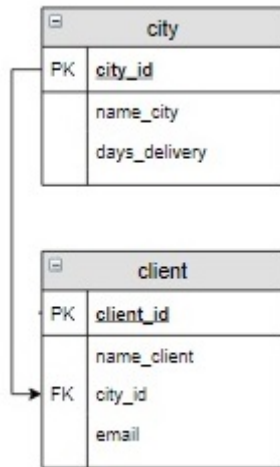


Задание

Запрос

Включить нового человека в таблицу с клиентами. Его имя **Попов Илья**, его email **popov@test**, проживает он в **Москве**.



| client_id | name_client | city_id | email |
|-----------|-----------------|---------|----------------|
| 1 | Баранов Павел | 3 | baranov@test |
| 2 | Абрамова Катя | 1 | abramova@test |
| 3 | Семенов Иван | 2 | semenov@test |
| 4 | Яковлева Галина | 1 | yakovleva@test |
| 5 | Попов Илья | 1 | popov@test |

```

INSERT INTO client (name_client, city_id, email)
SELECT 'Попов Илья', city_id, 'popov@test'
FROM city
WHERE name_city = 'Москва';
    
```

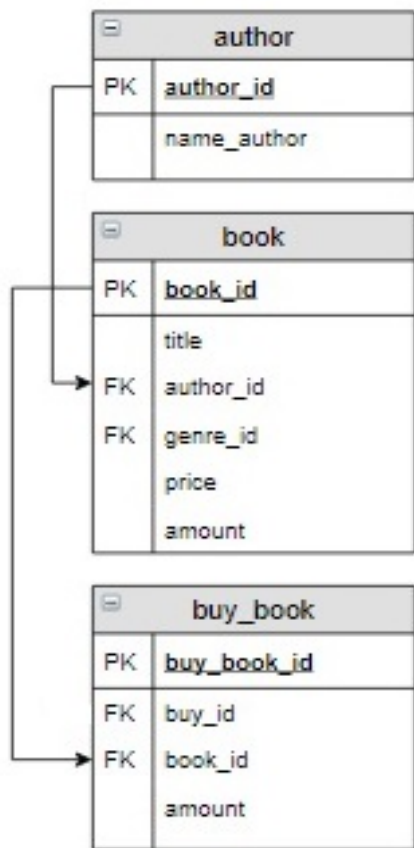
Создать новый заказ для Попова Ильи. Его комментарий для заказа: «Связаться со мной по вопросу доставки».

| buy_id | buy_description | client_id |
|--------|---------------------------------------|-----------|
| 1 | Доставка только вечером | 1 |
| 2 | NULL | 3 |
| 3 | Упаковать каждую книгу по отдельности | 2 |
| 4 | NULL | 1 |
| 5 | Связаться со мной по вопросу доставки | 5 |

```

INSERT INTO buy (buy_description, client_id)
SELECT 'Связаться со мной по вопросу доставки', client_id
FROM client
WHERE name_client = 'Попов Илья';
    
```

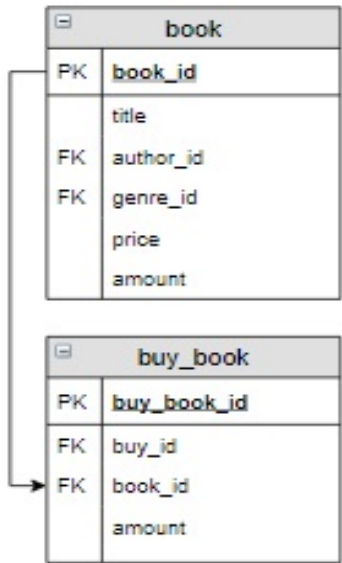
В таблицу **buy_book** добавить заказ с номером 5. Этот заказ должен содержать книгу Пастернака «Лирика» в количестве двух экземпляров и книгу Булгакова «Белая гвардия» в одном экземпляре.



| buy_book_id | buy_id | book_id | amount |
|-------------|--------|---------|--------|
| 1 | 1 | 1 | 1 |
| 2 | 1 | 7 | 2 |
| 3 | 1 | 4 | 1 |
| 4 | 2 | 8 | 2 |
| 5 | 3 | 3 | 2 |
| 6 | 3 | 2 | 1 |
| 7 | 3 | 1 | 1 |
| 8 | 4 | 5 | 1 |
| 9 | 5 | 8 | 2 |
| 10 | 5 | 2 | 1 |

```
INSERT INTO buy_book (buy_id, book_id, amount)
SELECT buy_id, book_id, 2
FROM buy, book
WHERE buy_id=5 AND title='Лирика'
UNION ALL
SELECT buy_id, book_id, 1
FROM buy, book
WHERE buy_id=5 AND title='Белая гвардия'
```

Количество тех книг на складе, которые были включены в заказ с номером 5, уменьшить на то количество, которое в заказе с номером 5 указано.



| book_id | title | author_id | genre_id | price | amount |
|---------|-----------------------|-----------|----------|--------|--------|
| 1 | Мастер и Маргарита | 1 | 1 | 670.99 | 3 |
| 2 | Белая гвардия | 1 | 1 | 540.50 | 4 |
| 3 | Идиот | 2 | 1 | 460.00 | 10 |
| 4 | Братья Карамазовы | 2 | 1 | 799.01 | 3 |
| 5 | Игрок | 2 | 1 | 480.50 | 10 |
| 6 | Стихотворения и поэмы | 3 | 2 | 650.00 | 15 |
| 7 | Черный человек | 3 | 2 | 570.20 | 6 |
| 8 | Лирика | 4 | 2 | 518.99 | 0 |

```
UPDATE book
JOIN buy_book USING(book_id)
SET book.amount = book.amount - buy_book.amount
WHERE buy_book.buy_id=5
```

