

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
"КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО"

Кафедра програмного забезпечення комп'ютерних систем

Лабораторна робота №2
із дисципліни «Бази даних»
на тему «Створення додатку бази даних, орієнтованого на взаємодію з
СУБД PostgreSQL»

Виконав
студент 2 курсу
групи КП-92

Фенченко Ігор Юрійович

Перевірів
“ ____ ” “ ____ ” 20__ р.
к. т. н. доцент

Петрашенко Андрій Васильович

Київ 2020

Мета роботи: здобуття вмінь програмування прикладних додатків баз даних PostgreSQL.

Завдання:

1. Реалізувати функції внесення, редагування та видалення 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).

URL репозиторію з вихідним кодом:

https://github.com/bujhmt/DB_labs/tree/master/lab2

Завдання 1

- ілюстрації обробки виняткових ситуацій (помилки) при введенні/вилучення даних;

```
try:
    if '' in value: raise Exception

    # my ORM types:
    if type == 'string':
        self.__varSetUpString(key, value)

    if type == 'number':
        self.__varSetUpNumber(key, value)

    if type == 'float':
        self.__varSetUpFloat(key, value)

    if type == 'money':
        self.__varSetUpMoney(key, value)

    if type == 'date':
        self.__varSetUpDate(key, value)

    if len(args) == 0: self.__fillMenu.renameField(key, key + f' ({value})')
except Exception:
    if len(args) == 0: self.__fillMenu.setError(f"ERROR! Incorrect {key} input")
```

```
try:
    client_id = int(input('Enter client_id: '))

    if not (isinstance(client_id, int) and client_id > 0):
        raise Exception('Invalid input')
```

- ілюстрації валідації даних при введенні користувачем.

```

def __varSetUpDate(self, key: str, value: str):
    datetime.datetime.strptime(value, "%Y-%m-%d")
    exec("self.%s = \"'%s'\"" % (key, value))

def __varSetUpNumber(self, key: str, value: str):
    if not isinstance(int(value), int): raise Exception("Invalid input")
    exec("self.%s = %i" % (key, int(value)))

def __varSetUpFloat(self, key: str, value: str):
    if not isinstance(float(value), float): raise Exception("Invalid input")
    exec("self.%s = %f" % (key, float(value)))

def __varSetUpMoney(self, key: str, value: str):
    if not isinstance(int(value), int): raise Exception("Invalid input")
    exec("self.%s = %i" % (key, math.fabs(int(value))))

def __varSetUpString(self, key: str, value: str):
    if len(value) < 2: raise Exception
    exec("self.%s = \"'%s'\"" % (key, value))

```

Завдання 2

- ілюстрації з фрагментами згенерованих даних таблиць;

