

ГУАП

КАФЕДРА № 43

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

старший преподаватель

должность, уч. степень, звание

подпись, дата

Н.В. Путилова

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №7

Хранимые процедуры. Управление доступом

по курсу: ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. № 4236

подпись, дата

А.В. Панченко

инициалы, фамилия

Санкт-Петербург 2024

Цель работы: Получения навыков по серверному программированию и управлению доступом в базе данных.

Описание задания

1) Создать в БД хранимые процедуры, реализующие:

— вставку с пополнением справочников (вставляется информация о студенте, если указанный номер группы отсутствует в БД, запись добавляется в таблицу с 84 перечнем групп) (получаем ссылку на внешний ключ по значению данного из родительской таблицы);

— хранимую функцию любого назначения, соответствующего предметной области.

— каскадное удаление: удаление всех зависимых данных (перед удалением записи о группе удаляются записи обо всех студентах этой группы и их секциях);

2) Создать пользователя, который обладает правами, указанными в варианте задания

Вариант 18:

18. служба доставки: адрес доставки, контактное лицо, стоимость посылки, диапазон желаемого времени доставки. время доставки фактическое, вес посылки, отметка о доставке, фирма отправитель

а. все посылки, отправляемые в район, в названии которой есть начало «моск», но это не единственные буквы в нем

б. Улица, на которой нет адресов доставки

в. контактное лицо, получавшее посылки на улице Строителей и на проспекте Ленина

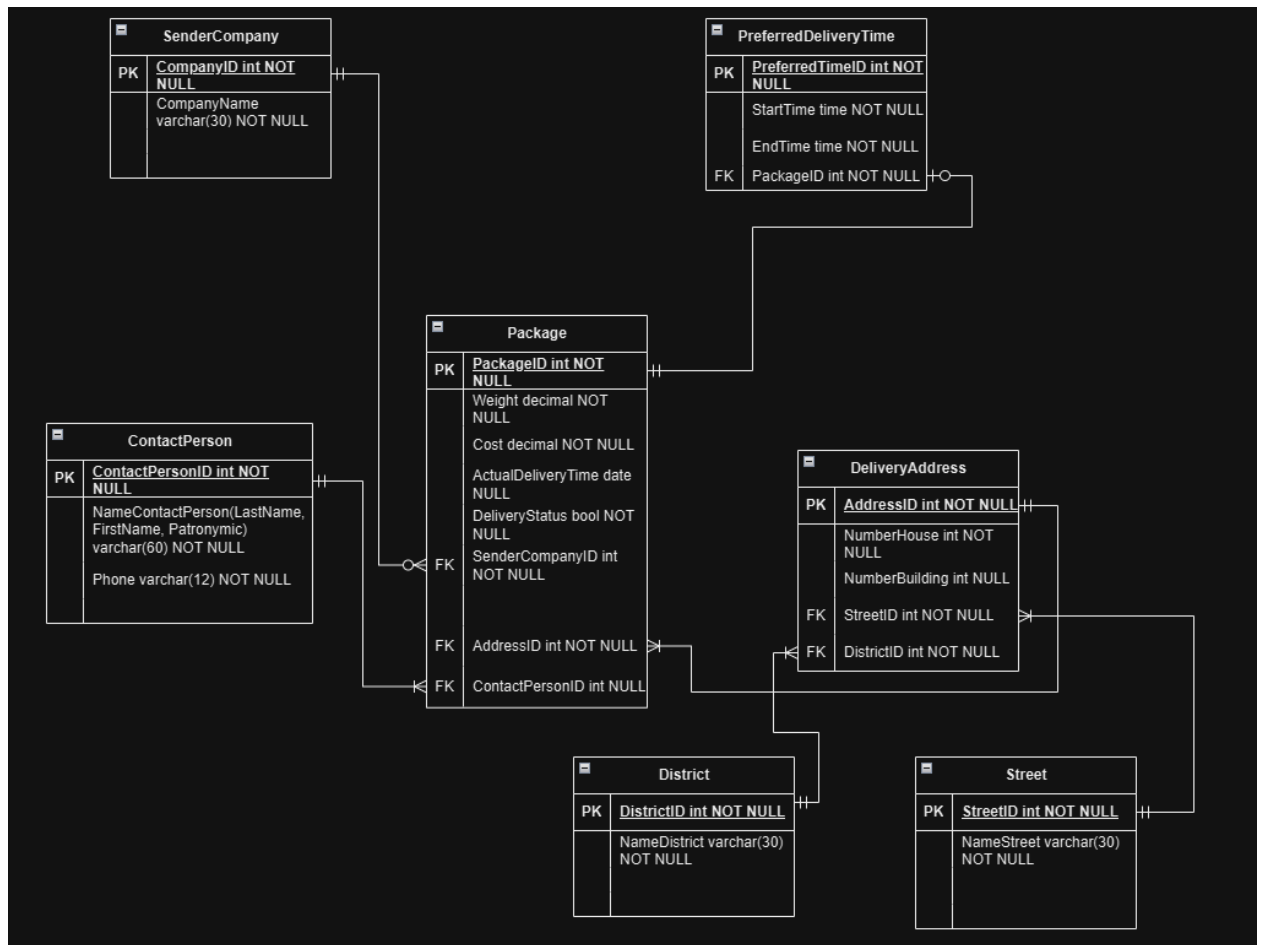
г. фирма, отправившая больше среднего посылок

д. посылка с весом меньше среднего

е. контактное лицо, получавшее посылки от всех фирм на букву Б

ж. контактное лицо, никогда не получавшее посылок в прошлом году, но получавшее посылки в феврале текущего

Физическая модель БД:



Назначение разработанных хранимых процедур или функций текстом:

1. Процедура `add_package_with_sender`

Эта процедура предназначена для добавления новой посылки в таблицу `Package`, а также для создания новой компании-отправителя в таблице `SenderCompany`, если такая компания еще не существует. Она принимает параметры, такие как вес и стоимость посылки, статус доставки, имя компании-отправителя, идентификатор адреса доставки и идентификатор контактного лица. При вызове процедуры осуществляется проверка на наличие указанной компании, и если ее нет, она добавляется, после чего создается запись о новой посылке.

2. Функция `count_delivered_packages`

Эта функция предназначена для подсчета количества доставленных посылок для заданной компании-отправителя. Она принимает имя компании в качестве параметра и выполняет выборку из таблицы `Package`, фильтруя записи по статусу доставки и имени компании. Функция возвращает целое число, соответствующее количеству доставленных посылок, что позволяет быстро получать статистику по работе компаний.

