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

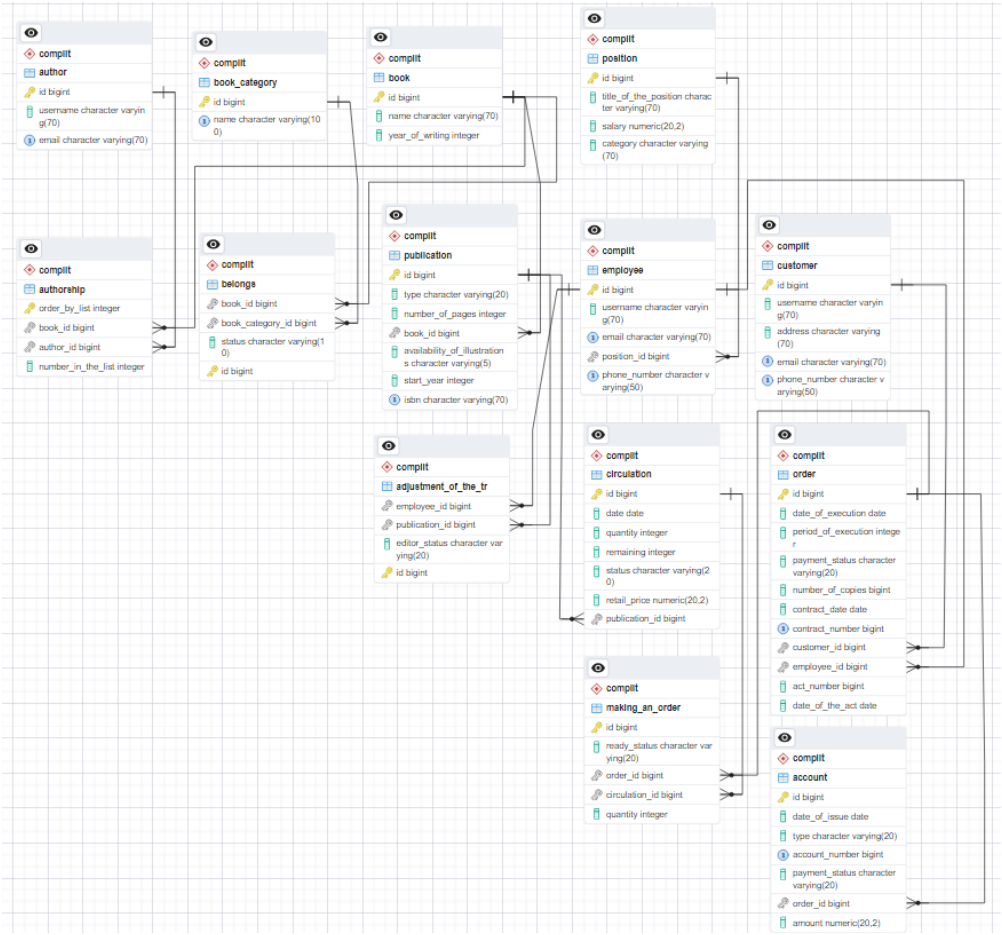
Вариант 2 (max - 8 баллов)

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
)
AS $$
BEGIN
    INSERT INTO complit.book (id, name, year_of_writing)
    VALUES (
        (SELECT MAX(id)::integer + 1 FROM complit.book),
        name,
        year_of_writing);
END;
$$ LANGUAGE plpgsql;
```

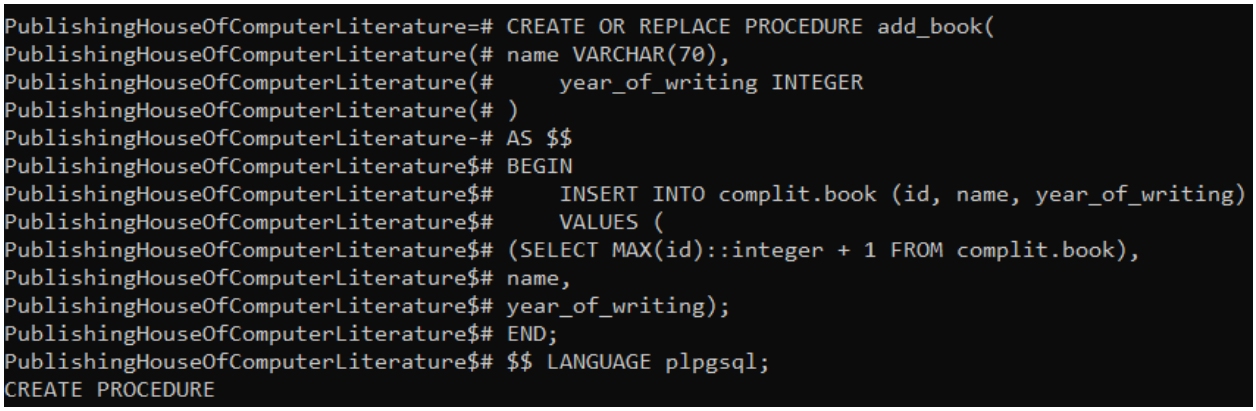


Рисунок 5. Создание процедуры add_book()

