

Лабораторная работа №5-6**Тема «Создание запросов на языке SQL. Операторы CREATE, INSERT, UPDATE, DELETE»**

Кол-во часов: 4

Ход работы:**Оператор SQL CREATE TABLE**

Оператор CREATE TABLE используется для создания новой таблицы в базе данных.

Синтаксис

```
CREATE [TEMPORARY] TABLE [IF NOT EXISTS] tbl_name  
    (create_definition,...)  
    [table_options]  
    [partition_options]
```

или

```
CREATE TABLE table_name (  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
);
```

Параметры столбцов определяют имена столбцов таблицы.

Параметр datatype указывает тип данных, которые может содержать столбец (например, varchar, integer, date и т. д.).


Задание

1. Создайте таблицу с именем «Persons», содержащую пять столбцов: PersonID, LastName, FirstName, Address и City:

```
CREATE TABLE Persons (  
    PersonID int NOT NULL PRIMARY KEY,  
    LastName varchar(255),  
    FirstName varchar(255),  
    Address varchar(255),  
    City varchar(255)  
);
```

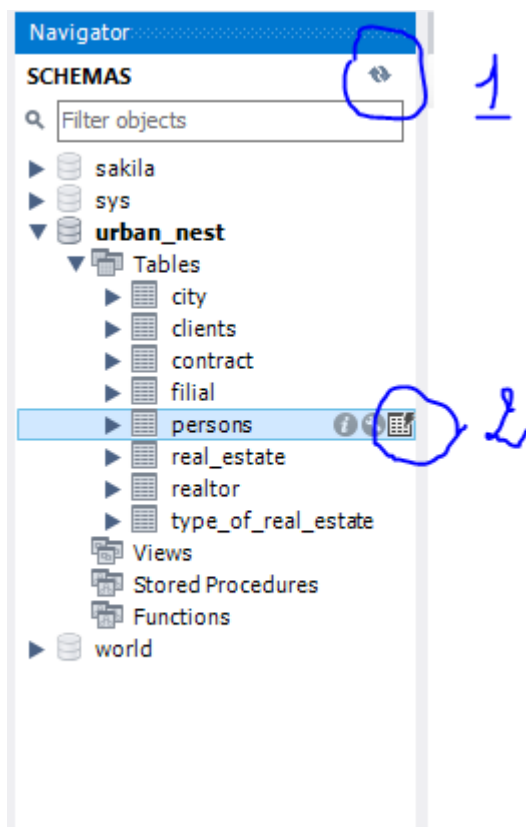
Столбец PersonID имеет тип int и будет содержать целое число, NOT NULL – обозначает недопустимость пустых значений, PRIMARY KEY – первичный ключ.

Столбцы LastName, FirstName, Address и City имеют тип varchar и могут содержать символы, а максимальная длина этих полей составляет 255 символов.

Чтобы запустить скрипт необходимо нажать кнопку «выполнения»  И в нижней части экрана в окне «Output» отобразится сообщение об успешном выполнении скрипта или об ошибках.

Output				
Action Output				
#	Time	Action	Message	
2	12:43:04	CREATE TABLE Persons (PersonID int, LastName varchar(255), FirstName varchar(255), Address varchar(255), City varchar(255))	0 row(s) affected	

Чтобы увидеть результат выполнения скрипта обновите все базы и нажмите на кнопку напротив названия таблицы



Пустая таблица «Персоны» теперь будет выглядеть так:

PersonID	LastName	FirstName	Address	City
----------	----------	-----------	---------	------

Создать таблицу, используя другую таблицу

Копию существующей таблицы также можно создать с помощью CREATE TABLE.

Новая таблица получает те же определения столбцов. Можно выбрать все столбцы или определенные столбцы.

Если вы создаете новую таблицу, используя существующую таблицу, новая таблица будет заполнена существующими значениями из старой таблицы.

Синтаксис

```
CREATE TABLE new_table_name AS
SELECT column1, column2,...
FROM existing_table_name
WHERE ....;
```

2. Создайте новую таблицу с именем «TestTable» (которая является копией таблицы «Persons»):

```
CREATE TABLE TestTable AS  
SELECT PersonID, LastName, FirstName, Address, City  
FROM Persons;
```

Оператор SQL ALTER TABLE

Оператор ALTER TABLE используется для добавления, удаления или изменения столбцов в существующей таблице.

ALTER TABLE оператор также используется для добавления и удаления различных ограничений для существующей таблицы.

Чтобы добавить столбец в таблицу, используйте следующий *синтаксис*:

```
ALTER TABLE table_name  
ADD column_name datatype;
```

3. Добавьте в таблицу «Persons» столбец «DateOfBirth».

```
ALTER TABLE Persons  
ADD DateOfBirth date;
```

Теперь таблица «Persons» выглядит следующим образом

	PersonID	LastName	FirstName	Address	City	DateOfBirth
--	----------	----------	-----------	---------	------	-------------

Обратите внимание, что новый столбец "DateOfBirth" имеет тип date и будет содержать дату.

4. Измените тип данных столбца «DateOfBirth» в таблице «Persons».

