

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
КАФЕДРА МО ЭВМ

ОТЧЕТ
по лабораторной работе №2
по дисциплине «Базы данных»
Тема: Реализация базы данных в СУБД PostgreSQL

Студент гр. 1303 _____ Коренев Д. А.

Преподаватель _____ Заславский М. М.

Санкт-Петербург

2023

Цель работы.

Развернуть локально PostgreSQL, написать запросы для создания и заполнения таблиц, написать запросы к БД, отвечающие на вопросы в заданиях.

Задание.

Вариант 11

Пусть требуется создать программную систему, предназначенную для работников почтового отделения. Такая система должна обеспечивать хранение сведений о подписчиках газет и журналов, обслуживаемых отделением связи, и о почтальонах. Каждое подписное издание характеризуется индексом, названием и подписной ценой. Данные о подписчиках включают в себя: фамилию, имя, отчество, домашний адрес, индексы получаемых изданий, дату, начиная с которой оформлена подписка, и срок подписки на каждое издание. Несколько домов объединяются в участок, который обслуживается одним почтальоном. Каждый почтальон может обслуживать несколько участков. В БД должны содержаться сведения о том, к каким участкам относятся подписчики газет, и об обслуживающем их почтальоне. Заведующий почтовым отделением может принять на работу и уволить почтальона, при этом участки не должны оставаться без обслуживания. Оператор почтовой связи должен иметь возможность по просьбе клиента оформить подписку, а также добавить в БД сведения о новом подписном издании. Оформление подписки связано с выдачей клиенту квитанции, в которой указывается общая стоимость подписки, что выписано, и на какой срок. Возможны следующие запросы к БД:

- Определить наименование и количество экземпляров всех изданий, получаемых отделением связи.
- По заданному адресу определить фамилию почтальона, обслуживающего подписчика.

- Какие газеты выписывает гражданин с указанной фамилией, именем, отчеством?
- Сколько почтальонов работает в почтовом отделении?
- На каком участке количество экземпляров подписных изданий максимально?
- Каков средний срок подписки по каждому изданию?

Необходимо развернуть PostgreSQL локально:

- Написать запросы для создания таблиц из предыдущей лабораторной работы
- Заполнить тестовыми данными: 5-10 строк на каждую таблицу, обязательно наличие связи между ними, данные приближены к реальности.
- Написать запросы к БД, отвечающие на вопросы из предыдущей лабораторной работы
- Исходный код выложить на www.db-fiddle.com для проверки работоспособности
- Исходный код в виде .sql файла загрузить в виде PR в репо

Выполнение работы.

Создание базы данных:

Создана база данных «Delivery»

Create - Database

General Definition Security Parameters Advanced SQL

Database

Delivery

Owner

postgres

Comment

Close Reset Save

Рисунок 1 создание базы данных

Написаны запросы для создания и заполнения таблиц (см. приложение Б).

```

1 CREATE TABLE postman (
2     postmanid SERIAL PRIMARY KEY,
3     firstname TEXT NOT NULL,
4     middlename TEXT,
5     lastname TEXT
6 );
7
8 INSERT INTO postman (firstname, middlename, lastname) VALUES
9     ('Рубеус', NULL, 'Хагрид'),
10    ('Букля', NULL, NULL),
11    ('Альбус', 'Персиваль', 'Дамблдор'),
12    ('Миневра', NULL, 'МакГонагалл'),
13    ('Северус', NULL, 'Снегг');|

```

```

1 CREATE TABLE district (
2     districtid SERIAL PRIMARY KEY,
3     postmanid INTEGER NOT NULL,
4     FOREIGN KEY (postmanid) REFERENCES postman(postmanid) ON DELETE CASCADE,
5     name TEXT NOT NULL
6 );
7
8 INSERT INTO district (postmanid, name) VALUES
9     (3, 'Хогвартс'),
10    (4, 'Косой Переулок'),
11    (1, 'Хогсмид'),
12    (5, 'Азкабан'),
13    (1, 'Литтл Уингин'),
14    (2, 'Оттери-Сент-Кэчпоул');

```

```

1 CREATE TABLE house (
2     houseid SERIAL PRIMARY KEY,
3     districtid INTEGER NOT NULL,
4     address TEXT NOT NULL,
5     FOREIGN KEY (districtid) REFERENCES district(districtid) ON DELETE CASCADE
6 );
7
8 INSERT INTO house (districtid, address) VALUES
9     (3, 'Хогсмид, Три Метлы'),
10    (3, 'Хогсмид, Волшебные палочки от Олливандера'),
11    (5, 'Литтл Уингин, Тисовая улица, дом 4'),
12    (4, 'Азкабан, 7 камера'),
13    (6, 'Оттери-Сент-Кэчпоул, Нора'),
14    (2, 'Косой Переулок, Гринготтс'),
15    (2, 'Косой Переулок, Лавка Олливандера'),
16    (1, 'Хогвартс, Башня Гриффиндор');

