



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Системы обработки информации и управления

Отчет по лабораторной работе № 4
«Нечеткие запросы в PostgreSQL»

по дисциплине «Оперативный анализ данных»

Студент ИУ5-51Б
(Группа)

Е.И. Бирюкова
(Подпись, дата) (И.О.Фамилия)

Преподаватель

К.Ю. Маслеников
(Подпись, дата) (И.О.Фамилия)

Москва

2024

Цель работы

Целью работы является:

- изучение подходов к реализации нечетких запросов к реляционным базам данных средствами СУБД PostgreSQL
- получение навыков адаптации реляционных баз данных к реализации нечетких запросов

Полученный вариант задания – Вариант 3

Вы работаете в ломбарде. Вашей задачей является отслеживание финансовой стороны его работы. Деятельность компании организована следующим образом: к вам обращаются различные лица с целью получения денежных средств под залог определенных товаров. У каждого из приходящих к вам клиентов вы запрашиваете фамилию, имя, отчество и другие паспортные данные. После оценивания стоимости принесенного в качестве залога товара вы определяете сумму, которую готовы выдать на руки клиенту, а также свои комиссионные. Кроме того, определяете срок возврата денег. Если клиент согласен, то ваши договоренности фиксируются в виде документа, деньги выдаются клиенту, а товар остается у вас. В случае если в указанный срок не происходит возврата денег, товар переходит в вашу собственность.

Ход лабораторной работы

1. Разработать физическую схему базы данных согласно варианту задания

- Client (Клиент) - Хранит информацию о клиентах, которые обращаются в ломбард.
 - ID (SERIAL, PRIMARY KEY) - уникальный идентификатор клиента.
 - LastName (VARCHAR) - фамилия клиента.
 - FirstName (VARCHAR) - имя клиента.
 - MiddleName (VARCHAR) - отчество клиента.

- PassportData (VARCHAR) - паспортные данные клиента.
- Pledge (Залог) - Хранит информацию о заложенных товарах.
 - ID (SERIAL, PRIMARY KEY) - уникальный идентификатор залога.
 - ClientID (INTEGER, FOREIGN KEY REFERENCES Client(ID)) - идентификатор клиента, которому принадлежит залог.
 - Item (VARCHAR) - название заложенного товара.
 - ItemValue (NUMERIC) - оценочная стоимость товара.
 - IssueAmount (NUMERIC) - сумма, выданная клиенту.
 - Commission (NUMERIC) - комиссия ломбарда.
 - ReturnDate (DATE) - дата возврата товара.
 - Status (VARCHAR) - статус залога (например, 'Выдан', 'Возвращен', 'Просрочен').
- History (История) - Хранит информацию о всех операциях, связанных с залогами.
 - ID (SERIAL, PRIMARY KEY) - уникальный идентификатор операции.
 - PledgeID (INTEGER, FOREIGN KEY REFERENCES Pledge(ID)) - идентификатор залога, к которому относится операция.
 - OperationDate (DATE) - дата проведения операции.
 - OperationType (VARCHAR) - тип операции (например, 'Выдача', 'Возврат', 'Просрочка').
 - Amount (NUMERIC) - сумма, участвующая в операции.

Query	Query History
1	-- Создание таблицы Client
2	
3	CREATE TABLE Client (
4	ID SERIAL PRIMARY KEY,
5	LastName VARCHAR(255),
6	FirstName VARCHAR(255),
7	MiddleName VARCHAR(255),
8	PassportData VARCHAR(255)
9);
10	
11	
12	-- Создание таблицы Pledge
13	
14	CREATE TABLE Pledge (
15	ID SERIAL PRIMARY KEY,
16	ClientID INTEGER REFERENCES Client(ID),
17	Item VARCHAR(255),
18	ItemValue NUMERIC(10, 2),
19	IssueAmount NUMERIC(10, 2),
20	Commission NUMERIC(10, 2),
21	ReturnDate DATE,
22	Status VARCHAR(20)
23);
24	
25	
26	-- Создание таблицы History
27	
28	CREATE TABLE History (
29	ID SERIAL PRIMARY KEY,
30	PledgeID INTEGER REFERENCES Pledge(ID),
31	OperationDate DATE,
32	OperationType VARCHAR(20),
33	Amount NUMERIC(10, 2)
34);