	id	name	cost	brand	manufacture_date	manufacturer
56133	56816	bcakonlkblrrgsm	795,00грн.	nakdvernmuwbmsa	2017-01-15	ysycqvuoraxvitp
56134	56817	scdndflnrkqchlj	1NNBSP601,00r	uksydxwxqlwsjfp	2012-06-15	gulsbedimriqdqu
56135	56818	qamblejxftcgsij	788,00грн.	yvpvxfjllxidvot	2000-08-24	hnbjhlyqdglaylf
56136	56819	nurrlaavjpgghpq	1NNBSP916,00r	najjdkunqmuvubu	2015-08-14	mavgtvvogpvnknj
56137	56820	phodofungnnhann	622,00грн.	kdbgetkbitjeuhp	1995-08-25	iurvmgfgmqmntwf
56138	56821	jffxuepcrfumauf	1NNBSP143,00r	dwpgudrgcyfgrdu	2001-03-20	avfgiomkygjkhhc
56139	56822	veydmibhuourthf	647,00грн.	unchyesiqeswhon	2013-09-28	agrwm1nejcflhoa
56140	56823	tdumsedipfmxgoh	1NNBSP345,00r	lrvkftqcgxcphog	1994-06-23	uuspggjwknrmvw
56141	56824	hgqdsibupfhrbop	1NNBSP596,00r	jgklwxppfiugayp	2008-02-22	anfneaohrbdqddq
56142	56825	qgevtensnvedgbv	1NNBSP918,00r	knnjulqwfkygiu	2003-05-19	ibyatimnfjchkob
56143	56826	ytqwovrhgjdarr	518,00грн.	epqhvookuwxhxaa	2007-08-20	tuqodqdobwvxfcx
56144	56827	fppnbchxjeamrrd	1NNBSP807,00r	rsysyrqqytnpgt	2017-06-11	pfgtlgsvppcxkni
56145	56828	jtcabsfboxoqfui	1NNBSP577,00r	rtcmhhdronhiwmh	2011-09-21	ovvueehhgnyxabg
56146	56829	takkxobdbegxmbj	57,00грн.	cvlkyxgcaweixud	2020-02-12	gwrribkkvaysuepf
56147	56830	vgmsmxltuijpcs	256,00грн.	kqnadfwbjknkaxt	2002-04-17	oyvtbqlrwyoiwd
56148	56831	ocjxbisuibncysc	1NNBSP303,00r	txnfaedhjnfxil	2006-07-10	yhrhrjgcpxlevul
56149	56832	hihljratdfahewk	1NNBSP771,00r	klinkhkmfpowxy	2006-04-10	jggfqioggsrtre
56150	56833	focwdbwlxxientu	117,00грн.	ytjpcpnxjsfcgdh	2003-08-27	dfgmmqeowyxhcpj
56151	56834	ropminsanskmbks	1NNBSP434,00r	mimqjdouflvljbn	2002-09-10	vjvtournrqqifxq
56152	56835	kdgfsfplryesmuxl	771,00грн.	jdtbtblcrvbyio	2003-03-10	gcjvigheevbgaxh
56153	56836	llavtukprlkxnit	791,00грн.	qibetnkvrlnoyq	2000-01-21	puaxcxphwhvhybm
56154	56837	rawxuscvitjtyaf	1NNBSP463,00r	amuorkdfvshnkiy	2019-01-24	vaislrqqkojjchb
56155	56838	dmhfajifglvgxvh	82,00грн.	dcngfyxvdywgiys	2018-09-09	dxLutlsrretsfnw
56156	56839	jwkwboxopgxmyc	188,00грн.	ttjqbskbjhedffh	2007-07-21	xpnwxttomjhxmdp
56157	56840	pwatvuvnrnrwkyb	1NNBSP015,00r	gcffddqqrarogtw	2000-08-26	xstqrsrlhftfwml

	id	transaction_date	client_id	taxes_sum
60738	102715	2004-02-15	30702	14,00грн.
60739	102715	2001-01-19	51786	78,00грн.
60740	102716	2016-08-17	28105	38,00грн.
60741	102718	1993-03-25	98615	74,00грн.
60742	102719	2003-01-24	128603	52,00грн.
60743	102721	2010-03-09	27740	87,00грн.
60744	102723	2017-09-27	25879	52,00грн.
60745	102724	2011-01-12	4968	49,00грн.
60746	102726	2001-04-21	14780	92,00грн.
60747	102728	2012-02-11	13141	81,00грн.
60748	102730	1996-08-17	76154	4,00грн.
60749	102732	1991-04-17	107436	84,00грн.
60750	102733	2012-08-18	83878	26,00грн.
60751	102735	2007-01-21	160237	89,00грн.
60752	102737	2014-09-14	107037	79,00грн.
60753	102738	1992-01-17	6057	2,00грн.
60754	102740	2004-05-22	113996	72,00грн.
60755	102742	2009-04-21	76932	71,00грн.
60756	102743	2015-04-12	58879	58,00грн.
60757	102745	2002-07-10	154743	35,00грн.
60758	102747	1995-04-12	72895	94,00грн.
60759	102748	2014-08-21	38687	15,00грн.
60760	102750	2002-05-11	93319	5,00грн.
60761	102752	2016-07-23	48746	7,00грн.
60762	102754	2015-03-20	108869	57,00грн.
60763	102755	1994-04-28	119931	35,00грн.

- копії SQL-запитів, що ілюструють генерацію при визначених вхідних параметрах.

```
try:
    self.db.cursor.execute(f"INSERT INTO \"Order\" (client_id,transaction_date,taxes_sum)"
                           f"SELECT getrandomrow('Client')::int,"
                           f"generatedate()::date,"
                           f"generateint(100)::money "
                           f"FROM generate_series(1, {entitiesNum})")
    self.db.connect.commit()
```

```
try:
    self.db.cursor.execute(f"INSERT INTO \"Client\" (name,birthday_date,email)"
                           f" SELECT generatestring(15),"
                           f"generatedate()::date,"
                           f"concat(generatestring(15)::text, '@gmail.com')"
                           f"FROM generate_series(1, {entitiesNum})")
    self.db.connect.commit()
```

```

self.db.cursor.execute(f"INSERT INTO \"Product\" (name,cost,brand,manufacture_date,\"
                        f\"manufacturer,category_id, order_id) \"
                        f\"SELECT generatestring(15),\"
                        f\"generateint(2000)::money,\"
                        f\"generatestring(15),\"
                        f\"generatedate()::date,\"
                        f\"generatestring(15),\"
                        f\"getrandomrow('Category')::int,\"
                        f\"getrandomrow('Order')::int \"
                        f\"FROM generate_series(1, {entitiesNum})")
self.db.connect.commit()