```

```

1 CREATE TABLE subscriber (
2     subscriberid SERIAL PRIMARY KEY,
3     houseid INTEGER NOT NULL,
4     firstname TEXT NOT NULL,
5     middlename TEXT,
6     lastname TEXT,
7     FOREIGN KEY (houseid) REFERENCES house(houseid) ON DELETE CASCADE
8 );
9
10 INSERT INTO subscriber (houseid, firstname, middlename, lastname) VALUES
11    (1, 'Мадам', NULL, 'Розмерта'),
12    (2, 'Гаррик', NULL, 'Олливандер'),
13    (3, 'Гарри', 'Джеймс', 'Поттер'),
14    (4, 'Сириус', NULL, 'Блэк'),
15    (5, 'Молли', NULL, 'Уизли'),
16    (6, 'Гринготт', NULL, NULL),
17    (7, 'Джервейс', NULL, 'Олливандер'),
18    (8, 'Гермиона', NULL, 'Грейнджер'),
19    (8, 'Рон', NULL, 'Уизли');

```

```

1 CREATE TABLE publication (
2     publicationid SERIAL PRIMARY KEY,
3     index INTEGER NOT NULL,
4     title TEXT NOT NULL,
5     price NUMERIC(10,2)
6 );
7
8 INSERT INTO publication (index, title, price) VALUES
9     (93, 'Загадки Темного Искусства', 10.49),
10    (89, 'Энциклопедия Зельеварения', 13.79),
11    (88, 'Дейли Профет', 10.99),
12    (96, 'Тайные Существа и Где Они Обитают', 8.99),
13    (83, 'Заклинания для Начинающих', 6.99),
14    (84, 'Колдовство в Повседневной Жизни', 9.99),
15    (91, 'Путеводитель по Магическим Местам', 12.99);

```

```

1 CREATE TABLE subscription (
2     subscriptionid SERIAL PRIMARY KEY,
3     publicationid INTEGER NOT NULL,
4     subscriberid INTEGER NOT NULL,
5     startdate DATE NOT NULL,
6     duration INTEGER NOT NULL,
7     FOREIGN KEY (publicationid) REFERENCES publication(publicationid) ON DELETE CASCADE,
8     FOREIGN KEY (subscriberid) REFERENCES subscriber(subscriberid) ON DELETE CASCADE
9 );
10
11 INSERT INTO subscription (publicationid, subscriberid, startdate, duration) VALUES
12     (3, 6, '2022-11-01', 365),
13     (6, 5, '2023-03-20', 300),
14     (1, 4, '2020-06-15', 90),
15     (4, 1, '2022-04-12', 1000),
16     (3, 2, '2023-01-01', 365),
17     (1, 3, '2023-10-10', 180),
18     (2, 8, '2023-08-21', 90),
19     (5, 9, '2023-03-15', 120),
20     (7, 7, '2023-05-13', 150),
21     (3, 3, '2023-10-09', 90);

```

Запросы к БД:

Запросы к БД так же можно увидеть в приложении В.

1. Определить наименование и количество экземпляров всех изданий, получаемых отделением связи.

```
Select Title From publication
```

	title character varying (50)
1	Загадки Темного Искусства
2	Энциклопедия Зельеварения
3	Дейли Профет
4	Тайные Существа и Где Они Обитают
5	Заклинания для Начинающих
6	Колдовство в Повседневной Жизни
7	Путеводитель по Магическим Местам

2. По заданному адресу определить фамилию почтальона, обслуживающего подписчика.

```

Select lastname From postman
where postmanid = (
    Select postmanid
    From district
    Where districtid = (
        Select districtid
        From house
        Where address = 'Хогсмид, Три Метлы'
    )
)

