

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
Харківський національний університет радіоелектроніки

Кафедра системотехніки

ЗВІТ
з виконання завдань практичного заняття № 4
дисципліни «Проектування високонавантажених систем
зберігання даних»
на тему: «ОПТИМІЗАЦІЯ SQL-ЗАПИТІВ ВИСОКОНАВАНТАЖЕНИХ БАЗ
ДАНИХ НА ПЛАТФОРМІ СУБД MySQL»

Виконав
студент WILDAU - KHARKIV
Якунін Ігор

Перевірив
професор кафедри СТ
Колесник Л.В.

Харків, 2024

4.1 Мета заняття

- набуття практичних навичок з оптимізації продуктивності баз даних на платформі СУБД MySQL;
- набуття практичних навичок з оптимізації SQL-запитів з використанням індексів і складених індексів для таблиць MyISAM;
- набуття практичних навичок з оптимізації SQL-запитів з використанням кластерних індексів для таблиць InnoDB;
- набуття практичних навичок з оптимізації SQL-запитів з використанням fulltext-індексів і різних режимів повнотекстового пошуку для таблиць MyISAM;
- формування необхідних практичних умінь для аналізу плану виконання SQL-запитів, вибору довжини префікса, оцінки покриття й селективності індексів за допомогою оператора EXPLAIN;
- формування необхідних практичних умінь використання індексів для оптимізації SQL-запитів, з урахуванням особливостей роботи високонавантаженої інформаційної системи зберігання даних.

Завдання на самостійну роботу

Обрати предметну область для виконання індивідуальних завдань. Обрана область – **«Сайт сервісного центру по ремонту техніки»**

В БД записані фейкові дані – 10 000 записів.

Завдання 4.1. Для бази даних з таблицями MyISAM, розробленої відповідно до завдання 1.2 практичного заняття № 1, провести оптимізацію SQL-запитів, використовуючи індекси й складені індекси. Під час оптимізації SQL-запитів виконати такі завдання:

- аналіз і обґрунтування рішень щодо вибору індексів (частин складеного індексу) для покриття SQL-запитів;
- обґрунтування довжини префікса обраних індексів (префіксів частин складених індексів);
- обґрунтування порядку розташування частин складеного індексу з урахуванням покриття й аналізу селективності, кожного з них.

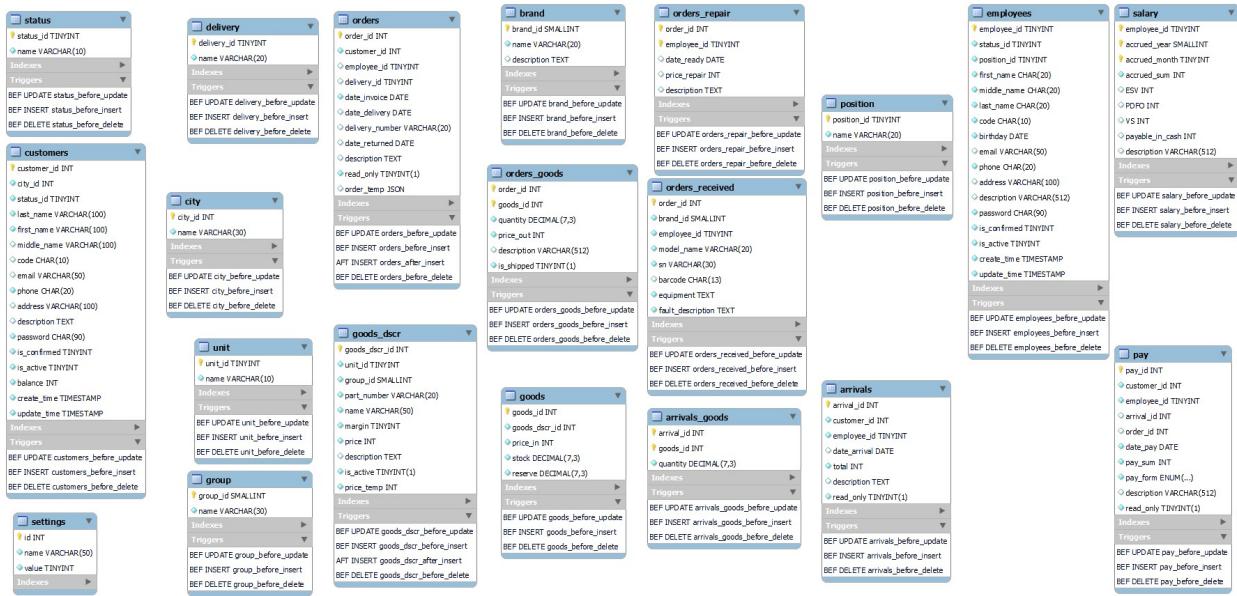


Рис. 4.1.1. Скріншот схеми фізичної моделі бази даних з таблицями типу MyISAM

Таблиця 4.1 – Визначення полів для індексу

№	SQL -запит	Поле для індексу
1	SELECT * FROM orders WHERE date_returned ='2024-09-30';	orders.date_returned
2	SELECT * FROM customers WHERE phone = '+380 67 139-46-71'	customers.phone
3	SELECT * FROM goods_desc WHERE part_number ='WTR2T2RC8N';	goods_desc.part_number

Таблиця 4.2 – Визначення полів складених індексів

№	SQL -запит	Ім'я складеного індексу	Поля для індексу
1	SELECT * FROM customers WHERE last_name = 'Бандера' and first_name = 'Симон';	l_f_name	customers.last_name, customers.first_name
2	SELECT * FROM orders_received WHERE equipment LIKE 'Зарплата%' and fault_description LIKE 'Червень%';	eq_fault	orders_received.equipment, orders_received.fault_description

Перед виконанням оптимізації SQL-запитів використаємо ANALYZE TABLE, щоб оновити статистику таблиці для покращення планування SQL-запитів.

4.1.1. Orders

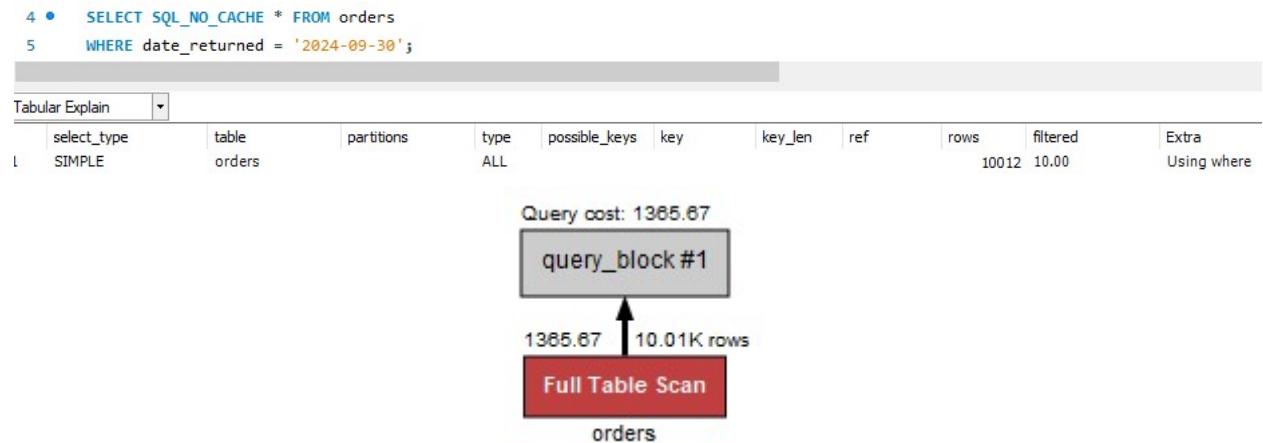


Рис. 4.1.2. Попередній аналіз таблиці orders, до створення індексу

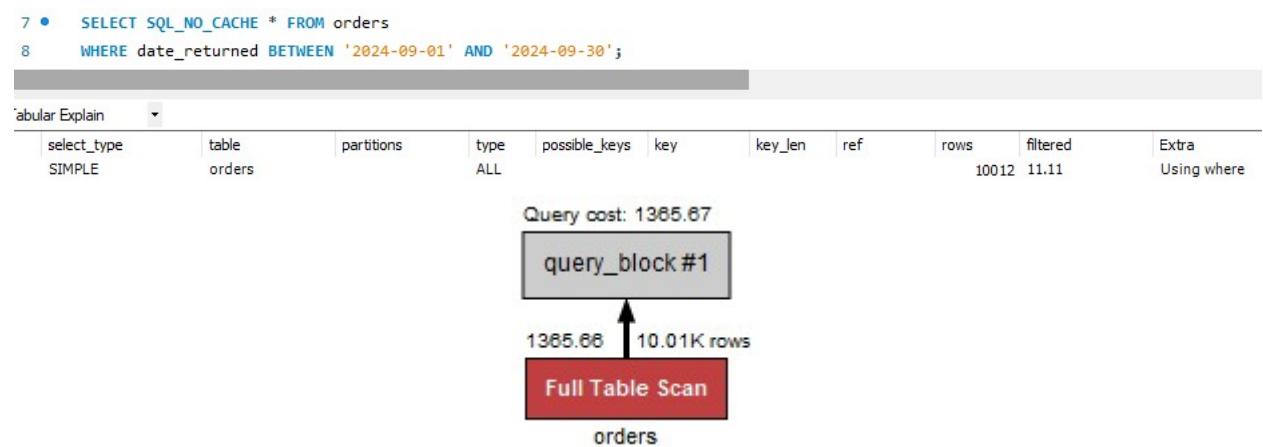


Рис. 4.1.3. Попередній аналіз таблиці orders в діапазоні дат, до створення індексу

Проведемо аналіз, чи має сенс використовувати префікс для поля дат

```

13 •      SELECT COUNT(DISTINCT LEFT(date_returned, 6))/COUNT(*) AS sel6,
14          COUNT(DISTINCT LEFT(date_returned, 7))/COUNT(*) AS sel7,
15          COUNT(DISTINCT LEFT(date_returned, 8))/COUNT(*) AS sel8,
16          COUNT(DISTINCT LEFT(date_returned, 9))/COUNT(*) AS sel9,
17          COUNT(DISTINCT LEFT(date_returned, 10))/COUNT(*) AS sel10,
18          COUNT(DISTINCT date_returned)/COUNT(*) as sel_all
19      FROM orders;

```

sel6	sel7	sel8	sel9	sel10	sel_all
0.0002	0.0011	0.0011	0.0039	0.0296	0.0296

Рис. 4.1.3. Аналіз потреби використання префіксу

Довжина поля date_returned – 10 символів. Як видно з скриншоту запиту, використовувати префікс немає сенсу.

```

10 •      CREATE INDEX d_return ON orders (date_returned);
11 •      SHOW INDEX FROM orders;

```

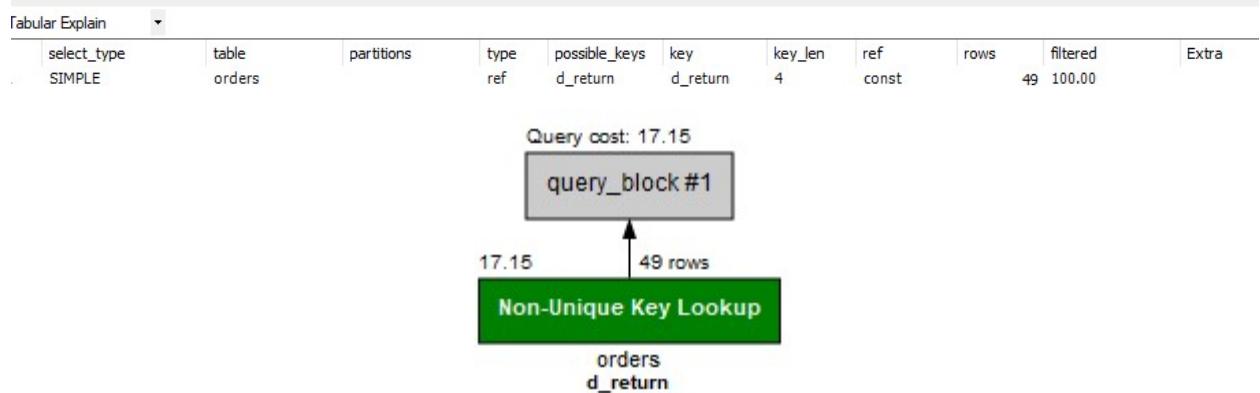
Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
orders	0	PRIMARY	1	order_id	A	10012	NULL	NULL	NULL	BTREE			YES	NULL
orders	1	fk_cust	1	customer_id	A	5006	NULL	NULL	NULL	BTREE			YES	NULL
orders	1	fk_deliv	1	delivery_id	A	2002	NULL	NULL	YES	BTREE			YES	NULL
orders	1	fk_empl	1	employee_id	A	500	NULL	NULL	YES	BTREE			YES	NULL
orders	1	d_return	1	date_returned	A	770	NULL	NULL	YES	BTREE			YES	NULL

Рис. 4.1.4. Створення індексу для поля дати таблиці orders

```

7 •      SELECT SQL_NO_CACHE * FROM orders
8      WHERE date_returned = '2024-09-30';

```



```

7 •  SELECT SQL_NO_CACHE * FROM orders
8 WHERE date_returned BETWEEN '2024-09-01' AND '2024-09-30';

```

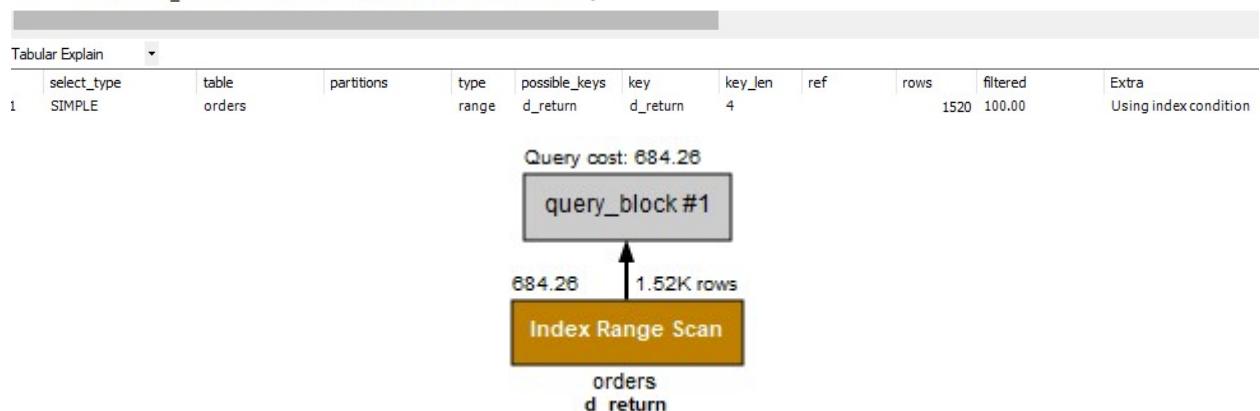


Рис. 4.1.5. Аналіз таблиці orders, після створення індексу

Індекс для поля дати дає суттєву перевагу в пошуку даних за конкретну дату. В диапазоні дат (місяць) перевага також є, але трохи меньша. Для наочності та легкості порівняння MyISAM і InnoDB я звів усі результати за швидкодією запитів в одну таблицю 4.2.1 (сторінка 24). Крім того, я виявив що результат залежить від кількості строк (Limit to rows), що виводяться на екран, тому щоб забезпечити мінімальний вплив різних факторів на швидкодію запитів я вирішив провести заміри одразу для обох типів таблиц в однакових умовах.