Создать счет (таблицу **buy_pay**) на оплату заказа с номером 5, в который включить название книг, их автора, цену, количество заказанных книг и стоимость. Последний столбец назвать **Стоимость**. Информацию в таблицу занести в отсортированном по названиям книг виде.

| title | name_author | price | amount | Стоимость |
|---------------|----------------|--------|--------|-----------|
| Белая гвардия | Булгаков М.А. | 540.50 | 1 | 540.50 |
| Лирика | Пастернак Б.Л. | 518.99 | 2 | 1037.98 |

```
CREATE TABLE buy_pay AS
SELECT title, name_author, price, buy_book.amount, (
    price*buy_book.amount) AS Стоимость
FROM buy_book
    INNER JOIN book USING(book_id)
    INNER JOIN author USING(author_id)
WHERE buy_id=5
ORDER BY title;
```


Создать общий счет (таблицу **buy_pay**) на оплату заказа с номером 5. Куда включить номер заказа, количество книг в заказе (название столбца **Количество**) и его общую стоимость (название столбца **Итого**). Для решения используйте ОДИН запрос.

| buy_id | Количество | Итого |
|--------|------------|---------|
| 5 | 3 | 1578.48 |

```
CREATE TABLE buy_pay AS
SELECT buy_id, SUM(buy_book.amount) AS Количество,
SUM(price*buy_book.amount) AS Итого
FROM buy_book
CROSS JOIN book USING(book_id)
WHERE buy_id=5;
```

В таблицу **buy_step** для заказа с номером 5 включить все этапы из таблицы **step**, которые должен пройти этот заказ. В столбцы **date_step_beg** и **date_step_end** всех записей занести Null.

| buy_step_id | buy_id | step_id | date_step_beg | date_step_end |
|-------------|--------|---------|---------------|---------------|
| 1 | 1 | 1 | 2020-02-20 | 2020-02-20 |
| 2 | 1 | 2 | 2020-02-20 | 2020-02-21 |
| 3 | 1 | 3 | 2020-02-22 | 2020-03-07 |
| 4 | 1 | 4 | 2020-03-06 | 2020-03-06 |
| 5 | 2 | 1 | 2020-02-28 | 2020-02-28 |
| 6 | 2 | 2 | 2020-02-29 | 2020-03-01 |
| 7 | 2 | 3 | 2020-03-02 | NULL |
| 8 | 2 | 4 | NULL | NULL |
| 9 | 3 | 1 | 2020-03-05 | 2020-03-05 |
| 10 | 3 | 2 | 2020-03-05 | 2020-03-06 |
| 11 | 3 | 3 | 2020-03-06 | 2020-03-10 |
| 12 | 3 | 4 | 2020-03-11 | NULL |
| 13 | 4 | 1 | 2020-03-20 | NULL |
| 14 | 4 | 2 | NULL | NULL |
| 15 | 4 | 3 | NULL | NULL |
| 16 | 4 | 4 | NULL | NULL |
| 17 | 5 | 1 | NULL | NULL |
| 18 | 5 | 2 | NULL | NULL |
| 19 | 5 | 3 | NULL | NULL |
| 20 | 5 | 4 | NULL | NULL |

```
INSERT INTO buy_step (buy_id, step_id)
SELECT buy.buy_id, step_id
FROM buy, step
WHERE buy.buy_id=5;
```

В таблицу **buy_step** занести дату 12.04.2020 выставления счета на оплату заказа с номером 5.

Правильнее было бы занести не конкретную, а текущую дату. Это можно сделать с помощью функции **Now()**. Но при этом в разные дни будут вставляться разная дата, и задание нельзя будет проверить, поэтому вставим дату 12.04.2020.

| buy_step_id | buy_id | step_id | date_step_beg | date_step_end |
|-------------|--------|---------|---------------|---------------|
| 17 | 5 | 1 | 2020-04-12 | NULL |
| 18 | 5 | 2 | NULL | NULL |
| 19 | 5 | 3 | NULL | NULL |
| 20 | 5 | 4 | NULL | NULL |



```
UPDATE buy_step
JOIN step USING(step_id)
SET date_step_beg = '2020-04-12'
WHERE buy_id=5 AND name_step='Оплата';
```

Завершить этап «Оплата» для заказа с номером 5, вставив в столбец **date_step_end** дату 13.04.2020, и начать следующий этап («Упаковка»), задав в столбце **date_step_beg** для этого этапа ту же дату.

Реализовать два запроса для завершения этапа и начала следующего. Они должны быть записаны в общем виде, чтобы его можно было применять для любых этапов, изменив только текущий этап. Для примера пусть это будет этап «Оплата».

| buy_step_id | buy_id | step_id | date_step_beg | date_step_end |
|-------------|--------|---------|---------------|---------------|
| 17 | 5 | 1 | 2020-04-12 | 2020-04-13 |
| 18 | 5 | 2 | 2020-04-13 | NULL |
| 20 | 5 | 4 | NULL | NULL |

Придумайте один или несколько запросов корректировки данных для предметной области «Интернет-магазин книг».

```
UPDATE buy_step
JOIN step USING(step_id)
SET date_step_end = '2020-04-13'
WHERE buy_id = 5 AND name_step = 'Оплата';

UPDATE buy_step
SET date_step_beg = '2020-04-13'
WHERE buy_id = 5 AND step_id = (
    SELECT step_id + 1
    FROM step
    WHERE name_step = 'Оплата');
```

```
UPDATE buy
JOIN client USING(client_id)
SET buy_description = 'Доставка только утром'
WHERE name_client='Семенов Иван';
```

Логическая схема базы данных:

