

**Міністерство освіти та науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет прикладної математики
Кафедра системного програмування і спеціалізованих комп’ютерних
систем**

Розрахунково-графічна робота
з дисципліни
“Бази даних та засоби управління”
**Тема: «Створення додатку бази даних, орієнтованого на взаємодію з
СУБД PostgreSQL»**

Виконав: Тимченко В.М.
Студент групи КВ-23
Перевірів(ла):
Телеграм: @Vova_Tym

Київ 2024

Загальне завдання

1. Реалізувати функції перегляду, внесення, редагування та вилучення даних у таблицях бази даних, створених у лабораторній роботі №1, засобами консольного інтерфейсу.
2. Передбачити автоматичне пакетне генерування «рандомізованих» даних у базі.
3. Забезпечити реалізацію пошуку за декількома атрибутами з двох та більше сутностей одночасно: для числових атрибутів – у рамках діапазону, для рядкових – як шаблон функції LIKE оператора SELECT SQL, для логічного типу – значення True/False, для дат – у рамках діапазону дат.
4. Програмний код виконати згідно шаблону MVC (модель-подання-контролер).

Мова програмування: Python.

Використані бібліотеки: pycorpg2, time, datetime, sys, os.

URL репозиторію:

https://github.com/VolodymyrTymchenko/Lab_Work_Databases

Діаграму сутність-зв'язок

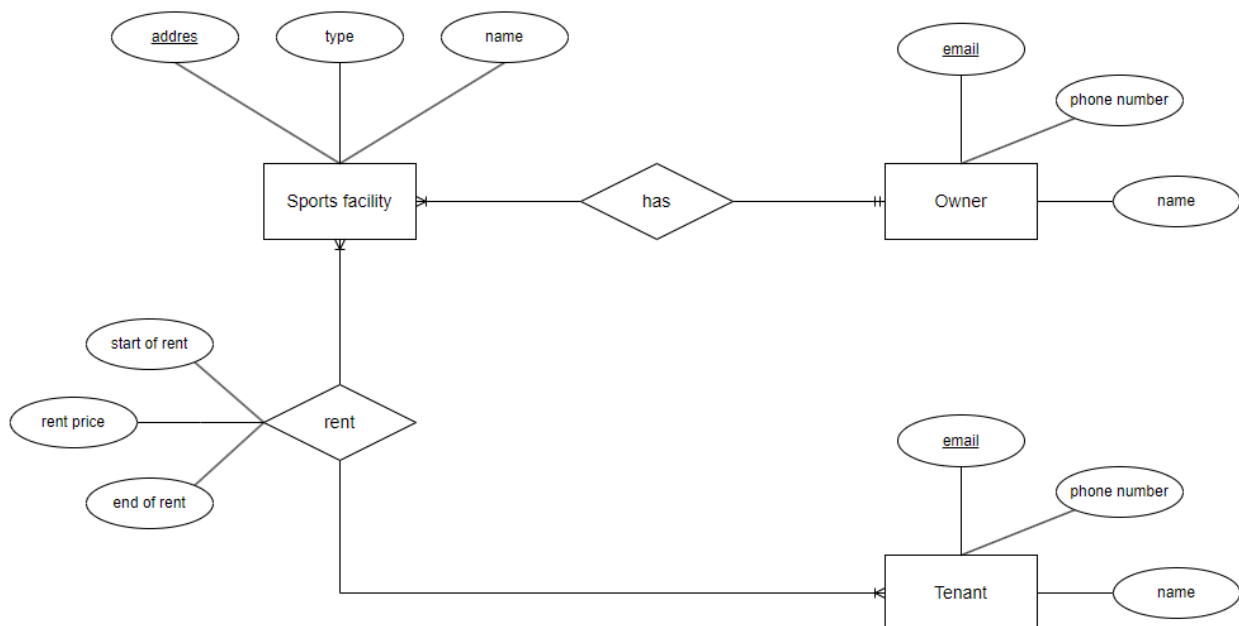
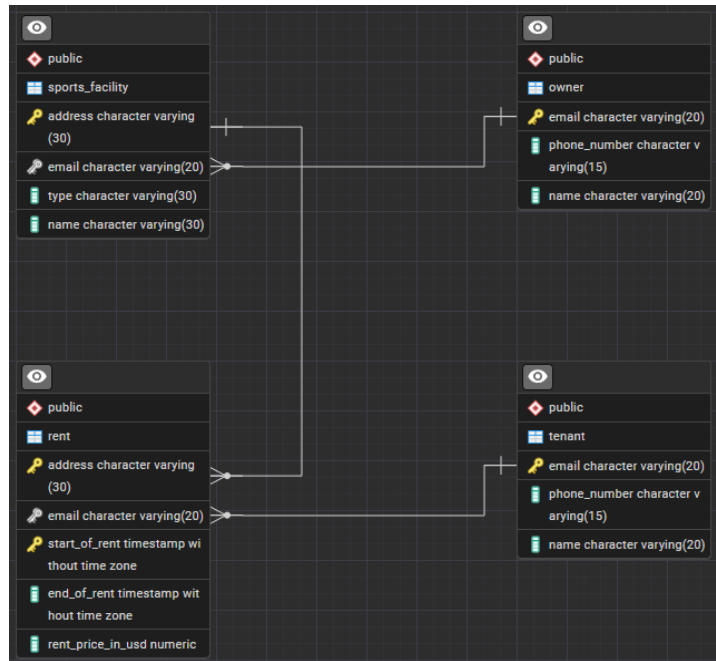


Схема БД



Опис БД

Спортивний об'єкт має одного власника, при цьому власник може мати декілька спортивних об'єктів. Орендар може орендувати багато спортивних об'єктів в різний час, при цьому один спортивний об'єкт можуть орендувати багато орендарів в різний час.

Опис меню програми

```
Tables:  
  
1. Owner  
2. Rent  
3. Sports_facility  
4. Tenant  
5. Interesting requests  
0. Quit  
Enter your choice: █
```

Пункти 1-4 відкривають підменю для вибору функцій роботи з таблицею:

```
1. Add row  
2. Show all rows  
3. Update row  
4. Delete row  
5. Generate random rows  
0. Quit  
Enter your choice: █
```

Пункт 5 відкриває підменю для вибору “цікавих” запитів:

```
1. Rating of the total income of the owners  
2. Rating of the total number of leased objects of tenants  
0. Quit  
Enter your choice: █
```

Пункт 0 – вихід з програми

Завдання №1

На прикладі батьківської таблиці **owner** та дочірньої **sports_facility**:

Вилучення запису таблиці **owner** та виведення вмісту таблиці **sports_facility** після цього вилучення:

address	email	type	name
Lviv, 73 Hrushevskogo st.	paul.987@mail.com	swimming pool	LvivAqua
Kyiv, 45 Shevchenko st.	ann.qwerty@ukr.net	swimming pool	AquaSport
Poltava, 574 Zalisna st.	paul.987@mail.com	rugby field	PoltavaSport
Kovel, 81c Abrikosova st.	ann.qwerty@ukr.net	rugby field	Kovellife
Uman, 343b Lipneva st.	denis.58021@ukr.net	football field	UmanLife
Nizhyn, 482b Hrushevskogo st.	tom.84923@gmail.com	gym	NizhynSport

```
Enter owner email: denis.58021@ukr.net
Owner: delted successfully!
0. Quit
Enter your choice: █
```

sports_facility			
address	email	type	name
Lviv, 73 Hrushevskogo st.	paul.987@mail.com	swimming pool	LvivAqua
Kyiv, 45 Shevchenko st.	ann.qwerty@ukr.net	swimming pool	AquaSport
Poltava, 574 Zalisna st.	paul.987@mail.com	rugby field	PoltavaSport
Kovel, 81c Abrikosova st.	ann.qwerty@ukr.net	rugby field	Kovellife
Nizhyn, 482b Hrushevskogo st.	tom.84923@gmail.com	gym	NizhynSport

Вставка запису в таблицю **sports_facility** та виведення повідомлення про її неможливість, якщо в таблиці **owner** немає відповідного запису:

owner		
email	phone_number	name
ann.qwerty@ukr.net	+380 95 4601593	Ann
paul.987@mail.com	+380 50 3791306	Paul
tom.84923@gmail.com	+380 96 9552390	Tom
jack.71578@ukr.net	+380 08 6065134	Jack

```
Enter sports_facility address: Lviv, 3 Hrushevskogo st.
Enter sports_facility email: john.268676@ukr.net
Enter sports_facility type: gym
Enter sports_facility name: LvivSport
Error: Sports_facility: Owner email is not exist
```

Завдання №2

Генерованих рядків таблиць:

Owner:

```
Enter the number of rows: 100000
Owner: generated successfully!
0. Quit
Enter your choice: █
```

mary.030089@ukr.net +380 74 3743774	Mary
eddy.312806@ukr.net +380 88 7384555	Eddy
alice.917414@ukr.net +380 84 5819035	Alice
alice.245280@ukr.net +380 96 3634818	Alice
john.603324@ukr.net +380 55 4857166	John
bob.292645@ukr.net +380 76 9097917	Bob
jack.092391@ukr.net +380 60 8819051	Jack
john.974752@ukr.net +380 65 2109513	John