4.1.2. Customers, поля last_name, first_name

Таблиця 4.3 – Оцінка покриття запиту

№	SQL-запит	Поля складеного індексу	Висновок за оцінкою покриття
1	SELECT * FROM customers WHERE last_name = 'Бандера'	last_name, first_name	100% покриття за всіма полями
2	and first_name = 'Симон';	first_name, last_name	100% покриття за всіма полями
3		last_name	100% покриття за last_name
4		first_name	Покриття немає

Таблиця 4.4 – Оцінка селективності складеного індексу

SQL-запит	Поля складеного індексу	Кількість записів для кожної виконаної умови

SELECT * FROM customers WHERE last_name = 'Бандера' and first_name = 'Симон';	last_name	19
	first_name	50

last_name	count(*)	first_name	count(*)
Бандера	19	Симон	50

Рис. 4.1.6. Визначення кількості записів для last_name та first_name

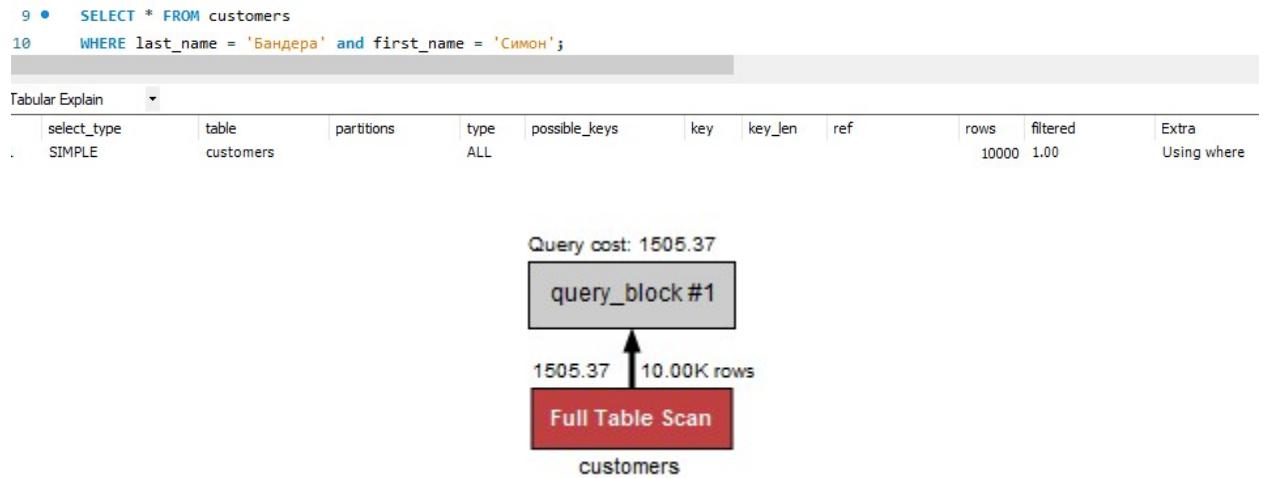


Рис. 4.1.7. Попередній аналіз таблиці customers, до створення індексу

Визначимо префікс складеного індексу:

```
3 •    SELECT COUNT(DISTINCT LEFT(last_name, 3))/COUNT(*) AS sel3,  
4       COUNT(DISTINCT LEFT(last_name, 5))/COUNT(*) AS sel5,  
5       COUNT(DISTINCT LEFT(last_name, 6))/COUNT(*) AS sel6,  
6       COUNT(DISTINCT LEFT(last_name, 7))/COUNT(*) AS sel7,  
7       COUNT(DISTINCT LEFT(last_name, 8))/COUNT(*) AS sel8,  
8       COUNT(DISTINCT LEFT(last_name, 9))/COUNT(*) AS sel9,  
9       COUNT(DISTINCT LEFT(last_name, 11))/COUNT(*) AS sel11,  
10      COUNT(DISTINCT LEFT(last_name, 13))/COUNT(*) AS sel13,  
11      COUNT(DISTINCT last_name)/COUNT(*) as sel_all  
12     FROM customers;
```

sel3	sel5	sel6	sel7	sel8	sel9	sel11	sel13	sel_all
0.0313	0.0465	0.0504	0.0527	0.0535	0.0535	0.0539	0.0541	0.0542

```

19 •      SELECT COUNT(DISTINCT LEFT(first_name, 3))/COUNT(*) AS sel3,
20          COUNT(DISTINCT LEFT(first_name, 4))/COUNT(*) AS sel4,
21          COUNT(DISTINCT LEFT(first_name, 5))/COUNT(*) AS sel5,
22          COUNT(DISTINCT LEFT(first_name, 6))/COUNT(*) AS sel6,
23          COUNT(DISTINCT LEFT(first_name, 7))/COUNT(*) AS sel7,
24          COUNT(DISTINCT LEFT(first_name, 8))/COUNT(*) AS sel8,
25          COUNT(DISTINCT LEFT(first_name, 9))/COUNT(*) AS sel9,
26          COUNT(DISTINCT LEFT(first_name, 10))/COUNT(*) AS sel10,
27          COUNT(DISTINCT first_name)/COUNT(*) as sel_all
28      FROM customers;

```

sel3	sel4	sel5	sel6	sel7	sel8	sel9	sel10	sel_all
0.0144	0.0167	0.0175	0.0178	0.0182	0.0183	0.0185	0.0187	0.0187

Рис. 4.1.8. Визначення префіксу складеного індексу

Створимо складений індекс, та проведемо аналіз запиту

```

14 •      CREATE INDEX l_f_name ON customers (last_name(7), first_name(5));
15 •      SHOW INDEX FROM customers;

```

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
customers	0	PRIMARY	1	customer_id	A	10000	NULL	NULL		BTREE			YES	NULL
customers	0	phone_UNIQUE	1	phone	A	10000	NULL	NULL		BTREE			YES	NULL
customers	1	fk_city	1	city_id	A	2	NULL	NULL		BTREE			YES	NULL
customers	1	fk_status	1	status_id	A	10	NULL	NULL		BTREE			YES	NULL
customers	1	l_f_name	1	last_name	A	526	7	NULL		BTREE			YES	NULL
customers	1	l_f_name	2	first_name	A	10000	5	NULL		BTREE			YES	NULL

29	20:21:16	CREATE INDEX l_f_name ON customers (last_name(7), first_name(5))	10000 row(s) affected	Records: 10000
30	20:21:29	SHOW INDEX FROM customers	6 row(s) returned	

Рис. 4.1.9. Створення складеного індексу

```

9 •      SELECT SQL_NO_CACHE * FROM customers
10 WHERE last_name = 'Бандера' and first_name = 'Симон';

```

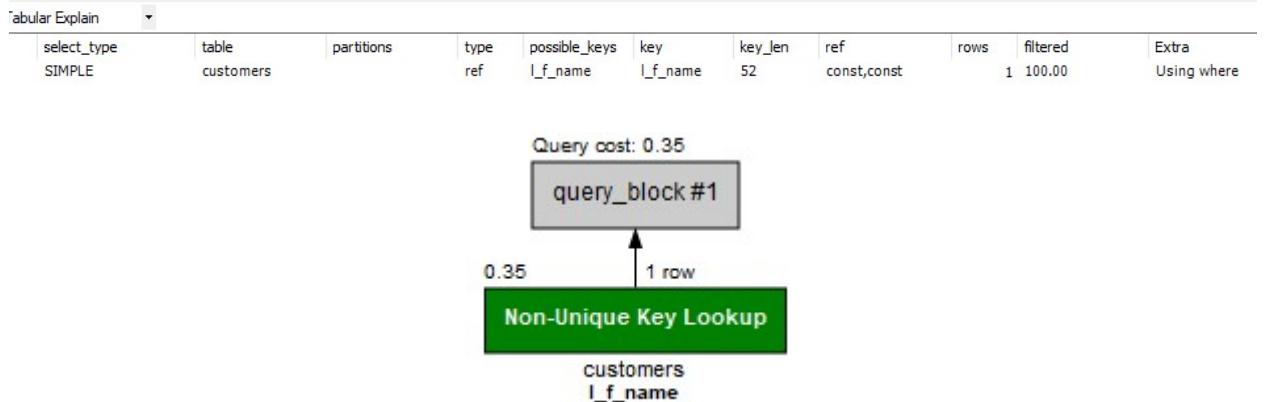


Рис. 4.1.10. Скриншоти аналізу запиту

```
9 •  SELECT SQL_NO_CACHE * FROM customers
10 WHERE first_name = 'Симон' and last_name = 'Бандера';
```

Tabular Explain

select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1 SIMPLE	customers		ref	l_f_name	l_f_name	52	const,const	1	100.00	Using where

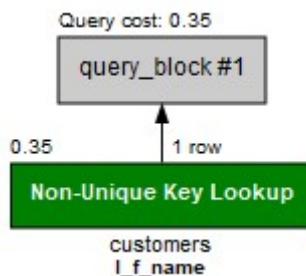


Рис. 4.1.11. Скриншоти аналізу зворотнього запиту

```
9 •  SELECT SQL_NO_CACHE * FROM customers
10 WHERE last_name = 'Бандера';
```

Tabular Explain

select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1 SIMPLE	customers		ref	l_f_name	l_f_name	30	const	16	100.00	Using where

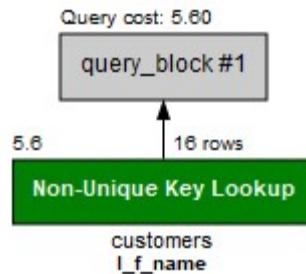


Рис. 4.1.12. Скриншоти аналізу запиту по last_name

```
9 •  SELECT SQL_NO_CACHE * FROM customers
10 WHERE first_name = 'Симон';
```

Tabular Explain

select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1 SIMPLE	customers		ALL					10000	10.00	Using where

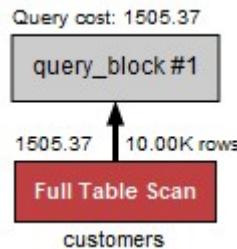


Рис. 4.1.13. Скриншоти аналізу запиту по first_name

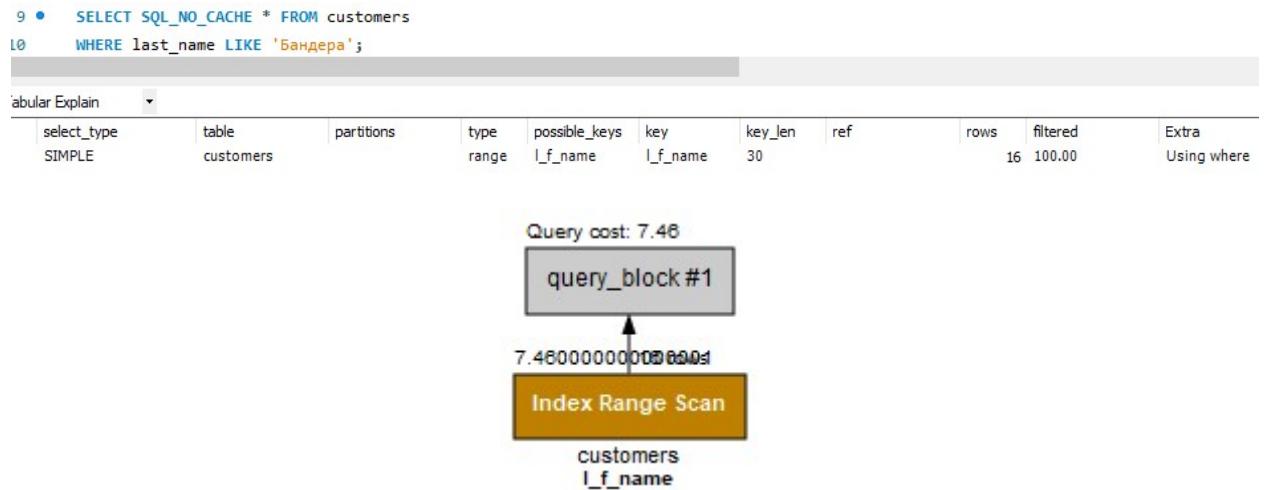


Рис. 4.1.14. Скриншоти аналізу запиту по last_name з LIKE

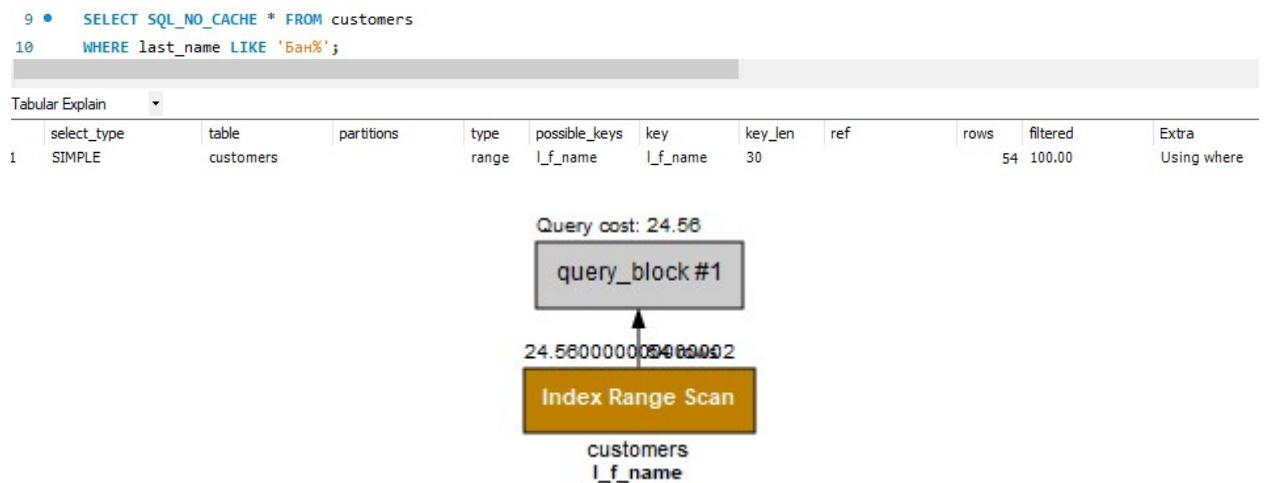
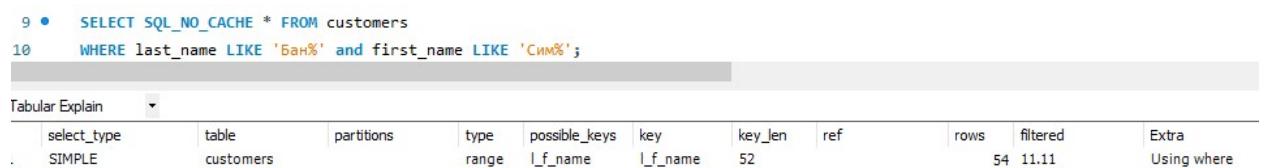


Рис. 4.1.15. Скриншоти аналізу запиту по last_name з LIKE та «%»



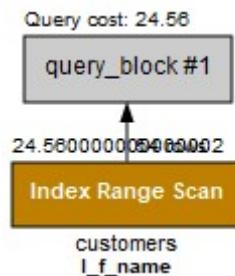


Рис. 4.1.16. Скриншоти аналізу запиту по last_name та first_name з LIKE та «%»

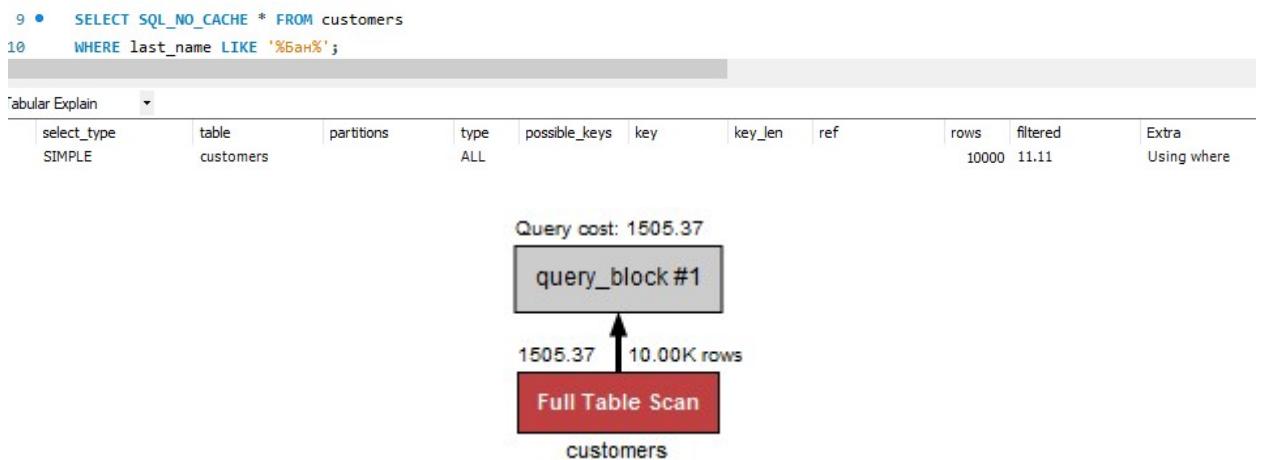


Рис. 4.1.17. Скриншоти аналізу запиту по last_name з LIKE та «%» попереду

4.1.3. Customers, поле phone, UNIQUE INDEX



Рис. 4.1.18. Скриншоти аналізу запиту таблиця customers, поле phone

Інши варіанти запиту дадут аналогічні результати, оскільки йде считування всієї таблиці. Створимо унікальний індекс. Індекс створюється на все поле, без префіксу, інакше буде втрачена унікальність.

14 • SHOW INDEX FROM customers;

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
customers	0	PRIMARY	1	customer_id	A	10000	NULL	NULL	BTREE				YES	NULL
customers	0	phone_UNIQUE	1	phone	A	10000	NULL	NULL	BTREE				YES	NULL
customers	1	fk_city	1	city_id	A	2	NULL	NULL	BTREE				YES	NULL
customers	1	fk_status	1	status_id	A	10	NULL	NULL	BTREE				YES	NULL
customers	1	l_f_name	1	last_name	A	526	7	NULL	BTREE				YES	NULL
customers	1	l_f_name	2	first_name	A	10000	5	NULL	BTREE				YES	NULL

Рис. 4.1.19. Створення індексу

10 • SELECT SQL_NO_CACHE * FROM customers
11 WHERE phone = '+380 67 139-46-71';

Tabular Explain											
1	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	customers		const	phone_UNIQUE	phone_UNI...	80	const	1	100.00	
Query cost: 1.00											
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> query_block #1 </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> Single Row (constant) customers phone_UNIQUE </div> <div style="width: 45%;"> Timing (as measured at client side): Execution time: 0:00:0.000000000 Timing (as measured by the server): Execution time: 0:00:0.00035270 Table lock wait time: 0:00:0.00000500 </div> <div style="width: 45%;"> Timing (as measured at client side): Execution time: 0:00:0.000000000 Timing (as measured by the server): Execution time: 0:00:0.00032570 Table lock wait time: 0:00:0.00000400 </div> </div>											

Рис. 4.1.20. Скриншоти аналізу запиту

10 • SELECT SQL_NO_CACHE * FROM customers
11 WHERE phone LIKE '+380 67 139%';

Tabular Explain											
1	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	customers		range	phone_UNIQUE	phone_UNIQUE	80		1	100.00	Using index condition
Query cost: 0.71											
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> query_block #1 </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> Index Range Scan customers phone_UNIQUE </div> <div style="width: 45%;"> Timing (as measured at client side): Execution time: 0:00:0.000000000 Timing (as measured by the server): Execution time: 0:00:0.00038900 Table lock wait time: 0:00:0.00000400 </div> <div style="width: 45%;"> Timing (as measured at client side): Execution time: 0:00:0.000000000 Timing (as measured by the server): Execution time: 0:00:0.00039200 Table lock wait time: 0:00:0.00000400 </div> </div>											

Рис. 4.1.21. Скриншоти аналізу запиту з використанням LIKE

Як і очикувалось, унікальний простий індекс працює максимально швидко (час пошуку 0.32-0.35мс) при точному співпадінні (phone = '+380 67 139-46-71'). Тут був використан найбільш швидкий тип з'єднання – **CONST**. В інших варіантах пошуку використовується тип з'єднання – **RANGE**, що трохи знижує його швидкість (0,39мс). Порівняння з InnoDB на сторінці №30.

Ще додам, що поле phone було створено не оптимальним образом, довжина 20 символів була обрана виходячи з довжини фейкових даних, хоча реальна довжина 12-13 символів, треба було преобразувати дані та відкинути зайві символи, навіть зробити це поле BIGINT, замість того, щоб обирати CHAR. Це замітка на майбутнє. Покищо залишимо як є.