Заполнение таблиц данными

```

37 -- Вставка данных в таблицу Client
38 INSERT INTO Client (LastName, FirstName, MiddleName, PassportData) VALUES
39 ('Иванов', 'Иван', 'Иванович', '1234 567890'),
40 ('Петров', 'Петр', 'Петрович', '9876 543210'),
41 ('Сидоров', 'Сидор', 'Сидорович', '1011 121314'),
42 ('Кузнецов', 'Кузьма', 'Кузьмич', '2222 333344'),
43 ('Смирнов', 'Сергей', 'Сергеевич', '5555 666677'),
44 ('Васильев', 'Василий', 'Васильевич', '8888 999900'),
45 ('Соколов', 'Софрон', 'Софронович', '1111 222233'),
46 ('Михайлов', 'Михаил', 'Михайлович', '4444 555566'),
47 ('Федоров', 'Федор', 'Федорович', '7777 888899'),
48 ('Попов', 'Павел', 'Павлович', '0000 111122');
49
50 -- Вставка данных в таблицу Pledge
51 INSERT INTO Pledge (ClientID, Item, ItemValue, IssueAmount, Commission, ReturnDate, Status) VALUES
52 (1, 'Золотые часы', 5000, 4000, 500, '2023-12-31', 'Выдан'),
53 (2, 'Ноутбук', 15000, 12000, 1000, '2024-01-15', 'Выдан'),
54 (3, 'Телевизор', 8000, 6000, 400, '2023-11-20', 'Возвращен'),
55 (1, 'Кольцо с бриллиантом', 12000, 9000, 800, '2024-02-10', 'Выдан'),
56 (2, 'Смартфон', 3000, 2500, 200, '2023-12-20', 'Выдан'),
57 (3, 'Фотоаппарат', 7000, 5000, 300, '2023-11-25', 'Возвращен'),
58 (1, 'Серебряная цепочка', 2000, 1500, 100, '2024-01-05', 'Выдан'),
59 (2, 'Планшет', 10000, 8000, 500, '2024-02-01', 'Выдан'),
60 (3, 'Видеокамера', 12000, 9000, 600, '2023-12-10', 'Выдан'),
61 (1, 'Золотая подвеска', 4000, 3000, 200, '2024-03-15', 'Выдан');
62
63 -- Вставка данных в таблицу History
64 INSERT INTO History (PledgeID, OperationDate, OperationType, Amount) VALUES
65 (1, '2023-11-10', 'Выдача', 4000),
66 (2, '2023-12-01', 'Выдача', 12000),
67 (3, '2023-11-15', 'Возврат', 6000)

```

Data Output Messages Notifications

INSERT 0 10

2. Предложить для физической схемы базы данных лингвистические переменные (не менее трех).

- Стоимость товара (ItemValue):
 - Диапазон: 0 - 20000
 - Лингвистические значения:
 - Низкая (Low): 0 - 4000
 - Средняя (Medium): 4000 - 10000
 - Высокая (High): 10000 - 20000
- Сумма выдачи (IssueAmount):
 - Диапазон: 0 - 15000
 - Лингвистические значения:
 - Малая (Small): 0 - 2500
 - Средняя (Medium): 2500 - 7500
 - Большая (Large): 7500 - 15000
- Срок возврата (ReturnDate):
 - Диапазон: 0 - 365 (в днях)
 - Лингвистические значения:
 - Короткий (Short): 0 - 30 (дней)
 - Средний (Medium): 30 - 90 (дней)
 - Длинный (Long): 90 - 365 (дней)