3. Процедура `delete_company_and_packages`

Эта процедура служит для удаления записи о компании-отправителе из таблицы `SenderCompany` и всех связанных с ней записей о посылках в таблице `Package`. При вызове процедуры сначала выбирается идентификатор компании, после чего выполняется каскадное удаление всех связанных посылок, что помогает поддерживать целостность данных и избегать "сиротских" записей о посылках без соответствующих компаний.

Скрипт для создания хранимых процедур или функций:

Процедура

```
CREATE OR REPLACE PROCEDURE add_package_with_sender(
    p_weight DECIMAL,
    p_cost DECIMAL(10, 2),
    p_delivery_status BOOLEAN,
    p_sender_company_name VARCHAR(30),
    p_address_id INT,
    p_contact_person_id INT
)
LANGUAGE plpgsql
SECURITY DEFINER
AS $$
DECLARE
    company_id INT;
BEGIN
    -- Проверяем, существует ли компания с таким названием
    SELECT CompanyID INTO company_id
    FROM SenderCompany
    WHERE CompanyName = p_sender_company_name;

    -- Если компании нет, добавляем ее в таблицу SenderCompany
    IF company_id IS NULL THEN
        INSERT INTO SenderCompany (CompanyName)
        VALUES (p_sender_company_name)
        RETURNING CompanyID INTO company_id;
    END IF;

    -- Вставляем данные о новом пакете
```

```
INSERT INTO Package (Weight, Cost, DeliveryStatus,  
SenderCompanyID, AddressID, ContactPersonID)  
VALUES (p_weight, p_cost, p_delivery_status, company_id,  
p_address_id, p_contact_person_id);  
END;  
$$;
```

Функция

```
CREATE OR REPLACE FUNCTION  
count_delivered_packages(p_company_name VARCHAR(30))  
RETURNS INT AS $$  
DECLARE  
    delivered_count INT;  
BEGIN  
    SELECT COUNT(*)  
    INTO delivered_count  
    FROM Package p  
    JOIN SenderCompany sc ON p.SenderCompanyID = sc.CompanyID  
    WHERE sc.CompanyName = p_company_name AND p.DeliveryStatus =  
TRUE;  
  
    RETURN delivered_count;  
END;  
$$ LANGUAGE plpgsql  
SECURITY DEFINER;
```

Процедура

```
CREATE OR REPLACE PROCEDURE  
delete_company_and_packages(p_company_name VARCHAR(30))  
LANGUAGE plpgsql
```

```
SECURITY DEFINER
AS $$
DECLARE
    company_id INT;
BEGIN
    -- Получаем ID компании
    SELECT CompanyID INTO company_id
    FROM SenderCompany
    WHERE CompanyName = p_company_name;

    -- Проверяем, существует ли компания
    IF company_id IS NOT NULL THEN

        DELETE FROM PreferredDeliveryTime
        WHERE PackageID IN (
            SELECT PackageID
            FROM Package
            WHERE SenderCompanyID = company_id
        );

        DELETE FROM Package
        WHERE SenderCompanyID = company_id;

        -- Удаляем компанию
        DELETE FROM SenderCompany
        WHERE CompanyID = company_id;
    END IF;
END;
$$;
```


Скрипт создания пользователя, с возможностью изменять данные только через процедуры и функции

```
-- Создание пользователя

CREATE USER delivery_user WITH PASSWORD 'password';

-- Ограничиваем доступ к таблицам

REVOKE ALL ON ALL TABLES IN SCHEMA public FROM
delivery_user;

-- Предоставляем доступ к выполнению процедур и функций

GRANT EXECUTE ON PROCEDURE
add_package_with_sender(DECIMAL, DECIMAL(10, 2), BOOLEAN,
VARCHAR(30), INT, INT) TO delivery_user;

GRANT EXECUTE ON FUNCTION
count_delivered_packages(VARCHAR(30)) TO delivery_user;

GRANT EXECUTE ON PROCEDURE
delete_company_and_packages(VARCHAR(30)) TO delivery_user;

GRANT SELECT ON ALL TABLES IN SCHEMA public TO
delivery_user;
```

SQL операторы и скриншоты наборов данных, иллюстрирующие работу процедур:

Вставка новой посылки через add_package_with_sender

-- Добавление посылки с новой компанией

```
CALL add_package_with_sender(4.5, 120.00, TRUE, 'NewDelivery', 1, 1);
```

-- Добавление посылки для существующей компании

```
CALL add_package_with_sender(3.0, 75.00, FALSE, 'FastDelivery', 2, 2);
```

Результат:

The image shows two screenshots of a PostgreSQL IDE interface. The top screenshot displays a query window with a SQL script that uses the `CALL` function to add two packages. The bottom screenshot shows the same IDE with a `SELECT` query executed, displaying the results in a table.

Query 1 (Top Screenshot):

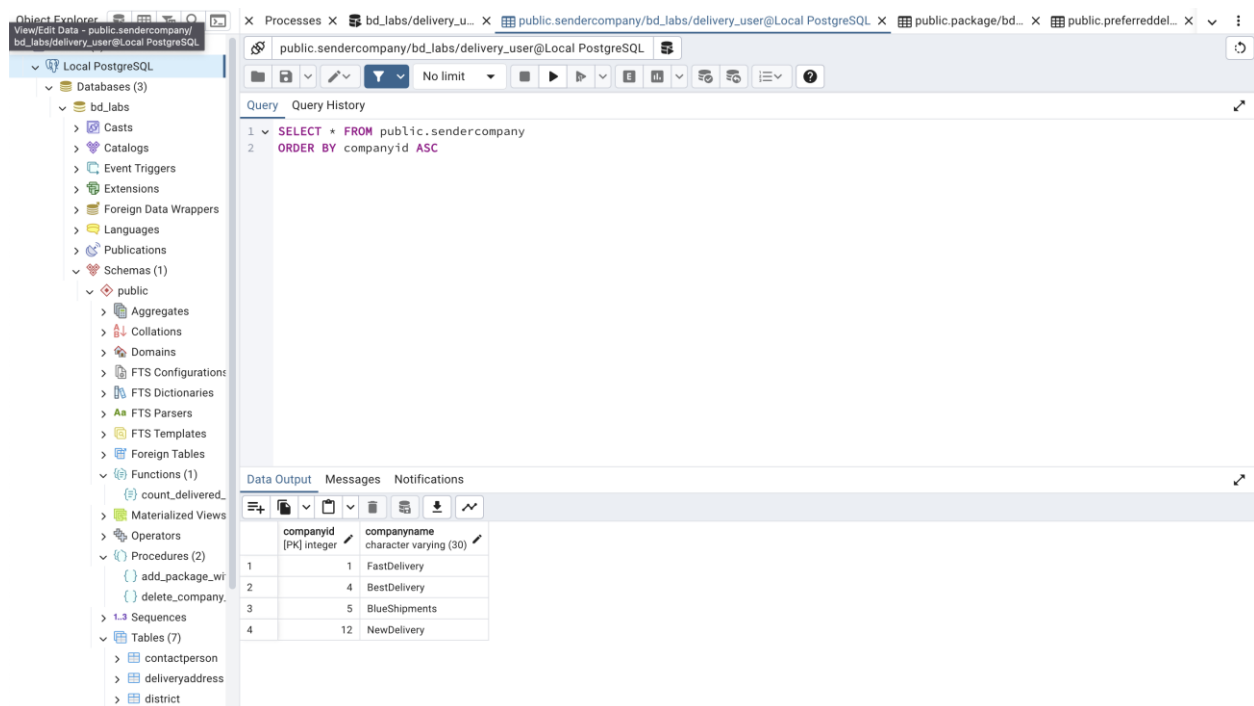
```
1 -- Добавление посылки с новой компанией
2 CALL add_package_with_sender(4.5, 120.00, TRUE, 'NewDelivery', 1, 1);
3
4 -- Добавление посылки для существующей компании
5 CALL add_package_with_sender(3.0, 75.00, FALSE, 'FastDelivery', 2, 2);
6
```