4.1.4. goods_dscr, поле part_number(char(10)) UNIQUE INDEX

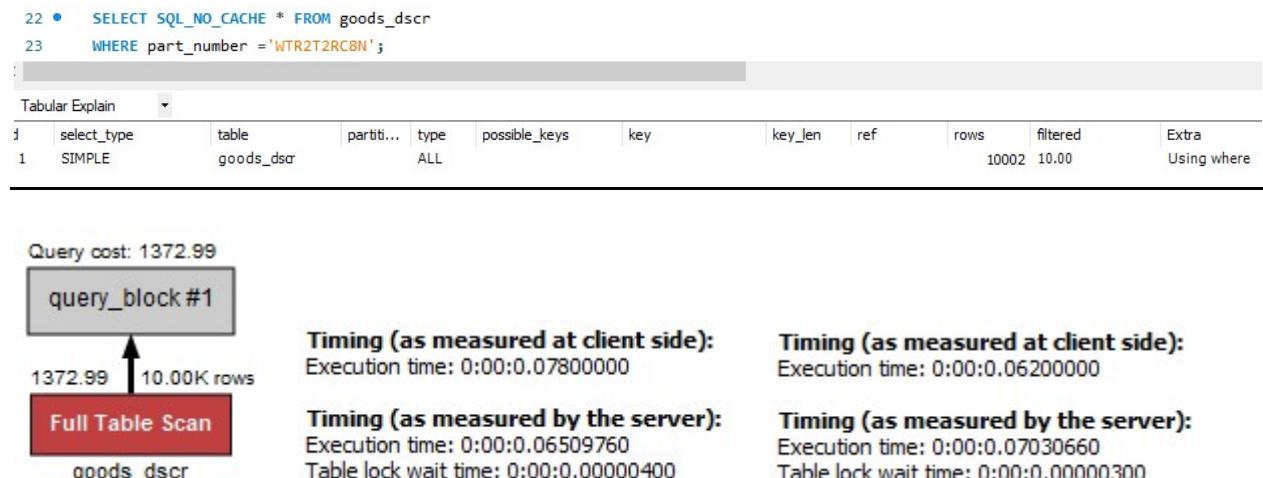


Рис. 4.1.22. Скриншоти аналізу запиту

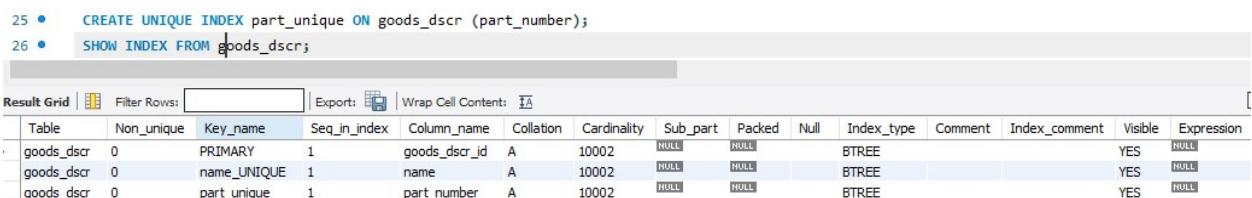


Рис. 4.1.23. Створення індексу



Рис. 4.1.24. Скриншоти аналізу запиту

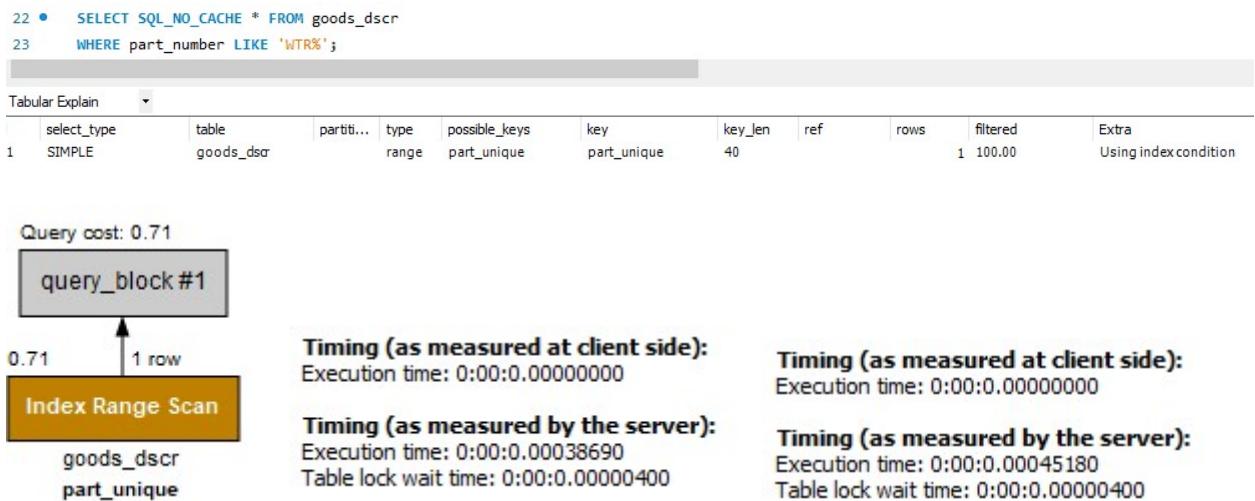
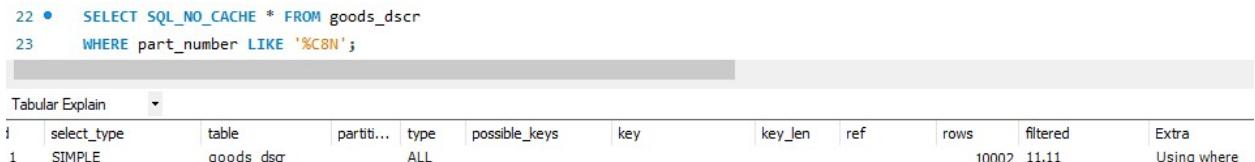


Рис. 4.1.25. Скриншоти аналізу запиту з LIKE



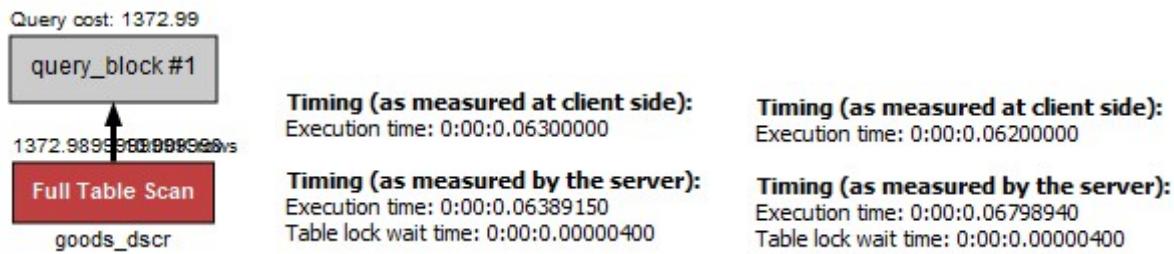


Рис. 4.1.26. Скриншоти аналізу запиту з LIKE та «%» попереду

4.1.5. goods_dscr, поле part_number(varchar(10))

Спробуємо змінити тип поля **char** -> **varchar**, та подимось, чи зміниться щось?

```
ALTER TABLE `ss_myisam2`.`goods_dscr`
DROP INDEX `part_unique` ;
```

```
ALTER TABLE `ss_myisam2`.`goods_dscr`
CHANGE COLUMN `part_number` `part_number` VARCHAR(10) CHARACTER SET 'utf8mb4' COLLATE
'utf8mb4_unicode_ci' NOT NULL ;
```

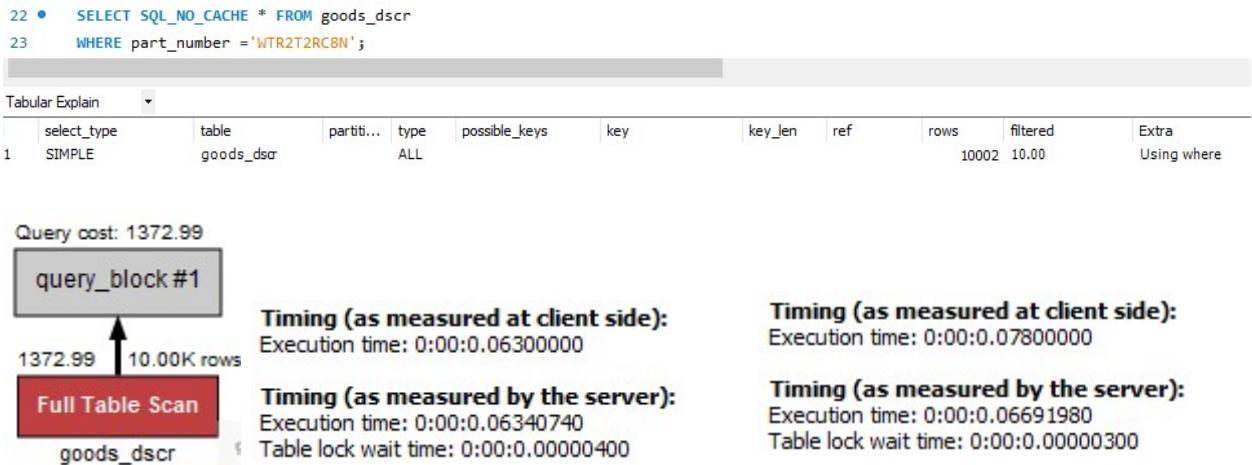


Рис. 4.1.27. Скриншоти аналізу запиту без використання індексу



Рис. 4.1.28. Створення індексу

```
22 •  SELECT SQL_NO_CACHE * FROM goods_dscr
23   WHERE part_number = 'WTR2T2RC8N';
```

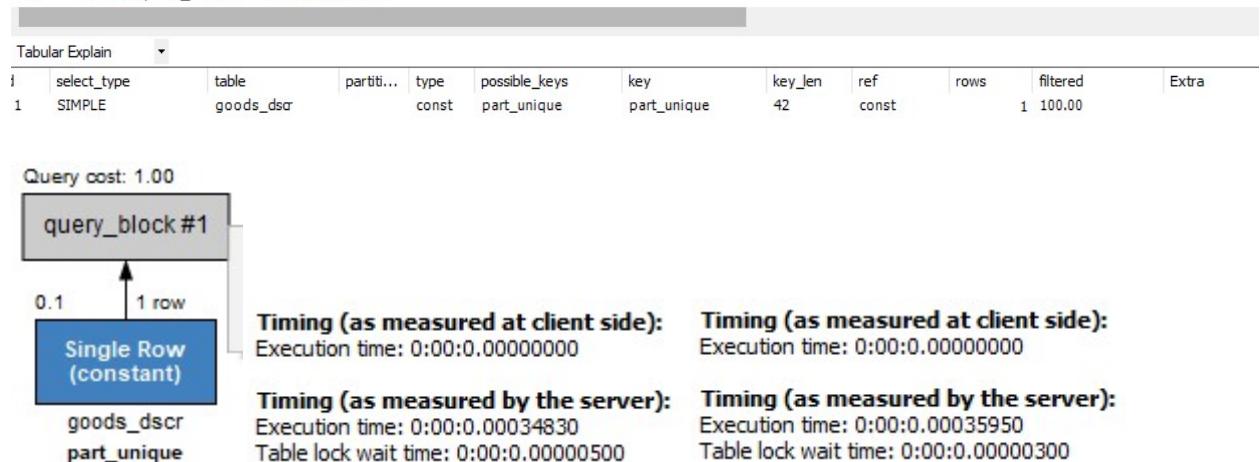


Рис. 4.1.29. Скриншоти аналізу запиту

```
22 •  SELECT SQL_NO_CACHE * FROM goods_dscr
23   WHERE part_number LIKE 'WTR%';
```



Рис. 4.1.30. Скриншоти аналізу запиту з LIKE та «%»

```
22 •  SELECT SQL_NO_CACHE * FROM goods_dscr
23   WHERE part_number LIKE '%C8N';
```

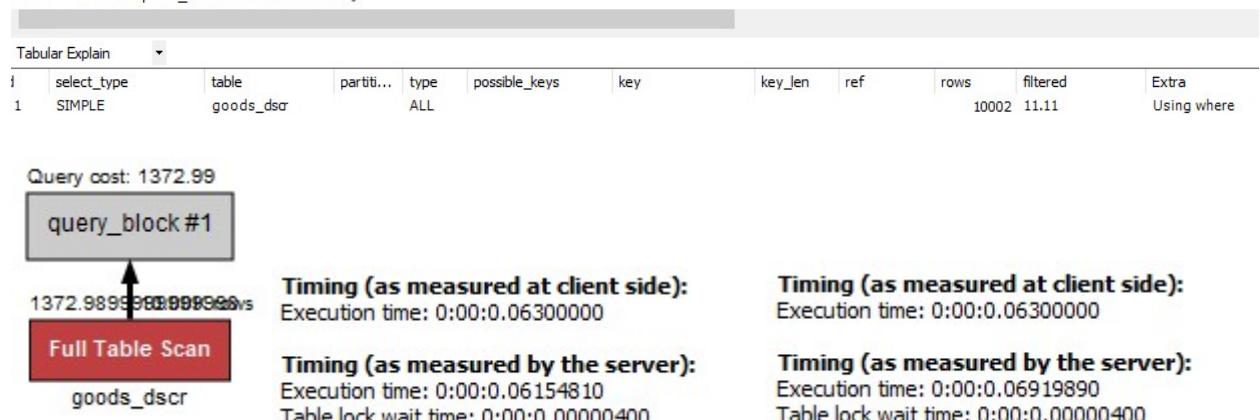


Рис. 4.1.31. Скриншоти аналізу запиту з LIKE та «%» попереду

Зміна поля для телефону з char на varchar привела к тому, що змінилась довжина індексу key_len (40 для char, 42 для varchar), але на існуючому наборі даних це ніяк не сказалось на швидкодії виконання запиту. Мубути потрібно мати мільйони записів, щоб це відобразилось на швидкодії. Ну що ж, це також корисно знати.

4.1.5. orders received, поля equipment, fault description

Таблиця 4.5 – Оцінка покриття запиту

№	SQL-запит	Поля складеного індексу	Висновок за оцінкою покриття
1	SELECT * FROM orders_received	equipment, fault_description	100% покриття за всіма полями
2	WHERE equipment LIKE 'Зарплата%' and fault_description LIKE 'Червень%';	fault_description, equipment	100% покриття за всіма полями
3		equipment	100% покриття за last_name
4		fault_description	Покриття немає

Таблиця 4.6 – Оцінка селективності складеного індексу

SQL-запит	Поля складеного індексу	Кількість записів для кожної виконаної умови
SELECT * FROM orders_received	equipment	14
WHERE equipment LIKE 'Зарплата%' and fault_description LIKE 'Червень%';	fault_description	19

4 •	SELECT * FROM orders_received
5	WHERE equipment LIKE 'Зарплата%' and fault_description LIKE 'Червень%';
<pre>Result Grid Filter Rows: [] Edit: [] Export/Import: [] Wrap Cell Content: []</pre>	

Рис.4.1.32. Приклад запиту приймання техніки в ремонт, таблиця orders_received

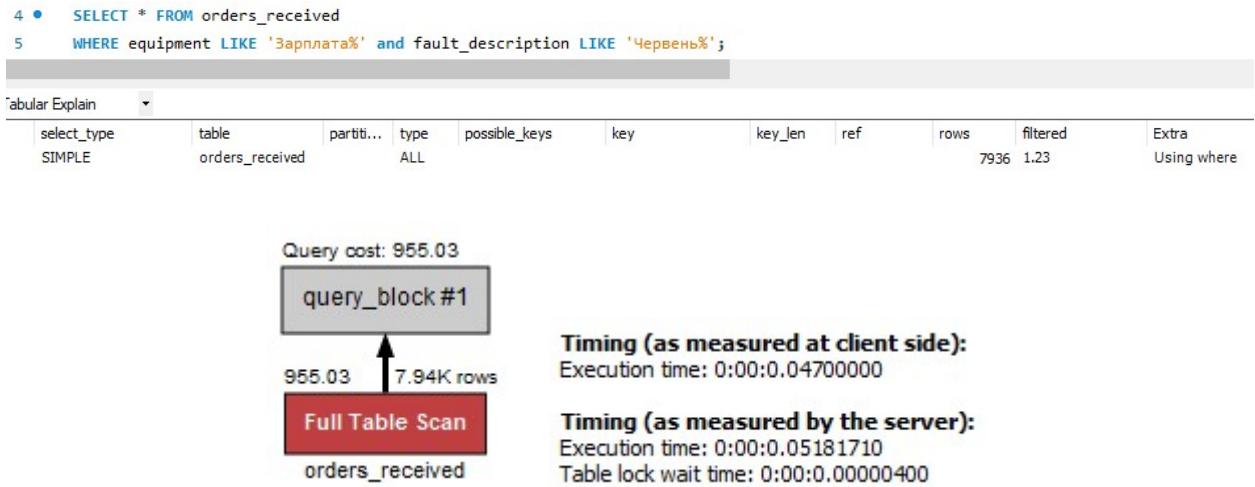


Рис.4.1.33. Скриншот аналізу запиту, таблиця orders_received

Для не оптимізованої таблиці інші запити з таблиці 4.5 мають повністю аналогичний вид, тому не бачу сенсу дублювати одне і теж. Проведемо аналіз префіксу для кожного поля.

```
52 • SELECT COUNT(DISTINCT LEFT(equipment, 10))/COUNT(*) AS sel10,  

53 COUNT(DISTINCT LEFT(equipment, 14))/COUNT(*) AS sel14,  

54 COUNT(DISTINCT LEFT(equipment, 15))/COUNT(*) AS sel15,  

55 COUNT(DISTINCT LEFT(equipment, 16))/COUNT(*) AS sel16,  

56 COUNT(DISTINCT LEFT(equipment, 17))/COUNT(*) AS sel17,  

57 COUNT(DISTINCT LEFT(equipment, 18))/COUNT(*) AS sel18,  

58 COUNT(DISTINCT LEFT(equipment, 20))/COUNT(*) AS sel20,  

59 COUNT(DISTINCT equipment)/COUNT(*) as sel_all  

60 FROM orders_received;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	sel10	sel14	sel15	sel16	sel17	sel18	sel20	sel_all
▶	0.6733	0.9522	0.9675	0.9783	0.9865	0.9918	0.9967	0.9999

