отвечать за правдивые значения — `true`.

NULL в SQL используется для обозначения отсутствующих данных.

IS NULL и IS NOT NULL — специально созданные операторы, которые осуществляют сравнение имеющихся NULLов. IS NULL возвращает истину, если операнда является NULLом. Соответственно, если операнд не является NULLом, то значение будет ложным.

ЧАСТЬ II.

1. Ход выполнения задания 0.

Запрос на создание начальной таблицы, выполнять задания на основе этих данных.

Вы можете воспользоваться заготовкой fiddle <https://dbfiddle.uk/hZWhxR8S>.

Или использовать код ниже для создания таблицы:

```
DROP TABLE IF EXISTS books;
```

```
CREATE TABLE books (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  title VARCHAR(255) NOT NULL,  
  author VARCHAR(255) NOT NULL,  
  year INT NOT NULL,  
  price DECIMAL(10, 2) NOT NULL  
);
```

```
INSERT INTO books (title, author, year, price)  
VALUES
```

```
('Убить пересмешника', 'Харпер Ли', 1960, 300.00),  
(1984, 'Джордж Оруэлл', 1949, 250.00),  
( 'Великий Гэтсби', 'Ф. Скотт Фицджеральд', 1925, 200.00),  
( 'Над пропастью во ржи', 'Дж. Д. Сэлинджер', 1951, 280.00),  
( 'Моби Дик', 'Герман Мелвилл', 1851, 400.00),  
( 'Скотный двор', 'Джордж Оруэлл', 1945, 220.00),  
( 'Почти на Каталонии', 'Джордж Оруэлл', 1938, 180.00),  
( 'Собака Баскервиль', 'Артур Конан Дойл', 1902, 350.00),  
( 'Дневник Анны Франк', 'Анна Франк', 1947, 300.00),  
( 'Краткая история времени', 'Стивен Хокинг', 1988, 320.00);
```

id	title	author	year	price
1	Убить пересмешника	Харпер Ли	1960	300.00
2	1984	Джордж Оруэлл	1949	250.00
3	Великий Гэтсби	Ф. Скотт Фицджеральд	1925	200.00
4	Над пропастью во ржи	Дж. Д. Сэлинджер	1951	280.00
5	Моби Дик	Герман Мелвилл	1851	400.00
6	Скотный двор	Джордж Оруэлл	1945	220.00
7	Почти на Каталонии	Джордж Оруэлл	1938	180.00
8	Собака Баскервиль	Артур Конан Дойл	1902	350.00
9	Дневник Анны Франк	Анна Франк	1947	300.00
10	Краткая история времени	Стивен Хокинг	1988	320.00

2. Ход выполнения задания 1.

Создание таблицы с жанрами книг.

Создайте таблицу (сущность) с жанрами книг *GENRES*.

Перечень полей (атрибутов):

- *id* – числовой тип, автоинкремент, первичный ключ;
- *name* – строковый тип, обязательный к заполнению.

Заполните сущность *genres* следующими данными:

name
Художественная литература
Нехудожественная литература
Детектив
Биография
Наука

```
26 • DROP TABLE IF EXISTS genres;
27
28 • CREATE TABLE genres
29 (
30     id INT PRIMARY KEY NOT NULL AUTO_INCREMENT,
31     name VARCHAR(50) NOT NULL
32 );
33
34 • INSERT INTO genres (name)
35 VALUES
36 ('Художественная литература'),
37 ('Нехудожественная литература'),
38 ('Детектив'),
39 ('Биография'),
40 ('Наука');
```

	id	name
►	1	Художественная литература
	2	Нехудожественная литература
	3	Детектив
	4	Биография
	5	Наука

3. Ход выполнения задания 2.

Создайте таблицу (сущность) с авторами *authors*.

Перечень полей (атрибутов):

- *id* – числовой тип, автоинкремент, первичный ключ;
- *name* – строковый тип, обязательный к заполнению.

