Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчет

по лабораторной работе №5 «Процедуры, функции, триггеры в PostgreSQL»

по дисциплине **«Проектирование и реализация баз данных»**

Автор: Красюк К.А.

Факультет: ИКТ

Группа: К3241

Преподаватель: Говорова М.М.

Санкт-Петербург 2023

Оглавление

Цель работы

Овладеть практическими создания и использования процедур, функций и триггеров в базе данных PostgreSQL.

Практическое задание

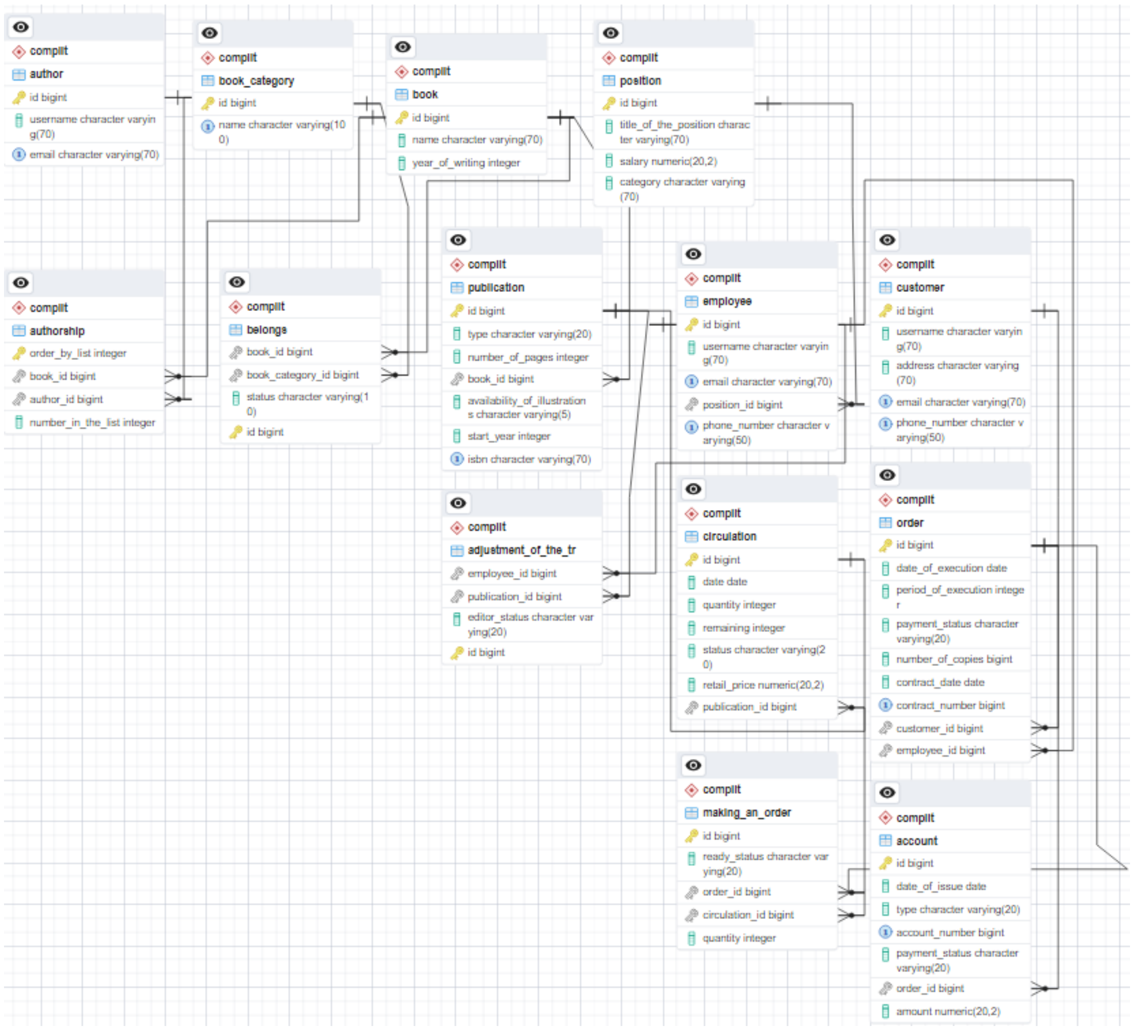
Вариант 2 (max - 8 баллов)

1. Создать процедуры/функции согласно индивидуальному заданию (часть 4).

2.1. Модифицировать триггер (триггерную функцию) на проверку корректности входа и выхода сотрудника (см. Практическое задание 1 Лабораторного практикума (Приложение)) с максимальным учетом «узких» мест некорректных данных по входу и выходу).

2.2. Создать авторский триггер по варианту индивидуального задания.

Схема базы данных (Вариант 5 - БД «Издательство компьютерной литературы»)

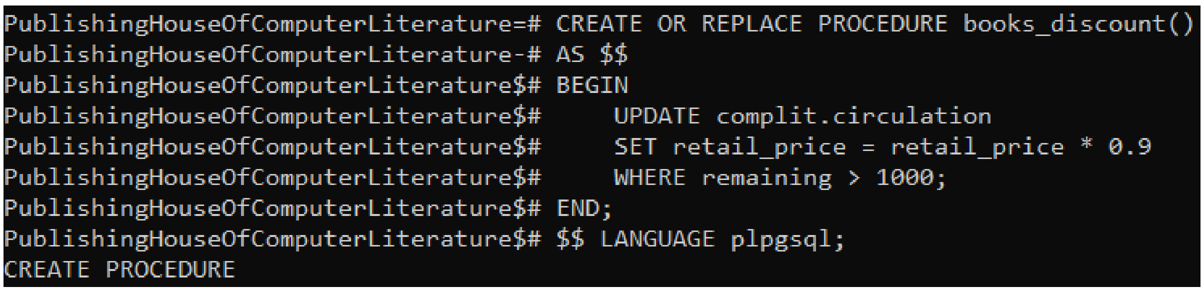


Выполнение

- **Создание процедур/функций согласно индивидуальному заданию**

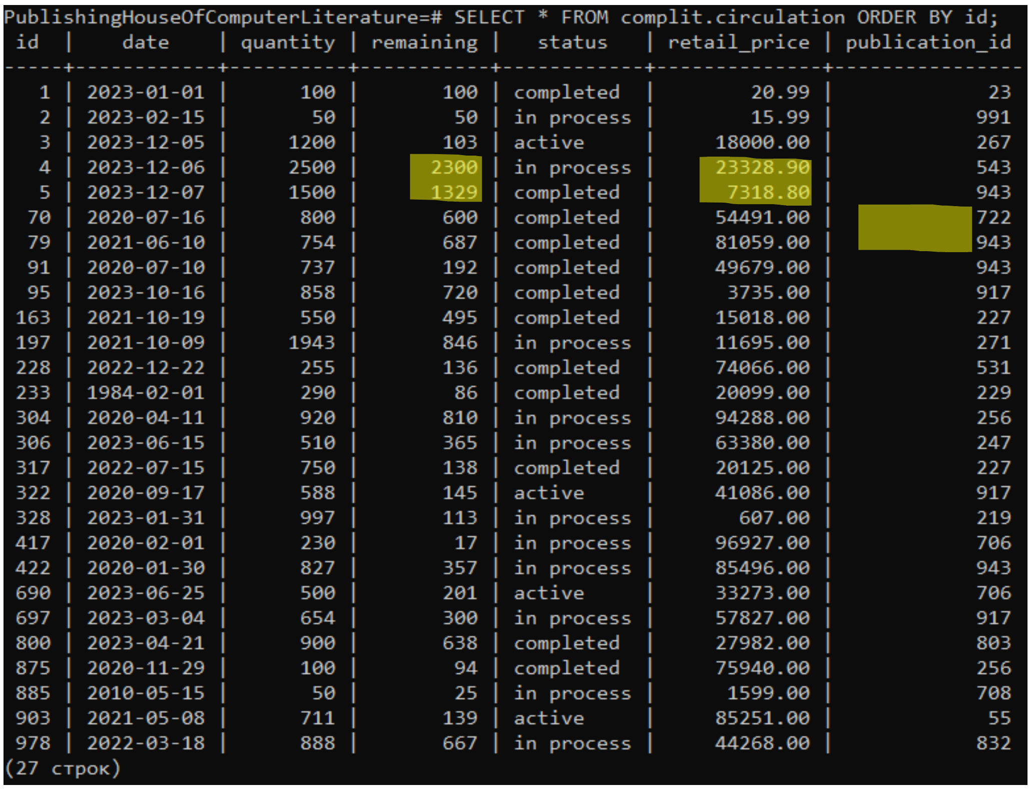
1) Для снижения цен на книги, которые находятся на базе в количестве, превышающем 1000 штук.

```
CREATE OR REPLACE PROCEDURE books_discount()

AS $$

BEGIN

   UPDATE complit.circulation

   SET retail_price = retail_price * 0.9

   WHERE remaining > 1000;

END;

$$ LANGUAGE plpgsql;
```



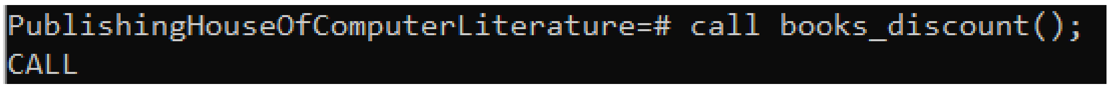
```
PublishingHouseOfComputerLiterature=# CREATE OR REPLACE PROCEDURE books_discount()
PublishingHouseOfComputerLiterature-# AS $$
PublishingHouseOfComputerLiterature$# BEGIN
PublishingHouseOfComputerLiterature$#     UPDATE complit.circulation
PublishingHouseOfComputerLiterature$#     SET retail_price = retail_price * 0.9
PublishingHouseOfComputerLiterature$#     WHERE remaining > 1000;
PublishingHouseOfComputerLiterature$# END;
PublishingHouseOfComputerLiterature$# $$ LANGUAGE plpgsql;
CREATE PROCEDURE
```

Рисунок 1. Создание процедуры books_discount()