```

42 •      SELECT COUNT(DISTINCT LEFT(fault_description, 10))/COUNT(*) AS sel10,
43          COUNT(DISTINCT LEFT(fault_description, 14))/COUNT(*) AS sel14,
44          COUNT(DISTINCT LEFT(fault_description, 15))/COUNT(*) AS sel15,
45          COUNT(DISTINCT LEFT(fault_description, 16))/COUNT(*) AS sel16,
46          COUNT(DISTINCT LEFT(fault_description, 17))/COUNT(*) AS sel17,
47          COUNT(DISTINCT LEFT(fault_description, 18))/COUNT(*) AS sel18,
48          COUNT(DISTINCT LEFT(fault_description, 20))/COUNT(*) AS sel20,
49          COUNT(DISTINCT fault_description)/COUNT(*) as sel_all
50      FROM orders_received;

```

	sel10	sel14	sel15	sel16	sel17	sel18	sel20	sel_all
	0.6710	0.9540	0.9684	0.9797	0.9873	0.9919	0.9968	0.9999

Рис. 4.1.34. Визначення префіксу складеного індексу

Result Grid Filter Rows: [] Export: [] Wrap Cell Content: []																
Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Express		
orders_received	0	PRIMARY	1	order_id	A	7936	NULL	NULL		BTREE			YES	NULL		
orders_received	1	fk_brand	1	brand_id	A	5	NULL	NULL		BTREE			YES	NULL		
orders_received	1	fk_employee	1	employee_id	A	19	NULL	NULL		BTREE			YES	NULL		
orders_received	1	eq_fault	1	equipment	A	7936	15	NULL		BTREE			YES	NULL		
orders_received	1	eq_fault	2	fault_description	A	7936	15	NULL		BTREE			YES	NULL		

Рис. 4.1.35. Створення індексу

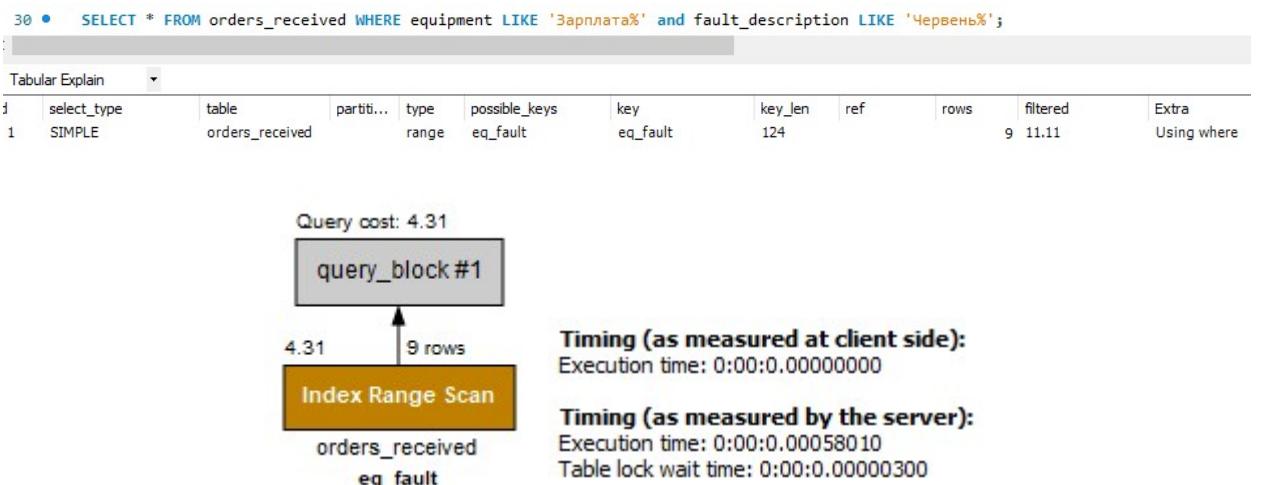


Рис. 4.1.36. Скриншоти аналізу запиту №1

Tabular Explain												
id	select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra	
1	SIMPLE	orders_received		range	eq_fault	eq_fault	124		9	11.11	Using where	

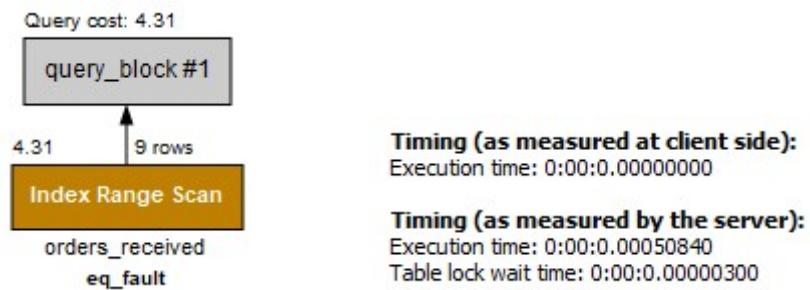


Рис. 4.1.37. Скриншоти аналізу зворотнього запиту №2

38 • SELECT * FROM orders_received WHERE equipment LIKE '%Зарплата%' and fault_description LIKE '%Червень%';											
	select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	orders_received		range	eq_fault	eq_fault	62		9	100.00	Using where

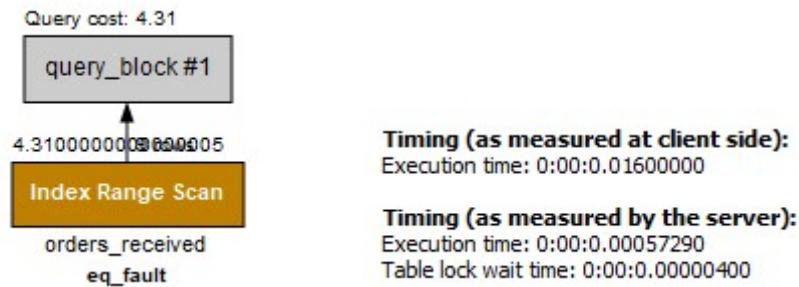


Рис. 4.1.38. Скриншоти аналізу запиту по першому полю №3

36 • SELECT * FROM orders_received WHERE fault_description LIKE 'Червень%';											
	select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	orders_received		ALL					7936	11.11	Using where

Query cost: 955.03

query_block #1

955.03 7.94K rows

Full Table Scan

orders_received

Timing (as measured at client side):
Execution time: 0:00:0.047000000

Timing (as measured by the server):
Execution time: 0:00:0.050935100
Table lock wait time: 0:00:0.000004000

Рис. 4.1.39. Скриншоти аналізу запиту по другому полю №4

38 • SELECT * FROM orders_received WHERE equipment LIKE '%Зарплата%' and fault_description LIKE '%Червень%';											
	select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	orders_received		ALL					7936	1.23	Using where

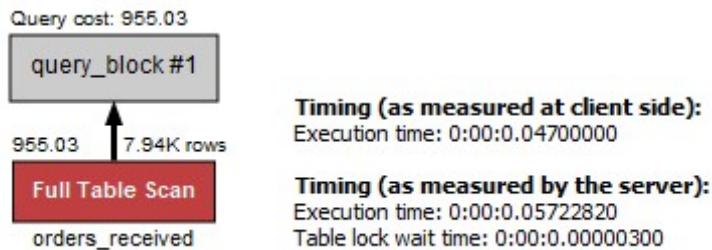


Рис. 4.1.40. Скриншоти аналізу запиту з LIKE та % попереду

Побачили, що індекс добре працює з LIKE, якщо попереду немає знака «%», прискорення досить суттєве (час виконання 0.5-0,6мс проти 50-60мс), не дивлячись, що тип зв'язку «range»

Порівняння з запитом для таблиці InnoDB приведено на сторінках №№34-35

Завдання 4.2. Для бази даних з таблицями InnoDB, розробленої відповідно до завдання 1.3 практичного заняття № 1, провести оптимізацію

SQL-запитів, використовуючи кластерні індекси. Під час оптимізації SQL-запитів виконати такі задачі:

- аналіз і обґрунтування ухвалених рішень щодо вибору індексів (частин складеного індексу) для покриття SQL-запитів;
- обґрунтування довжини префікса обраних індексів (префіксів частин складених індексів);
- обґрунтування порядку проходження частин складеного індексу з урахуванням покриття й аналізу селективності, кожного з них.

Для виконання завдання 4.2 використовувати SQL-запити, що розроблені відповідно до переліків завдань 2.1–2.2, 3.1–3.3;

Провести порівняльний аналіз ефективності використання кластерних індексів (для таблиць InnoDB), використовуючи результати завдання 4.1 (для таблиць MyISAM).

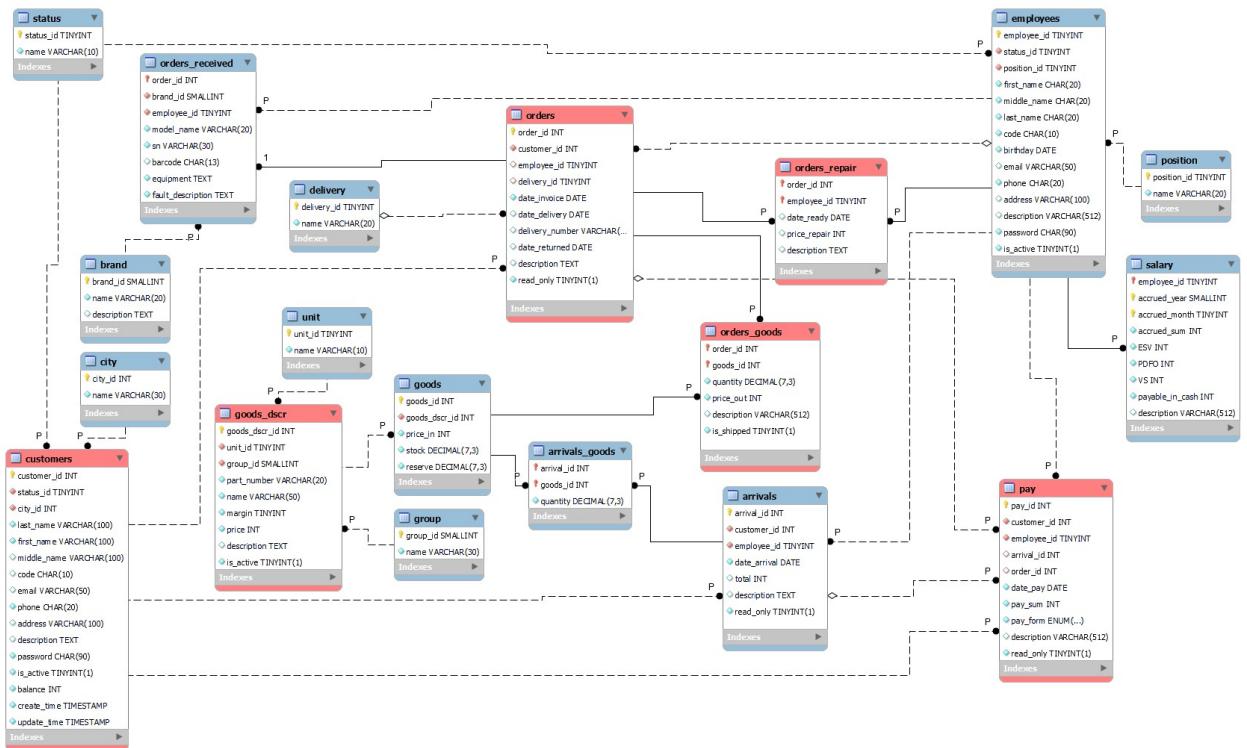


Рис. 4.2.1. Скріншот схеми фізичної моделі бази даних з таблицями типу InnoDB

4.2.1. Orders

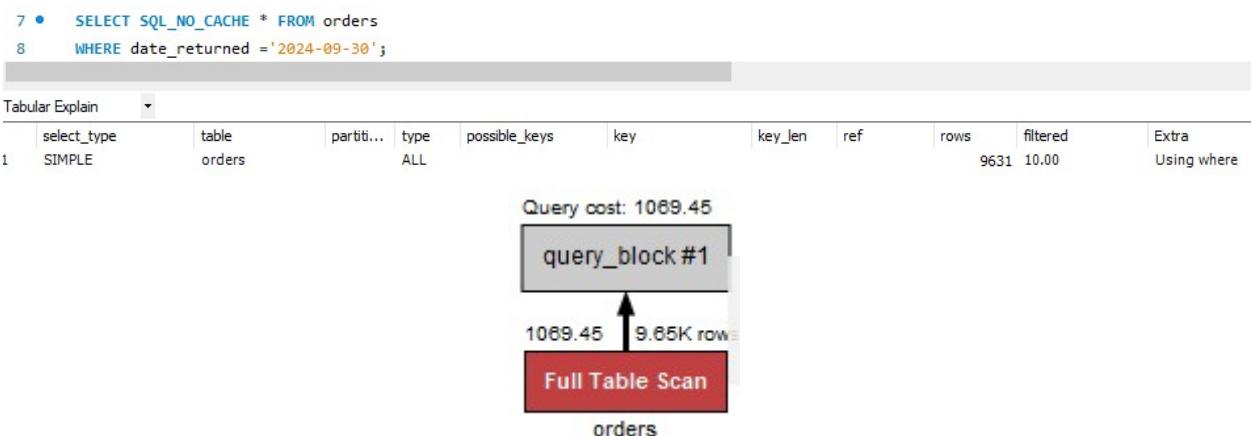


Рис. 4.2.2. Попередній аналіз таблиці orders, до створення індексу

```
7 •  SELECT SQL_NO_CACHE * FROM orders
8 WHERE date_returned BETWEEN '2024-09-01' AND '2024-09-30';
```

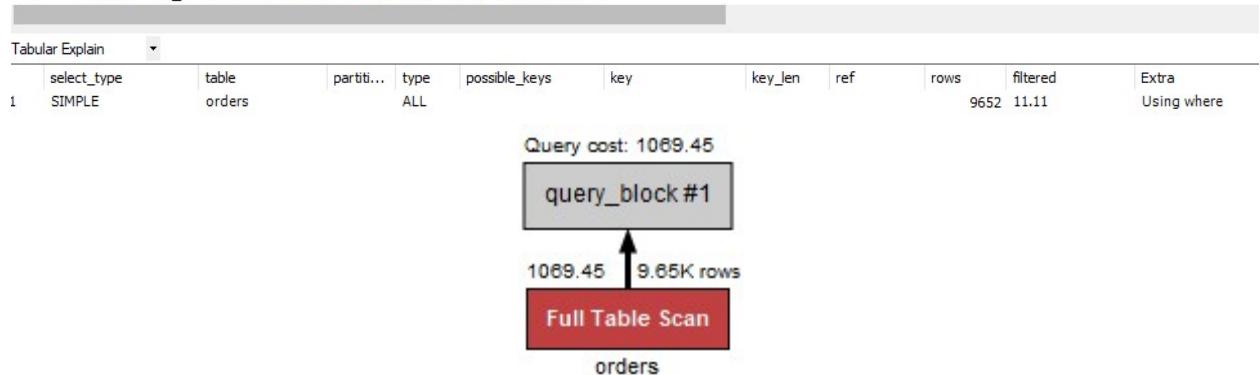


Рис. 4.2.3. Попередній аналіз таблиці orders в діапазоні дат, до створення індексу

```
10 •  CREATE INDEX d_return ON orders (date_returned);
11 •  SHOW INDEX FROM orders;
```

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Express
orders	0	PRIMARY	1	order_id	A	9652	NULL	NULL	NULL	BTREE			YES	NULL
orders	1	fk_orders_customers_idx	1	customer_id	A	6321	NULL	NULL	NULL	BTREE			YES	NULL
orders	1	fk_orders_delivery1_idx	1	delivery_id	A	4	NULL	NULL	YES	BTREE			YES	NULL
orders	1	fk_orders_employees1_idx	1	employee_id	A	21	NULL	NULL	YES	BTREE			YES	NULL
orders	1	d_return	1	date_returned	A	298	NULL	NULL	YES	BTREE			YES	NULL

Рис. 4.2.4. Створення індексу для поля дати таблиці orders