Tenant:

```
Enter the number of rows: 100000
Tenant: generated successfully!
0. Quit
Enter your choice: █
```

ann.183834@ukr.net +380 31 7627166	Ann
antony.002761@ukr.net +380 01 1355715	Antony
antony.513081@ukr.net +380 39 4565663	Antony
garry.076796@ukr.net +380 01 1905823	Garry
garry.937939@ukr.net +380 81 3930851	Garry
fred.301451@ukr.net +380 04 3334793	Fred
john.659561@ukr.net +380 90 4973648	John
kevin.755525@ukr.net +380 51 7116445	Kevin

Sports_facility:

```
Enter the number of rows: 100000
Sports_facility: generated successfully!
0. Quit
Enter your choice: █
```

Kyiv, 595e Azovska st. bob.076332@ukr.net	rugby field	KyivLife
Kovel, 393e Dachna st. tom.289829@ukr.net	basketball court	KovelFitness
Nizhyn, 918d Dachna st. linda.309972@ukr.net	volleyball court	NizhynLife
Irpın, 395 Grecheskya st. alice.878326@ukr.net	volleyball court	IrpınFitness
Odesa, 780 Baikova st. eddy.463079@ukr.net	ice skating rink	OdesaFitness
Obukhov, 556c Vyshneva st. bob.092325@ukr.net	volleyball court	ObukhovFitness
Brovary, 455b Berezova st. eddy.318197@ukr.net	football field	BrovaryFitness

Rent:

```
Enter the number of rows: 10000
Rent: generated successfully!
0. Quit
Enter your choice: █
```

	Lutsk, 900b Lipneva st.		garry.632774@ukr.net		2021-09-19 11:38:18		2021-09-21 02:38:18		815.89	
	Uman, 224e Shevchenko st.		garry.632774@ukr.net		2021-02-26 05:51:39		2021-02-26 13:51:39		807.44	
	Lutsk, 674c Druzhby st.		garry.632774@ukr.net		2024-01-11 22:23:52		2024-01-13 10:23:52		734.61	
	Uman, 529 Azovska st.		garry.632774@ukr.net		2020-05-14 07:56:25		2020-05-15 09:56:25		705.74	
	Lutsk, 201 Azovska st.		garry.632774@ukr.net		2023-05-31 17:22:05		2023-06-02 01:22:05		662.55	
	Dnipro, 282c Galitska st.		garry.632774@ukr.net		2022-06-16 13:25:05		2022-06-17 11:25:05		445.98	
	Obukhov, 240 Azovska st.		garry.632774@ukr.net		2021-10-28 14:28:41		2021-10-29 04:28:41		162.56	
	Dnipro, 318d Druzhby st.		garry.632774@ukr.net		2021-09-08 02:53:51		2021-09-10 00:53:51		798.10	

SQL-запити для генерування рядків таблиць:

Owner:

```
curs.execute("""
...INSERT INTO owner (email, phone_number, name)
...SELECT email1, phone_number1, name1
...FROM
... (
... ..SELECT (Lower(name1) || '.' || to_char(trunc(random()*999999)::int, 'FM000000') || '@ukr.net')::varchar(20) AS
... ..email1,
... ..('+380' || to_char(trunc(random()*99)::int, '00') || to_char(trunc(random()*9999999)::int, '0000000'))
... ..::varchar(20) AS phone_number1,
... ..(name1)::varchar(20)
... ..FROM
... .. (
... .. ..SELECT (ARRAY['Mary', 'Eddy', 'Denis', 'Jack', 'Garry', 'Alice', 'Bob', 'Tom', 'Linda', 'John'])[trunc
... .. ..(random()*10+1)::int] AS name1
... .. ..FROM generate_series(1,%s)
... .. ..) subquery1
... ..) subquery2
... ON CONFLICT(email) DO NOTHING;
""", (row_count,))
```