Заполните сущность *authors* данными из таблицы *books*.

```
45 • DROP TABLE IF EXISTS authors;
46
47 • CREATE TABLE authors
48   (
49     id INT PRIMARY KEY NOT NULL AUTO_INCREMENT,
50     name VARCHAR(50) NOT NULL
51   );
52
53
54 • INSERT INTO authors(name)
55     SELECT DISTINCT author FROM books;
```

	id	name
►	1	Харпер Ли
	2	Джордж Оруэлл
	3	Ф. Скотт Фицджеральд
	4	Дж. Д. Сэлинджер
	5	Герман Мелвилл
	6	Артур Конан Дойл
	7	Анна Франк
	8	Стивен Хокинг

4. Ход выполнения задания 3.

Добавление жанров и авторов в таблицу книг и вывод книг по жанрам и авторам.

Имеется таблица (сущность) с книгами *books*. У сущности имеются следующие поля (атрибуты):

- *id* – идентификатор;
- *itle* – название;
- *author* – автор;
- *year* – год издания;
- *price* – цена.

Создайте новые столбцы *genre_id* и *author_id* в таблице *books* и обновите значения этих столбцов в соответствии с жанром и автором каждой книги.

title	genre	author
To Kill a Mockingbird	Fiction	Harper Lee
1984	Fiction	George Orwell
The Great Gatsby	Fiction	F. Scott Fitzgerald
The Catcher in the Rye	Fiction	J.D. Salinger
Moby-Dick	Fiction	Herman Melville
Animal Farm	Fiction	George Orwell
Homage to Catalonia	Non-Fiction	George Orwell
The Hound of the Baskervilles	Mystery	Arthur Conan Doyle
The Diary of a Young Girl	Biography	Anne Frank
A Brief History of Time	Science	Stephen Hawking

Необходимо вывести название книг, их жанры и авторов.

```

63 • UPDATE books
64     SET genre_id = (SELECT id FROM genres WHERE name = 'Художественная литература')
65     WHERE title IN ('Убить пересмешника', '1984', 'Великий Гэтсби', 'Над пропастью во ржи', 'Моби Дик');
66
67 • UPDATE books
68     SET genre_id = (SELECT id FROM genres WHERE name = 'Нехудожественная литература')
69     WHERE title IN ('Скотный двор', 'Почти на Каталонии');
70
71 • UPDATE books
72     SET genre_id = (SELECT id FROM genres WHERE name = 'Детектив')
73     WHERE title = 'Собака Баскервией';
74 • UPDATE books
75     SET genre_id = (SELECT id FROM genres WHERE name = 'Биография')
76     WHERE title = 'Дневник Анны Франк';
77
78 • UPDATE books
79     SET genre_id = (SELECT id FROM genres WHERE name = 'Наука')
80     WHERE title = 'Краткая история времени';

```



```



83 • UPDATE books
84     SET author_id = (SELECT id FROM authors WHERE name = 'Харпер Ли')
85     WHERE title = 'Убить пересмешника';
86
87 • UPDATE books
88     SET author_id = (SELECT id FROM authors WHERE name = 'Джордж Оруэлл')
89     WHERE title IN ('1984', 'Скотный двор', 'Почти на Каталонии');
90
91 • UPDATE books
92     SET author_id = (SELECT id FROM authors WHERE name = 'Ф. Скотт Фицджеральд')
93     WHERE title = 'Великий Гэтсби';
94
95 • UPDATE books
96     SET author_id = (SELECT id FROM authors WHERE name = 'Дж. Д. Сэлинджер')
97     WHERE title = 'Над пропастью во ржи';
98
99 • UPDATE books
100    SET author_id = (SELECT id FROM authors WHERE name = 'Герман Мелвилл')
101    WHERE title = 'Моби Дик';
102
103 • UPDATE books
104    SET author_id = (SELECT id FROM authors WHERE name = 'Артур Конан Дойл')
105    WHERE title = 'Собака Баскервиль';
106
107 • UPDATE books
108    SET author_id = (SELECT id FROM authors WHERE name = 'Анна Франк')
109    WHERE title = 'Дневник Анны Франк';
110
111 • UPDATE books
112    SET author_id = (SELECT id FROM authors WHERE name = 'Стивен Хокинг')
113    WHERE title = 'Краткая история времени';