```
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('My lovely life', 2016);
CALL
```

Рисунок 7. Выполнение 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
 975 | My lovely life                                  |            2016
(26 строк)
```

Рисунок 8. Таблица book после выполнения процедуры add_book()

3) Для ввода нового заказа.

```
CREATE OR REPLACE PROCEDURE add_order(
    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,
    act_number BIGINT = 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,
        act_number,
        date_of_the_act
    )
```

```
    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),
        act_number,
        CURRENT_DATE + date_inter
    );
END;
$$ LANGUAGE plpgsql;
```

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

Рисунок 10. Создание процедуры add_order()

```
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 | act_number | date_of_the_act
-----+-------------------+---------------------+----------------+------------------+---------------+-----------------+-------------+-------------+------------+-----------------
   1 | 2023-01-01        |                  30 | paid           |              100 | 2022-12-31    |           12215 |         677 |         306 |       6081 | 2023-01-02
   2 | 2023-02-15        |                  45 | not paid       |               50 | 2023-02-14    |           59311 |         185 |         765 |       1856 | 2023-02-16
  78 | 2021-08-13        |                  37 | paid           |               86 | 2020-10-02    |            1196 |         290 |         306 |        192 | 2020-06-12
  93 | 2023-11-17        |                  24 | paid           |              100 | 2023-11-16    |           12345 |         880 |         263 |       7219 | 2023-11-17
 142 | 2020-06-10        |                 185 | paid           |               86 | 2023-04-25    |            7910 |         822 |         912 |        144 | 2021-02-03
 237 | 2021-04-16        |                 201 | not paid       |               41 | 2020-07-31    |            3362 |         880 |         306 |        641 | 2021-11-19
 241 | 2022-06-20        |                  10 | not paid       |               23 | 2023-10-18    |            7640 |         822 |         211 |        904 | 2021-08-10
 255 | 2023-04-01        |                 330 | paid           |               28 | 2020-10-03    |            1473 |         303 |         304 |        414 | 2021-10-12
 261 | 2023-11-23        |                  45 | not paid       |               50 | 2023-11-22    |           54321 |         204 |          29 |       9876 | 2023-11-23
 283 | 2021-10-12        |                 227 | paid           |               45 | 2022-12-01    |            6536 |         559 |         263 |        602 | 2021-03-07
 321 | 2022-12-09        |                 209 | not paid       |               44 | 2020-08-03    |            7998 |         190 |         780 |        812 | 2020-03-18
 485 | 2022-08-14        |                 222 | not paid       |               97 | 2022-07-11    |            1528 |         851 |         765 |        746 | 2020-03-30
 526 | 2022-09-17        |                 239 | paid           |               62 | 2023-04-01    |            1576 |         677 |         304 |        302 | 2023-03-05
 578 | 2022-12-17        |                 259 | not paid       |               73 | 2022-09-22    |            6766 |         962 |          92 |        527 | 2020-05-30
 629 | 2022-02-16        |                  23 | paid           |               62 | 2021-02-07    |            8902 |         185 |         252 |        271 | 2021-01-27
 640 | 2020-01-28        |                 188 | paid           |               83 | 2020-10-24    |            8663 |         185 |         244 |        364 | 2022-07-13
 667 | 2022-05-12        |                 165 | paid           |               77 | 2022-04-24    |            8527 |         677 |         780 |        544 | 2022-03-14
 684 | 2022-07-27        |                 258 | not paid       |               99 | 2021-01-04    |            1145 |         592 |         689 |        658 | 2022-09-26
 800 | 2023-06-16        |                   7 | paid           |               44 | 2021-03-19    |            4462 |         868 |         765 |        917 | 2022-10-27
 810 | 2020-06-27        |                 135 | paid           |               64 | 2022-08-13    |            9368 |         810 |         900 |        852 | 2022-01-05
 832 | 2020-11-22        |                  42 | paid           |               39 | 2023-06-24    |            4476 |         459 |          92 |        953 | 2022-03-19
 847 | 2023-04-13        |                  23 | not paid       |               89 | 2023-08-20    |            6505 |          26 |         912 |        350 | 2020-05-02
 894 | 2021-01-28        |                 137 | not paid       |               56 | 2022-11-16    |            1389 |         810 |         263 |        550 | 2022-02-18
 973 | 2021-09-08        |                 128 | paid           |               94 | 2023-09-09    |            1090 |         784 |          92 |        825 | 2020-03-15
(24 строки)
```