Используйте следующий оператор SQL:

```
ALTER TABLE Persons  
MODIFY COLUMN DateOfBirth YEAR;
```

Столбец «DateOfBirth» теперь имеет тип «год» и будет содержать год в двух- или четырехзначном формате.

5. Удаление столбца

Удалите столбец «DateOfBirth» с помощью следующего оператора SQL:

```
ALTER TABLE Persons  
DROP COLUMN DateOfBirth;
```

Таблица «Персоны» теперь будет выглядеть так:

	PersonID	LastName	FirstName	Address	City
--	----------	----------	-----------	---------	------

6. Удаление таблицы

Синтаксис

```
DROP TABLE table_name;
```

Удалите, созданные нами учебные таблицы Persons и TestTable.

Оператор SQL INSERT INTO

Оператор INSERT INTO используется для вставки новых записей в таблицу.

Синтаксис INSERT INTO. Укажите имена столбцов и значения, которые необходимо вставить:

```
INSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
```

Если вы добавляете значения для всех столбцов таблицы, вам не нужно указывать имена столбцов в SQL-запросе. Однако убедитесь, что порядок значений соответствует порядку столбцов в таблице. Здесь синтаксис INSERT INTO будет следующим:

```
INSERT INTO table_name
VALUES (value1, value2, value3, ...);
```

7. Внесем записи в таблицу Clients базы данных «urbane_nest»

а)

```
insert into clients (clients_Id, cl_lname, cl_fname, phone, city, state, email, date_of_birth)
value (1, 'Bennet', 'Abraham', '415 658-9932', 'Berkley', 'CA', 'abraham@gmail.com', '1985-01-01');
```

И таким способом

б)

```
insert into clients
value (3, 'Green', 'Marjorie', '415 986-7020', 'Oakland', 'CA', 'Marjorie@gmail.com', '1985-01-01');
```

В результате увидим обе эти записи внесены в таблицу:

	clients_ID	cl_lname	cl_fname	phone	city	state	email	date_of_birth
▶	1	Bennet	Abraham	415 658-9932	Berkley	CA	abraham@gmail.com	1985-01-01
	3	Green	Marjorie	415 986-7020	Oakland	CA	Marjorie@gmail.com	1986-03-01
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Обратите внимание, что, хотя для поля clients_Id мы определили заполнение auto-increment, но при заполнении можем указывать разные значения.

Также можно вставлять данные только в определенные столбцы.

Пример

```
INSERT INTO Customers (CustomerName, City, Country)
VALUES ('Cardinal', 'Stavanger', 'Norway');
```

Для того чтобы внести в таблицу несколько записей одновременно используется следующий синтаксис:

в)

```

insert into clients
value (2, 'Carson', 'Cheryl', '415 548-7723', 'Berkley', 'CA', 'Cheryl@icloud.com', '1987-03-05'),
      (4, 'Ringer', 'Albert', '801 826-0752', 'Salt Lake City', 'UT', 'Albertl@company.com', '2000-07-25'),
      (5, 'Ringer', 'Anne', '801 826-0752', 'Salt Lake City', 'UT', 'Ringer@icloud.com', '2002-12-05'),
      (6, 'Defrance', 'Michel', '219 547-9982', 'Gary', 'IN', NULL, '1992-03-25'),
      (7, 'Panteley', 'Sylvia', '301 946-8853', 'Rockville', 'MD', 'Panteley@company.com', '1988-03-05'),
      (8, 'McBadden', 'Heather', '707 448-4982', 'Nashville', 'TN', NULL, '1995-10-08'),
      (9, 'Stringer', 'Dirk', '415 843-2991', 'Oakland', 'CA', 'Stringer@gmail.com', '1987-12-05'),
      (10, 'DeLongue', 'Martinella', '415 843-2222', 'Denver', 'CO', 'DeLongue_m@company.com', '1983-12-15'),
      (11, 'Dull', 'Ann', '415 836-7128', 'Corvallis', 'OR', NULL, '1993-08-15');

```

Результат:

	clients_ID	d_l_name	d_f_name	phone	city	state	email	date_of_birth
►	1	Bennet	Abraham	415 658-9932	Berkley	CA	abraham@gmail.com	1985-01-01
	2	Carson	Cheryl	415 548-7723	Berkley	CA	Cheryl@icloud.com	1987-03-05
	3	Green	Marjorie	415 986-7020	Oakland	CA	Marjorie@gmail.com	1986-03-01
	4	Ringer	Albert	801 826-0752	Salt Lake City	UT	Albertl@company.com	2000-07-25
	5	Ringer	Anne	801 826-0752	Salt Lake City	UT	Ringer@icloud.com	2002-12-05
	6	Defrance	Michel	219 547-9982	Gary	IN	NULL	1992-03-25
	7	Panteley	Sylvia	301 946-8853	Rockville	MD	Panteley@company.com	1988-03-05
	9	Stringer	Dirk	415 843-2991	Oakland	CA	Stringer@gmail.com	1987-12-05
	10	DeLongue	Martinella	415 843-2222	Denver	CO	DeLongue_m@company.com	1983-12-15
	11	Dull	Ann	415 836-7128	Corvallis	OR	NULL	1993-08-15
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Table 1 clients