3. Разработать SQL-реализации функций принадлежности и лингвистических переменных.

```

85 -- Функция принадлежности для "Стоимость товара (ItemValue)"
86 --/*
87 CREATE FUNCTION fun_ItemValue(cur_cost numeric)
88 RETURNS VARCHAR AS $$
89 BEGIN
90     IF cur_cost < 4000 THEN
91         RETURN 'low';
92     ELSEIF cur_cost BETWEEN 4000 AND 10000 THEN
93         RETURN 'medium';
94     ELSE
95         RETURN 'high';
96     END IF;
97 END;
98 $$ LANGUAGE plpgsql;
99 --*/

100
101 -- Функция принадлежности для "Сумма выдачи (IssueAmount)"
102 --/*
103 CREATE OR REPLACE FUNCTION fun_IssueAmount(cur_amount numeric)
104 RETURNS VARCHAR AS $$
105 BEGIN
106     IF cur_amount < 2500 THEN
107         RETURN 'small';
108     ELSEIF cur_amount BETWEEN 2500 AND 7500 THEN
109         RETURN 'medium';
110     ELSE
111         RETURN 'large';
112     END IF;
113 END;
114 $$ LANGUAGE plpgsql;
115 --*/

116
117 -- Функция принадлежности для "Срок возврата (ReturnDate)"
118 --/*
119 CREATE FUNCTION fun_ReturnDate(cur_date date)
120 RETURNS VARCHAR AS $$
121 BEGIN
122     IF cur_date - CURRENT_DATE <= 30 THEN
123         RETURN 'short';
124     ELSEIF cur_date - CURRENT_DATE BETWEEN 31 AND 90 THEN
125         RETURN 'medium';
126     ELSE
127         RETURN 'long';
128     END IF;
129 END;
130 $$ LANGUAGE plpgsql;
131 --*/

```

4. На основе SQL реализации лингвистических переменных разработать нечеткие запросы к базе данных (не менее пяти).

Пример 1: “Найти клиентов, которые заложили товар с высокой стоимостью”

```

134 --Пример 1: "Найти клиентов, которые заложили товар с высокой стоимостью":
135 SELECT
136     c.LastName,
137     c.FirstName,
138     c.MiddleName,
139     p.itemvalue,
140     p.Item
141 FROM
142     Client c
143 JOIN Pledge p
144     ON c.ID = p.ClientID
145 WHERE
146     fun_ItemValue(p.itemvalue) = 'high';
147

```

Data Output Messages Notifications

	lastname character varying (255)	firstname character varying (255)	middlename character varying (255)	itemvalue numeric (10,2)	item character varying (255)
1	Петров	Петр	Петрович	15000.00	Ноутбук
2	Иванов	Иван	Иванович	12000.00	Кольцо с бриллиантом
3	Сидоров	Сидор	Сидорович	12000.00	Видеокамера

Пример 2: “Найти залоги с малой суммой выдачи и коротким сроком возврата”

```

150 --Пример 2: "Найти залоги с малой суммой выдачи и коротким сроком возврата":
151 SELECT
152     p.Item,
153     p.IssueAmount,
154     p.ReturnDate
155 FROM
156     Pledge p
157 WHERE
158     fun_IssueAmount(p.IssueAmount) = 'small' AND fun_ReturnDate(p.ReturnDate) = 'short';
159

```

Data Output Messages Notifications

	item character varying (255)	issueamount numeric (10,2)	returndate date
1	Золотые часы	4000.00	2023-12-31
2	Смартфон	2500.00	2023-12-20
3	Серебряная цепочка	1500.00	2024-01-05
4	Золотая подвеска	3000.00	2024-03-15

Пример 3: “Найти клиентов, которые заложили товар с высокой стоимостью и получили большую сумму выдачи”

```

--Пример 3: "Найти клиентов, которые заложили товар с высокой стоимостью и получили большую сумму выдачи":
SELECT
    c.LastName,
    c.FirstName,
    c.MiddleName
FROM
    Client c
JOIN
    Pledge p ON c.ID = p.ClientID
WHERE
    fun_ItemValue(p.itemvalue) = 'high' AND fun_IssueAmount(p.IssueAmount) = 'large';

```

Пример 4: “Найти залоги с высокой стоимостью, но с низкой суммой выдачи”

```
176 --Пример 4: “Найти залоги с высокой стоимостью, но с низкой суммой выдачи”:
177 --/*
178 SELECT
179     p.Item,
180     p.itemvalue,
181     p.IssueAmount
182 FROM
183     Pledge p
184 WHERE
185     fun_ItemValue(p.itemvalue) = 'high' AND fun_IssueAmount(p.IssueAmount) = 'small';
186 --*/
187
188
189
```