Рисунок 11. Таблица order до выполнения процедуры add_order()

```
PublishingHouseOfComputerLiterature=# call add_order('5 days', 25, 'paid', 190, 52398, 'Shawna Stout', 'Devin Serrano', 91245);
CALL
```

Рисунок 12. Выполнение процедуры add_order()

```
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 | act_number | date_of_the_act
-----+-------------------+---------------------+----------------+------------------+---------------+-----------------+-------------+-------------+------------+-----------------
   1 | 2023-01-01        |                  30 | paid           |              100 | 2022-12-31    |           12215 |         677 |         306 |       6081 | 2023-01-02
   2 | 2023-02-15        |                  45 | not paid       |               50 | 2023-02-14    |           59311 |         185 |         765 |       1856 | 2023-02-16
  78 | 2021-08-13        |                  37 | paid           |               86 | 2020-10-02    |            1196 |         290 |         306 |        192 | 2020-06-12
  93 | 2023-11-17        |                  24 | paid           |              100 | 2023-11-16    |           12345 |         880 |         263 |       7219 | 2023-11-17
 142 | 2020-06-10        |                 185 | paid           |               86 | 2023-04-25    |            7910 |         822 |         912 |        144 | 2021-02-03
 237 | 2021-04-16        |                 201 | not paid       |               41 | 2020-07-31    |            3362 |         880 |         306 |        641 | 2021-11-19
 241 | 2022-06-20        |                  10 | not paid       |               23 | 2023-10-18    |            7640 |         822 |         211 |        904 | 2021-08-10
 255 | 2023-04-01        |                 330 | paid           |               28 | 2020-10-03    |            1473 |         303 |         304 |        414 | 2021-10-12
 261 | 2023-11-23        |                  45 | not paid       |               50 | 2023-11-22    |           54321 |         204 |          29 |       9876 | 2023-11-23
 283 | 2021-10-12        |                 227 | paid           |               45 | 2022-12-01    |            6536 |         559 |         263 |        602 | 2021-03-07
 321 | 2022-12-09        |                 209 | not paid       |               44 | 2020-08-03    |            7998 |         190 |         780 |        812 | 2020-03-18
 485 | 2022-08-14        |                 222 | not paid       |               97 | 2022-07-11    |            1528 |         851 |         765 |        746 | 2020-03-30
 526 | 2022-09-17        |                 239 | paid           |               62 | 2023-04-01    |            1576 |         677 |         304 |        302 | 2023-03-05
 578 | 2022-12-17        |                 259 | not paid       |               73 | 2022-09-22    |            6766 |         962 |          92 |        527 | 2020-05-30
 629 | 2022-02-16        |                  23 | paid           |               62 | 2021-02-07    |            8902 |         185 |         252 |        271 | 2021-01-27
 640 | 2020-01-28        |                 188 | paid           |               83 | 2020-10-24    |            8663 |         185 |         244 |        364 | 2022-07-13
 667 | 2022-05-12        |                 165 | paid           |               77 | 2022-04-24    |            8527 |         677 |         780 |        544 | 2022-03-14
 684 | 2022-07-27        |                 258 | not paid       |               99 | 2021-01-04    |            1145 |         592 |         689 |        658 | 2022-09-26
 800 | 2023-06-16        |                   7 | paid           |               44 | 2021-03-19    |            4462 |         868 |         765 |        917 | 2022-10-27
 810 | 2020-06-27        |                 135 | paid           |               64 | 2022-08-13    |            9368 |         810 |         900 |        852 | 2022-01-05
 832 | 2020-11-22        |                  42 | paid           |               39 | 2023-06-24    |            4476 |         459 |          92 |        953 | 2022-03-19
 847 | 2023-04-13        |                  23 | not paid       |               89 | 2023-08-20    |            6505 |          26 |         912 |        350 | 2020-05-02
 894 | 2021-01-28        |                 137 | not paid       |               56 | 2022-11-16    |            1389 |         810 |         263 |        550 | 2022-02-18
 973 | 2021-09-08        |                 128 | paid           |               94 | 2023-09-09    |            1090 |         784 |          92 |        825 | 2020-03-15
 974 | 2023-12-15        |                  25 | paid           |              190 | 2023-12-10    |           52398 |         851 |         211 |      91245 | 2023-12-20
(25 строк)
```