```

```

try:
    self.db.cursor.execute(f"INSERT INTO \"Category\" (name,type)'
                            f' SELECT generatestring(15),'
                            f'generatestring(15)'
                            f'FROM generate_series(1, {entitiesNum})')
    self.db.connect.commit()

```

Завдання 3

- ілюстрації введення пошукового запиту та результатів виконання запитів;

```

-----Search menu-----
[Search products cost by cost range]
[Search all client orders]
[Search all category products]
[EXIT]
Enter min value: 100
Enter max value: 150

```



```

Elapsed time: 0.0665459s
Rows num: 2029
[NEXT]
[<ID> | Product name | Product cost | Category name]
[<18951> bqknkdapdyastxc 101,00грн. bqhfvsohmcyfvbr]
[<23341> naxlvsqkwkrvfjq 101,00грн. nqojkraxgumvawv]
[<7429> vnavuvnkexgsimo 101,00грн. orxnnlmpnvtahae]
[<16605> vxnctmtbsdghxdf 101,00грн. yrxasvoudwtpona]
[<12791> voimрмаqidqrlhl 101,00грн. ofbmwcayjbmlqjn]
[<11183> fllpxtkbqgecgju 101,00грн. fpqkpmwgfgcqwbby]
[<1399> msbpwafgghmnsdp 101,00грн. wxfdjfpachwifow]
[<3710> wkyklwjseyxawun 101,00грн. ldcfigisxtqdgge]
[<7209> alfplpsftlprybo 101,00грн. iaoggptalfmrxhu]
[<72202> sukimwrylhgxvyy 101,00грн. ekagwjtvwvywrqp]
[Return to prev menu]

```

- копії SQL-запитів, що ілюструють пошук з зазначеними початковими параметрами

```

self.db.cursor.execute(
    f'SELECT p.id, p.name, p.cost, c.name as category from "Product" as p '
    f'inner join "Category" C on C.id = p.category_id '
    f'Where p.cost::numeric::int > {min} and p.cost::numeric::int < {max} '
    f'ORDER BY p.cost::numeric::int LIMIT {args[1]} OFFSET {args[0] * args[1]}')
return self.db.cursor.fetchall()

```

```

try:
    self.db.cursor.execute(
        f'SELECT O.id as OrderId, O.transaction_date, O.taxes_sum from "Order" as O '
        f'INNER JOIN "Client" C on C.id = O.client_id '
        f'WHERE client_id = {client_id}')
    return self.db.cursor.fetchall()

```

```

try:
    self.db.cursor.execute(
        f'SELECT P.id as ProductId, P.name, P.cost, P.manufacturer from "Product" as P '
        f'INNER JOIN "Category" C on C.id = P.category_id '
        f'WHERE category_id = {category_id}')
    return self.db.cursor.fetchall()

```

Завдання 4

- ілюстрації програмного коду з репозиторію Git.

The screenshot displays the GitHub interface for the repository 'bujhmt / DB_labs'. The top navigation bar includes links for Pull requests, Issues, Marketplace, and Explore. The repository name 'bujhmt / DB_labs' is shown with 1 Unwatch, 0 Stars, and 0 Forks. The 'Code' tab is selected, showing the file structure for the 'master' branch at 'DB_labs / lab2 /'. The file list includes:

File	Commit	Time
..	93f63df	31 minutes ago
CUI	category/client models, generate random data	2 days ago
controllers	final	31 minutes ago
models	a lot of...	2 days ago
utils	Prod controller 2	19 days ago
views	final	31 minutes ago
README.md	Update README.md	32 minutes ago
config.json	jsonReader	26 days ago
database.py	pre-fina	2 hours ago
main.py	pre-fina	2 hours ago

Below the file list, a snippet of the README.md file is visible, containing the text: "Створення додатку бази даних, орієнтованого на взаємодію з СУБД PostgreSQL".

The second part of the screenshot shows the 'controllers' directory. The file list includes:

File	Commit	Time
..	93f63df	32 minutes ago
categoryController.py	pre-fina	2 hours ago
clientController.py	pre-fina	2 hours ago
orderController.py	pre-fina	2 hours ago
productController.py	pre-fina	2 hours ago
searchController.py	final	32 minutes ago