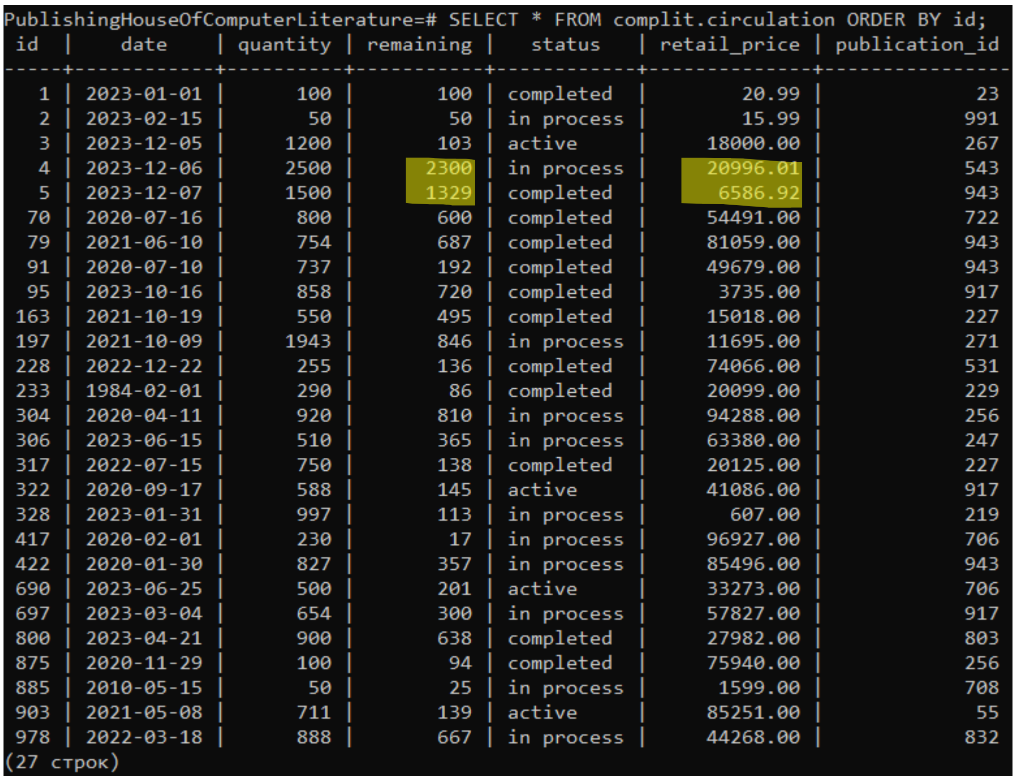
```
PublishingHouseOfComputerLiterature=# SELECT * FROM complit.circulation ORDER BY id;
 id  |    date    | quantity | remaining |   status    | retail_price | publication_id
-----+------------+----------+-----------+-------------+--------------+----------------
   1 | 2023-01-01 |      100 |       100 | completed   |        20.99 |             23
   2 | 2023-02-15 |       50 |        50 | in process  |        15.99 |            991
   3 | 2023-12-05 |     1200 |       103 | active      |     18000.00 |            267
   4 | 2023-12-06 |     2500 |      2300 | in process  |     23328.90 |            543
   5 | 2023-12-07 |     1500 |      1329 | completed   |      7318.80 |            943
  70 | 2020-07-16 |      800 |       600 | completed   |     54491.00 |            722
  79 | 2021-06-10 |      754 |       687 | completed   |     81059.00 |            943
  91 | 2020-07-10 |      737 |       192 | completed   |     49679.00 |            943
  95 | 2023-10-16 |      858 |       720 | completed   |      3735.00 |            917
 163 | 2021-10-19 |      550 |       495 | completed   |     15018.00 |            227
 197 | 2021-10-09 |     1943 |       846 | in process  |     11695.00 |            271
 228 | 2022-12-22 |      255 |       136 | completed   |     74066.00 |            531
 233 | 1984-02-01 |      290 |        86 | completed   |     20099.00 |            229
 304 | 2020-04-11 |      920 |       810 | in process  |     94288.00 |            256
 306 | 2023-06-15 |      510 |       365 | in process  |     63380.00 |            247
 317 | 2022-07-15 |      750 |       138 | completed   |     20125.00 |            227
 322 | 2020-09-17 |      588 |       145 | active      |     41086.00 |            917
 328 | 2023-01-31 |      997 |       113 | in process  |       607.00 |            219
 417 | 2020-02-01 |      230 |        17 | in process  |     96927.00 |            706
 422 | 2020-01-30 |      827 |       357 | in process  |     85496.00 |            943
 690 | 2023-06-25 |      500 |       201 | active      |     33273.00 |            706
 697 | 2023-03-04 |      654 |       300 | in process  |     57827.00 |            917
 800 | 2023-04-21 |      900 |       638 | completed   |     27982.00 |            803
 875 | 2020-11-29 |      100 |        94 | completed   |     75940.00 |            256
 885 | 2010-05-15 |       50 |        25 | in process  |      1599.00 |            708
 903 | 2021-05-08 |      711 |       139 | active      |     85251.00 |             55
 978 | 2022-03-18 |      888 |       667 | in process  |     44268.00 |            832
(27 строк)
```

Рисунок 2. Таблица circulation до выполнения процедуры books_discount()



```
PublishingHouseOfComputerLiterature=# call books_discount();
CALL
```

Рисунок 3. Выполнение процедуры books_discount()



```
PublishingHouseOfComputerLiterature=# SELECT * FROM complit.circulation ORDER BY id;
 id  |    date    | quantity | remaining |   status    | retail_price | publication_id
-----+------------+----------+-----------+-------------+--------------+----------------
   1 | 2023-01-01 |      100 |       100 | completed   |        20.99 |             23
   2 | 2023-02-15 |       50 |        50 | in process  |        15.99 |            991
   3 | 2023-12-05 |     1200 |       103 | active      |     18000.00 |            267
   4 | 2023-12-06 |     2500 |      2300 | in process  |     20996.01 |            543
   5 | 2023-12-07 |     1500 |      1329 | completed   |      6586.92 |            943
  70 | 2020-07-16 |      800 |       600 | completed   |     54491.00 |            722
  79 | 2021-06-10 |      754 |       687 | completed   |     81059.00 |            943
  91 | 2020-07-10 |      737 |       192 | completed   |     49679.00 |            943
  95 | 2023-10-16 |      858 |       720 | completed   |      3735.00 |            917
 163 | 2021-10-19 |      550 |       495 | completed   |     15018.00 |            227
 197 | 2021-10-09 |     1943 |       846 | in process  |     11695.00 |            271
 228 | 2022-12-22 |      255 |       136 | completed   |     74066.00 |            531
 233 | 1984-02-01 |      290 |        86 | completed   |     20099.00 |            229
 304 | 2020-04-11 |      920 |       810 | in process  |     94288.00 |            256
 306 | 2023-06-15 |      510 |       365 | in process  |     63380.00 |            247
 317 | 2022-07-15 |      750 |       138 | completed   |     20125.00 |            227
 322 | 2020-09-17 |      588 |       145 | active      |     41086.00 |            917
 328 | 2023-01-31 |      997 |       113 | in process  |       607.00 |            219
 417 | 2020-02-01 |      230 |        17 | in process  |     96927.00 |            706
 422 | 2020-01-30 |      827 |       357 | in process  |     85496.00 |            943
 690 | 2023-06-25 |      500 |       201 | active      |     33273.00 |            706
 697 | 2023-03-04 |      654 |       300 | in process  |     57827.00 |            917
 800 | 2023-04-21 |      900 |       638 | completed   |     27982.00 |            803
 875 | 2020-11-29 |      100 |        94 | completed   |     75940.00 |            256
 885 | 2010-05-15 |       50 |        25 | in process  |      1599.00 |            708
 903 | 2021-05-08 |      711 |       139 | active      |     85251.00 |             55
 978 | 2022-03-18 |      888 |       667 | in process  |     44268.00 |            832
(27 строк)
```

Рисунок 4. Таблица circulation псоле выполнения процедуры books_discount()

2) Для ввода новой книги.

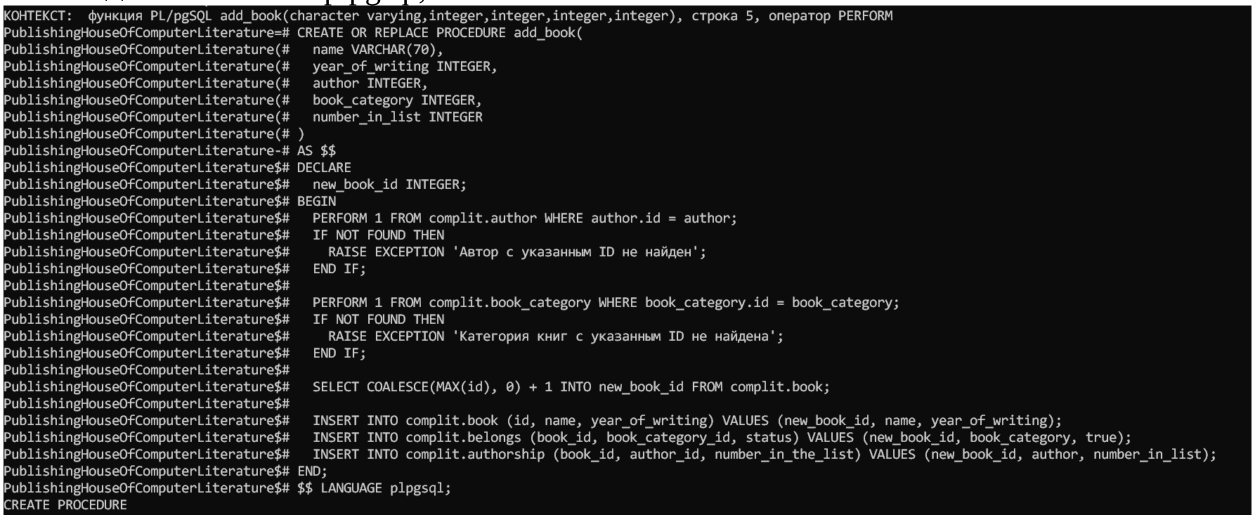
```
CREATE OR REPLACE PROCEDURE add_book(
  name VARCHAR(70),
  year_of_writing INTEGER,
  author_id INTEGER,
  book_category_id INTEGER,
  number_in_list INTEGER
)
AS $$
DECLARE
  new_book_id INTEGER;
BEGIN
  PERFORM 1 FROM complit.author WHERE author.id = author_id;
  IF NOT FOUND THEN
    RAISE EXCEPTION 'Автор с указанным ID не найден';
  END IF;

  PERFORM 1 FROM complit.book_category WHERE book_category.id =
book_category_id;
  IF NOT FOUND THEN
    RAISE EXCEPTION 'Категория книг с указанным ID не найдена';
  END IF;

  SELECT COALESCE(MAX(id), 0) + 1 INTO new_book_id FROM complit.book;

  INSERT INTO complit.book (id, name, year_of_writing) VALUES (new_book_id,
name, year_of_writing);
```