Рисунок 13. Таблица order после выполнения процедуры add_order()

```
PublishingHouseOfComputerLiterature=# CREATE OR REPLACE FUNCTION update_payment_status()
PublishingHouseOfComputerLiterature-# RETURNS TRIGGER AS
PublishingHouseOfComputerLiterature-# $$
PublishingHouseOfComputerLiterature$# BEGIN
PublishingHouseOfComputerLiterature$#     IF NEW.payment_status = 'not paid' AND NEW.amount >= 0.9 * (SELECT SUM(amount) FROM complit.account WHERE order_id = NEW.id) THEN
PublishingHouseOfComputerLiterature$#         NEW.payment_status := 'paid';
PublishingHouseOfComputerLiterature$#     END IF;
PublishingHouseOfComputerLiterature$#     RETURN NEW;
PublishingHouseOfComputerLiterature$# END;
PublishingHouseOfComputerLiterature$# $$ LANGUAGE plpgsql;
CREATE FUNCTION
```

```
PublishingHouseOfComputerLiterature=# CREATE TRIGGER payment_status_trigger
PublishingHouseOfComputerLiterature-# AFTER INSERT OR UPDATE ON complit.account
PublishingHouseOfComputerLiterature-# FOR EACH ROW
PublishingHouseOfComputerLiterature-# EXECUTE FUNCTION update_payment_status();
CREATE TRIGGER
```

- Создание авторского триггера для обновления списков покупателей (их удаления), у которых статус оплаты 'not paid' более 3 раз.

```
CREATE OR REPLACE FUNCTION delete_customer()

RETURNS TRIGGER AS

$$

BEGIN

   IF (

      SELECT COUNT(*)

      FROM complit."order"

      WHERE customer_id = NEW.customer_id AND payment_status = 'not paid'

   ) >= 3 THEN


     DELETE FROM complit.account

      WHERE order_id IN (

      SELECT id

      FROM complit."order"

      WHERE customer_id = NEW.customer_id AND payment_status = 'not paid'

      );


      DELETE FROM complit.making_an_order

      WHERE order_id IN (

      SELECT id

      FROM complit."order"

      WHERE customer_id = NEW.customer_id AND payment_status = 'not paid'

      );
```

```
        DELETE FROM complit."order"
        WHERE customer_id = NEW.customer_id AND payment_status = 'not paid';

        DELETE FROM complit.customer
        WHERE id = NEW.customer_id;
    END IF;
    RETURN NEW;
END;
$$
LANGUAGE plpgsql;
```

```
PublishingHouseOfComputerLiterature=# CREATE OR REPLACE FUNCTION delete_customer()
PublishingHouseOfComputerLiterature-# RETURNS TRIGGER AS
PublishingHouseOfComputerLiterature-# $$
PublishingHouseOfComputerLiterature$# BEGIN
PublishingHouseOfComputerLiterature$#     IF (
PublishingHouseOfComputerLiterature$#         SELECT COUNT(*)
PublishingHouseOfComputerLiterature$#         FROM complit."order"
PublishingHouseOfComputerLiterature$#         WHERE customer_id = NEW.customer_id AND payment_status = 'not paid'
PublishingHouseOfComputerLiterature$#     ) >= 3 THEN
PublishingHouseOfComputerLiterature$#
PublishingHouseOfComputerLiterature$#     DELETE FROM complit.account
PublishingHouseOfComputerLiterature$#         WHERE order_id IN (
PublishingHouseOfComputerLiterature$#         SELECT id
PublishingHouseOfComputerLiterature$#         FROM complit."order"
PublishingHouseOfComputerLiterature$#         WHERE customer_id = NEW.customer_id AND payment_status = 'not paid'
PublishingHouseOfComputerLiterature$#         );
PublishingHouseOfComputerLiterature$#
PublishingHouseOfComputerLiterature$# DELETE FROM complit.making_an_order
PublishingHouseOfComputerLiterature$#         WHERE order_id IN (
PublishingHouseOfComputerLiterature$#         SELECT id
PublishingHouseOfComputerLiterature$#         FROM complit."order"
PublishingHouseOfComputerLiterature$#         WHERE customer_id = NEW.customer_id AND payment_status = 'not paid'
PublishingHouseOfComputerLiterature$#         );
PublishingHouseOfComputerLiterature$#
PublishingHouseOfComputerLiterature$#     DELETE FROM complit."order"
PublishingHouseOfComputerLiterature$#         WHERE customer_id = NEW.customer_id AND payment_status = 'not paid';
PublishingHouseOfComputerLiterature$#
PublishingHouseOfComputerLiterature$#         DELETE FROM complit.customer
PublishingHouseOfComputerLiterature$#         WHERE id = NEW.customer_id;
PublishingHouseOfComputerLiterature$#     END IF;
PublishingHouseOfComputerLiterature$#
PublishingHouseOfComputerLiterature$#     RETURN NEW;
PublishingHouseOfComputerLiterature$# END;
PublishingHouseOfComputerLiterature$# $$
PublishingHouseOfComputerLiterature-# LANGUAGE plpgsql;
CREATE FUNCTION
```