Data Output Messages Notifications			
	item	itemvalue	issueamount
	character varying (255)	numeric (10,2)	numeric (10,2)
1	Серебряная цепочка	11000.00	1500.00

Пример 5: “Найти клиентов, которые заложили товар с низкой стоимостью, но получили большую сумму выдачи”

```
188 --Пример 5: “Найти клиентов, которые заложили товар с низкой стоимостью, но получили большую сумму выдачи”:
189 SELECT
190     c.LastName,
191     c.FirstName,
192     c.MiddleName
193 FROM
194     Client c
195 JOIN
196     Pledge p ON c.ID = p.ClientID
197 WHERE
198     fun_ItemValue(p.itemvalue) = 'low' AND fun_IssueAmount(p.IssueAmount) = 'large';
199
200
201
202
```

Data Output Messages Notifications			
	lastname	firstname	middlename
	character varying (255)	character varying (255)	character varying (255)
1	Петров	Петр	Петрович

5. На основе SQL реализации модификации лингвистических переменных разработать нечеткие запросы к базе данных

Найти клиентов, которые заложили товар с высокой стоимостью (ItemValue), но получили сумму выдачи (IssueAmount), которая является “не очень большой” и “не очень маленькой”, то есть не больше 10000, но и не меньше 2500.


```

204 CREATE OR REPLACE FUNCTION fun_IssueAmount_mod(cur_amount numeric)
205 RETURNS VARCHAR AS $$
206 BEGIN
207     IF cur_amount < 2500 THEN
208         RETURN 'small';
209     ELSIF cur_amount BETWEEN 2500 AND 5000 THEN
210         RETURN 'medium_low';
211     ELSIF cur_amount BETWEEN 5000 AND 10000 THEN
212         RETURN 'medium_high';
213     ELSE
214         RETURN 'large';
215     END IF;
216 END;
217 $$ LANGUAGE plpgsql;
218 --*/
219
220

```

Data Output Messages Notifications

CREATE FUNCTION

Query returned successfully in 62 msec.

```

220 --*/
221 SELECT
222     c.LastName,
223     c.FirstName,
224     c.MiddleName,
225     p.Item,
226     p.itemvalue,
227     p.IssueAmount
228 FROM
229     Client c
230 JOIN
231     Pledge p ON c.ID = p.ClientID
232 WHERE
233     fun_ItemValue(p.itemvalue) = 'high'
234     AND fun_IssueAmount_mod(p.IssueAmount) IN ('medium_high', 'medium_low');
235 --*/
236
237

```

Data Output Messages Notifications

	lastname character varying (255)	firstname character varying (255)	middlename character varying (255)	item character varying (255)	itemvalue numeric (10,2)	issueamount numeric (10,2)
1	Иванов	Иван	Иванович	Кольцо с бриллиантом	12000.00	9000.00
2	Сидоров	Сидор	Сидорович	Видеокамера	12000.00	9000.00

6. Выбрать параметр для индекса соответствия (CI), разработать нечеткие запросы к базе данных (не менее пяти) с учетом индекса соответствия.

В качестве параметра CI можно выбрать IssueAmount.

Пример 1: “Найти клиентов, у которых уровень соответствия высокой стоимости товара и большой суммы выдачи выше 0.8”

```

2  SELECT
3      c.LastName,
4      c.FirstName,
5      c.MiddleName,
6      p.Item,
7      p.itemvalue,
8      p.IssueAmount,
9      p.IssueAmount / 12000 AS CI
10 FROM
11     Client c
12 JOIN
13     Pledge p ON c.ID = p.ClientID
14 WHERE
15     fun_ItemValue(p.itemvalue) = 'high'
16     AND (p.IssueAmount / 12000) > 0.8;
17

```

	lastname character varying (255)	firstname character varying (255)	middlename character varying (255)	item character varying (255)	itemvalue numeric (10,2)	issueamount numeric (10,2)	ci numeric
1	Петров	Петр	Петрович	Ноутбук	15000.00	12000.00	1.00000000000000000000