8. Самостоятельно заполните остальные таблицы по ниже приведенным данным.

	city
►	Berkley
	Corvallis
	Covelo
	Gary
	Nashville
	Oakland
	Palo Alto
	Rockville
	Salt Lake City
*	NULL

Table 2 city

	filial_ID	filial_Name	city	address
►	10122	urban_nest1	Berkley	6223 Bateman St.
	10152	urban_nest2	Corvallis	589 Darwin Ln.
	11104	urban_nest3	Oakland	1956 Arlington Pl.
	12002	urban_nest4	Rockville	3 Silver Ct.
	12010	urban_nest6	Palo Alto	5 Upland Hts.
*	NULL	NULL	NULL	NULL

Table 3 filials

	realtor_ID	r_lname	r_fname	phone	filial_ID	salary	city	state
▶	10	Yokomoto	Akiko	415 935-4228	10122	1200.00	Palo Alto	CA
	11	Green	Morningstar	615 297-2723	10122	1500.00	Nashville	TN
	12	White	Johnson	408 496-7223	10152	900.00	Berkley	CA
	13	del Castillo	Innes	615 996-8275	NULL	650.00	Palo Alto	CA
	14	Hunter	Sheryl	415 836-7128	11104	1350.00	Salt Lake City	UT
	15	Locksley	Chastity	415 585-4620	11104	1800.00	Salt Lake City	UT
	16	Smith	Meander	913 843-0462	11104	1100.00	Salt Lake City	UT
	17	Karsen	Livia	415 534-9219	12002	1350.00	Gary	IN
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Table 4 realtors

	type_of_re
▶	Detached House
	office
	one-room apartment
	studio apartment
	three-room apartment
	Townhouse
	two-room apartment
*	NULL

Table 5 type_of_re

	contract_ID	date	clients_ID	real_estate_ID	realtor_ID
▶	1	2024-03-13	2	1	10
	2	2022-04-05	3	2	10
	3	2023-10-20	4	3	12
	4	2024-08-18	5	5	13
	5	2024-05-30	7	5	11
	6	2024-07-11	8	6	14
*	NULL	NULL	NULL	NULL	NULL

Table 6 contract

	realestate_ID	type_of_re	price	city	address	area	description	discount	agency_commission
▶	1	one-room apartment	2500.00	Berkley	6223 Bateman St.	43	Charming studio apartment available for rent. Modern design, open-plan living and fully equipped kitchen. Prime location, close to amenities	100	200
	2	two-room apartment	5000.00	Berkley	67 Seventh Av.	63	Spacious 2-bedroom apartment for rent. Bright living area, modern kitchen. Convenient location, close to shops and transport.	100	300
	3	two-room apartment	5700.00	Salt Lake City	309 63rd St. #411	73	Cozy 2-bedroom apartment for rent. Features a spacious living area, vintage kitchen. Affordable and conveniently located near shops.	200	300
	4	office	7200.00	Salt Lake City	3 Balding Av.	125	Office space available for rent. Spacious and well-lit with flexible layout options. Ideal for small to medium businesses. Convenient location, close to transport	200	500
	5	Detached House	3700.00	Covelo	44 Upland Hts	110	Private house for rent. Needs renovation but offers spacious rooms and a large yard. Great potential for those looking to customize their living space.	200	500
	6	three-room apartment	4200.00	Rockville	PO Box 792	93	NULL	NULL	200
	18	one-room apartment	2500.00	Berkley	62 Bateman St.	43	Spacious apartment available for rent. Offers two bedrooms, a full kitchen, and a comfortable living room. Located near schools and shopping centers	100	200
	19	two-room apartment	2500.00	Berkley	97 Bleaker	63	long term apartment rental	100	200
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Table 7 realestate

Оператор SQL UPDATE

Оператор UPDATE используется для изменения существующих записей в таблице.

Синтаксис

```
UPDATE table_name
SET column1 = value1, column2 = value2, ...
WHERE condition;
```

9. Используя команду UPDATE измените записи в таблице Clients**а)**

```
UPDATE clients  
SET state = 'TN', City= 'Frankfurt';
```

!* При выполнении данной команды может возникнуть ошибка

Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses a KEY column. To disable safe mode, toggle the option in Preferences -> SQL Editor and reconnect.

Поскольку включена опция *Safe mode*. Чтобы её отключить выполните команду
SET SQL_SAFE_UPDATES = 0;

б) Теперь попробуйте обновить только одну запись, подумайте, как это сделать.**Оператор SQL DELETE**

Оператор DELETE используется для удаления существующих записей в таблице.

```
DELETE FROM table_name;
```

10.**а)** Удалите все записи в таблице и внесите заново.**б)** Затем удалите только одну запись, где значение state = TN.**Индивидуальное задание** – создать и заполнить данными таблицы для своего варианта**Литература:**

1. Боуман Дж. Практическое руководство по SQL/ Дж. Боуман, С. Эмерсон, М. Мардновски, 2017. – 322 с.
2. <https://www.w3schools.com/sql>

Инструменты:

1. MySQLWorkbench.
2. MySQL.