Рисунок 13. Создание функции delete_customer()

```
CREATE TRIGGER customer_unpayment
AFTER INSERT OR UPDATE ON complit."order"
FOR EACH ROW
EXECUTE FUNCTION delete_customer();
```

```
PublishingHouseOfComputerLiterature=# CREATE TRIGGER customer_unpayment
PublishingHouseOfComputerLiterature-# AFTER INSERT OR UPDATE ON complit."order"
PublishingHouseOfComputerLiterature-# FOR EACH ROW
PublishingHouseOfComputerLiterature-# EXECUTE FUNCTION delete_customer();
CREATE TRIGGER
```

Рисунок 14. Создание триггера customer_unpayment

```
PublishingHouseOfComputerLiterature=# SELECT * FROM complit.customer ORDER BY id;
 id  |     username     |           address            |            email            |   phone_number
-----+------------------+------------------------------+-----------------------------+------------------
  26 | Patrick Delgado  | 69614 Smith Street          +| cherylhernandez@example.com | +1(783)-348-47-57
     |                  | Martinburgh, CT 99623        |                             |
 185 | Lori Norton      | 01234 Teresa Lake Suite 239 +| lisabeck@example.net        | +6(134)-660-49-48
     |                  | Gomezville, MS 34718         |                             |
 190 | Mark Mcdonald    | PSC 5387, Box 5797          +| stephenlynn@example.net     | +5(582)-376-37-75
     |                  | APO AA 82606                 |                             |
 204 | Harold Adams     | 9998 Melissa Turnpike       +| carolyncallahan@example.org | +5(116)-760-36-36
     |                  | Davidfurt, OR 28807          |                             |
 290 | Jason Montgomery | 3095 Stephanie Centers      +| jasonweaver@example.com     | +5(116)-294-63-52
     |                  | Cynthiamouth, IA 15574       |                             |
 303 | Alan Evans       | 248 Miller Point Apt. 911   +| zmartinez@example.net       | +4(183)-423-37-82
     |                  | Oscarfurt, MI 44711          |                             |
 459 | Sydney Hardy     | 07374 Lee Views             +| westmichael@example.org     | +1(181)-669-23-93
     |                  | Lake Austin, GA 36591        |                             |
 559 | Jonathan Smith   | Unit 0207 Box 4847          +| lnichols@example.com        | +9(231)-162-95-35
     |                  | DPO AA 99523                 |                             |
 592 | Travis Mack      | 249 Moore Port Apt. 844     +| kellybrian@example.net      | +5(600)-104-38-22
     |                  | North Janet, VT 72170        |                             |
 677 | Kim Wilson       | 461 Flores Spurs Suite 838  +| wnewman@example.org         | +1(653)-171-17-96
     |                  | South Jenniferview, ND 82189 |                             |
 784 | Jessica Beck     | 162 Medina Park             +| marshdanielle@example.net   | +4(499)-306-31-28
     |                  | Timbury, DE 12206            |                             |
 810 | Kelli Key        | 809 Parsons Points Suite 370+| lisa57@example.net          | +2(344)-880-13-71
     |                  | North Emily, CO 94540        |                             |
 822 | Nathan Gilmore   | 32827 Wells Points Apt. 892 +| ivaughn@example.net         | +4(475)-912-38-17
     |                  | Port Jessicamouth, WV 16023  |                             |
 851 | Shawna Stout     | 4750 Jaclyn Plains Apt. 490 +| stephaniemccoy@example.net  | +9(815)-718-19-44
     |                  | New Charlesborough, OH 90135 |                             |
 868 | Rhonda Hughes    | 2122 Brown Mill Apt. 006    +| steven29@example.net        | +4(172)-431-19-11
     |                  | North Kevin, CT 43868        |                             |
 880 | Shawn Whitaker   | 7590 Bishop Radial Apt. 942 +| pamelamitchell@example.com  | +6(556)-774-58-87
     |                  | West Raymond, NH 68526       |                             |
 962 | Dylan Thompson   | 6919 Larry Pines            +| garciadiane@example.org     | +6(720)-834-36-70
     |                  | North Michaelbury, UT 44073  |                             |
(17 строк)
```