Query 2 (Bottom Screenshot):

```
1 SELECT * FROM public."package"
2 ORDER BY packageid ASC
```

Data Output (Bottom Screenshot):

	packageid [PK] integer	weight numeric	cost numeric (10,2)	actualdeliverytime date	deliverystatus boolean	sendercompanyid integer	addressid integer	contactpersonid integer
1	1	5.5	100.00	2023-09-15	true	1	1	1
2	4	20.0	300.00	2023-10-01	true	1	1	1
3	5	3.5	150.00	2023-10-05	true	1	4	2
4	7	2.0	120.00	2023-10-07	true	1	6	3
5	8	1.0	80.00	2023-10-08	true	1	7	3
6	10	4.0	90.00	2023-10-10	true	1	1	1
7	11	3.0	110.00	2023-10-11	true	1	1	2
8	13	3.5	100.00	2023-10-13	true	4	2	2
9	14	2.0	60.00	2023-10-14	true	5	3	2
10	16	20.0	300.00	2023-10-01	true	1	1	1
11	17	5.0	200.00	2024-02-15	true	1	1	4
12	30	3.0	75.00	[null]	false	1	2	2
13	31	4.5	120.00	[null]	true	12	1	1
14	32	4.5	120.00	[null]	true	12	1	1
15	33	3.0	75.00	[null]	false	1	2	2

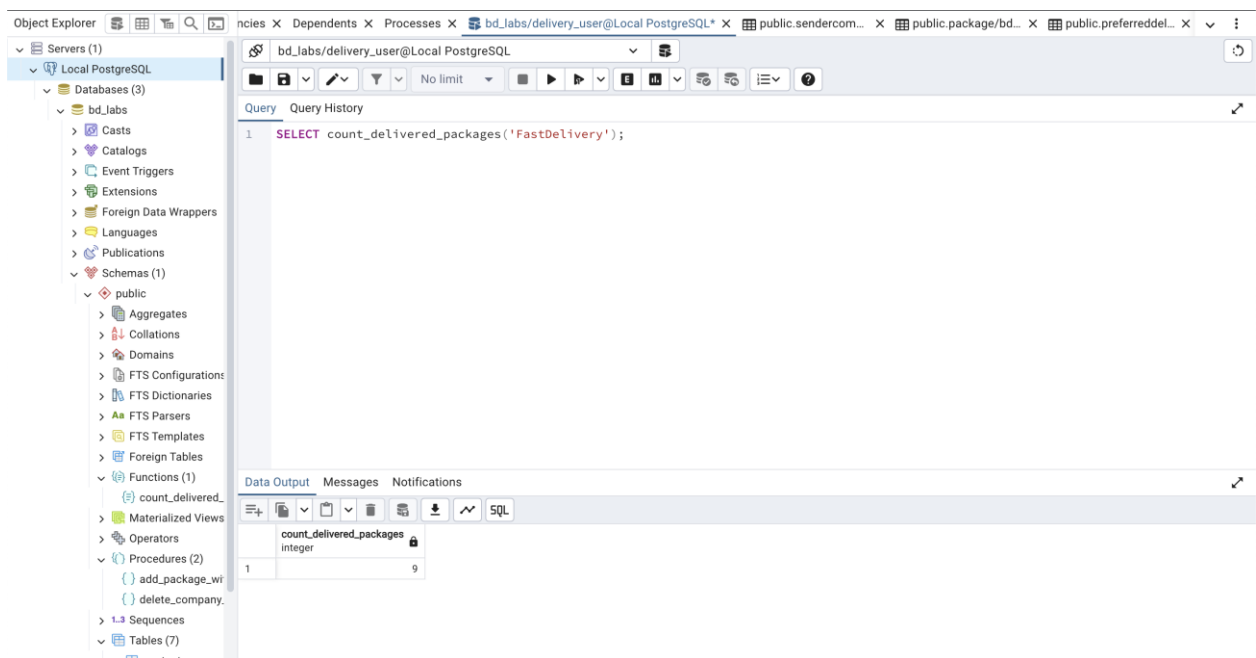


Расчет стоимости доставки:

-- Подсчет доставленных посылок для компании 'FastDelivery'

SELECT count_delivered_packages('FastDelivery');

Результат:



Каскадное удаление адреса и связанных данных:

-- Удаление компании и всех ее посылок, так же удаление желаемого времени доставки

CALL delete_company_and_packages('ExpressCo');

Данные до выполнения:

Object Explorer

- Extensions
- Foreign Data Wrappers
- Languages
- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi
 - delete_company.
 - Sequences
 - Tables (7)
 - contactperson
 - deliveryaddress
 - district
 - package
 - preferreddeliver
 - sendercompany
 - street
 - Trigger Functions
 - Types
 - Views
- Subscriptions
- postgres

Processes

bd_labs/andrejpan...

public.sendercompany/bd_labs/delivery_user@Local PostgreSQL

public.package/bd...

public.preferreddel...