```

	lastname character varying (50) 🔒
1	Хагрид

3. Какие газеты выписывает гражданин с указанной фамилией, именем, отчеством?

```
Select title
From publication
    Inner join subscription On subscription.publicationid =
publication.publicationid
    Inner join subscriber On subscriber.subscriberid =
subscription.subscriberid
Where
    firstname = 'Гарри' AND
    middlename = 'Джеймс' AND
    lastname = 'Поттер'
```

	title character varying (50) 🔒
1	Загадки Темного Искусства
2	Дейли Профет

4. Сколько почтальонов работает в почтовом отделении?

```
Select Count(*) From postman
```

	count bigint 🔒
1	5

5. На каком участке количество экземпляров подписных изданий максимально?

```
WITH MaxSubscriptionCount AS (
    SELECT MAX(subscription_count) AS max_count
    FROM (
        SELECT COUNT(*) AS subscription_count
        FROM district d
        JOIN house h ON d.districtid = h.districtid
        JOIN subscriber s ON h.houseid = s.houseid
        JOIN subscription sub ON s.subscriberid =
sub.subscriberid
        GROUP BY d.name
    ) AS counts
)
SELECT d.name AS district_name, COUNT(*) AS subscription_count
FROM district d
JOIN house h ON d.districtid = h.districtid
```

```

JOIN subscriber s ON h.houseid = s.houseid
JOIN subscription sub ON s.subscriberid = sub.subscriberid
GROUP BY d.name
HAVING COUNT(*) = (SELECT max_count FROM MaxSubscriptionCount);

```

	district_name character varying (50)	subscription_count bigint
1	Хогсмид	2
2	Литтл Уингин	2
3	Косой Переулок	2
4	Хогвартс	2

6. Каков средний срок подписки по каждому изданию?

Для всех существующих подписок:

```
Select round(avg(duration), 2) From subscription
```

	round numeric
1	275.00

Только для действующих подписок:

```
Select round(avg(duration), 2) From subscription
WHERE startdate + interval '1 day' * duration >= NOW();
```

	round numeric
1	341.43

Вывод.

В ходе выполнения работы была развернуто локально PostgreSQL, написаны запросы для создания и заполнения таблицы в соответствии со структурой БД, написаны запросы к БД, отвечающие на вопросы задания.

ПРИЛОЖЕНИЕ А

PullRequest: <https://github.com/moevm/sql-2023-1303/pull/23>
db-fiddle : <https://www.db-fiddle.com/f/qeQNArneSPKg99quqKmPZz/0>

ПРИЛОЖЕНИЕ Б

```
--почтальон
CREATE TABLE postman (
    postmanid SERIAL PRIMARY KEY,
    firstname TEXT NOT NULL,
    middlename TEXT,
    lastname TEXT
);

--район
CREATE TABLE district (
    districtid SERIAL PRIMARY KEY,
    postmanid INTEGER NOT NULL,
    FOREIGN KEY (postmanid) REFERENCES postman(postmanid) ON DELETE
    CASCADE,
    name TEXT NOT NULL
);

--дом
CREATE TABLE house (
    houseid SERIAL PRIMARY KEY,
    districtid INTEGER NOT NULL,
    address TEXT NOT NULL,
    FOREIGN KEY (districtid) REFERENCES district(districtid) ON DELETE
    CASCADE
);

--подписчик
CREATE TABLE subscriber (
    subscriberid SERIAL PRIMARY KEY,
    houseid INTEGER NOT NULL,
    firstname TEXT NOT NULL,
    middlename TEXT,
    lastname TEXT,
    FOREIGN KEY (houseid) REFERENCES house(houseid) ON DELETE CASCADE
);

--публикация
CREATE TABLE publication (
    publicationid SERIAL PRIMARY KEY,
    index INTEGER NOT NULL,
    title TEXT NOT NULL,
    price NUMERIC(10,2)
);

--подписка
CREATE TABLE subscription (
    subscriptionid SERIAL PRIMARY KEY,
    publicationid INTEGER NOT NULL,
    subscriberid INTEGER NOT NULL,
    startdate DATE NOT NULL,
    duration INTEGER NOT NULL,
    FOREIGN KEY (publicationid) REFERENCES publication(publicationid)
    ON DELETE CASCADE,
    FOREIGN KEY (subscriberid) REFERENCES subscriber(subscriberid) ON
    DELETE CASCADE
);
```

```

INSERT INTO postman (firstname, middlename, lastname) VALUES
    ('Рубеус', NULL, 'Хагрид'),
    ('Букля', NULL, NULL),
    ('Альбус', 'Персиваль', 'Дамблдор'),
    ('Миневра', NULL, 'МакГонагалл'),
    ('Северус', NULL, 'Снегг');

INSERT INTO district (postmanid, name) VALUES
    (3, 'Хогвартс'),
    (4, 'Косой Переулок'),
    (1, 'Хогсмид'),
    (5, 'Азкабан'),
    (1, 'Литтл Уингин'),
    (2, 'Оттери-Сент-Кэчпоул');

INSERT INTO house (districtid, address) VALUES
    (3, 'Хогсмид, Три Метлы'),
    (3, 'Хогсмид, Волшебные палочки от Олливандера'),
    (5, 'Литтл Уингин, Тисовая улица, дом 4'),
    (4, 'Азкабан, 7 камера'),
    (6, 'Оттери-Сент-Кэчпоул, Нора'),
    (2, 'Косой Переулок, Гринготтс'),
    (2, 'Косой Переулок, Лавка Олливандера'),
    (1, 'Хогвартс, Башня Гриффиндор');

INSERT INTO subscriber (houseid, firstname, middlename, lastname)
VALUES
    (1, 'Мадам', NULL, 'Розмерта'),
    (2, 'Гаррик', NULL, 'Олливандер'),
    (3, 'Гарри', 'Джеймс', 'Поттер'),
    (4, 'Сириус', NULL, 'Блэк'),
    (5, 'Молли', NULL, 'Уизли'),
    (6, 'Гринготт', NULL, NULL),
    (7, 'Джервейс', NULL, 'Олливандер'),
    (8, 'Гермиона', NULL, 'Грейнджер'),
    (8, 'Рон', NULL, 'Уизли');

INSERT INTO publication (index, title, price) VALUES
    (93, 'Загадки Темного Искусства', 10.49),
    (89, 'Энциклопедия Зельеварения', 13.79),
    (88, 'Дейли Профет', 10.99),
    (96, 'Тайные Существа и Где Они Обитают', 8.99),
    (83, 'Заклинания для Начинающих', 6.99),
    (84, 'Колдовство в Повседневной Жизни', 9.99),
    (91, 'Путеводитель по Магическим Местам', 12.99);

INSERT INTO subscription (publicationid, subscriberid, startdate,
duration) VALUES
    (3, 6, '2022-11-01', 365),
    (6, 5, '2023-03-20', 300),
    (1, 4, '2020-06-15', 90),
    (4, 1, '2022-04-12', 1000),
    (3, 2, '2023-01-01', 365),
    (1, 3, '2023-10-10', 180),
    (2, 8, '2023-08-21', 90),