```

```

116 • SELECT books.title AS title, genres.name AS name, authors.name AS name
117        FROM books
118        INNER JOIN genres ON books.genre_id = genres.id
119        INNER JOIN authors ON books.author_id = authors.id;
120
121

```

Result Grid			
Filter Rows: <input type="text"/>			
Export: 			
Wrap Cell Content: 			
	title	name	name
►	Убить пересмешника	Художественная литература	Харпер Ли
	1984	Художественная литература	Джордж Оруэлл
	Великий Гэтсби	Художественная литература	Ф. Скотт Фицджеральд
	Над пропастью во ржи	Художественная литература	Дж. Д. Сэлинджер
	Моби Дик	Художественная литература	Герман Мелвилл
	Скотный двор	Нехудожественная литература	Джордж Оруэлл
	Почти на Каталонии	Нехудожественная литература	Джордж Оруэлл
	Собака Баскервиль	Детектив	Артур Конан Дойл
	Дневник Анны Франк	Биография	Анна Франк
	Краткая история времени	Наука	Стивен Хокинг

5. Ход выполнения задания 4.

Обновление цен на книги.

Имеется таблица (сущность) с книгами *books*. У сущности имеются следующие поля (атрибуты):

- *id* – идентификатор;
- *title* – название;
- *author* – автор;
- *year* – год издания;
- *price* – цена.

Необходимо увеличить цену на все книги, изданные до 2000 года, на 10%:

```
123 • UPDATE books
124     SET price = price + (price * 0.1)
125     WHERE year < 2000;
126
127 • SELECT * FROM books;
128
```

	id	title	author	year	price	genre_id	author_id
▶	1	Убить пересмешника	Харпер Ли	1960	330.00	1	1
	2	1984	Джордж Оруэлл	1949	275.00	1	2
	3	Великий Гэтсби	Ф. Скотт Фицджеральд	1925	220.00	1	3
	4	Над пропастью во ржи	Дж. Д. Сэлинджер	1951	308.00	1	4
	5	Моби Дик	Герман Мелвилл	1851	440.00	1	5
	6	Скотный двор	Джордж Оруэлл	1945	242.00	2	2
	7	Почти на Каталонии	Джордж Оруэлл	1938	198.00	2	2
	8	Собака Баскервилей	Артур Конан Дойл	1902	385.00	3	6
	9	Дневник Анны Франк	Анна Франк	1947	330.00	4	7
	10	Краткая история времени	Стивен Хокинг	1988	352.00	5	8

6. Ход выполнения задания 5.

Выборка книг по средней цене жанра.

Имеется таблица (сущность) с книгами *books* и таблица с жанрами *genres*. У сущностей имеются следующие поля (атрибуты):

- *books.id* – идентификатор;
- *books.title* – название;
- *books.author* – автор;
- *books.year* – год издания;
- *books.price* – цена;
- *books.genre_id* – идентификатор жанра;
- *genres.id* – идентификатор;
- *genres.name* – название жанра.

Необходимо вывести название жанра и среднюю цену книг в этом жанре для жанров, где средняя цена выше 300.

```
129 • SELECT genres.name AS name, AVG(books.price) AS price
130     FROM books
131     INNER JOIN genres ON books.genre_id = genres.id
132     GROUP BY genres.name
133     HAVING AVG(books.price) > 300;
```

	name	price
▶	Художественная литература	314.600000
	Детектив	385.000000
	Биография	330.000000
	Наука	352.000000