Query

Query History

```
1 SELECT * FROM public.sendercompany
2 ORDER BY companyid ASC
```

Data Output

	companyid [PK] integer	companyname character varying (30)
1	1	FastDelivery
2	2	ExpressCo
3	4	BestDelivery
4	5	BlueShipments
5	12	NewDelivery

Messages

Notifications

✓ Server disconnected. ✕

✓ Server disconnected. ✕

Object Explorer

- Extensions
- Foreign Data Wrappers
- Languages
- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi
 - delete_company.
 - Sequences
 - Tables (7)
 - contactperson
 - deliveryaddress
 - district
 - package
 - preferreddeliver
 - sendercompany
 - street
 - Trigger Functions
 - Types
 - Views
- Subscriptions
- postgres

Processes

bd_labs/andrejpan...

public.sendercom...

public.package/bd_labs/delivery_user@Local PostgreSQL

public.preferreddel...

Query

Query History

```
1 SELECT * FROM public."package"
2 ORDER BY packageid ASC
```

Data Output

	packageid [PK] integer	weight numeric	cost numeric (10,2)	actualdeliverytime date	deliverystatus boolean	sendercompanyid integer	addressid integer	contactpersonid integer
1	1	5.5	100.00	2023-09-15	true	1	1	1
2	2	10.0	200.00	2023-10-01	true	2	2	2
3	4	20.0	300.00	2023-10-01	true	1	1	1
4	5	3.5	150.00	2023-10-05	true	1	4	2
5	6	5.0	100.00	2023-10-06	true	2	5	3
6	7	2.0	120.00	2023-10-07	true	1	6	3
7	8	1.0	80.00	2023-10-08	true	1	7	3
8	9	4.0	100.00	2023-10-09	true	2	3	2
9	10	4.0	90.00	2023-10-10	true	1	1	1
10	11	3.0	110.00	2023-10-11	true	1	1	2
11	12	0.5	50.00	2023-10-12	true	2	2	2
12	13	3.5	100.00	2023-10-13	true	4	2	2
13	14	2.0	60.00	2023-10-14	true	5	3	2
14	15	2.5	70.00	2024-02-10	true	2	2	2
15	16	20.0	300.00	2023-10-01	true	1	1	1
16	17	5.0	200.00	2024-02-15	true	1	1	4
17	30	3.0	75.00	[null]	false	1	2	2
18	31	4.5	120.00	[null]	true	12	1	1

Messages

Notifications

Object Explorer

- Extensions
- Foreign Data Wrappers
- Languages
- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi
 - delete_company.
 - Sequences
 - Tables (7)
 - contactperson
 - deliveryaddress
 - district
 - package
 - preferreddeliver
 - sendercompany
 - street
 - Trigger Functions
 - Types
 - Views
 - Subscriptions
 - postgres

Processes

- bd_labs/andrejpan...
- public.sendercom...
- public.package/bd...
- public.preferredeliverytime/bd_labs/delivery_user@Local PostgreSQL

Query

```

1 SELECT * FROM public.preferredeliverytime
2 ORDER BY preferredtimeid ASC

```

Data Output

	preferredtimeid [PK] integer	starttime time without time zone	endtime time without time zone	packageid integer
1	1	09:00:00	12:00:00	1
2	2	14:00:00	18:00:00	2

Данные после выполнения:

Object Explorer

- Servers (1)
 - Local PostgreSQL
 - Databases (3)
 - bd_labs
 - Casts
 - Catalogs
 - Event Triggers
 - Extensions
 - Foreign Data Wrappers
 - Languages
 - Publications
 - Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi
 - delete_company.
 - Sequences
 - Tables (7)
 - contactperson
 - deliveryaddress
 - district
 - package
 - preferreddeliver

Dashboard

- Properties
- SQL
- Statistics
- Dependencies
- Dependents
- Processes
- bd_labs/delivery_user@Local PostgreSQL*
- public.sendercom...

Query

```

1 CALL delete_company_and_packages('ExpressCo');

```

Data Output

CALL

Query returned successfully in 66 msec.

Total rows: 1 of 1 Query complete 00:00:00.066 Ln 1. Col 47

Object Explorer

- Servers (1)
 - Local PostgreSQL
 - Databases (3)
 - bd_labs
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_...
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi...
 - delete_company...
 - Sequences
 - Tables (7)
 - contactperson
 - deliveryaddress
 - district
 - package
 - preferreddeliver...

Query

```
1 SELECT * FROM public.sendercompany
2 ORDER BY companyid ASC
```

Data Output

companyid [PK] integer	companyname character varying (30)
1	FastDelivery
2	BestDelivery
3	BlueShipments
4	NewDelivery

Object Explorer

- Servers (1)
 - Local PostgreSQL
 - Databases (3)
 - bd_labs
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_...
 - Materialized Views
 - Operators

Query

```
1 SELECT * FROM public."package"
2 ORDER BY packageid ASC
```

Data Output

packageid [PK] integer	weight numeric	cost numeric (10,2)	actualdeliverytime date	deliverystatus boolean	sendercompanyid integer	addressid integer	contactpersonid integer	
1	1	5.5	100.00	2023-09-15	true	1	1	1
2	4	20.0	300.00	2023-10-01	true	1	1	1
3	5	3.5	150.00	2023-10-05	true	1	4	2
4	7	2.0	120.00	2023-10-07	true	1	6	3
5	8	1.0	80.00	2023-10-08	true	1	7	3
6	10	4.0	90.00	2023-10-10	true	1	1	1
7	11	3.0	110.00	2023-10-11	true	1	1	2
8	13	3.5	100.00	2023-10-13	true	4	2	2
9	14	2.0	60.00	2023-10-14	true	5	3	2
10	16	20.0	300.00	2023-10-01	true	1	1	1
11	17	5.0	200.00	2024-02-15	true	1	1	4
12	30	3.0	75.00	[null]	false	1	2	2
13	31	4.5	120.00	[null]	true	12	1	1

Object Explorer

- Servers (1)
 - Local PostgreSQL
 - Databases (3)
 - bd_labs
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_...
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi...
 - delete_company...