INSERT INTO complit.belongs (book_id, book_category_id, status) VALUES (new_book_id, book_category, true);

INSERT INTO complit.authorship (book_id, author_id, number_in_the_list) VALUES (new_book_id, author, number_in_list);

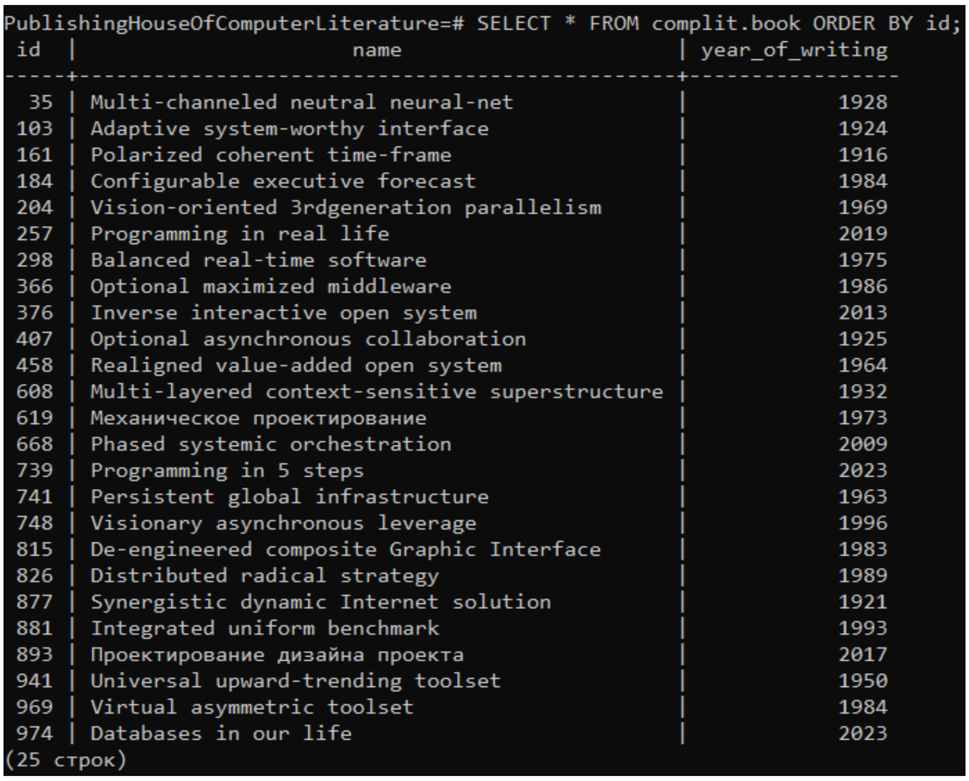
END;

$$ LANGUAGE plpgsql;



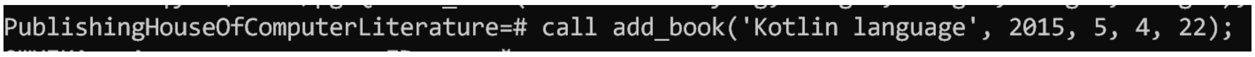
```
КОНТЕКСТ:  функция PL/pgSQL add_book(character varying,integer,integer,integer,integer), строка 5, оператор PERFORM
PublishingHouseOfComputerLiterature=# CREATE OR REPLACE PROCEDURE add_book(
PublishingHouseOfComputerLiterature(#   name VARCHAR(70),
PublishingHouseOfComputerLiterature(#   year_of_writing INTEGER,
PublishingHouseOfComputerLiterature(#   author INTEGER,
PublishingHouseOfComputerLiterature(#   book_category INTEGER,
PublishingHouseOfComputerLiterature(#   number_in_list INTEGER
PublishingHouseOfComputerLiterature(# )
PublishingHouseOfComputerLiterature-# AS $$
PublishingHouseOfComputerLiterature$# DECLARE
PublishingHouseOfComputerLiterature$#   new_book_id INTEGER;
PublishingHouseOfComputerLiterature$# BEGIN
PublishingHouseOfComputerLiterature$#   PERFORM 1 FROM complit.author WHERE author.id = author;
PublishingHouseOfComputerLiterature$#   IF NOT FOUND THEN
PublishingHouseOfComputerLiterature$#     RAISE EXCEPTION 'Автор с указанным ID не найден';
PublishingHouseOfComputerLiterature$#   END IF;
PublishingHouseOfComputerLiterature$#
PublishingHouseOfComputerLiterature$#   PERFORM 1 FROM complit.book_category WHERE book_category.id = book_category;
PublishingHouseOfComputerLiterature$#   IF NOT FOUND THEN
PublishingHouseOfComputerLiterature$#     RAISE EXCEPTION 'Категория книг с указанным ID не найдена';
PublishingHouseOfComputerLiterature$#   END IF;
PublishingHouseOfComputerLiterature$#
PublishingHouseOfComputerLiterature$#   SELECT COALESCE(MAX(id), 0) + 1 INTO new_book_id FROM complit.book;
PublishingHouseOfComputerLiterature$#
PublishingHouseOfComputerLiterature$#   INSERT INTO complit.book (id, name, year_of_writing) VALUES (new_book_id, name, year_of_writing);
PublishingHouseOfComputerLiterature$#   INSERT INTO complit.belongs (book_id, book_category_id, status) VALUES (new_book_id, book_category, true);
PublishingHouseOfComputerLiterature$#   INSERT INTO complit.authorship (book_id, author_id, number_in_the_list) VALUES (new_book_id, author, number_in_list);
PublishingHouseOfComputerLiterature$# END;
PublishingHouseOfComputerLiterature$# $$ LANGUAGE plpgsql;
CREATE PROCEDURE
```

Рисунок 5. Создание процедуры add_book()



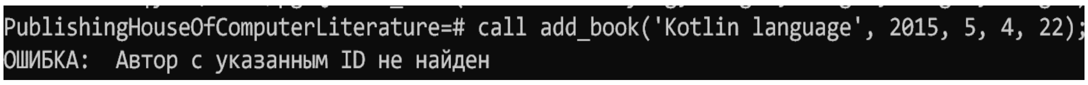
```
PublishingHouseOfComputerLiterature=# SELECT * FROM complit.book ORDER BY id;
 id  |                     name                      | year_of_writing
-----+-----------------------------------------------+-----------------
  35 | Multi-channeled neutral neural-net            |            1928
 103 | Adaptive system-worthy interface              |            1924
 161 | Polarized coherent time-frame                 |            1916
 184 | Configurable executive forecast               |            1984
 204 | Vision-oriented 3rdgeneration parallelism     |            1969
 257 | Programming in real life                      |            2019
 298 | Balanced real-time software                   |            1975
 366 | Optional maximized middleware                 |            1986
 376 | Inverse interactive open system               |            2013
 407 | Optional asynchronous collaboration           |            1925
 458 | Realigned value-added open system             |            1964
 608 | Multi-layered context-sensitive superstructure |            1932
 619 | Механическое проектирование                   |            1973
 668 | Phased systemic orchestration                 |            2009
 739 | Programming in 5 steps                        |            2023
 741 | Persistent global infrastructure              |            1963
 748 | Visionary asynchronous leverage               |            1996
 815 | De-engineered composite Graphic Interface     |            1983
 826 | Distributed radical strategy                  |            1989
 877 | Synergistic dynamic Internet solution         |            1921
 881 | Integrated uniform benchmark                  |            1993
 893 | Проектирование дизайна проекта                |            2017
 941 | Universal upward-trending toolset             |            1950
 969 | Virtual asymmetric toolset                    |            1984
 974 | Databases in our life                         |            2023
(25 строк)
```

Рисунок 6. Таблица book до выполнения процедуры add_book()



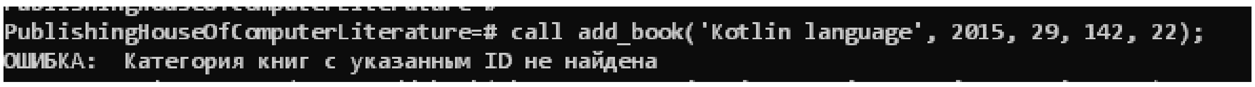
```
PublishingHouseOfComputerLiterature=# call add_book('Kotlin language', 2015, 5, 4, 22);
```

Рисунок 7. Выполнение процедуры add_book()