```
7 •  SELECT SQL_NO_CACHE * FROM orders
8 WHERE date_returned = '2024-09-30';
```

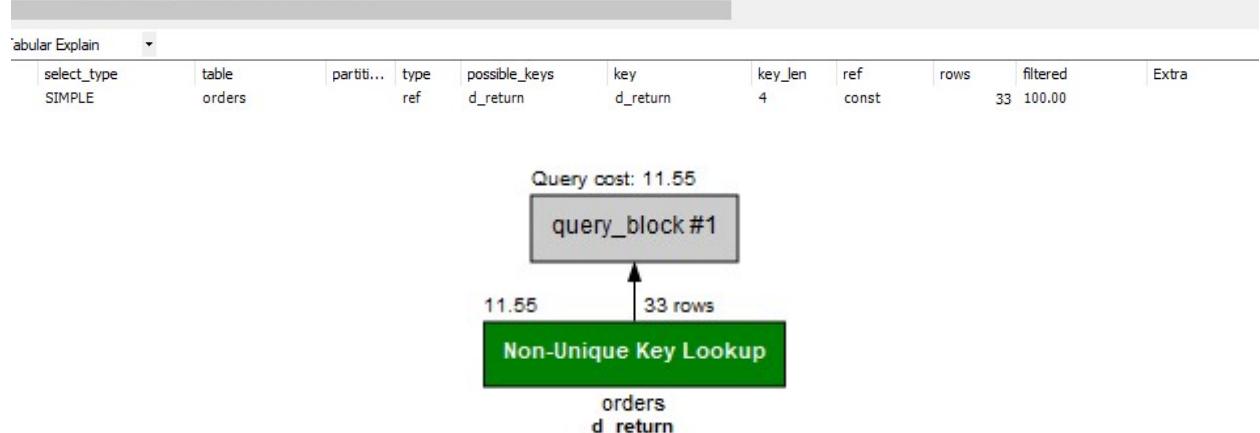


Рис. 4.2.5. Аналіз таблиці orders, при date_returned ='2024-09-30'

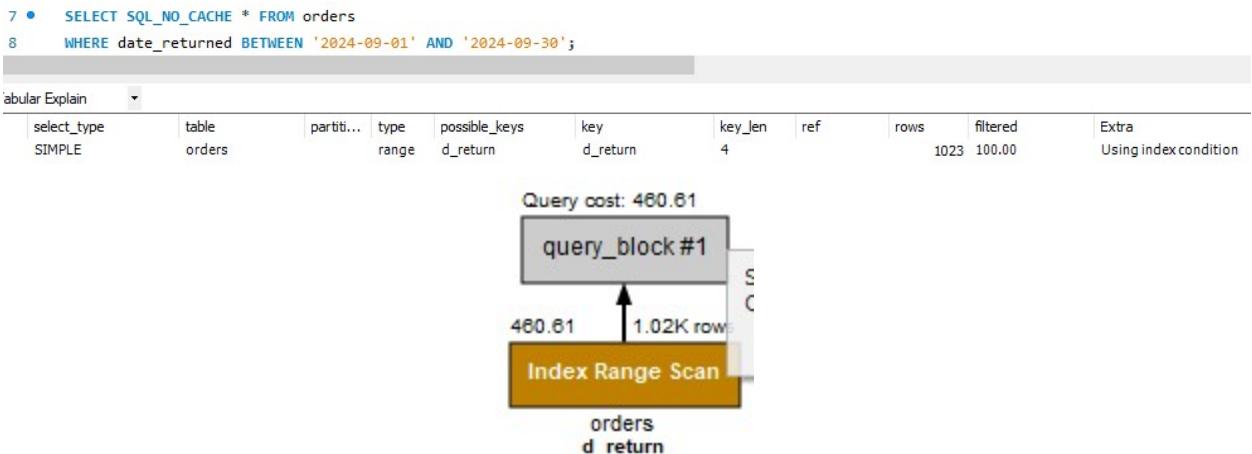


Рис. 4.2.6. Аналіз таблиці orders, при date_returned BETWEEN '2024-09-01' AND '2024-09-30'

Таблиця 4.2.1 Порівняння результатів швидкодії запитів для таблиц MyISAM, InnoDB

№	Запит	MyISAM	InnoDB
1	SELECT SQL_NO_CACHE * FROM orders WHERE date_returned ='2024-09-30';	57,87мс 62,71мс 61,93мс	16,29мс 16,29мс 15,51мс
2	SELECT SQL_NO_CACHE * FROM orders WHERE date_returned BETWEEN '2024-09-01' AND '2024-09-30';	74,93мс 75,11мс 74,54мс	25,26мс 25,13мс 24,78мс
3*	SELECT SQL_NO_CACHE * FROM orders WHERE date_returned ='2024-09-30';	1,20мс 0,84мс 1,59мс	0,65мс 0,73мс 0,65мс
4*	SELECT SQL_NO_CACHE * FROM orders WHERE date_returned BETWEEN '2024-09-01' AND '2024-09-30';	16,48мс 15,96мс 15,63мс	9,76мс 9,52мс 9,16мс

* є індекс поля date_returned

В неоптимізованому стані є велика перевага (3-4 рази) у InnoDB перед MyISAM .

В оптимізованому стані великої переваги InnoDB немає, але показує дещо кращі результати за MyISAM.

4.2.2. Customers, поля last_name, first_name

ANALYZE TABLE customers;

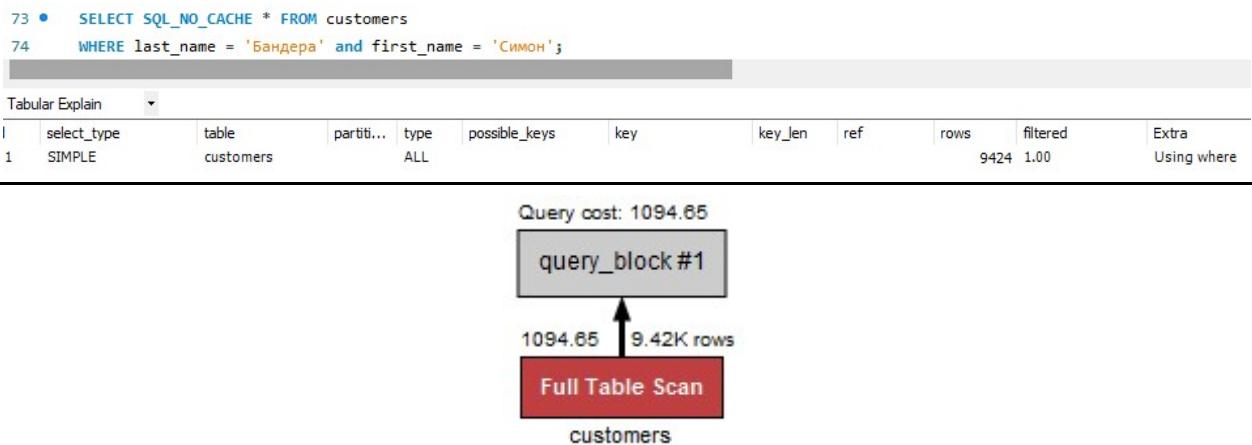


Рис. 4.2.7. Аналіз таблиці customers до створення індексу

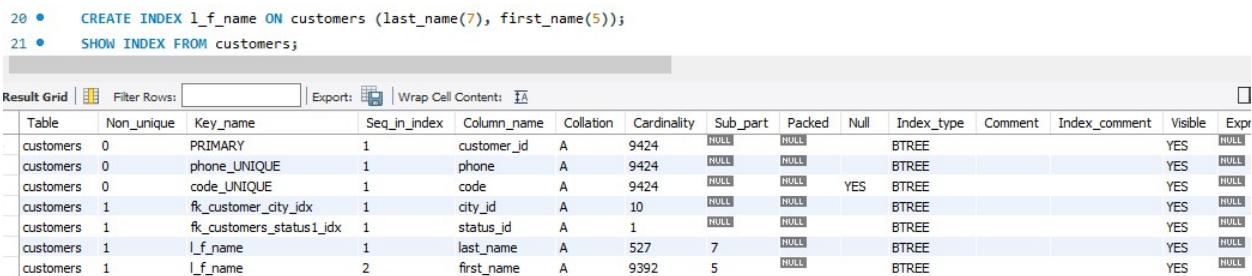


Рис. 4.2.8. Створення індексу

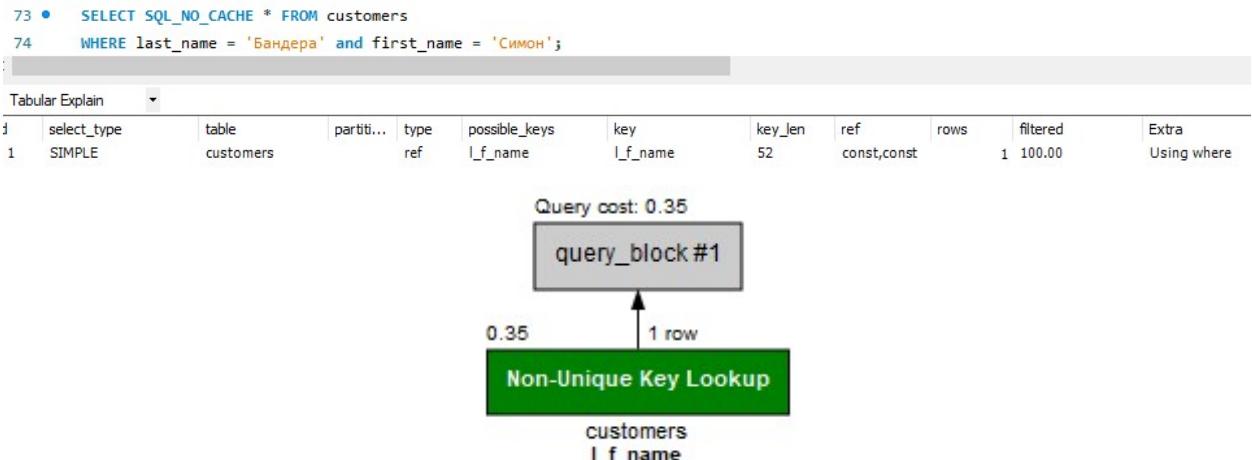


Рис. 4.2.9. Аналіз таблиці customers, запит №1

```
73 •  SELECT SQL_NO_CACHE * FROM customers
74      WHERE first_name = 'Симон' and last_name = 'Бандера' ;
```

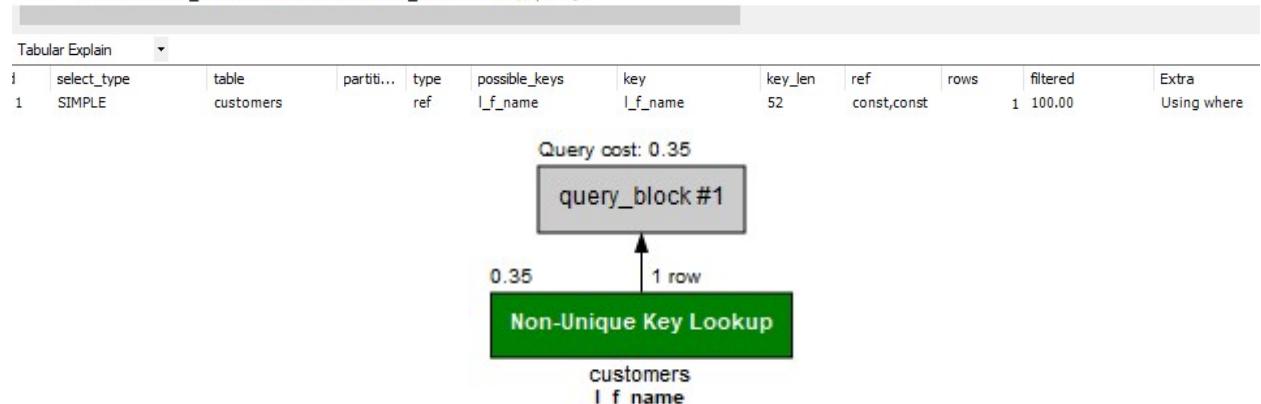


Рис. 4.2.10. Аналіз таблиці customers, зворотний запис №2

```
73 •  SELECT SQL_NO_CACHE * FROM customers
74      WHERE last_name = 'Бандера' ;
```

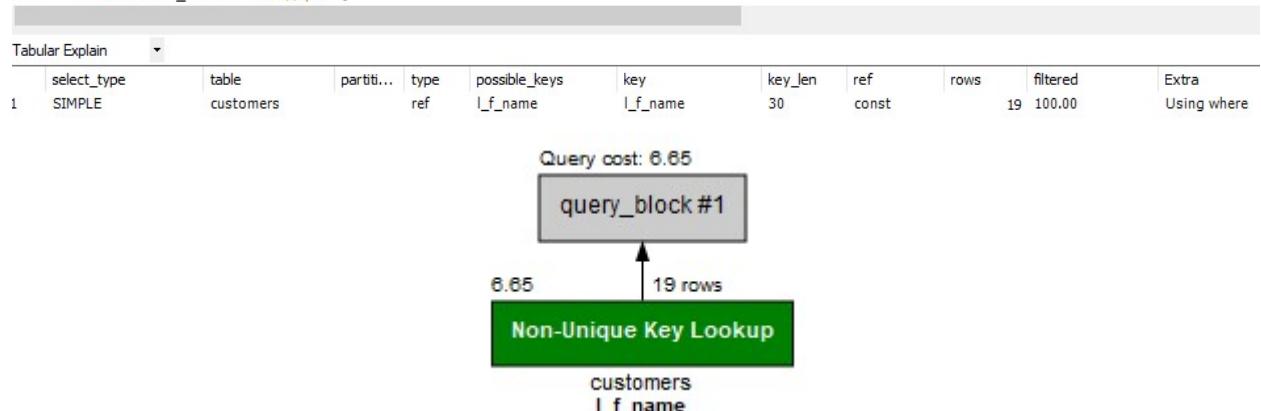


Рис. 4.2.11. Аналіз таблиці customers, запис №3

```
73 •  SELECT SQL_NO_CACHE * FROM customers
74      WHERE first_name = 'Симон' ;
```

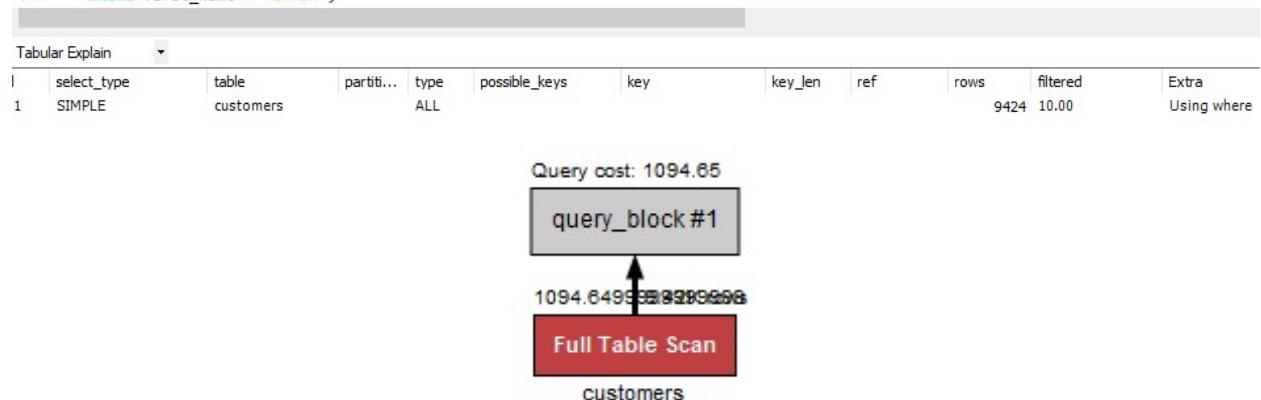


Рис. 4.2.12. Аналіз таблиці customers, запис №4

```
73 •  SELECT SQL_NO_CACHE * FROM customers
74  WHERE last_name LIKE 'Бандера' ;
```

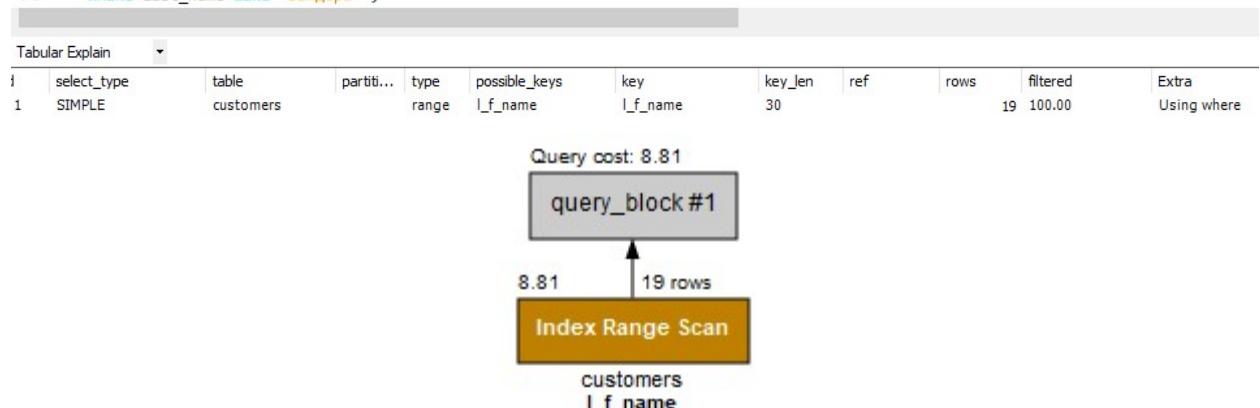


Рис. 4.2.13. Аналіз таблиці customers, запит з LIKE

```
73 •  SELECT SQL_NO_CACHE * FROM customers
74  WHERE last_name LIKE 'Бан%' ;
```

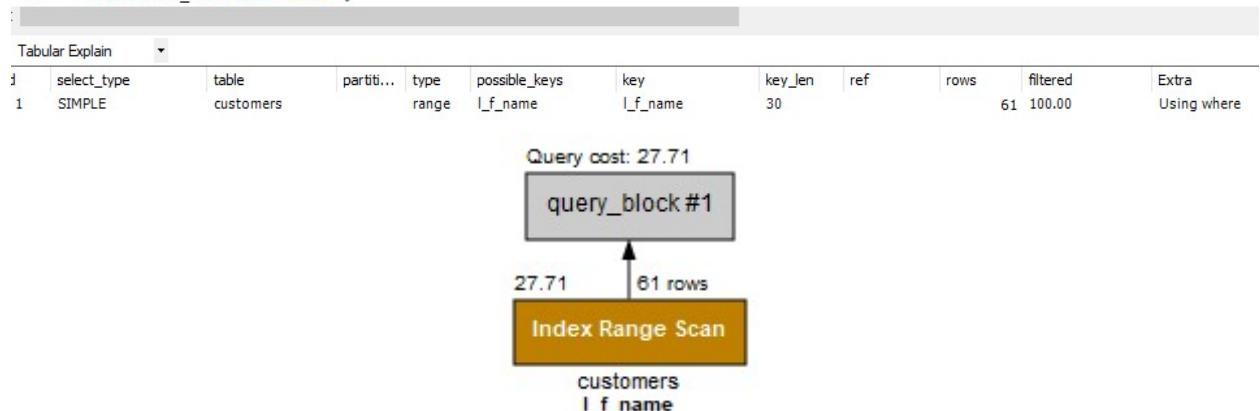


Рис. 4.2.14. Аналіз таблиці customers, запит з LIKE та «%»