Query

```
1 SELECT * FROM public.preferreddeliverytime
2 ORDER BY preferredtimeid ASC
```

Data Output

preferredtimeid [PK] integer	starttime time without time zone	endtime time without time zone	packageid integer
1	1 09:00:00	12:00:00	1

SQL операторы и скриншоты наборов данных, отображающие права пользователя и демонстрирующие его работу с процедурными работами:

Доступ к таблицам:

SELECT

grantee,

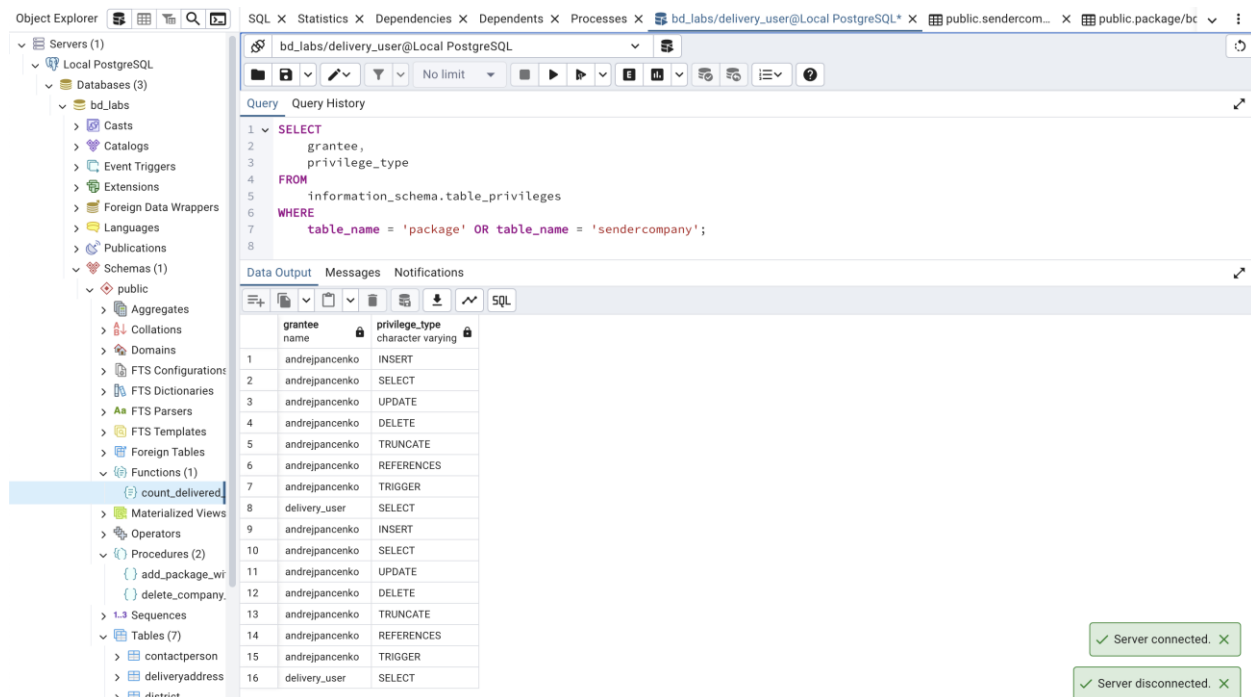
privilege_type

FROM

information_schema.table_privileges

WHERE

table_name = 'package' OR table_name = 'sendercompany';



The screenshot shows a PostgreSQL IDE interface. On the left, the 'Object Explorer' pane displays the database structure, including 'Schemas (1)' with 'public' selected. The main query editor shows the following SQL query:

```
1 SELECT
2   grantee,
3   privilege_type
4 FROM
5   information_schema.table_privileges
6 WHERE
7   table_name = 'package' OR table_name = 'sendercompany';
8
```

Below the query editor, the 'Data Output' pane displays the results of the query in a table format:

	grantee name	privilege_type character varying
1	andrejpancencko	INSERT
2	andrejpancencko	SELECT
3	andrejpancencko	UPDATE
4	andrejpancencko	DELETE
5	andrejpancencko	TRUNCATE
6	andrejpancencko	REFERENCES
7	andrejpancencko	TRIGGER
8	delivery_user	SELECT
9	andrejpancencko	INSERT
10	andrejpancencko	SELECT
11	andrejpancencko	UPDATE
12	andrejpancencko	DELETE
13	andrejpancencko	TRUNCATE
14	andrejpancencko	REFERENCES
15	andrejpancencko	TRIGGER
16	delivery_user	SELECT

At the bottom right, there are two status indicators: 'Server connected.' and 'Server disconnected.', both with a green checkmark and a close button.

Проверку доступа к функциям и процедурам:

SELECT

n.nspname as function_schema,

p.proname as function_name,

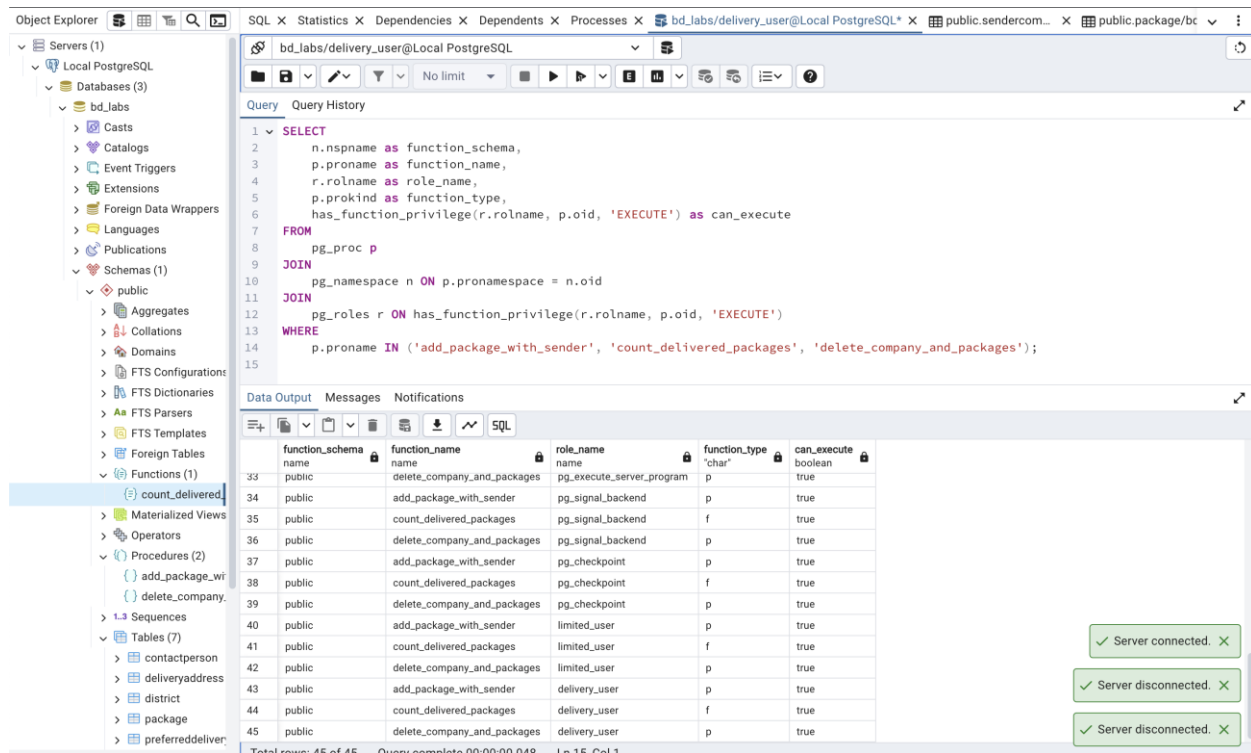
r.rolname as role_name,

p.prokind as function_type,

```

has_function_privilege(r.rolname, p.oid, 'EXECUTE') as can_execute
FROM
    pg_proc p
JOIN
    pg_namespace n ON p.pronamespace = n.oid
JOIN
    pg_roles r ON has_function_privilege(r.rolname, p.oid, 'EXECUTE')
WHERE
    p.proname IN ('add_package_with_sender', 'count_delivered_packages',
'delete_company_and_packages');

```



The screenshot shows a PostgreSQL IDE interface. On the left is the 'Object Explorer' showing the database structure. The main window displays a SQL query and its results in a table.

Query:

```

SELECT
    n.nsname as function_schema,
    p.proname as function_name,
    r.rolname as role_name,
    p.prokind as function_type,
    has_function_privilege(r.rolname, p.oid, 'EXECUTE') as can_execute
FROM
    pg_proc p
JOIN
    pg_namespace n ON p.pronamespace = n.oid
JOIN
    pg_roles r ON has_function_privilege(r.rolname, p.oid, 'EXECUTE')
WHERE
    p.proname IN ('add_package_with_sender', 'count_delivered_packages', 'delete_company_and_packages');

```

Data Output:

	function_schema	function_name	role_name	function_type	can_execute
33	public	delete_company_and_packages	pg_execute_server_program	p	true
34	public	add_package_with_sender	pg_signal_backend	p	true
35	public	count_delivered_packages	pg_signal_backend	f	true
36	public	delete_company_and_packages	pg_signal_backend	p	true
37	public	add_package_with_sender	pg_checkpoint	p	true
38	public	count_delivered_packages	pg_checkpoint	f	true
39	public	delete_company_and_packages	pg_checkpoint	p	true
40	public	add_package_with_sender	limited_user	p	true
41	public	count_delivered_packages	limited_user	f	true
42	public	delete_company_and_packages	limited_user	p	true
43	public	add_package_with_sender	delivery_user	p	true
44	public	count_delivered_packages	delivery_user	f	true
45	public	delete_company_and_packages	delivery_user	p	true

Total rows: 45 of 45. Query complete 00:00:00.048. In 15 Cnl 1.

On the right side of the IDE, there are three status messages:

- ✓ Server connected. ✕
- ✓ Server disconnected. ✕
- ✓ Server disconnected. ✕

Вставка новой посылки через add_package_with_sender

-- Добавление посылки с новой компанией

```
CALL add_package_with_sender(4.5, 120.00, TRUE, 'NewDelivery', 1, 1);
```

-- Добавление посылки для существующей компании

```
CALL add_package_with_sender(3.0, 75.00, FALSE, 'FastDelivery', 2, 2);
```


Результат:

The image shows two screenshots of a PostgreSQL IDE interface. The top screenshot displays a query window with a SQL script that uses the `CALL` function to add two packages. The bottom screenshot shows the same IDE with a `SELECT` query executed, displaying the results in a table.

Query 1 (Top Screenshot):

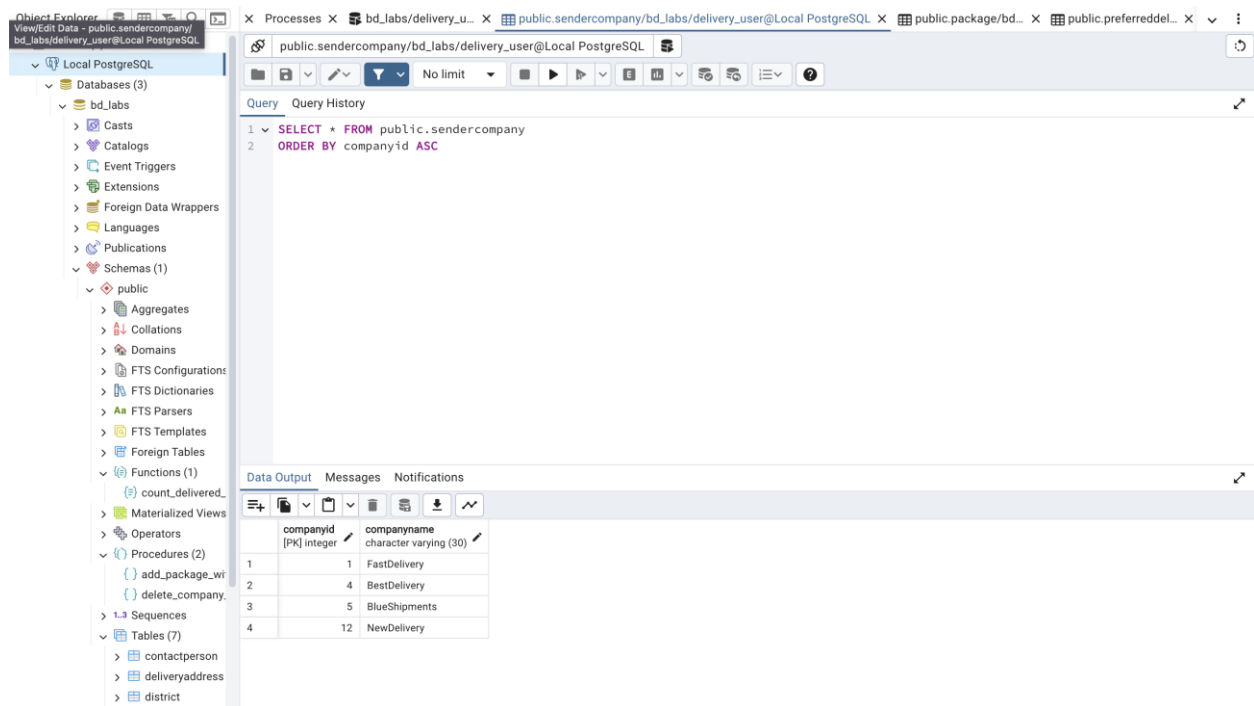
```
1 -- Добавление посылки с новой компанией
2 CALL add_package_with_sender(4.5, 120.00, TRUE, 'NewDelivery', 1, 1);
3
4 -- Добавление посылки для существующей компании
5 CALL add_package_with_sender(3.0, 75.00, FALSE, 'FastDelivery', 2, 2);
6
```