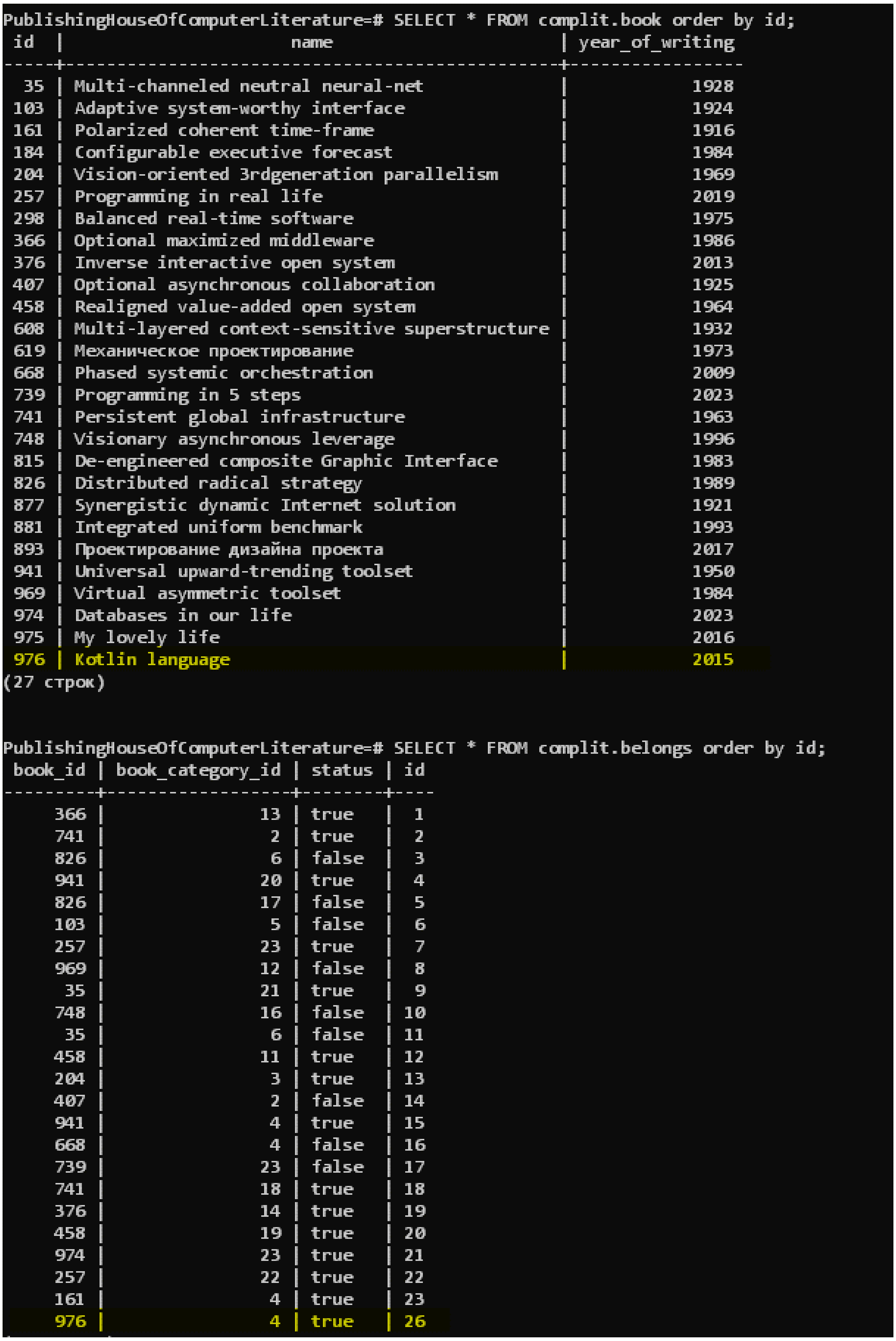
```
PublishingHouseOfComputerLiterature=# call add_book('Kotlin language', 2015, 5, 4, 22);
ОШИБКА:  Автор с указанным ID не найден
```

Рисунок 8. Вывод ошибки, если автора с указанным ID нет в таблице.



```
PublishingHouseOfComputerLiterature=# call add_book('Kotlin language', 2015, 29, 142, 22);
ОШИБКА:  Категория книг с указанным ID не найдена
```

Рисунок 9. Вывод ошибки, если категории книг с указанным ID нет в таблице.

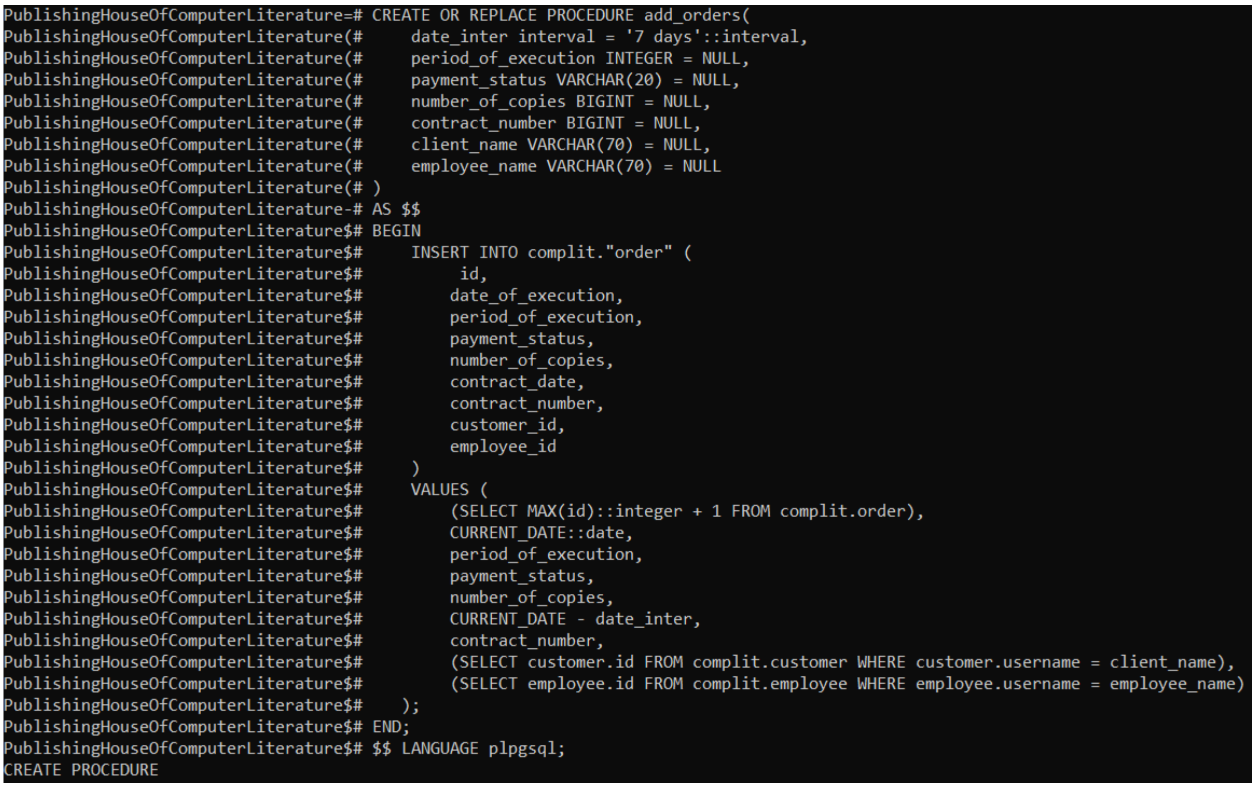


```
PublishingHouseOfComputerLiterature=# SELECT * FROM complit.book order by id;
 id  |                      name                       | year_of_writing
-----+-------------------------------------------------+-----------------
  35 | Multi-channeled neutral neural-net              |            1928
 103 | Adaptive system-worthy interface                |            1924
 161 | Polarized coherent time-frame                   |            1916
 184 | Configurable executive forecast                 |            1984
 204 | Vision-oriented 3rdgeneration parallelism       |            1969
 257 | Programming in real life                        |            2019
 298 | Balanced real-time software                     |            1975
 366 | Optional maximized middleware                   |            1986
 376 | Inverse interactive open system                 |            2013
 407 | Optional asynchronous collaboration             |            1925
 458 | Realigned value-added open system               |            1964
 608 | Multi-layered context-sensitive superstructure  |            1932
 619 | Механическое проектирование                     |            1973
 668 | Phased systemic orchestration                   |            2009
 739 | Programming in 5 steps                          |            2023
 741 | Persistent global infrastructure                |            1963
 748 | Visionary asynchronous leverage                 |            1996
 815 | De-engineered composite Graphic Interface       |            1983
 826 | Distributed radical strategy                    |            1989
 877 | Synergistic dynamic Internet solution           |            1921
 881 | Integrated uniform benchmark                    |            1993
 893 | Проектирование дизайна проекта                  |            2017
 941 | Universal upward-trending toolset               |            1950
 969 | Virtual asymmetric toolset                      |            1984
 974 | Databases in our life                           |            2023
 975 | My lovely life                                  |            2016
 976 | Kotlin language                                 |            2015
(27 строк)


PublishingHouseOfComputerLiterature=# SELECT * FROM complit.belongs order by id;
 book_id | book_category_id | status | id
---------+------------------+--------+----
     366 |               13 | true   |  1
     741 |                2 | true   |  2
     826 |                6 | false  |  3
     941 |               20 | true   |  4
     826 |               17 | false  |  5
     103 |                5 | false  |  6
     257 |               23 | true   |  7
     969 |               12 | false  |  8
      35 |               21 | true   |  9
     748 |               16 | false  | 10
      35 |                6 | false  | 11
     458 |               11 | true   | 12
     204 |                3 | true   | 13
     407 |                2 | false  | 14
     941 |                4 | true   | 15
     668 |                4 | false  | 16
     739 |               23 | false  | 17
     741 |               18 | true   | 18
     376 |               14 | true   | 19
     458 |               19 | true   | 20
     974 |               23 | true   | 21
     257 |               22 | true   | 22
     161 |                4 | true   | 23
     976 |                4 | true   | 26
```

Рисунок 10. Результат выполнения процедуры add_book()