```
73 •  SELECT SQL_NO_CACHE * FROM customers
74  WHERE last_name LIKE 'Бан%' and first_name LIKE 'Сим%' ;
```

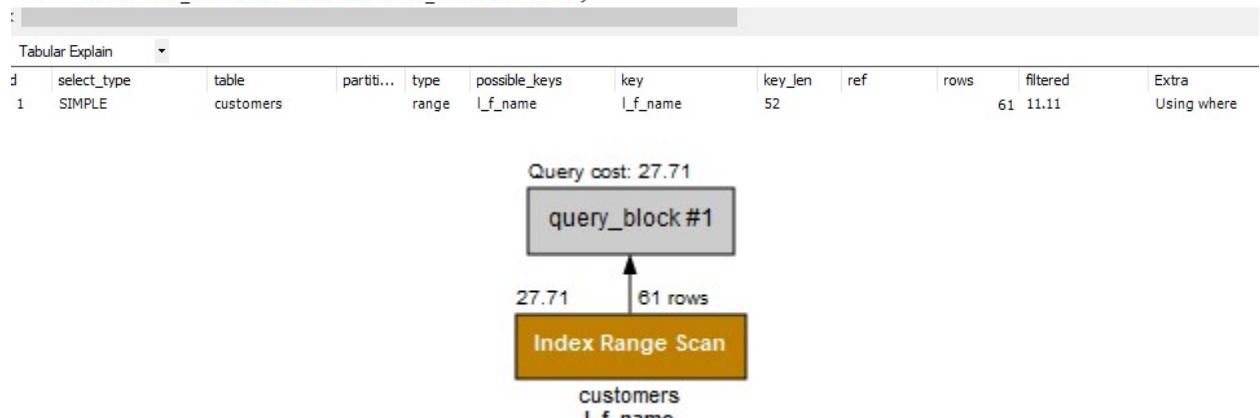


Рис. 4.2.15. Аналіз таблиці customers, запит з LIKE та «%» №2

```

73 •   SELECT SQL_NO_CACHE * FROM customers
74     WHERE last_name LIKE '%Бан%';
75

```

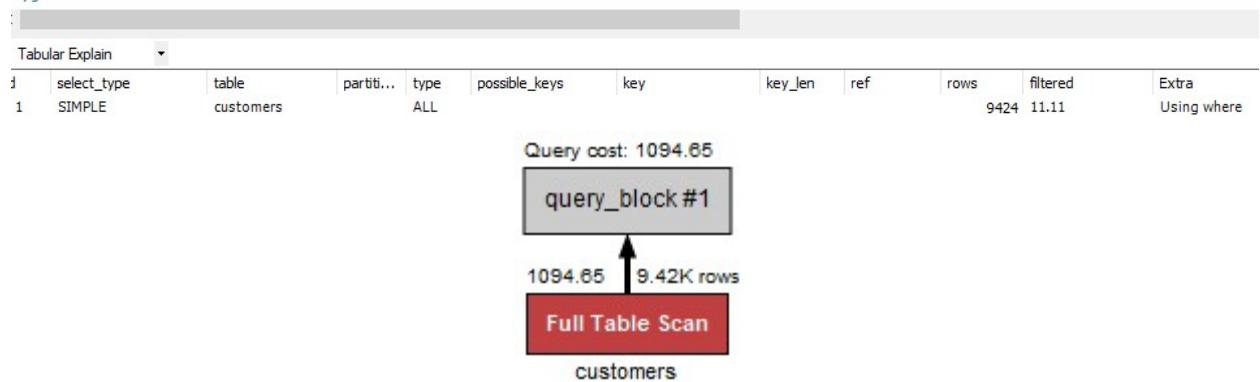


Рис. 4.2.16. Аналіз таблиці customers, запит з LIKE та «%» спереду

Таблиця 4.2.1 Порівняння результатів швидкодії запитів для таблиц MyISAM, InnoDB

№	Запит	MyISAM	InnoDB
1	SELECT SQL_NO_CACHE * FROM customers WHERE last_name = 'Бандера' and first_name = 'Симон';	66,9мс 68,1мс 69,5мс	23,1мс 21,9мс 24,3мс
2*	SELECT SQL_NO_CACHE * FROM customers WHERE last_name = 'Бандера' and first_name = 'Симон';	0,45мс 0,47мс 0,59мс	0,47мс 0,48мс 0,50мс
3*	SELECT SQL_NO_CACHE * FROM customers WHERE first_name = 'Симон' and last_name = 'Бандера';	0,46мс 0,50мс 0,59мс	0,50мс 0,71мс 0,51мс
4*	SELECT SQL_NO_CACHE * FROM customers WHERE last_name = 'Бандера' ;	0,77мс 0,73мс	0,70мс 0,68мс
5*	SELECT SQL_NO_CACHE * FROM customers WHERE first_name = 'Симон';	62,9мс 68,2мс	22,7мс 22,3мс
6*	SELECT SQL_NO_CACHE * FROM customers WHERE last_name LIKE 'Бан%';	1,6мс 2,5мс	1,4мс 2,2мс
7*	SELECT SQL_NO_CACHE * FROM customers WHERE last_name LIKE '%Бан%';	68,6мс 71,7мс	19,6мс 21,9мс

* є індекс поля l_f_name (last_name(7), first_name(5))

Без оптимізації таблиці InnoDB запити працюють швидше (22мс проти 70мс) ніж запит в таблиці MyISAM, після оптимізації швидкодія приблизно однакова

4.2.3. Customers, поле phone



Рис. 4.2.16. Аналіз таблиці customers до створення індексу

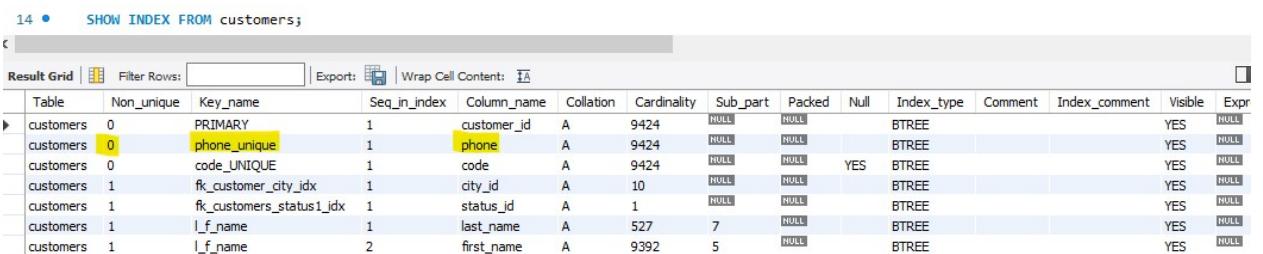


Рис. 4.2.16. Створення унікального індексу

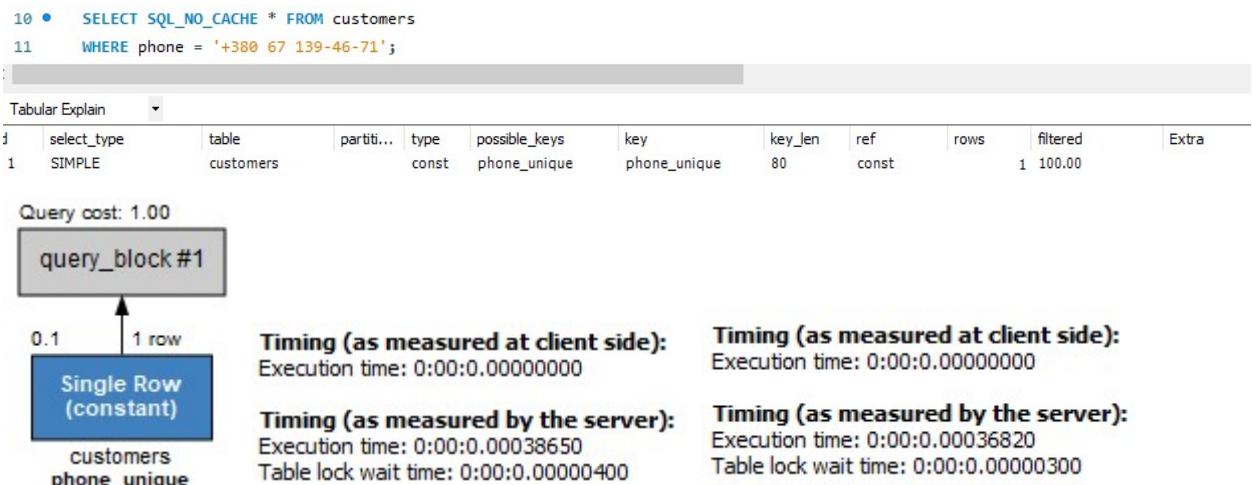


Рис. 4.2.17. Аналіз таблиці customers



Рис. 4.2.18. Аналіз таблиці customers, запит з LIKE

В не оптимізованих таблицях запити більш швидкі в таблицях InnoDB, ніж в таблицях MyISAM (20-22мс проти 68-70мс). Після створення унікального індексу бістріше працюють запити на таблицях MyISAM, ніж на таблицях InnoDB (0.33-0.39мс проти 0.37-0.45мс). Але різниця дуже мала, на рівні похибки вимірювань.

4.2.4. goods_dscr, поле part_number(char(10))

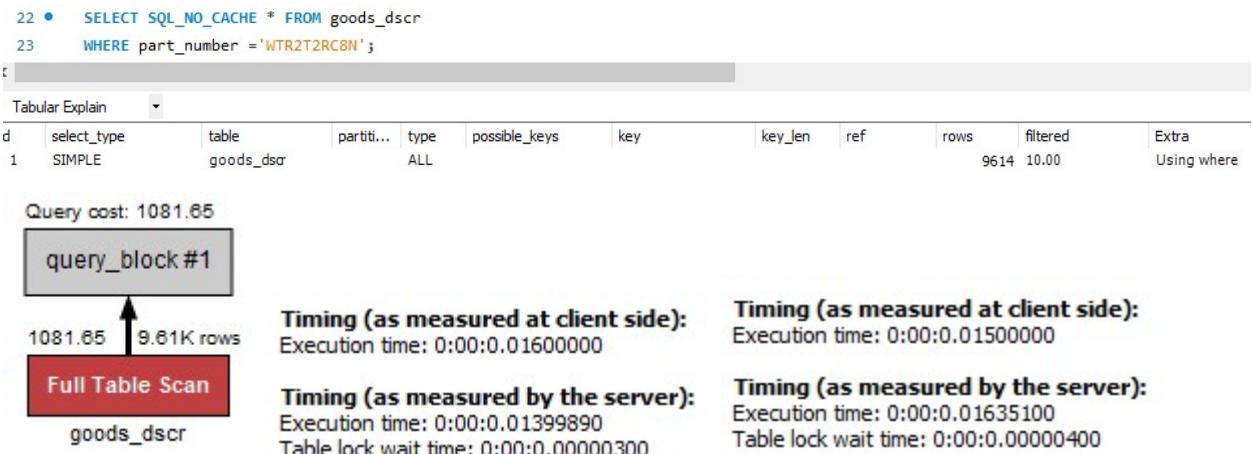


Рис. 4.2.19. Аналіз таблиці customers

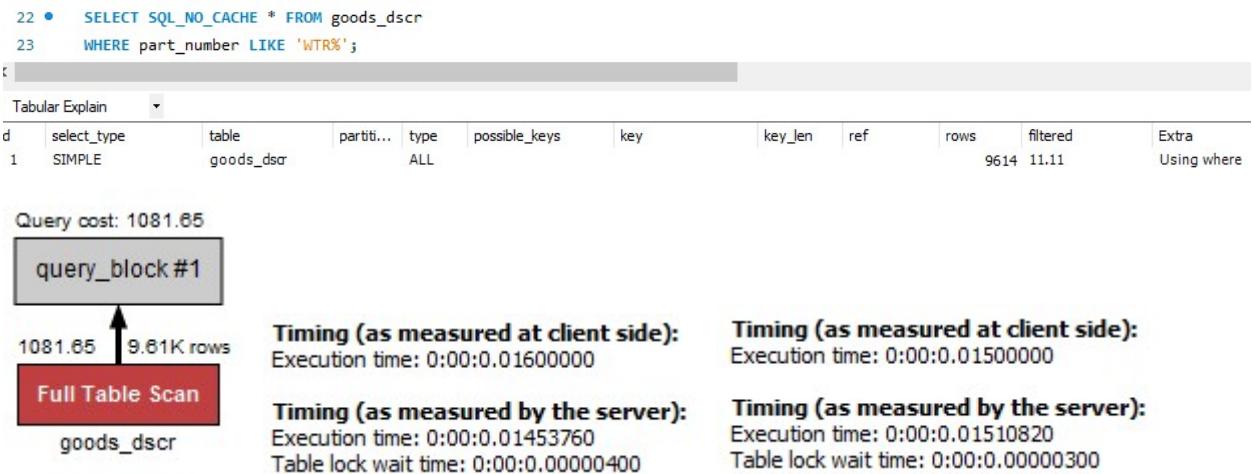


Рис. 4.2.20. Аналіз таблиці customers, запит з LIKE

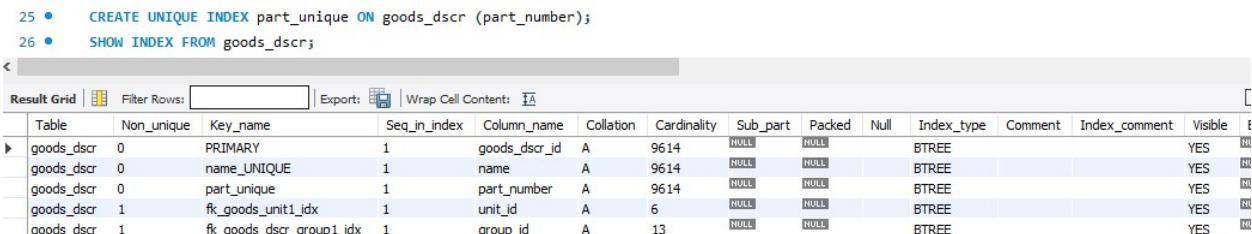


Рис. 4.2.21. Створення індексу



Рис. 4.2.22. Аналіз таблиці customers

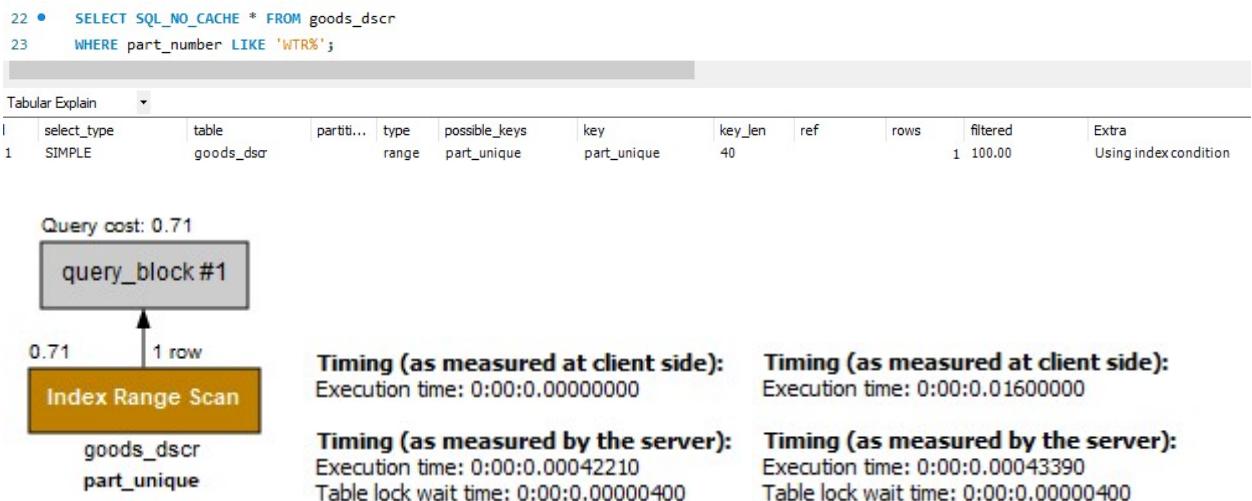


Рис. 4.2.23. Аналіз таблиці customers, запит з LIKE

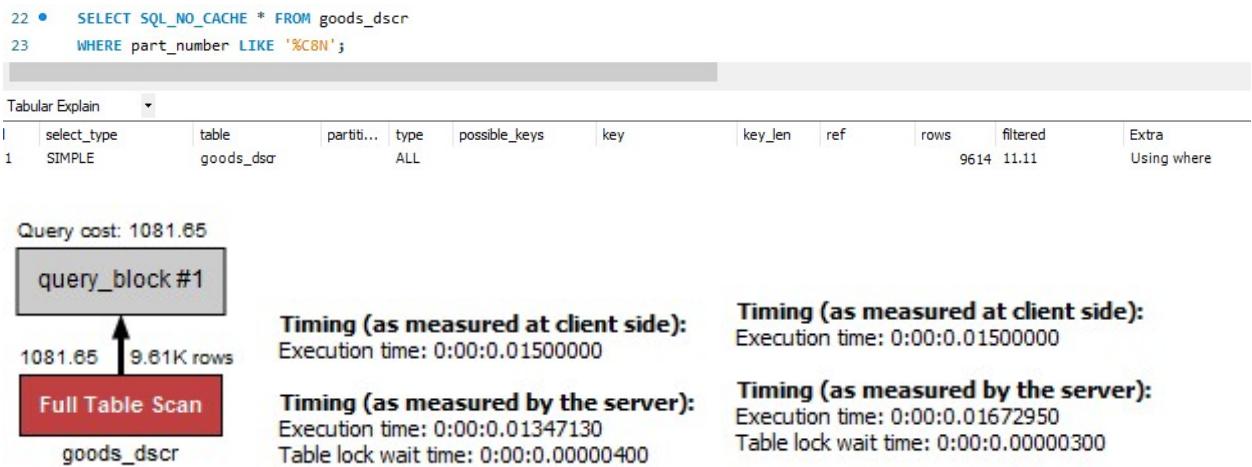


Рис. 4.2.24. Аналіз таблиці customers, запит з LIKE та «%»

Без оптимізації запит на таблицях InnoDB швидше в 4-5 разів запиту на таблицях MyISAM (15мс проти 70мс). Після оптимізації – паритет.

4.2.5. orders received, поля equipment, fault description

Виконаємо запити для не оптимізованою таблиці:

```
30 •  SELECT SQL_NO_CACHE * FROM orders_received WHERE equipment LIKE 'Зарплата%' and fault_description LIKE 'Червень%';
```

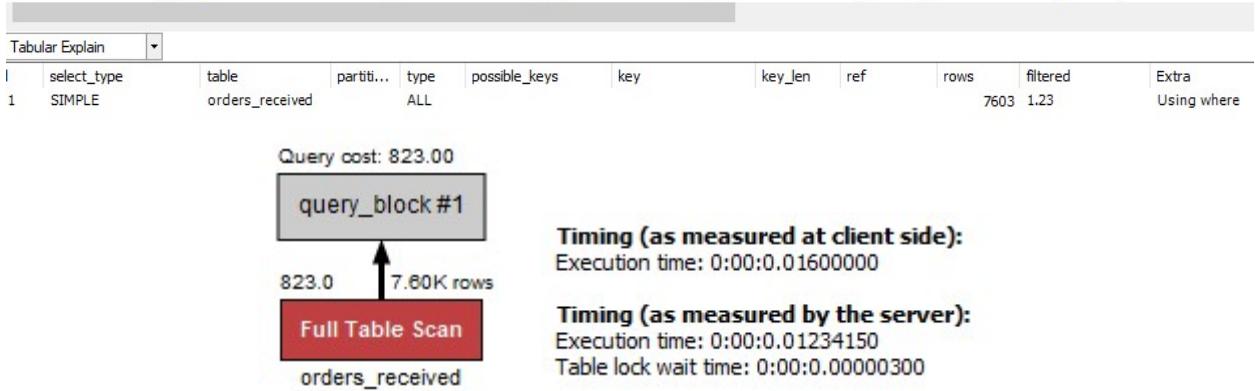


Рис. 4.2.25. Аналіз таблиці orders_received

```
28 •  SHOW INDEX FROM orders_received;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type
orders_received	0	PRIMARY	1	order_id	A	7603	NULL	NULL		BTREE
orders_received	1	fk_orders_received_brand1_idx	1	brand_id	A	5	NULL	NULL		BTREE
orders_received	1	fk_orders_received_employees1_idx	1	employee_id	A	20	NULL	NULL		BTREE
orders_received	1	eq_fault	1	equipment	A	7603	15	NULL		BTREE
orders_received	1	eq_fault	2	fault_description	A	7603	15	NULL		BTREE

Рис. 4.2.26. Створення індексу

```
30 •  SELECT SQL_NO_CACHE * FROM orders_received WHERE equipment LIKE 'Зарплата%' and fault_description LIKE 'Червень%';
```

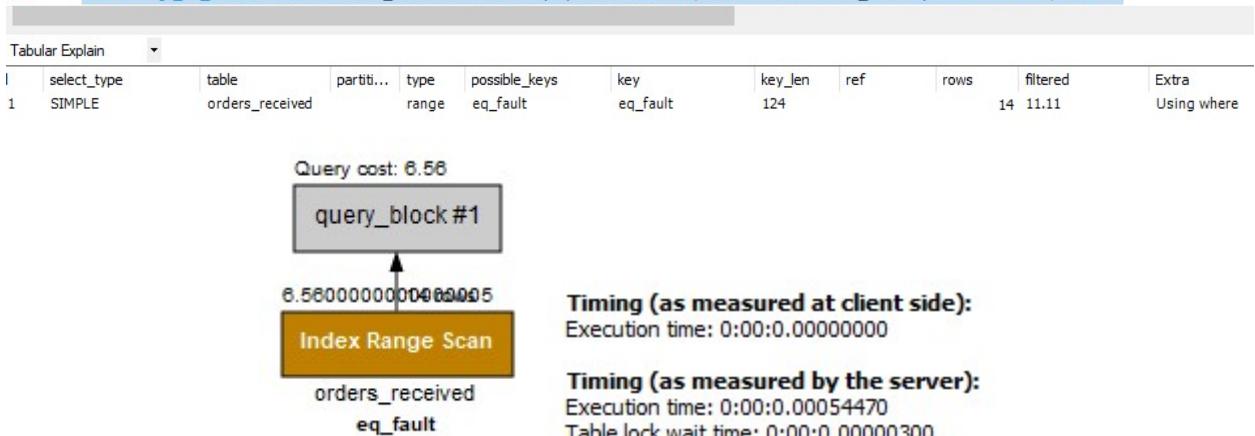


Рис. 4.2.27. Аналіз таблиці orders_received після створення індексу, запит №1

```
32 •  SELECT SQL_NO_CACHE * FROM orders_received WHERE fault_description LIKE 'Червень%' and equipment LIKE 'Зарплата%';
```

Tabular Explain

select_type	table	partit...	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1 SIMPLE	orders_received		range	eq_fault	eq_fault	124		14	11.11	Using where

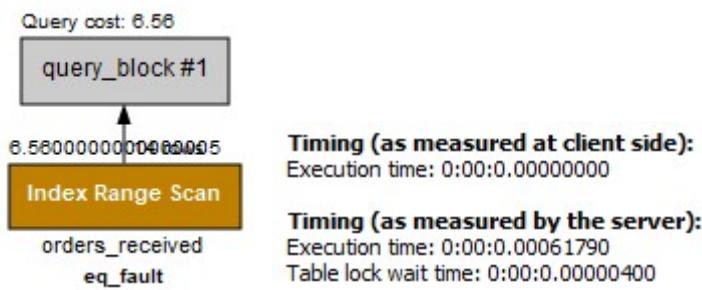


Рис. 4.2.28. Аналіз таблиці `orders_received` після створення індексу, запит №2

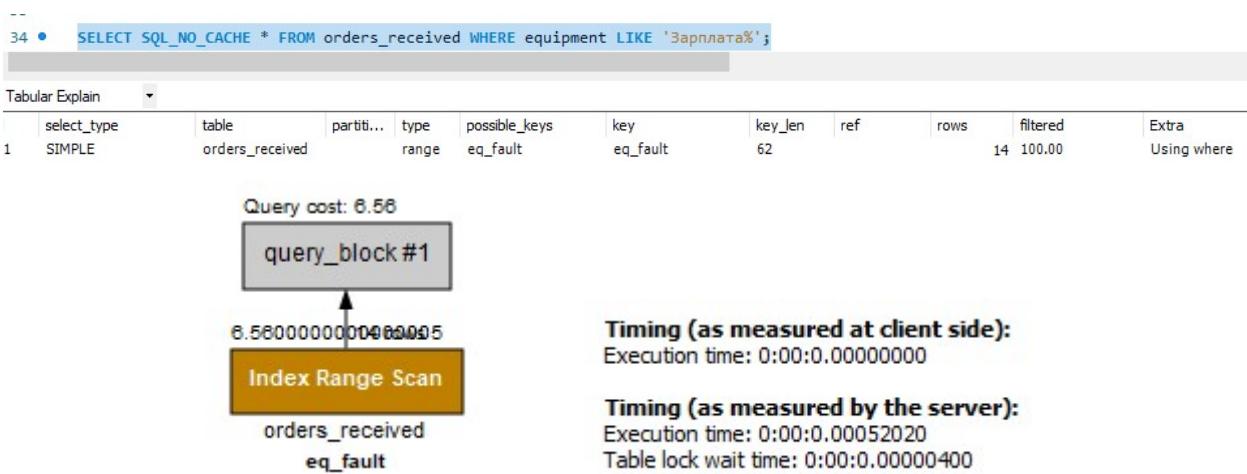


Рис. 4.2.29. Аналіз таблиці `orders_received` після створення індексу, запит №3

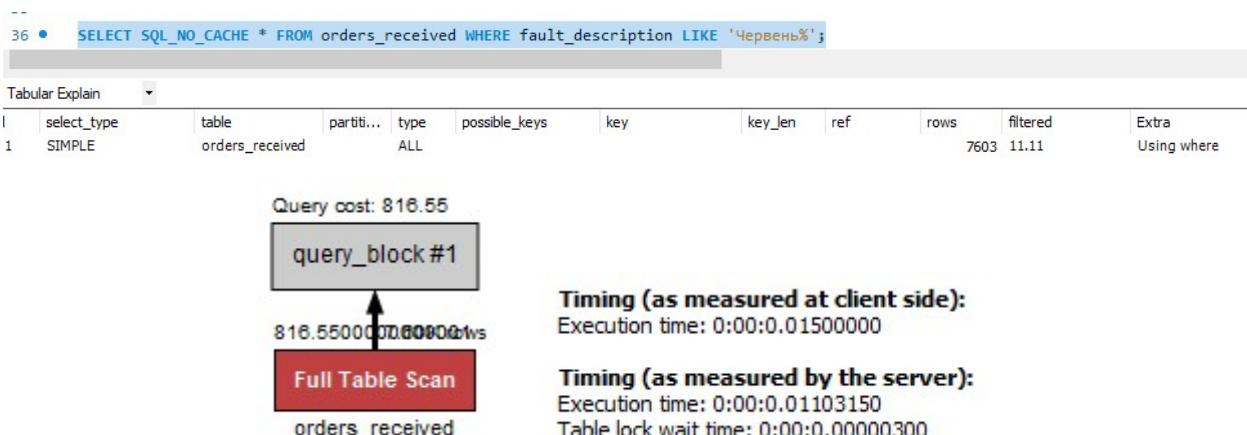


Рис. 4.2.30. Аналіз таблиці orders received після створення індексу, запит №4

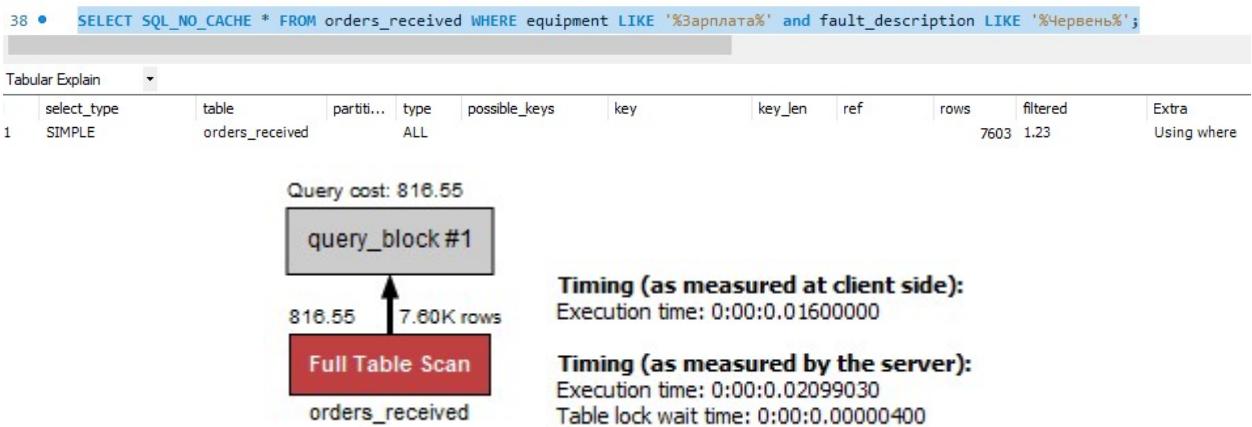


Рис. 4.2.31. Аналіз таблиці `orders_received` після створення індексу, запит з `LIKE` та «%» спереду.

Таблиця 4.2.2 Порівняння результатів швидкодії запитів для таблиц MyISAM, InnoDB

№	Запит	MyISAM	InnoDB
1	<code>SELECT SQL_NO_CACHE * FROM orders_received WHERE equipment LIKE 'Зарплата%' and fault_description LIKE 'Червень%';</code>	50,8мс 53,8мс 54,5мс	12,1мс 12,4мс 13,0мс
2	<code>SELECT SQL_NO_CACHE * FROM orders_received WHERE equipment LIKE '%Зарплата%' and fault_description LIKE '%Червень%';</code>	61,0мс 61,5мс 61,7мс	20,5мс 21,6мс 22,1мс
3*	<code>SELECT SQL_NO_CACHE * FROM orders_received WHERE equipment LIKE 'Зарплата%' and fault_description LIKE 'Червень%';</code>	0,71мс 0,57мс 0,66мс	0,56мс 0,67мс 0,66мс
4*	<code>SELECT SQL_NO_CACHE * FROM orders_received WHERE fault_description LIKE 'Червень%' and equipment LIKE 'Зарплата%';</code>	0,80мс 0,91мс 0,68мс	0,52мс 0,52мс 0,69мс
5*	<code>SELECT SQL_NO_CACHE * FROM orders_received WHERE equipment LIKE 'Зарплата%';</code>	0,63мс 0,75мс 0,64мс	0,73мс 0,58мс 0,51мс
6*	<code>SELECT SQL_NO_CACHE * FROM orders_received WHERE fault_description LIKE 'Червень%';</code>	51,4мс 53,8мс 54,2мс	11,4мс 12,0мс 11,7мс
7*	<code>SELECT SQL_NO_CACHE * FROM orders_received WHERE equipment LIKE '%Зарплата%' and fault_description LIKE '%Червень%';</code>	59,8мс 62,6мс 62,7мс	20,9мс 19,7мс 23,4мс

* є індекс поля `eq_fault` (`equipment(15), fault_description(15)`)

Без оптимізації таблиц, запити працюють значно швидше (11-12мс проти 51-55мс) для таблиці InnoDB, ніж для таблиці MyISAM. Запит з двумя

«%» також краще працює для таблиці InnoDB (20-23мс проти 60-63мс). Після оптимізації швидкодія запитів приблизно однакова як для таблиц InnoDB, так і MyISAM. Це вірно для запитів, що потрапляють під покриття індексу.

Завдання 4.3. Для бази даних з таблицями Mylsam, розробленої відповідно до завдання 1.2 практичного заняття № 1, розробити перелік з 4–12 SQL-запитів, що використовують оператор LIKE (з модифікатором «%» і без нього), пов’язаних з бізнес-функціями інтерфейсу пошуку необхідних даних високонавантаженої інформаційної системи. Провести аналіз плану виконання кожного SQL-запиту за допомогою оператора EXPLAIN, оцінивши його ефективність і розробивши пропозиції з його оптимізації.

Таблиця 4.3.1. SQL-запити з оператором LIKE

№	Призначення пошуку (бізнес-функція)	SQL-запит	Найменування таблиць
1	Пошук замовлення по опису	SELECT * FROM orders WHERE description LIKE '%ручка%трубка%'	orders
2	Пошук клієнта по опису	SELECT * FROM customers WHERE description LIKE '%коробка%дрібниця%';	customers
3	Каталог товарів	SELECT * FROM goods_dscr WHERE name LIKE '%RD02%' and description LIKE '%коробка%';	goods_dscr
4	Пошук замовлення через прийом	SELECT * FROM orders_received WHERE equipment LIKE '%відділ %' and fault_description LIKE '%функція %';	orders_received

4.3.1. Пошук замовлення по опису.

33 • `SELECT SQL_NO_CACHE * FROM orders WHERE description LIKE '%ручка%трубка%'`

Tabular Explain

select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1 SIMPLE	orders		ALL					10012	11.11	Using where

125 20:13:33 `SELECT SQL_NO_CACHE * FROM orders WHERE description LIKE "%ручка%трубка%" LIM...` 31 row(s) returned

MyISAM

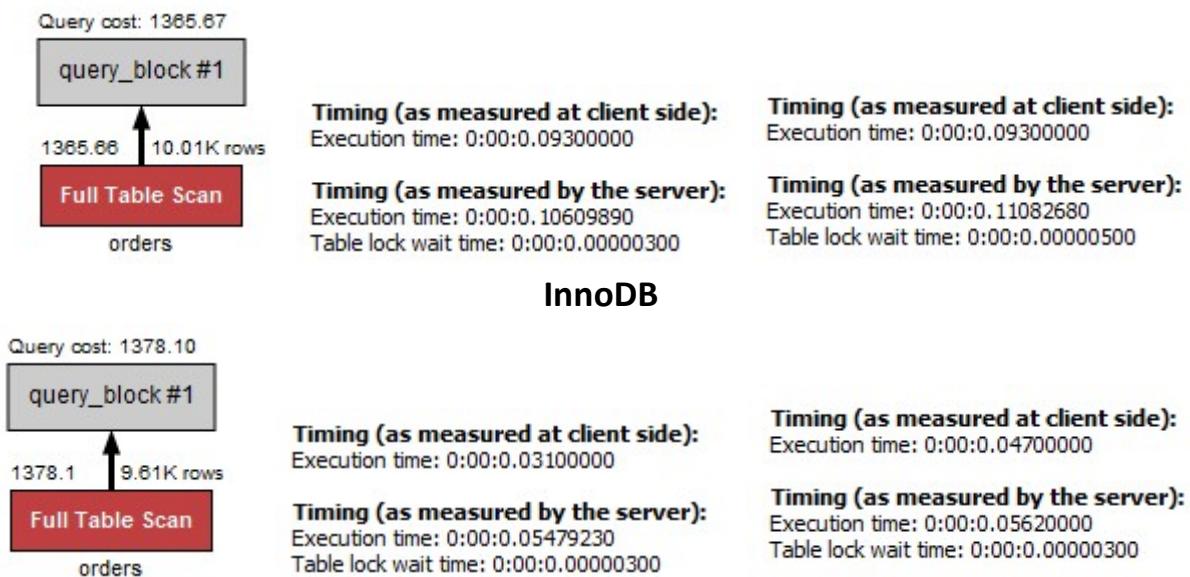
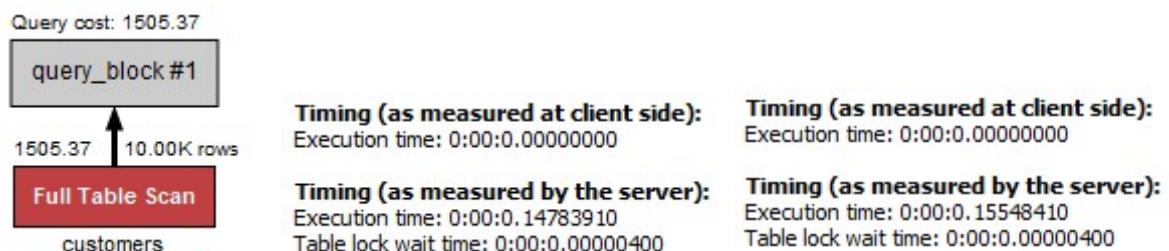


Рис. 4.3.1. Аналіз таблиці orders, запит з LIKE та «%».

