

Номер 1

Создать таблицу users с полями:

- id целочисленный ключ,
- first_name строка из 50 символов не может хранить NULL,
- last_name строка из 50 символов не может хранить NULL,
- date_joined хранит дату и время не может хранить NULL

)

```
CREATE table users
(id INT PRIMARY KEY,
first_name NVARCHAR(50) NOT NULL,
last_name NVARCHAR(50) NOT NULL);
```

Создайте таблицу users_data, в которой первичный ключ id будет также внешним ключом с ссылкой на таблицу users. Помимо первичного ключа, таблица должна содержать поле bio типа TEXT

```
CREATE table users_data
(id INT PRIMARY KEY foreign key references Users(id) not null on
delete cascade,
bio text NOT NULL);
```

Номер 2

В базе данных интернет-магазина по продаже красок есть таблица products, которая была создана с помощью следующего запроса:

```
CREATE TABLE products (
  id INT NOT NULL IDENTITY (1,1) PRIMARY KEY,
  name NVARCHAR(50) NULL,
  count INTEGER NULL,
  price INTEGER NULL
);
```

Создайте таблицу products_details, которая будет содержать дополнительные текстовые данные о краске и связана с таблицей products отношением один-к-одному. В таблице должно быть три поля: id, product_id и description.

Поле product_id должно быть внешним ключом, который ссылается на поле id в таблице products.

(обратите внимание, что внешний ключ должен быть точно такого же типа, что и поле, на которое он ссылается)

Поле description должно быть типа TEXT.

```
CREATE TABLE product_details (
  id INT IDENTITY (1,1) PRIMARY KEY not null,
  product_id INT foreign key references products(id) not NULL,
  description text NULL );
```

Номер 3

В таблице users хранится базовая информация о пользователях, а в таблице users_details — подробная. Таблицы связаны отношением один-к-одному так, что первичные ключи в таблицах совпадают, но могут быть рассинхронизированы (id в одной таблице могут опережать ключи в другой).

- 1) Создать 2 таблицы и связать их связью один к одному

id	first_name	last_name	age
1	Вероника	Петрова	20
2	Светлана	Иванова	17
3	Елена	Абрамова	18
4	Василиса	Кац	15
5	Алёна	Сорокина	22

id	bio
1	Биография Вероники Петровой
2	Биография Светланы Ивановой
3	Биография Елены Абрамовой
4	Биография Василисы Кац
5	Биография Алёны Сорокиной

```
CREATE TABLE users (  
    id INT IDENTITY (1,1) PRIMARY KEY not null,  
    first_name NVARCHAR(50) not NULL,  
    last_name NVARCHAR (50) not Null,  
    age INT not null  
);
```

```
INSERT INTO users(first_name, last_name, age)  
VALUES ('Вероника', 'Петрова', 20),  
('Светлана', 'Иванова', 17),  
('Елена', 'Абрамова', 18),  
('Василис' , 'Кац', 15),  
('Алёна', 'Сорокина', 22);
```

```
CREATE TABLE users_details (  
    id INT PRIMARY KEY foreign key references users(id) not  
null,  
    bio text not NULL);
```

```
INSERT INTO users_details (id, bio)
```

```
VALUES (1, 'Биография Вероники Петровой'),  
(2, 'Биография Светланы Ивановой'),  
(3, 'Биография Елены Абрамовой'),  
(4, 'Биография Василисы Кац'),  
(5, 'Биография Алёны Сорокиной');
```

2) Добавьте нового пользователя со следующими данными:

Имя: Антон

Фамилия: Дмитриев

Биография: «Антон родился в 1993 году.»

```
INSERT INTO users(first_name, last_name, age)  
VALUES ('Антон', 'Дмитриев', 29);
```

```
INSERT INTO users_details (id, bio)  
VALUES (6, 'Антон родился в 1993 году.');
```