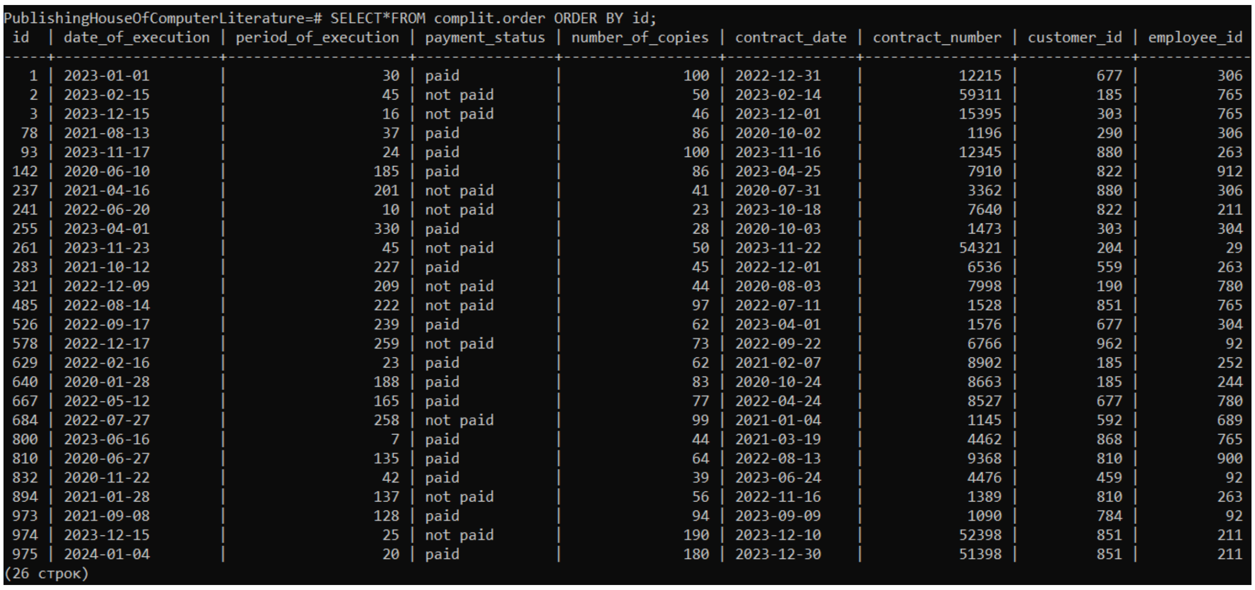
3) Для ввода нового заказа.

```
CREATE OR REPLACE PROCEDURE add_orders(
   date_inter interval = '7 days'::interval,
   period_of_execution INTEGER = NULL,
   payment_status VARCHAR(20) = NULL,
   number_of_copies BIGINT = NULL,
   contract_number BIGINT = NULL,
   client_name VARCHAR(70) = NULL,
   employee_name VARCHAR(70) = NULL
)
AS $$
BEGIN
   INSERT INTO complit."order" (
       id,
      date_of_execution,
      period_of_execution,
      payment_status,
      number_of_copies,
      contract_date,
      contract_number,
      customer_id,
      employee_id
   )
   VALUES (
      (SELECT MAX(id)::integer + 1 FROM complit.order),
      CURRENT_DATE::date,
      period_of_execution,

      payment_status,

      number_of_copies,

      CURRENT_DATE - date_inter,

      contract_number,

      (SELECT customer.id FROM complit.customer WHERE customer.username =
client_name),

      (SELECT employee.id FROM complit.employee WHERE employee.username =
employee_name)

   );

END;

$$ LANGUAGE plpgsql;
```



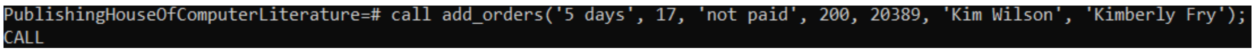
```
PublishingHouseOfComputerLiterature=# CREATE OR REPLACE PROCEDURE add_orders(
PublishingHouseOfComputerLiterature(#     date_inter interval = '7 days'::interval,
PublishingHouseOfComputerLiterature(#     period_of_execution INTEGER = NULL,
PublishingHouseOfComputerLiterature(#     payment_status VARCHAR(20) = NULL,
PublishingHouseOfComputerLiterature(#     number_of_copies BIGINT = NULL,
PublishingHouseOfComputerLiterature(#     contract_number BIGINT = NULL,
PublishingHouseOfComputerLiterature(#     client_name VARCHAR(70) = NULL,
PublishingHouseOfComputerLiterature(#     employee_name VARCHAR(70) = NULL
PublishingHouseOfComputerLiterature(# )
PublishingHouseOfComputerLiterature-# AS $$
PublishingHouseOfComputerLiterature$# BEGIN
PublishingHouseOfComputerLiterature$#     INSERT INTO complit."order" (
PublishingHouseOfComputerLiterature$#          id,
PublishingHouseOfComputerLiterature$#         date_of_execution,
PublishingHouseOfComputerLiterature$#         period_of_execution,
PublishingHouseOfComputerLiterature$#         payment_status,
PublishingHouseOfComputerLiterature$#         number_of_copies,
PublishingHouseOfComputerLiterature$#         contract_date,
PublishingHouseOfComputerLiterature$#         contract_number,
PublishingHouseOfComputerLiterature$#         customer_id,
PublishingHouseOfComputerLiterature$#         employee_id
PublishingHouseOfComputerLiterature$#     )
PublishingHouseOfComputerLiterature$#     VALUES (
PublishingHouseOfComputerLiterature$#         (SELECT MAX(id)::integer + 1 FROM complit.order),
PublishingHouseOfComputerLiterature$#         CURRENT_DATE::date,
PublishingHouseOfComputerLiterature$#         period_of_execution,
PublishingHouseOfComputerLiterature$#         payment_status,
PublishingHouseOfComputerLiterature$#         number_of_copies,
PublishingHouseOfComputerLiterature$#         CURRENT_DATE - date_inter,
PublishingHouseOfComputerLiterature$#         contract_number,
PublishingHouseOfComputerLiterature$#         (SELECT customer.id FROM complit.customer WHERE customer.username = client_name),
PublishingHouseOfComputerLiterature$#         (SELECT employee.id FROM complit.employee WHERE employee.username = employee_name)
PublishingHouseOfComputerLiterature$#    );
PublishingHouseOfComputerLiterature$# END;
PublishingHouseOfComputerLiterature$# $$ LANGUAGE plpgsql;
CREATE PROCEDURE
```

Рисунок 11. Создание процедуры add_orders()



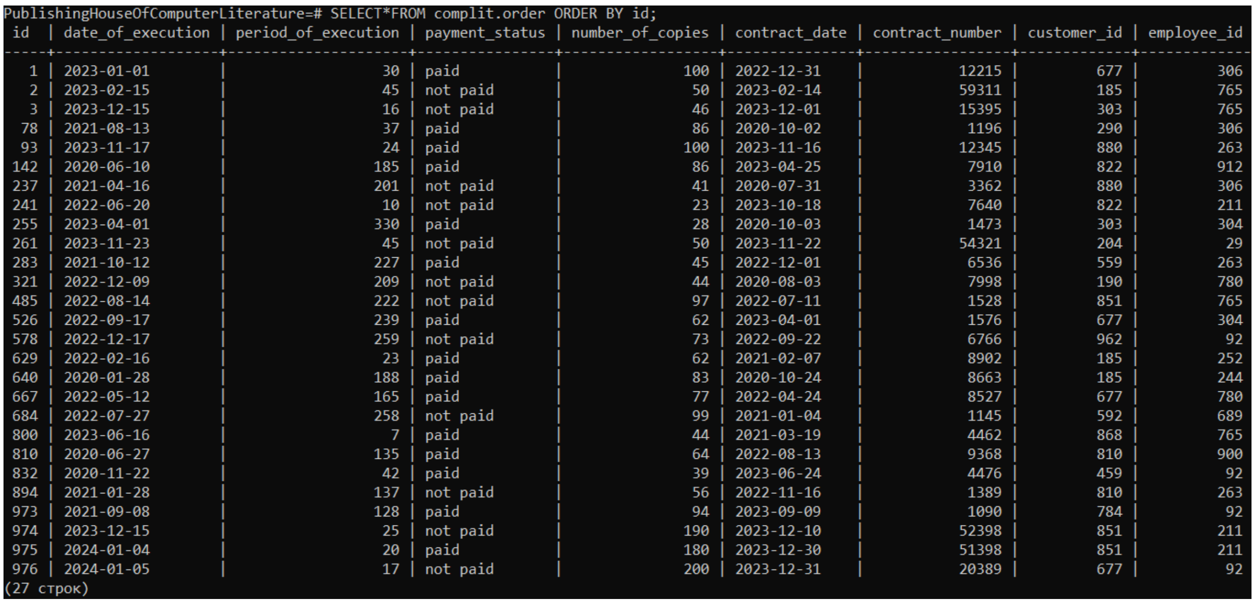
```
PublishingHouseOfComputerLiterature=# SELECT*FROM complit.order ORDER BY id;
 id  | date_of_execution | period_of_execution | payment_status | number_of_copies | contract_date | contract_number | customer_id | employee_id
-----+-------------------+---------------------+----------------+------------------+---------------+-----------------+-------------+-------------
   1 | 2023-01-01        |                  30 | paid           |              100 | 2022-12-31    |           12215 |         677 |         306
   2 | 2023-02-15        |                  45 | not paid       |               50 | 2023-02-14    |           59311 |         185 |         765
   3 | 2023-12-15        |                  16 | not paid       |               46 | 2023-12-01    |           15395 |         303 |         765
  78 | 2021-08-13        |                  37 | paid           |               86 | 2020-10-02    |            1196 |         290 |         306
  93 | 2023-11-17        |                  24 | paid           |              100 | 2023-11-16    |           12345 |         880 |         263
 142 | 2020-06-10        |                 185 | paid           |               86 | 2023-04-25    |            7910 |         822 |         912
 237 | 2021-04-16        |                 201 | not paid       |               41 | 2020-07-31    |            3362 |         880 |         306
 241 | 2022-06-20        |                  10 | not paid       |               23 | 2023-10-18    |            7640 |         822 |         211
 255 | 2023-04-01        |                 330 | paid           |               28 | 2020-10-03    |            1473 |         303 |         304
 261 | 2023-11-23        |                  45 | not paid       |               50 | 2023-11-22    |           54321 |         204 |          29
 283 | 2021-10-12        |                 227 | paid           |               45 | 2022-12-01    |            6536 |         559 |         263
 321 | 2022-12-09        |                 209 | not paid       |               44 | 2020-08-03    |            7998 |         190 |         780
 485 | 2022-08-14        |                 222 | not paid       |               97 | 2022-07-11    |            1528 |         851 |         765
 526 | 2022-09-17        |                 239 | paid           |               62 | 2023-04-01    |            1576 |         677 |         304
 578 | 2022-12-17        |                 259 | not paid       |               73 | 2022-09-22    |            6766 |         962 |          92
 629 | 2022-02-16        |                  23 | paid           |               62 | 2021-02-07    |            8902 |         185 |         252
 640 | 2020-01-28        |                 188 | paid           |               83 | 2020-10-24    |            8663 |         185 |         244
 667 | 2022-05-12        |                 165 | paid           |               77 | 2022-04-24    |            8527 |         677 |         780
 684 | 2022-07-27        |                 258 | not paid       |               99 | 2021-01-04    |            1145 |         592 |         689
 800 | 2023-06-16        |                   7 | paid           |               44 | 2021-03-19    |            4462 |         868 |         765
 810 | 2020-06-27        |                 135 | paid           |               64 | 2022-08-13    |            9368 |         810 |         900
 832 | 2020-11-22        |                  42 | paid           |               39 | 2023-06-24    |            4476 |         459 |          92
 894 | 2021-01-28        |                 137 | not paid       |               56 | 2022-11-16    |            1389 |         810 |         263
 973 | 2021-09-08        |                 128 | paid           |               94 | 2023-09-09    |            1090 |         784 |          92
 974 | 2023-12-15        |                  25 | not paid       |              190 | 2023-12-10    |           52398 |         851 |         211
 975 | 2024-01-04        |                  20 | paid           |              180 | 2023-12-30    |           51398 |         851 |         211
(26 строк)
```