Tenant:

```
curs.execute("""
INSERT INTO tenant (email, phone_number, name)
SELECT email1, phone_number1, name1
FROM
(
SELECT (Lower(name1) || '-' || to_char(trunc(random()*999999)::int, 'FM000000') || '@ukr.net')::varchar(20) AS
email1,
('+380' || to_char(trunc(random()*99)::int, '00') || to_char(trunc(random()*9999999)::int, '0000000'))
::varchar(20) AS phone_number1,
(name1)::varchar(20)
FROM
(
SELECT (ARRAY['Ann', 'John', 'Paul', 'Max', 'Tom', 'Carl', 'Antony', 'Garry', 'Kevin', 'Fred'])[trunc(random()
)* 10 + 1)::int] AS name1
FROM generate_series(1, %s)
) subquery1
) subquery2
ON CONFLICT(email) DO NOTHING;
""", (row_count,))
```

Sports_facility:

```
curs.execute("""
INSERT INTO sports_facility (address, email, type, name)
SELECT address1, email1, type1, name1
FROM
(
SELECT (city || '-' || (trunc(random()*999)::int)::text || letter || '-' || street || '-' || 'st.')
::varchar(32) AS address1,
email1, type1::varchar(30), (city || name2)::varchar(30) AS name1
FROM
(
SELECT (ARRAY['Kyiv', 'Lviv', 'Odesa', 'Kharkiv', 'Dnipro', 'Poltava', 'Sumy', 'Rivne', 'Lutsk', 'Uman',
'Brovary', 'Irpine', 'Nizhyn', 'Kovel', 'Obukhov'])[trunc(random()* 15 + 1)::int] AS city,
(ARRAY['', 'a', 'b', 'c', 'd', 'e'])[trunc(random()* 6 + 1)::int] AS letter,
(ARRAY['Shevchenko', 'Hrushevskogo', 'Grecheskya', 'Vyshneva', 'Parkova', 'Azovska', 'Abrikosova',
'Baikova', 'Berezova', 'Verbova', 'Galitska', 'Dachna', 'Druzhby', 'Zalisna', 'Lipneva'])[trunc
(random()* 15 + 1)::int] AS street,
(ARRAY['swimming pool', 'tennis court', 'football field', 'golf course', 'basketball court',
'volleyball court', 'rugby field', 'ice skating rink', 'gym'])[trunc(random()* 9 + 1)::int] AS
type1,
(ARRAY['Sport', 'Fitness', 'Life'])[trunc(random()* 3 + 1)::int] AS name2,
email AS email1
FROM generate_series(1, %(count)s),
(
SELECT email
FROM owner
ORDER BY random()
) subquery1
) subquery2
LIMIT %(count)s
) subquery3
ON CONFLICT(address) DO NOTHING;
""", {'count': row_count})
```


Rent:

```
curs.execute("""
INSERT INTO rent (address, email, start_of_rent, end_of_rent, rent_price_in_us)
SELECT address1, email1, start_of_rent1, end_of_rent1, rent_price_in_us1
FROM
(
SELECT address1, email1, random_timestamp AS start_of_rent1, random_timestamp + (floor(random() * 48)::int || '
hours')::INTERVAL AS end_of_rent1, (random() * 999)::numeric(5, 2) AS rent_price_in_us1
FROM
(
SELECT date_trunc('second', TIMESTAMP '2020-01-01 00:00:00' + (random() * (TIMESTAMP '2024-12-07 23:59:59'
- TIMESTAMP '2020-01-01 00:00:00'))) AS random_timestamp,
email1, address1
FROM generate_series(1, %(count)s),
(
SELECT
t1.email AS email1, t2.address AS address1
FROM
(SELECT email FROM tenant LIMIT 1000 OFFSET floor(random() * 99000)) t1
CROSS JOIN
(SELECT address FROM sports_facility LIMIT 1000 OFFSET floor(random() * 99000)) t2
) subquery1
)
LIMIT %(count)s
) subquery2
ON CONFLICT(address, start_of_rent) DO NOTHING;
""", {'count': row_count})
```

Завдання №3

введення пошукового запиту та результати виконання
“цікавих” запитів:

```
Enter the number of rows: 5
Enter the minimum sum: 5000

-----
|          Rating of the total income of the owners          |
-----
|   rating   |   name   |   email   | total_income |
-----
|           1|          Eddy| eddy.100058@ukr.net|    59752.15|
-----
|           2|          Linda| linda.768021@ukr.net|    58247.08|
-----
|           3|          John| john.813542@ukr.net|    58018.37|
-----
|           4|          John| john.400845@ukr.net|    57886.69|
-----
|           5|          Garry| garry.065080@ukr.net|    57641.34|
-----
Total query runtime: 85 msec.
```

```
Enter the number of rows: 5
Enter start timestamp: 2022-09-09 00:00:00
Enter end timestamp: 2024-07-07 00:00:00

-----
|          Rating of the total number of leased objects of tenants          |
-----
|   rating   |   name   |   email   | total_count |
-----
|           1|          Fred| fred.443872@ukr.net|        406|
-----
|           2|          Paul| paul.712880@ukr.net|        403|
-----
|           3|          Max| max.690299@ukr.net|        399|
-----
|           4|          Paul| paul.642750@ukr.net|        396|
-----
|           4|          Paul| paul.594873@ukr.net|        396|
-----
Total query runtime: 26 msec.
```