Пример 2: “Найти клиентов, у которых уровень соответствия низкой стоимости товара и маленькой суммы выдачи выше 0.7”

```

18 --Пример 2: “Найти клиентов, у которых уровень соответствия низкой стоимости товара и маленькой суммы выдачи выше 0.7”:
19 SELECT
20     c.LastName,
21     c.FirstName,
22     c.MiddleName,
23     p.Item,
24     p.itemvalue,
25     p.IssueAmount,
26     p.IssueAmount / 12000 AS CI
27 FROM
28     Client c
29 JOIN
30     Pledge p ON c.ID = p.ClientID
31 WHERE
32     fun_ItemValue(p.itemvalue) = 'low'
33     AND p.IssueAmount / 12000 > 0.7;
34
35 --Пример 3: “Найти залоги с высокой стоимостью и коротким сроком возврата, где уровень соответствия выше 0.6”:
36 SELECT

```

	lastname character varying (255)	firstname character varying (255)	middlename character varying (255)	item character varying (255)	itemvalue numeric (10,2)	issueamount numeric (10,2)	ci numeric
1	Петров	Петр	Петрович	Смартфон	3000.00	10000.00	0.83333333333333333333

Пример 3: “Найти залоги с высокой стоимостью и коротким сроком возврата, где уровень соответствия выше 0.6”

```

36 SELECT
37     p.Item,
38     p.itemvalue,
39     p.ReturnDate,
40     p.IssueAmount,
41     p.IssueAmount / 12000 AS CI
42 FROM
43     Pledge p
44 WHERE
45     fun_ItemValue(p.itemvalue) = 'high'
46     AND fun_ReturnDate(p.ReturnDate) = 'short'
47     AND p.IssueAmount / 12000 > 0.6;
48

```

Data Output Messages Notifications

	item character varying (255)	itemvalue numeric (10,2)	returndate date	issueamount numeric (10,2)	ci numeric
1	Ноутбук	15000.00	2024-01-15	12000.00	1.00000000000000000000
2	Кольцо с бриллиантом	12000.00	2024-02-10	9000.00	0.75000000000000000000
3	Видеокамера	12000.00	2023-12-10	9000.00	0.75000000000000000000

Пример 4: “Найти залоги с высокой стоимостью, но с “не очень большой” суммой выдачи, где уровень соответствия выше 0.5”

```

286 --Пример 4: “Найти залоги с высокой стоимостью, но с “не очень большой” суммой выдачи
287 SELECT
288     p.Item,
289     p.itemvalue,
290     p.IssueAmount,
291     p.IssueAmount / 12000 AS CI
292 FROM
293     Pledge p
294 WHERE
295     fun_ItemValue(p.itemvalue) = 'high'
296     AND fun_IssueAmount_mod(p.IssueAmount) IN ('medium_high', 'medium_low')
297     AND p.IssueAmount / 12000 > 0.5;
298
299 --Пример 5: “Найти залоги, которые соответствуют средней стоимости товара и средней с
300 SELECT

```

Data Output Messages Notifications

	item character varying (255)	itemvalue numeric (10,2)	issueamount numeric (10,2)	ci numeric
1	Кольцо с бриллиантом	12000.00	9000.00	0.75000000000000000000
2	Видеокамера	12000.00	9000.00	0.75000000000000000000

Пример 5: “Найти залогов, которые соответствуют средней стоимости товара и средней сумме выдачи, где уровень соответствия выше 0.4”

```

300 SELECT
301     p.Item,
302     p.itemvalue,
303     p.IssueAmount,
304     p.IssueAmount / 12000 AS CI
305 FROM
306     Pledge p
307 WHERE
308     fun_ItemValue(p.itemvalue) = 'medium'
309     AND fun_IssueAmount(p.IssueAmount) = 'medium'
310     AND p.IssueAmount / 12000 > 0.4;
311
312
313
314

```

Data Output		Messages	Notifications	
<div><div><div><div>≡</div><div>+</div></div><div><div>📄</div><div>▼</div></div><div><div>📋</div><div>▼</div></div><div><div>🗑️</div></div><div><div>🗄️</div></div><div><div>⬇️</div></div><div><div>📈</div></div></div></div>				
	item character varying (255) 🔒	itemvalue numeric (10,2) 🔒	issueamount numeric (10,2) 🔒	ci numeric 🔒
1	Телевизор	8000.00	6000.00	0.50000000000000000000
2	Фотоаппарат	7000.00	5000.00	0.41666666666666666667