Рисунок 12. Таблица order до выполнения процедуры add_orders()



```
PublishingHouseOfComputerLiterature=# call add_orders('5 days', 17, 'not paid', 200, 20389, 'Kim Wilson', 'Kimberly Fry');
CALL
```

Рисунок 13. Выполнение процедуры add_orders()

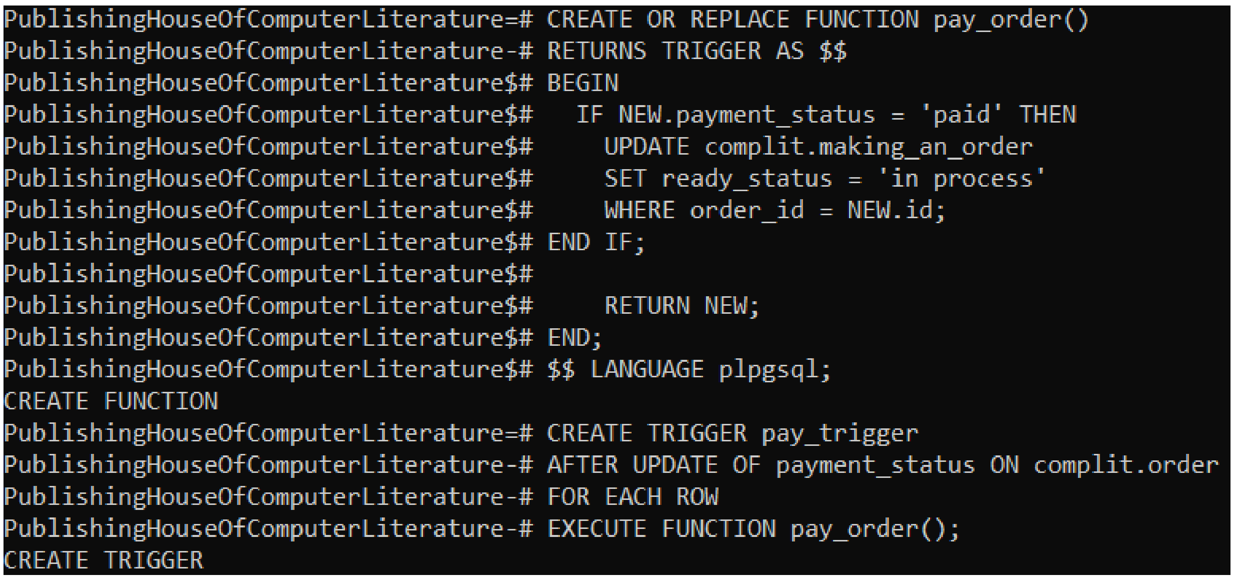


```
PublishingHouseOfComputerLiterature=# SELECT*FROM complit.order ORDER BY id;
 id  | date_of_execution | period_of_execution | payment_status | number_of_copies | contract_date | contract_number | customer_id | employee_id
-----+-------------------+---------------------+----------------+------------------+---------------+-----------------+-------------+-------------
   1 | 2023-01-01        |                  30 | paid           |              100 | 2022-12-31    |           12215 |         677 |         306
   2 | 2023-02-15        |                  45 | not paid       |               50 | 2023-02-14    |           59311 |         185 |         765
   3 | 2023-12-15        |                  16 | not paid       |               46 | 2023-12-01    |           15395 |         303 |         765
  78 | 2021-08-13        |                  37 | paid           |               86 | 2020-10-02    |            1196 |         290 |         306
  93 | 2023-11-17        |                  24 | paid           |              100 | 2023-11-16    |           12345 |         880 |         263
 142 | 2020-06-10        |                 185 | paid           |               86 | 2023-04-25    |            7910 |         822 |         912
 237 | 2021-04-16        |                 201 | not paid       |               41 | 2020-07-31    |            3362 |         880 |         306
 241 | 2022-06-20        |                  10 | not paid       |               23 | 2023-10-18    |            7640 |         822 |         211
 255 | 2023-04-01        |                 330 | paid           |               28 | 2020-10-03    |            1473 |         303 |         304
 261 | 2023-11-23        |                  45 | not paid       |               50 | 2023-11-22    |           54321 |         204 |          29
 283 | 2021-10-12        |                 227 | paid           |               45 | 2022-12-01    |            6536 |         559 |         263
 321 | 2022-12-09        |                 209 | not paid       |               44 | 2020-08-03    |            7998 |         190 |         780
 485 | 2022-08-14        |                 222 | not paid       |               97 | 2022-07-11    |            1528 |         851 |         765
 526 | 2022-09-17        |                 239 | paid           |               62 | 2023-04-01    |            1576 |         677 |         304
 578 | 2022-12-17        |                 259 | not paid       |               73 | 2022-09-22    |            6766 |         962 |          92
 629 | 2022-02-16        |                  23 | paid           |               62 | 2021-02-07    |            8902 |         185 |         252
 640 | 2020-01-28        |                 188 | paid           |               83 | 2020-10-24    |            8663 |         185 |         244
 667 | 2022-05-12        |                 165 | paid           |               77 | 2022-04-24    |            8527 |         677 |         780
 684 | 2022-07-27        |                 258 | not paid       |               99 | 2021-01-04    |            1145 |         592 |         689
 800 | 2023-06-16        |                   7 | paid           |               44 | 2021-03-19    |            4462 |         868 |         765
 810 | 2020-06-27        |                 135 | paid           |               64 | 2022-08-13    |            9368 |         810 |         900
 832 | 2020-11-22        |                  42 | paid           |               39 | 2023-06-24    |            4476 |         459 |          92
 894 | 2021-01-28        |                 137 | not paid       |               56 | 2022-11-16    |            1389 |         810 |         263
 973 | 2021-09-08        |                 128 | paid           |               94 | 2023-09-09    |            1090 |         784 |          92
 974 | 2023-12-15        |                  25 | not paid       |              190 | 2023-12-10    |           52398 |         851 |         211
 975 | 2024-01-04        |                  20 | paid           |              180 | 2023-12-30    |           51398 |         851 |         211
 976 | 2024-01-05        |                  17 | not paid       |              200 | 2023-12-31    |           20389 |         677 |          92
(27 строк)
```

Рисунок 14. Таблица order после выполнения процедуры add_orders()

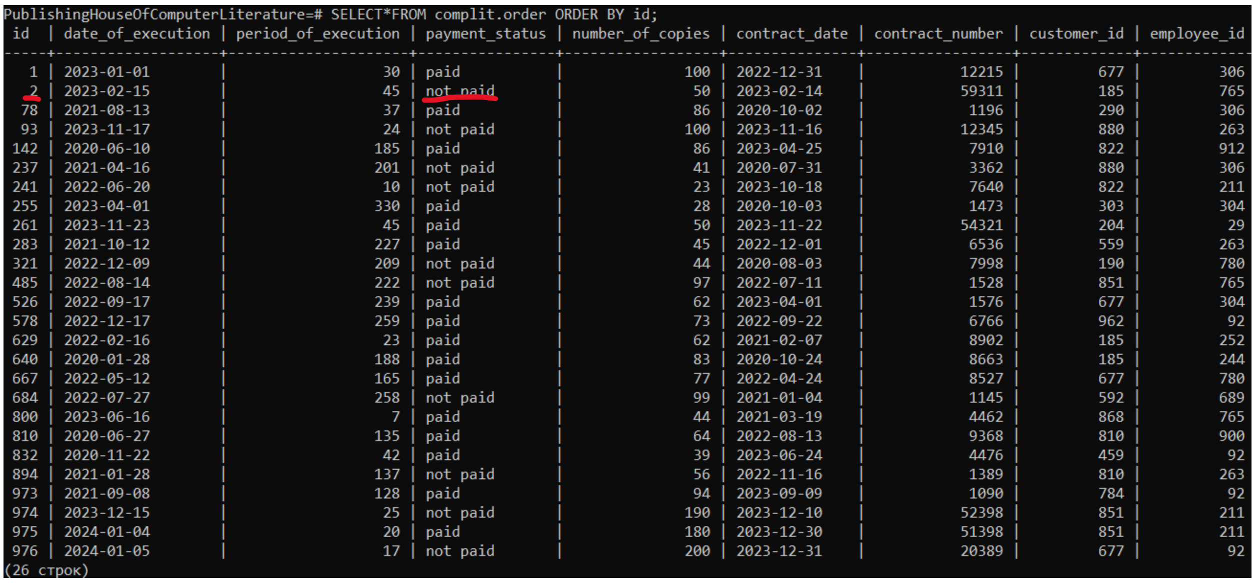
- Создание триггера, который после обновления статуса оплаты заказа на “Оплачено”, изменяет статус заказа на “in progress”

```
CREATE OR REPLACE FUNCTION pay_order()

RETURNS TRIGGER AS $$

BEGIN

  IF NEW.payment_status = 'paid' THEN

    UPDATE complit.making_an_order

    SET ready_status = 'in process'

    WHERE order_id = NEW.id;

END IF;



    RETURN NEW;

END;

$$ LANGUAGE plpgsql;



CREATE TRIGGER pay_trigger

AFTER UPDATE OF payment_status ON complit.order

FOR EACH ROW

EXECUTE FUNCTION pay_order();
```