Рисунок 15. Таблица customer до работы триггера

```
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 | act_number | date_of_the_act
-----+-------------------+---------------------+----------------+------------------+---------------+-----------------+-------------+-------------+------------+-----------------
   1 | 2023-01-01        |                  30 | paid           |              100 | 2022-12-31    |           12215 |         677 |         306 |       6081 | 2023-01-02
   2 | 2023-02-15        |                  45 | not paid       |               50 | 2023-02-14    |           59311 |         185 |         765 |       1856 | 2023-02-16
  78 | 2021-08-13        |                  37 | paid           |               86 | 2020-10-02    |            1196 |         290 |         306 |        192 | 2020-06-12
  93 | 2023-11-17        |                  24 | paid           |              100 | 2023-11-16    |           12345 |         880 |         263 |       7219 | 2023-11-17
 142 | 2020-06-10        |                 185 | paid           |               86 | 2023-04-25    |            7910 |         822 |         912 |        144 | 2021-02-03
 237 | 2021-04-16        |                 201 | not paid       |               41 | 2020-07-31    |            3362 |         880 |         306 |        641 | 2021-11-19
 241 | 2022-06-20        |                  10 | not paid       |               23 | 2023-10-18    |            7640 |         822 |         211 |        904 | 2021-08-10
 255 | 2023-04-01        |                 330 | paid           |               28 | 2020-10-03    |            1473 |         303 |         304 |        414 | 2021-10-12
 261 | 2023-11-23        |                  45 | not paid       |               50 | 2023-11-22    |           54321 |         204 |          29 |       9876 | 2023-11-23
 283 | 2021-10-12        |                 227 | paid           |               45 | 2022-12-01    |            6536 |         559 |         263 |        602 | 2021-03-07
 321 | 2022-12-09        |                 209 | not paid       |               44 | 2020-08-03    |            7998 |         190 |         780 |        812 | 2020-03-18
 485 | 2022-08-14        |                 222 | not paid       |               97 | 2022-07-11    |            1528 |         851 |         765 |        746 | 2020-03-30
 526 | 2022-09-17        |                 239 | paid           |               62 | 2023-04-01    |            1576 |         677 |         304 |        302 | 2023-03-05
 527 | 2021-10-26        |                  15 | not paid       |              140 | 2021-10-24    |           19355 |          26 |         809 |       8253 | 2021-10-25
 528 | 2023-09-17        |                  30 | not paid       |              100 | 2023-09-16    |           89294 |          26 |         451 |       1927 | 2023-10-01
 578 | 2022-12-17        |                 259 | not paid       |               73 | 2022-09-22    |            6766 |         962 |          92 |        527 | 2020-05-30
 629 | 2022-02-16        |                  23 | paid           |               62 | 2021-02-07    |            8902 |         185 |         252 |        271 | 2021-01-27
 640 | 2020-01-28        |                 188 | paid           |               83 | 2020-10-24    |            8663 |         185 |         244 |        364 | 2022-07-13
 667 | 2022-05-12        |                 165 | paid           |               77 | 2022-04-24    |            8527 |         677 |         780 |        544 | 2022-03-14
 684 | 2022-07-27        |                 258 | not paid       |               99 | 2021-01-04    |            1145 |         592 |         689 |        658 | 2022-09-26
 800 | 2023-06-16        |                   7 | paid           |               44 | 2021-03-19    |            4462 |         868 |         765 |        917 | 2022-10-27
 810 | 2020-06-27        |                 135 | paid           |               64 | 2022-08-13    |            9368 |         810 |         900 |        852 | 2022-01-05
 832 | 2020-11-22        |                  42 | paid           |               39 | 2023-06-24    |            4476 |         459 |          92 |        953 | 2022-03-19