```

```
(5, 9, '2023-03-15', 120),  
(7, 7, '2023-05-13', 150),  
(3, 3, '2023-10-09', 90);
```

ПРИЛОЖЕНИЕ В

--1.Определить наименование и количество экземпляров всех изданий, получаемых отделением связи.

```
SELECT title FROM publication
```

--2.По заданному адресу определить фамилию почтальона, обслуживающего подписчика.

```
SELECT lastname FROM postman
where postmanid = (
    SELECT postmanid
    FROM district
    WHERE districtid = (
        SELECT districtid
        FROM house
        WHERE address = 'Хогсмид, Три Метлы'
    )
)
```

--3.Какие газеты выписывает гражданин с указанной фамилией, именем, отчеством?

```
SELECT title
FROM publication
    INNER JOIN subscription ON subscription.publicationid =
publication.publicationid
    INNER JOIN subscriber ON subscriber.subscriberid =
subscription.subscriberid
WHERE
    firstname = 'Гарри' AND
    middlename = 'Джеймс' AND
    lastname = 'Поттер'
```

--4.Сколько почтальонов работает в почтовом отделении?

```
SELECT Count(*) FROM postman
```

--5.На каком участке количество экземпляров подписных изданий максимально?

```
WITH MaxSubscriptionCount AS (
    SELECT MAX(subscription_count) AS max_count
    FROM (
        SELECT COUNT(*) AS subscription_count
        FROM district d
        JOIN house h ON d.districtid = h.districtid
        JOIN subscriber s ON h.houseid = s.houseid
        JOIN subscription sub ON s.subscriberid = sub.subscriberid
        GROUP BY d.name
    ) AS counts
)
SELECT d.name AS district_name, COUNT(*) AS subscription_count
FROM district d
JOIN house h ON d.districtid = h.districtid
JOIN subscriber s ON h.houseid = s.houseid
JOIN subscription sub ON s.subscriberid = sub.subscriberid
GROUP BY d.name
HAVING COUNT(*) = (SELECT max_count FROM MaxSubscriptionCount);
```

--6.Каков средний срок подписки по каждому изданию?

--Для всех существующих подписок:

```
SELECT round(avg(duration), 2) FROM subscription
```

```
--6.Каков средний срок подписки по каждому изданию?
```

```
--Только для действующих подписок:
```

```
SELECT round(avg(duration), 2) FROM subscription
```

```
WHERE startdate + interval '1 day' * duration >= NOW();
```