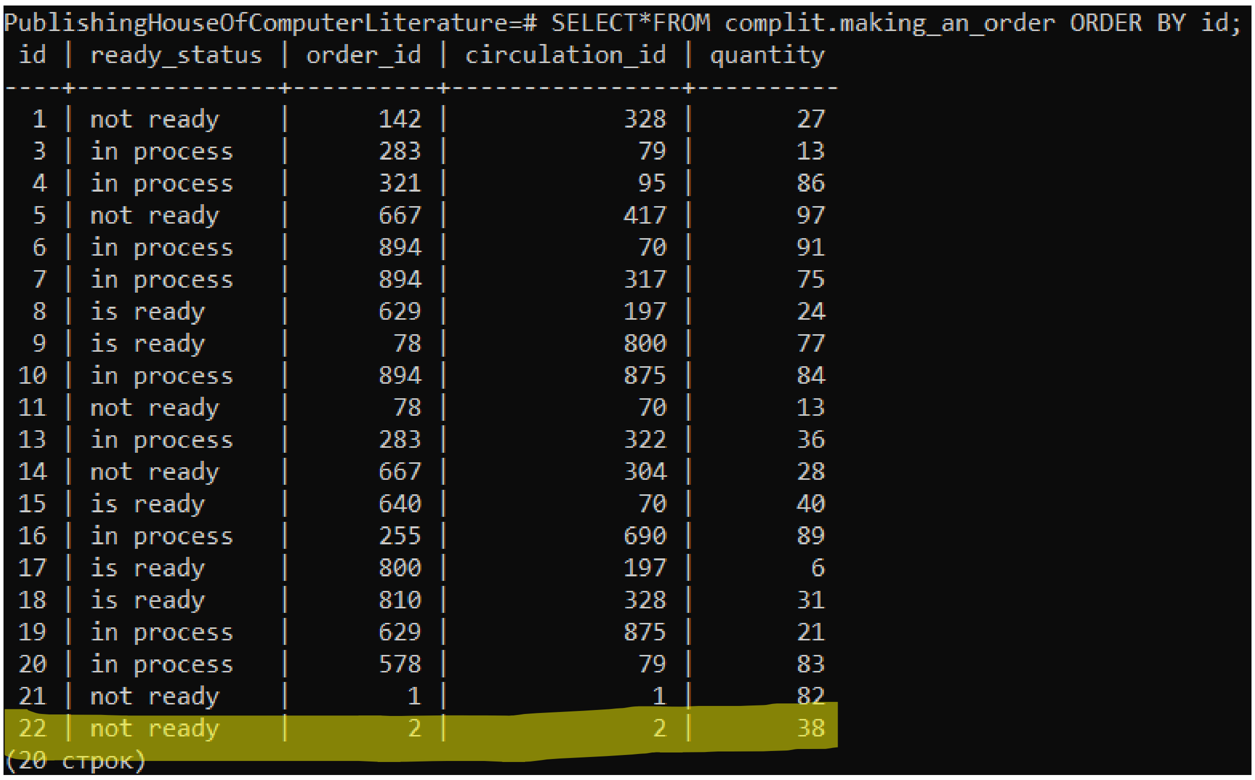
```
PublishingHouseOfComputerLiterature=# CREATE OR REPLACE FUNCTION pay_order()
PublishingHouseOfComputerLiterature-# RETURNS TRIGGER AS $$
PublishingHouseOfComputerLiterature$# BEGIN
PublishingHouseOfComputerLiterature$#   IF NEW.payment_status = 'paid' THEN
PublishingHouseOfComputerLiterature$#     UPDATE complit.making_an_order
PublishingHouseOfComputerLiterature$#     SET ready_status = 'in process'
PublishingHouseOfComputerLiterature$#     WHERE order_id = NEW.id;
PublishingHouseOfComputerLiterature$# END IF;
PublishingHouseOfComputerLiterature$#
PublishingHouseOfComputerLiterature$#     RETURN NEW;
PublishingHouseOfComputerLiterature$# END;
PublishingHouseOfComputerLiterature$# $$ LANGUAGE plpgsql;
CREATE FUNCTION
PublishingHouseOfComputerLiterature=# CREATE TRIGGER pay_trigger
PublishingHouseOfComputerLiterature-# AFTER UPDATE OF payment_status ON complit.order
PublishingHouseOfComputerLiterature-# FOR EACH ROW
PublishingHouseOfComputerLiterature-# EXECUTE FUNCTION pay_order();
CREATE TRIGGER
```

Рисунок 15. Создание функции pay_order() и триггера pay_trigger



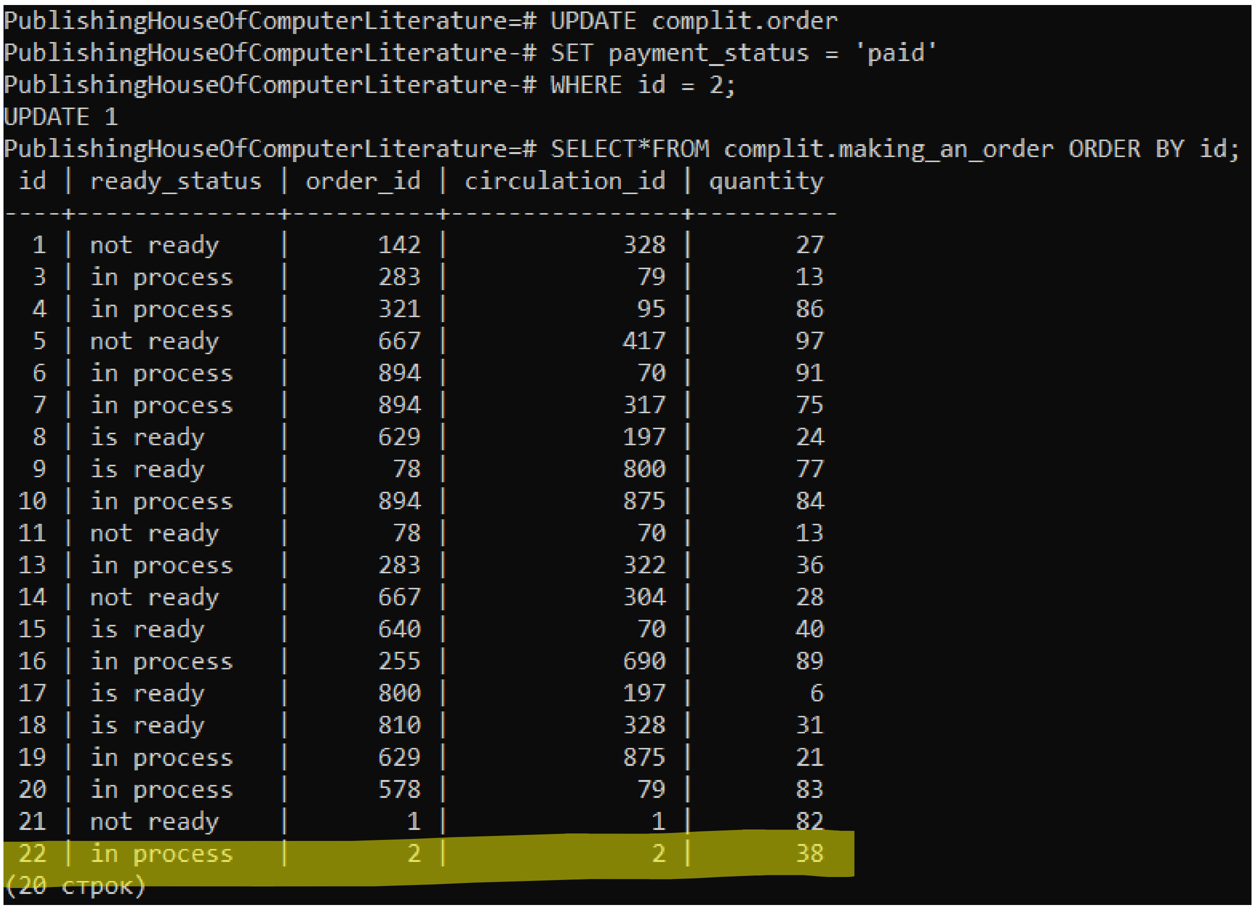
```
PublishingHouseOfComputerLiterature=# SELECT*FROM complit.order ORDER BY id;
 id  | date_of_execution | period_of_execution | payment_status | number_of_copies | contract_date | contract_number | customer_id | employee_id
-----+-------------------+---------------------+----------------+------------------+---------------+-----------------+-------------+------------
   1 | 2023-01-01        |                  30 | paid           |              100 | 2022-12-31    |           12215 |         677 |         306
   2 | 2023-02-15        |                  45 | not paid       |               50 | 2023-02-14    |           59311 |         185 |         765
  78 | 2021-08-13        |                  37 | paid           |               86 | 2020-10-02    |            1196 |         290 |         306
  93 | 2023-11-17        |                  24 | not paid       |              100 | 2023-11-16    |           12345 |         880 |         263
 142 | 2020-06-10        |                 185 | paid           |               86 | 2023-04-25    |            7910 |         822 |         912
 237 | 2021-04-16        |                 201 | not paid       |               41 | 2020-07-31    |            3362 |         880 |         306
 241 | 2022-06-20        |                  10 | not paid       |               23 | 2023-10-18    |            7640 |         822 |         211
 255 | 2023-04-01        |                 330 | paid           |               28 | 2020-10-03    |            1473 |         303 |         304
 261 | 2023-11-23        |                  45 | paid           |               50 | 2023-11-22    |           54321 |         204 |          29
 283 | 2021-10-12        |                 227 | paid           |               45 | 2022-12-01    |            6536 |         559 |         263
 321 | 2022-12-09        |                 209 | not paid       |               44 | 2020-08-03    |            7998 |         190 |         780
 485 | 2022-08-14        |                 222 | not paid       |               97 | 2022-07-11    |            1528 |         851 |         765
 526 | 2022-09-17        |                 239 | paid           |               62 | 2023-04-01    |            1576 |         677 |         304
 578 | 2022-12-17        |                 259 | paid           |               73 | 2022-09-22    |            6766 |         962 |          92
 629 | 2022-02-16        |                  23 | paid           |               62 | 2021-02-07    |            8902 |         185 |         252
 640 | 2020-01-28        |                 188 | paid           |               83 | 2020-10-24    |            8663 |         185 |         244
 667 | 2022-05-12        |                 165 | paid           |               77 | 2022-04-24    |            8527 |         677 |         780
 684 | 2022-07-27        |                 258 | not paid       |               99 | 2021-01-04    |            1145 |         592 |         689
 800 | 2023-06-16        |                   7 | paid           |               44 | 2021-03-19    |            4462 |         868 |         765
 810 | 2020-06-27        |                 135 | paid           |               64 | 2022-08-13    |            9368 |         810 |         900
 832 | 2020-11-22        |                  42 | paid           |               39 | 2023-06-24    |            4476 |         459 |          92
 894 | 2021-01-28        |                 137 | not paid       |               56 | 2022-11-16    |            1389 |         810 |         263
 973 | 2021-09-08        |                 128 | paid           |               94 | 2023-09-09    |            1090 |         784 |          92
 974 | 2023-12-15        |                  25 | not paid       |              190 | 2023-12-10    |           52398 |         851 |         211
 975 | 2024-01-04        |                  20 | paid           |              180 | 2023-12-30    |           51398 |         851 |         211
 976 | 2024-01-05        |                  17 | not paid       |              200 | 2023-12-31    |           20389 |         677 |          92
(26 строк)
```