Query 2 (Bottom Screenshot):

```
1 SELECT * FROM public."package"
2 ORDER BY packageid ASC
```

Data Output (Bottom Screenshot):

	packageid [PK] integer	weight numeric	cost numeric (10,2)	actualdeliverytime date	deliverystatus boolean	sendercompanyid integer	addressid integer	contactpersonid integer
1	1	5.5	100.00	2023-09-15	true	1	1	1
2	4	20.0	300.00	2023-10-01	true	1	1	1
3	5	3.5	150.00	2023-10-05	true	1	4	2
4	7	2.0	120.00	2023-10-07	true	1	6	3
5	8	1.0	80.00	2023-10-08	true	1	7	3
6	10	4.0	90.00	2023-10-10	true	1	1	1
7	11	3.0	110.00	2023-10-11	true	1	1	2
8	13	3.5	100.00	2023-10-13	true	4	2	2
9	14	2.0	60.00	2023-10-14	true	5	3	2
10	16	20.0	300.00	2023-10-01	true	1	1	1
11	17	5.0	200.00	2024-02-15	true	1	1	4
12	30	3.0	75.00	[null]	false	1	2	2
13	31	4.5	120.00	[null]	true	12	1	1
14	32	4.5	120.00	[null]	true	12	1	1
15	33	3.0	75.00	[null]	false	1	2	2

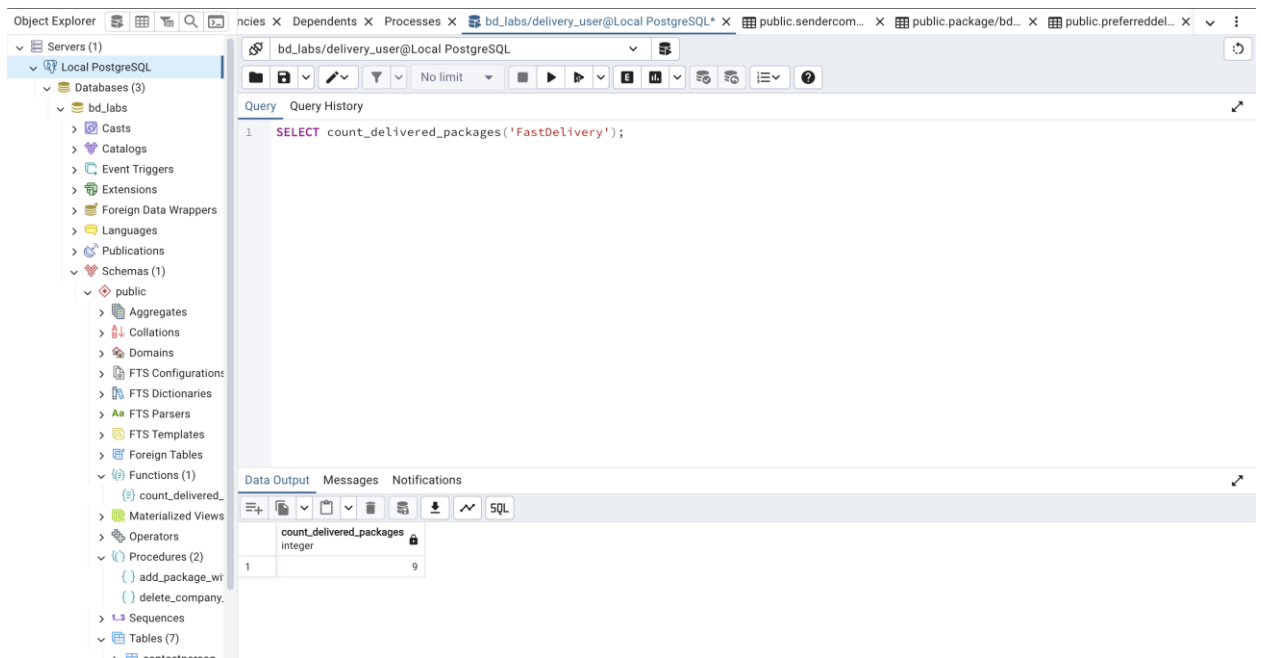


Расчет стоимости доставки:

-- Подсчет доставленных посылок для компании 'FastDelivery'

```
SELECT count_delivered_packages('FastDelivery');
```

Результат:



Каскадное удаление адреса и связанных данных:

-- Удаление компании и всех ее посылок, так же удаление желаемого времени доставки

```
CALL delete_company_and_packages('ExpressCo');
```

Данные до выполнения:

Object Explorer

- Extensions
- Foreign Data Wrappers
- Languages
- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi
 - delete_company.
 - Sequences
 - Tables (7)
 - contactperson
 - deliveryaddress
 - district
 - package
 - preferreddeliver
 - sendercompany
 - street
 - Trigger Functions
 - Types
 - Views
- Subscriptions
- postgres

Processes

bd_labs/andrejpan...

public.sendercompany/bd_labs/delivery_user@Local PostgreSQL

public.package/bd...

public.preferreddel...

Query

Query History

```
1 SELECT * FROM public.sendercompany
2 ORDER BY companyid ASC
```

Data Output

	companyid [PK] integer	companyname character varying (30)
1	1	FastDelivery
2	2	ExpressCo
3	4	BestDelivery
4	5	BlueShipments
5	12	NewDelivery

Messages

Notifications

✓ Server disconnected. ✕

✓ Server disconnected. ✕

Object Explorer

- Extensions
- Foreign Data Wrappers
- Languages
- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi
 - delete_company.
 - Sequences
 - Tables (7)
 - contactperson
 - deliveryaddress
 - district
 - package
 - preferreddeliver
 - sendercompany
 - street
 - Trigger Functions
 - Types
 - Views
- Subscriptions
- postgres

Processes

bd_labs/andrejpan...

public.sendercom...

public.package/bd_labs/delivery_user@Local PostgreSQL

public.preferreddel...