SQL-запити для пошуку із зазначеними початковими параметрами:

```
curs.execute("""
...SELECT DENSE_RANK() OVER (ORDER BY SUM(r.rent_price_in_us) DESC) AS rating, o.name, s.email, SUM(r.
...rent_price_in_us) AS total_income
...FROM owner o, rent r, sports_facility s
...WHERE o.email = s.email AND r.address = s.address
...GROUP BY s.email, o.name
...HAVING SUM(r.rent_price_in_us) > %s
...ORDER BY total_income DESC
...LIMIT %s
""", (min_sum, row_count))
```

```
curs.execute("""
...SELECT DENSE_RANK() OVER (ORDER BY COUNT(r.address) DESC) AS rating, t.name, r.email, COUNT(r.address) AS
...total_count
...FROM tenant t, rent r
...WHERE r.email = t.email AND r.start_of_rent BETWEEN %s AND %s
...GROUP BY r.email, t.name
...ORDER BY total_count DESC
...LIMIT %s
""", (start_timestamp, end_timestamp, row_count))
```

Завдання №4

методи “Model”:

```
1  import psycopg2
2  import time
3
4  class Model:
5  > ... def __init__(self): ...
6
7
8  # table 'owner':
9
10 > ... def owner_add(self, email, phone_number, name): ...
11
12
13
14
15 > ... def owner_get_all_rows(self): ...
16
17
18
19
20 > ... def owner_update(self, old_email, email, phone_number, name): ...
21
22
23
24
25 > ... def owner_delete(self, email): ...
26
27
28
29
30 > ... def owner_random_generate(self, row_count): ...
31
32
33
34
35 > ... def owner_row_search(self, email): ...
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55 # table 'tenant':
56
57 > ... def tenant_add(self, email, phone_number, name): ...
58
59
60
61
62 > ... def tenant_get_all_rows(self): ...
63
64
65
66
67 > ... def tenant_update(self, old_email, email, phone_number, name): ...
68
69
70
71
72 > ... def tenant_delete(self, email): ...
73
74
75
76
77 > ... def tenant_random_generate(self, row_count): ...
78
79
80
81
82 > ... def tenant_row_search(self, email): ...
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
```

```

102 # table 'sports_facility':
103
104 > ....def sports_facility_add(self, address, email, type, name):...
108
109 > ....def sports_facility_get_all_rows(self):...
113
114 > ....def sports_facility_update(self, old_address, address, email, type, name):...
118
119 > ....def sports_facility_delete(self, address):...
123
124 > ....def sports_facility_random_generate(self, row_count):...
154
155 > ....def sports_facility_row_search(self, address):...
159
160 # table 'rent':
161
162 > ....def rent_add(self, address, email, start_of_rent, end_of_rent, rent_price_in_us):...
166
167 > ....def rent_get_all_rows(self):...
171
172 > ....def rent_update(self, old_address, old_start_of_rent, address, email, start_of_rent, end_of_rent, rent_price_in_us):...
176
177 > ....def rent_delete(self, address, start_of_rent):...
181
182 > ....def rent_random_generate(self, row_count):...
209
210 > ....def rent_row_search(self, address, start_of_rent):...
214
215 # interesting requests:
216
217 > ....def owner_total_income_rating(self, min_sum, row_count):...
231
232 > ....def tenant_total_count_rented_facilities_rating(self, start_timestamp, end_timestamp, row_count):...
245

```

<table_name>_add() – додовання рядка в таблицю.

<table_name>_get_all_rows() – отримання всіх рядків таблиці.

<table_name>_update() – оновлення рядка таблиці.

<table_name>_delete() – видалення рядка таблиці.

<table_name>_random_generate() – генерування рядків таблиці заповнених випадковими даними.

<table_name>_search() – пошук рядка в таблиці.

owner_total_income_rating() – виведення рейтингу власників спортивних об'єктів за їх доходами, які перевищують задану суму.

tenant_total_count_rented_facilities_rating() – виведення рейтингу орендарів за кількістю орендованих спортивних об'єктів, в заданому часовому інтервалі.