Рисунок 16. Таблица order



```
PublishingHouseOfComputerLiterature=# SELECT*FROM complit.making_an_order ORDER BY id;
 id | ready_status | order_id | circulation_id | quantity
----+--------------+----------+----------------+----------
  1 | not ready    |      142 |            328 |       27
  3 | in process   |      283 |             79 |       13
  4 | in process   |      321 |             95 |       86
  5 | not ready    |      667 |            417 |       97
  6 | in process   |      894 |             70 |       91
  7 | in process   |      894 |            317 |       75
  8 | is ready     |      629 |            197 |       24
  9 | is ready     |       78 |            800 |       77
 10 | in process   |      894 |            875 |       84
 11 | not ready    |       78 |             70 |       13
 13 | in process   |      283 |            322 |       36
 14 | not ready    |      667 |            304 |       28
 15 | is ready     |      640 |             70 |       40
 16 | in process   |      255 |            690 |       89
 17 | is ready     |      800 |            197 |        6
 18 | is ready     |      810 |            328 |       31
 19 | in process   |      629 |            875 |       21
 20 | in process   |      578 |             79 |       83
 21 | not ready    |        1 |              1 |       82
 22 | not ready    |        2 |              2 |       38
(20 строк)
```

Рисунок 17. Таблица making_an_order до работы триггера



```
PublishingHouseOfComputerLiterature=# UPDATE complit.order
PublishingHouseOfComputerLiterature-# SET payment_status = 'paid'
PublishingHouseOfComputerLiterature-# WHERE id = 2;
UPDATE 1
PublishingHouseOfComputerLiterature=# SELECT*FROM complit.making_an_order ORDER BY id;
 id | ready_status | order_id | circulation_id | quantity
----+--------------+----------+----------------+----------
  1 | not ready    |      142 |            328 |       27
  3 | in process   |      283 |             79 |       13
  4 | in process   |      321 |             95 |       86
  5 | not ready    |      667 |            417 |       97
  6 | in process   |      894 |             70 |       91
  7 | in process   |      894 |            317 |       75
  8 | is ready     |      629 |            197 |       24
  9 | is ready     |       78 |            800 |       77
 10 | in process   |      894 |            875 |       84
 11 | not ready    |       78 |             70 |       13
 13 | in process   |      283 |            322 |       36
 14 | not ready    |      667 |            304 |       28
 15 | is ready     |      640 |             70 |       40
 16 | in process   |      255 |            690 |       89
 17 | is ready     |      800 |            197 |        6
 18 | is ready     |      810 |            328 |       31
 19 | in process   |      629 |            875 |       21
 20 | in process   |      578 |             79 |       83
 21 | not ready    |        1 |              1 |       82
 22 | in process   |        2 |              2 |       38
(20 строк)
```

Рисунок 18. Таблица making_an_order после работы триггера

- Модифицировать триггер на проверку корректности входа и выхода сотрудника (имеющиеся проблемы: может быть отрицательное время работы, человек зашел/вышел в будущем)

```
create or replace function fn_check_time_punch() returns trigger as $psql$ begin

if

new.is_out_punch = (select tps.is_out_punch from time_punch tps

      where tps.employee_id = new.employee_id order by tps.id desc limit 1 )

or

new.punch_time>now()

or

new.punch_time <= (select tps.punch_time from time_punch tps

where tps.employee_id = new.employee_id order by tps.id desc limit 1 )
```

then return null;

end if; return new;

end;

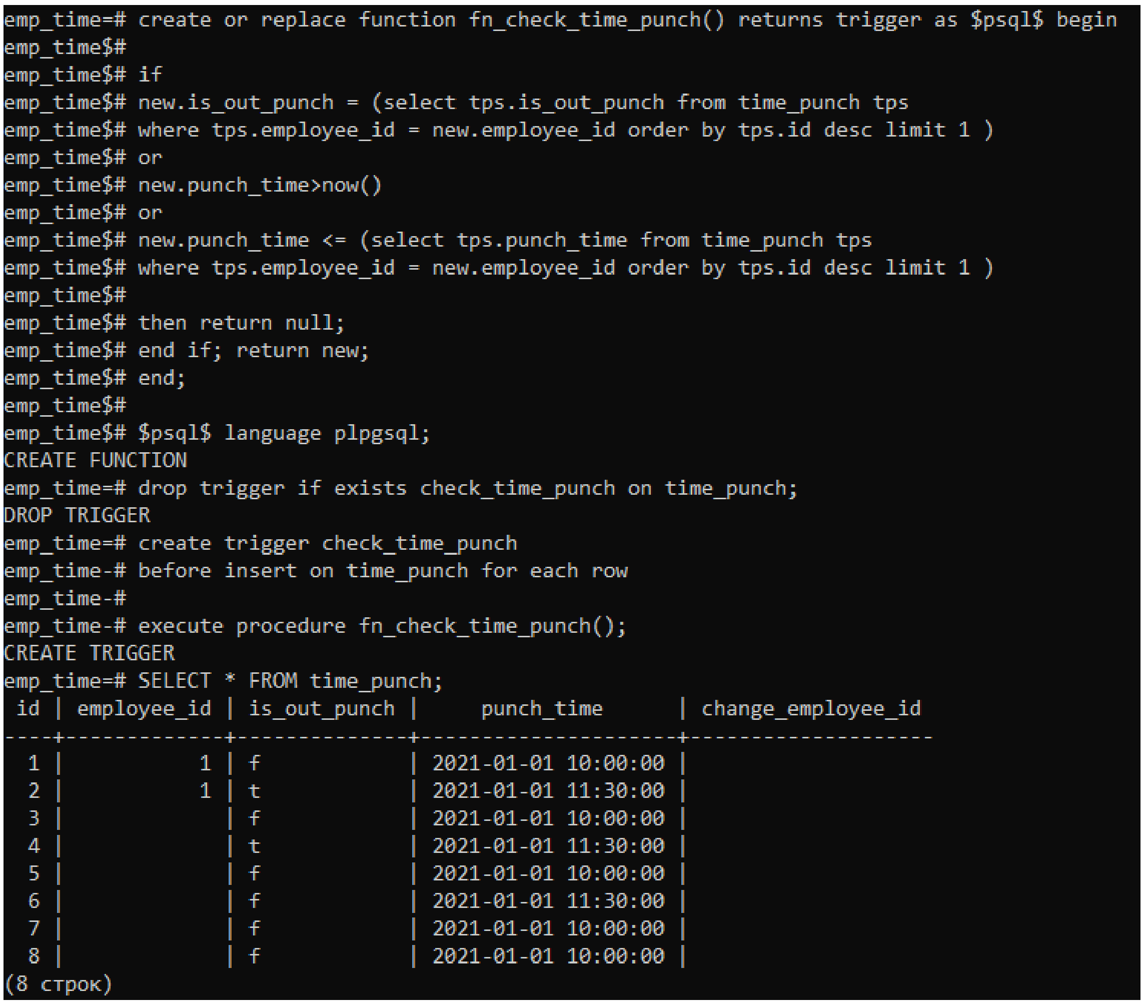
$psql$ language plpgsql;

drop trigger if exists check_time_punch on time_punch;

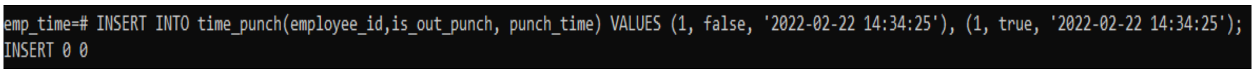
create trigger check_time_punch

before insert on time_punch for each row

execute procedure fn_check_time_punch();



```
emp_time=# create or replace function fn_check_time_punch() returns trigger as $psql$ begin
emp_time$#
emp_time$# if
emp_time$# new.is_out_punch = (select tps.is_out_punch from time_punch tps
emp_time$# where tps.employee_id = new.employee_id order by tps.id desc limit 1 )
emp_time$# or
emp_time$# new.punch_time>now()
emp_time$# or
emp_time$# new.punch_time <= (select tps.punch_time from time_punch tps
emp_time$# where tps.employee_id = new.employee_id order by tps.id desc limit 1 )
emp_time$#
emp_time$# then return null;
emp_time$# end if; return new;
emp_time$# end;
emp_time$#
emp_time$# $psql$ language plpgsql;
CREATE FUNCTION
emp_time=# drop trigger if exists check_time_punch on time_punch;
DROP TRIGGER
emp_time=# create trigger check_time_punch
emp_time-# before insert on time_punch for each row
emp_time-#
emp_time-# execute procedure fn_check_time_punch();
CREATE TRIGGER
emp_time=# SELECT * FROM time_punch;
 id | employee_id | is_out_punch |     punch_time      | change_employee_id
----+-------------+--------------+---------------------+--------------------
  1 |           1 | f            | 2021-01-01 10:00:00 |
  2 |           1 | t            | 2021-01-01 11:30:00 |
  3 |             | f            | 2021-01-01 10:00:00 |
  4 |             | t            | 2021-01-01 11:30:00 |
  5 |             | f            | 2021-01-01 10:00:00 |
  6 |             | f            | 2021-01-01 11:30:00 |
  7 |             | f            | 2021-01-01 10:00:00 |
  8 |             | f            | 2021-01-01 10:00:00 |
(8 строк)
```



```
emp_time=# INSERT INTO time_punch(employee_id,is_out_punch, punch_time) VALUES (1, false, '2022-02-22 14:34:25'), (1, true, '2022-02-22 14:34:25');
INSERT 0 0
```

Вывод

В ходе данной лабораторной работы я овладела практическими навыками создания и использования процедур, функций и триггеров в базе данных PostgreSQL. Взаимодействие с базой данных осуществлялось с помощью консольного клиента psql, который позволил мне научиться интерактивно вводить запросы, передавать их в PostgreSQL и видеть результаты.