Query

Query History

```
1 SELECT * FROM public."package"
2 ORDER BY packageid ASC
```

Data Output

	packageid [PK] integer	weight numeric	cost numeric (10,2)	actualdeliverytime date	deliverystatus boolean	sendercompanyid integer	addressid integer	contactpersonid integer
1	1	5.5	100.00	2023-09-15	true	1	1	1
2	2	10.0	200.00	2023-10-01	true	2	2	2
3	4	20.0	300.00	2023-10-01	true	1	1	1
4	5	3.5	150.00	2023-10-05	true	1	4	2
5	6	5.0	100.00	2023-10-06	true	2	5	3
6	7	2.0	120.00	2023-10-07	true	1	6	3
7	8	1.0	80.00	2023-10-08	true	1	7	3
8	9	4.0	100.00	2023-10-09	true	2	3	2
9	10	4.0	90.00	2023-10-10	true	1	1	1
10	11	3.0	110.00	2023-10-11	true	1	1	2
11	12	0.5	50.00	2023-10-12	true	2	2	2
12	13	3.5	100.00	2023-10-13	true	4	2	2
13	14	2.0	60.00	2023-10-14	true	5	3	2
14	15	2.5	70.00	2024-02-10	true	2	2	2
15	16	20.0	300.00	2023-10-01	true	1	1	1
16	17	5.0	200.00	2024-02-15	true	1	1	4
17	30	3.0	75.00	[null]	false	1	2	2
18	31	4.5	120.00	[null]	true	12	1	1

Messages

Notifications

Object Explorer

- Extensions
- Foreign Data Wrappers
- Languages
- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi
 - delete_company.
 - Sequences
 - Tables (7)
 - contactperson
 - deliveryaddress
 - district
 - package
 - preferreddeliver
 - sendercompany
 - street
 - Trigger Functions
 - Types
 - Views
- Subscriptions
- postgres

Processes

- bd_labs/andrejpan...
- public.sendercom...
- public.package/bd...
- public.preferredeliverytime/bd_labs/delivery_user@Local PostgreSQL

Query

```
1 SELECT * FROM public.preferredeliverytime
2 ORDER BY preferredtimeid ASC
```

Data Output

	preferredtimeid [PK] integer	starttime time without time zone	endtime time without time zone	packageid integer
1	1	09:00:00	12:00:00	1
2	2	14:00:00	18:00:00	2

Данные после выполнения:

Object Explorer

- Servers (1)
 - Local PostgreSQL
 - Databases (3)
 - bd_labs
 - Casts
 - Catalogs
 - Event Triggers
 - Extensions
 - Foreign Data Wrappers
 - Languages
 - Publications
 - Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi
 - delete_company.
 - Sequences
 - Tables (7)
 - contactperson
 - deliveryaddress
 - district
 - package
 - preferreddeliver

Dashboard

- Properties
- SQL
- Statistics
- Dependencies
- Dependents
- Processes
- bd_labs/delivery_user@Local PostgreSQL*
- public.sendercom...

Query

```
1 CALL delete_company_and_packages('ExpressCo');
```

Data Output

CALL

Query returned successfully in 66 msec.

Total rows: 1 of 1 Query complete 00:00:00.066 Ln 1. Col 47

Object Explorer

- Servers (1)
 - Local PostgreSQL
 - Databases (3)
 - bd_labs
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_...
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi...
 - delete_company...
 - Sequences
 - Tables (7)
 - contactperson
 - deliveryaddress
 - district
 - package
 - preferreddeliver...

Query

```
1 SELECT * FROM public.sendercompany
2 ORDER BY companyid ASC
```

Data Output

companyid [PK] integer	companyname character varying (30)
1	FastDelivery
2	BestDelivery
3	BlueShipments
4	NewDelivery

Object Explorer

- Servers (1)
 - Local PostgreSQL
 - Databases (3)
 - bd_labs
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_...
 - Materialized Views
 - Operators

Query

```
1 SELECT * FROM public."package"
2 ORDER BY packageid ASC
```

Data Output

packageid [PK] integer	weight numeric	cost numeric (10,2)	actualdeliverytime date	deliverystatus boolean	sendercompanyid integer	addressid integer	contactpersonid integer	
1	1	5.5	100.00	2023-09-15	true	1	1	1
2	4	20.0	300.00	2023-10-01	true	1	1	1
3	5	3.5	150.00	2023-10-05	true	1	4	2
4	7	2.0	120.00	2023-10-07	true	1	6	3
5	8	1.0	80.00	2023-10-08	true	1	7	3
6	10	4.0	90.00	2023-10-10	true	1	1	1
7	11	3.0	110.00	2023-10-11	true	1	1	2
8	13	3.5	100.00	2023-10-13	true	4	2	2
9	14	2.0	60.00	2023-10-14	true	5	3	2
10	16	20.0	300.00	2023-10-01	true	1	1	1
11	17	5.0	200.00	2024-02-15	true	1	1	4
12	30	3.0	75.00	[null]	false	1	2	2
13	31	4.5	120.00	[null]	true	12	1	1

Object Explorer

- Servers (1)
 - Local PostgreSQL
 - Databases (3)
 - bd_labs
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions (1)
 - count_delivered_...
 - Materialized Views
 - Operators
 - Procedures (2)
 - add_package_wi...
 - delete_company...

Query

```
1 SELECT * FROM public.preferreddeliverytime
2 ORDER BY preferredtimeid ASC
```

Data Output

preferredtimeid [PK] integer	starttime time without time zone	endtime time without time zone	packageid integer
1	1 09:00:00	12:00:00	1

Нет прав на вставку новых данных в таблицы:

The screenshot displays the PostgreSQL Enterprise Studio interface. On the left, the Object Explorer shows the database structure, including the 'public' schema. The main query editor on the right contains an SQL statement to insert data into the 'SenderCompany' table. The query is as follows:

```
1 INSERT INTO SenderCompany (CompanyID, CompanyName)
2 VALUES
3 (1, 'FastDelivery'),
4 (2, 'ExpressCo'),
5 (3, 'SpeedyShipping'),
6 (4, 'BestDelivery'), -- Новая компания на букву Б
7 (5, 'BlueShipments'); -- Новая компания на букву Б
8
```

Below the query editor, the 'Messages' tab shows an error message: 'ERROR: permission denied for table sendercompany'. The 'SQL state' is reported as '42501'.

Выводы об использовании процедур и управление доступом в разработанной мной базе данных:

В ходе работы с разработанной базой данных было изучено использование хранимых процедур и управление доступом для пользователей. Созданные процедуры позволяют выполнять операции вставки, обновления и удаления данных, что значительно упрощает взаимодействие с базой данных и повышает ее безопасность. Ограничив доступ к таблицам и предоставив права на выполнение только определенных процедур, мы обеспечили защиту данных от несанкционированного доступа, позволяя пользователю выполнять только необходимые действия. Это подтверждает важность правильно настроенного управления доступом для эффективной работы с базами данных.