 847 | 2023-04-13        |                  23 | not paid       |               89 | 2023-08-20    |            6505 |          26 |         912 |        350 | 2020-05-02
 894 | 2021-01-28        |                 137 | not paid       |               56 | 2022-11-16    |            1389 |         810 |         263 |        550 | 2022-02-18
 973 | 2021-09-08        |                 128 | paid           |               94 | 2023-09-09    |            1090 |         784 |          92 |        825 | 2020-03-15
 974 | 2023-12-15        |                  25 | paid           |              190 | 2023-12-10    |           52398 |         851 |         211 |      91245 | 2023-12-20
(27 строк)
```

```
PublishingHouseOfComputerLiterature=# SELECT * FROM complit.customer ORDER BY id;
 id  |     username     |           address            |            email             |   phone_number
-----+------------------+------------------------------+------------------------------+-------------------
 185 | Lori Norton      | 01234 Teresa Lake Suite 239 +| lisabeck@example.net         | +6(134)-660-49-48
     |                  | Gomezville, MS 34718         |                              |
 190 | Mark Mcdonald    | PSC 5387, Box 5797          +| stephenlynn@example.net      | +5(582)-376-37-75
     |                  | APO AA 82606                 |                              |
 204 | Harold Adams     | 9998 Melissa Turnpike       +| carolyncallahan@example.org  | +5(116)-760-36-36
     |                  | Davidfurt, OR 28807          |                              |
 290 | Jason Montgomery | 3095 Stephanie Centers      +| jasonweaver@example.com      | +5(116)-294-63-52
     |                  | Cynthiamouth, IA 15574       |                              |
 303 | Alan Evans       | 248 Miller Point Apt. 911   +| zmartinez@example.net        | +4(183)-423-37-82
     |                  | Oscarfurt, MI 44711          |                              |
 459 | Sydney Hardy     | 07374 Lee Views             +| westmichael@example.org      | +1(181)-669-23-93
     |                  | Lake Austin, GA 36591        |                              |
 559 | Jonathan Smith   | Unit 0207 Box 4847          +| lnichols@example.com         | +9(231)-162-95-35
     |                  | DPO AA 99523                 |                              |
 592 | Travis Mack      | 249 Moore Port Apt. 844     +| kellybrian@example.net       | +5(600)-104-38-22
     |                  | North Janet, VT 72170        |                              |
 677 | Kim Wilson       | 461 Flores Spurs Suite 838  +| wnewman@example.org          | +1(653)-171-17-96
     |                  | South Jenniferview, ND 82189 |                              |
 784 | Jessica Beck     | 162 Medina Park             +| marshdanielle@example.net    | +4(499)-306-31-28
     |                  | Timbury, DE 12206            |                              |
 810 | Kelli Key        | 809 Parsons Points Suite 370+| lisa57@example.net           | +2(344)-880-13-71
     |                  | North Emily, CO 94540        |                              |
 822 | Nathan Gilmore   | 32827 Wells Points Apt. 892 +| ivaughn@example.net          | +4(475)-912-38-17
     |                  | Port Jessicamouth, WV 16023  |                              |
 851 | Shawna Stout     | 4750 Jaclyn Plains Apt. 490 +| stephaniemccoy@example.net   | +9(815)-718-19-44
     |                  | New Charlesborough, OH 90135 |                              |
 868 | Rhonda Hughes    | 2122 Brown Mill Apt. 006    +| steven29@example.net         | +4(172)-431-19-11
     |                  | North Kevin, CT 43868        |                              |
 880 | Shawn Whitaker   | 7590 Bishop Radial Apt. 942 +| pamelamitchell@example.com   | +6(556)-774-58-87
     |                  | West Raymond, NH 68526       |                              |
 962 | Dylan Thompson   | 6919 Larry Pines            +| garciadiane@example.org      | +6(720)-834-36-70
     |                  | North Michaelbury, UT 44073  |                              |
(16 строк)
```

Рисунок 16. Таблица customer 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.