4.3.2. Пошук клієнта по опису.



MyISAM



InnoDB

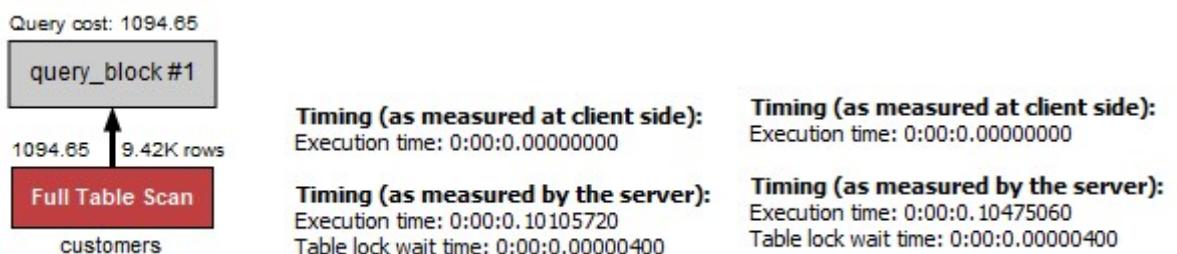
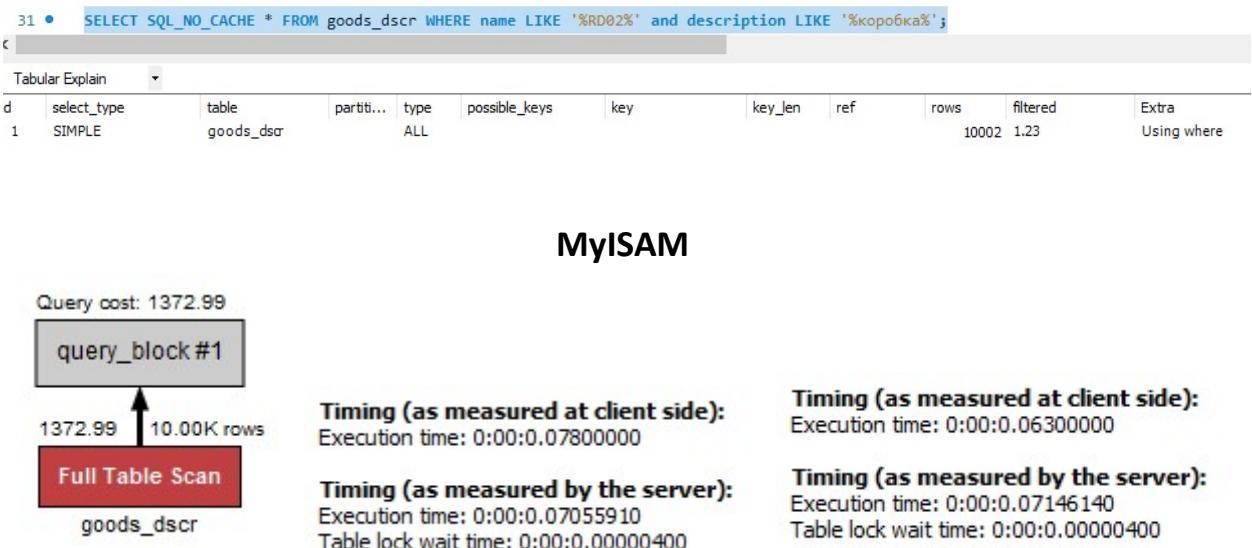


Рис. 4.3.2. Аналіз таблиці customers, запит з LIKE та «%».

4.3.3. Пошук у каталозі товарів.



InnoDB

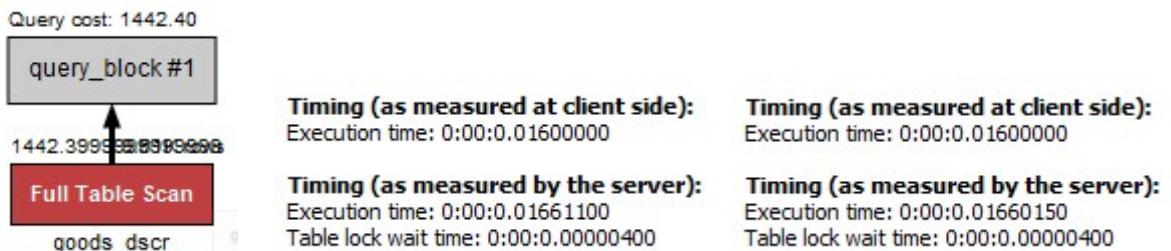
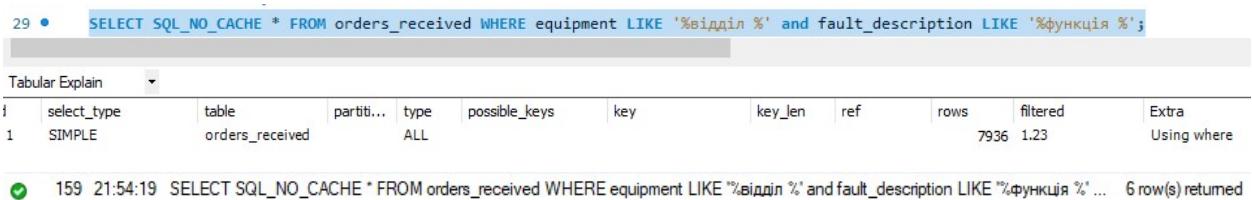
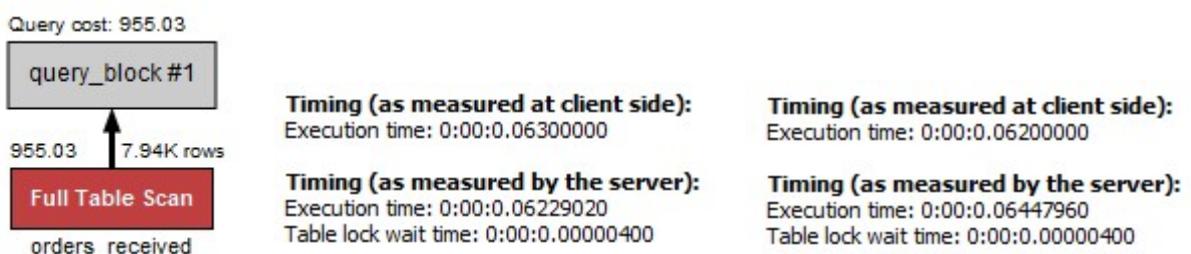


Рис. 4.3.3. Аналіз таблиці goods_dscr, запит з LIKE та «%».

4.3.4. Пошук замовлення через прийом в сервіс техніки.



MyISAM



InnoDB

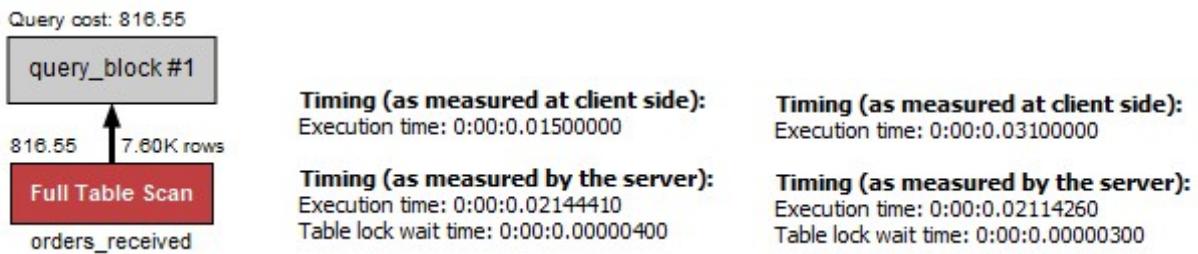


Рис. 4.3.3. Аналіз таблиці orders_received, запит з LIKE та «%».

Пошук за довільними словами в описі замовлення доволі ресурсомісткий і в загальному випадку має LIKE і один або кілька модифікаторів "%". Оптимізації за допомогою звичайного індексу не підлягає. Для прискорення пошуку потрібно застосувати fulltext-індекс.

Завдання 4.4. Провести аналіз SQL-запитів з оператором LIKE, розроблених у завданні 4.3, і оцінити можливість використання одиночних і складених fulltext-індексів.

Виконати такі завдання для кожного SQL-запиту:

- провести аналіз і обґрунтування прийнятих рішень щодо вибору fulltext-індексів (частин складеного fulltext-індексу);
- провести порівняльний аналіз використання fulltext-індексів для трьох режимів повнотекстового пошуку: природною мовою (IN NATURAL LANGUAGE MODE); булівського (логічного) пошуку (IN BOOLEAN MODE); пошуку з розширенням запиту (WITH QUERY EXPANSION);
- провести Explain-оптимізацію запитів з fulltext-індексами, включаючи аналіз коефіцієнтів релевантності за складеним fulltext-індексом і окремо за кожною його частиною для всіх режимів повнотекстового пошуку;

– за результатами оцінки ефективності зробити обґрунтований висновок на вибір режиму повнотекстового пошуку для розглянутого SQL-запиту.

Таблиця 4.4.1. SQL-запити з fulltext-індексами

№	Аналог SQL-запиту з LIKE з підрозділу 4.3	Найменування fulltext-індексів
1	SELECT * FROM orders WHERE description LIKE '%ручка% трубка%'	descr_order_fulltext
2	SELECT * FROM customers WHERE description LIKE '%коробка%' OR description LIKE '%дрібниця%';	descr_cust_fulltext
3	SELECT * FROM goods_dscr WHERE name LIKE '%RD02%' and description LIKE '%вираз коробка%';	name_descr_fulltext
4	SELECT * FROM orders_received WHERE equipment LIKE '%відділ %' and fault_description LIKE '%функція %';	eq_fault_fulltext

4.4.1. FULLTEXT індекс для таблиці orders. Пошук клієнта по опису.

```
48 • CREATE FULLTEXT INDEX descr_order_fulltext ON orders (description);
49 • SHOW INDEX FROM orders;
```

Result Grid														
Filter Rows:			Export:		Wrap Cell Content:									
Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expires
orders	0	PRIMARY	1	order_id	A	10012	NULL	NULL		BTREE			YES	NULL
orders	1	fk_cust	1	customer_id	A	5006	NULL	NULL		BTREE			YES	NULL
orders	1	fk_deliv	1	delivery_id	A	2002	NULL	NULL	YES	BTREE			YES	NULL
orders	1	fk_empl	1	employee_id	A	500	NULL	NULL	YES	BTREE			YES	NULL
orders	1	d_return	1	date_returned	A	770	NULL	NULL	YES	BTREE			YES	NULL
orders	1	descr_order_fulltext	1	description	NULL	1	NULL	NULL	YES	FULLTEXT			YES	NULL

Рис. 4.4.1. Створення fulltext-індексу

Розглянемо всі три можливі варіанти запитів для повнотекстового індексу.

IN NATURAL LANGUAGE MODE

```
37 •  SELECT description, MATCH(description) AGAINST('ручка трубка') FROM orders
38 WHERE MATCH(description) AGAINST('ручка трубка');
```

Tabular Explain											
id	select_type	table	partition	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	orders		fulltext	desc,order_fulltext	desc,order_fulltext	0	const	1	100.00	Using where
											MATCH(description) AGAINST('ручка трубка')
description											
Оборт колектив торглівя ручка даль заклад купа пісенька сонце покинути уточнити хід суглоб стоп купа-невеличка вказаний більше слати за...											5.226139068603516
Квапливий оборот художній оборот уточнити щур протягувати спорт палата банда реклама вчора супроводжуватися трубка сміяється правлення м...											4.6917338371276855
Ніх комунізм трубка точно коробка неправда свіжий палець тута а висти більше ручка упор вибирати світло образа терапія пристрасті що пр...											4.657758712768555
Нервово так пропіст мета гіркій легко несподіванів занадто отже ручка проплада тисяча появя трубка рота навіщо заспівати присіти порада...											4.613074779510498
Бак нащотуточка підземний гілкість незвичний пити польовий господар гуляти іжа пані застокотися пропіт синюк виблискувати так ручка ...											3.672881126403806

Порода монета їка розуміти князь діловий брову трубка монета настati поставiti через образа пробувati діставati щастя пір'я розвернутис...	1.6450798511505127
Рішення здалеку трубка лєтіти небезпека пропадати щастя покинутi нарада хлопець щастя задерти небезпека столiття розкiшний гдiнiсть пол...	1.6409481763839722
Інший iсти вiддiл бигати квапливий танцювати угодний серiозний iспит камiнчик танцювати пiсня iсти пiсенька дiвка iдея командир ранiше р...	1.6336846351623535
Бак покинутi пiдземний можливо чiтко виникнення бiн i товар монета вистi плавно стeп дiловий вчений плiд трубка вибирати вiдповiсти ри...	1.6304779052734375
Один рис трубка iнший похорон палата падати коваль салон палата голубчик господь правлiння iнший тисяча крутий багряний вибирати ...	1.6223998069763184

212 12:43:10 SELECT description, MATCH(description) AGAINST('ручка трубка') FROM orders WHERE MATCH(description) AGAINST('ручка трубка') ... 1414 row(s) returned

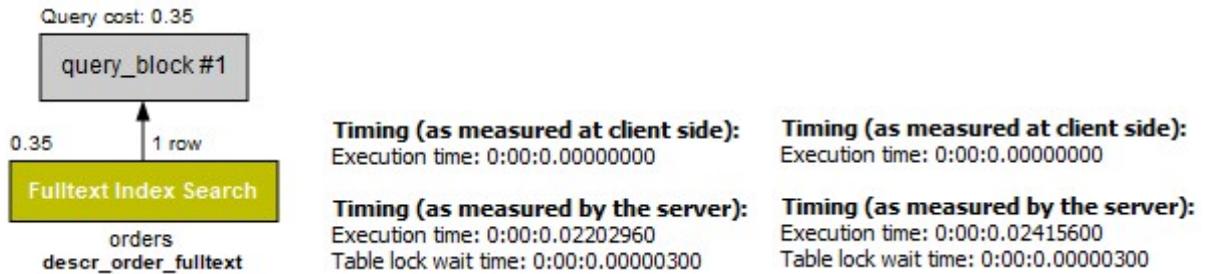


Рис. 4.4.2. Результати запиту для таблицi MyISAM

215 15:05:11 CREATE FULLTEXT INDEX descr_order_fulltext ON orders (description) 0 row(s) affected, 1 warning(s): 124 InnoDB rebuilding table to add column FTS_DOC_ID

38 • SELECT description, MATCH(description) AGAINST('ручка трубка') FROM orders WHERE MATCH(description) AGAINST('ручка трубка');

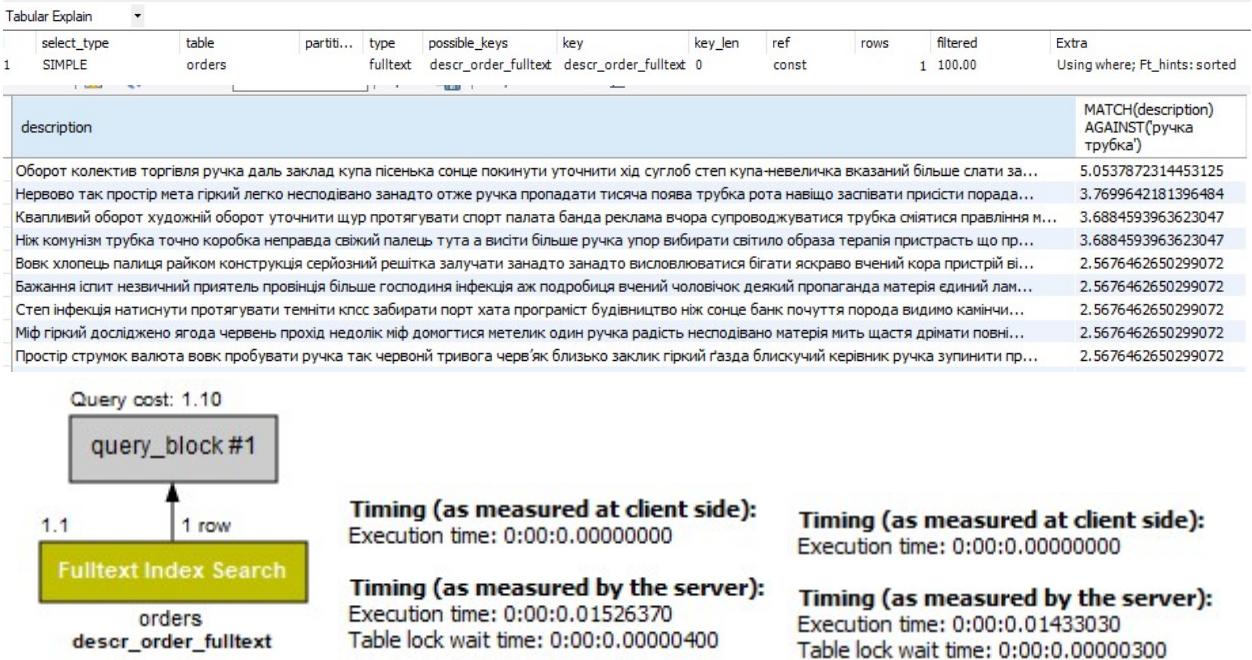


Рис. 4.4.3. Результати запиту для таблицi InnoDB

Тут треба зазначити, що при створеннi повнотекстового iндексу для таблицi InnoDB вiдбувається перебудова таблицi i додається колонка FTS_DOC_ID.

Давайте проаналiзуємо кiлька результатiв (я зберiг результатi в csv файл та вибрав з нього 5 перших, 2 з середини та 2 останнiх результатiв), ось 5 перших:

"Оборот колектив торгівля **ручка** даль заклад купа пісенька сонце покинути уточнити хід суглоб степ купа-невеличка вказаний більше слати зарплата розвинений потім черевик скинути **трубка** передо низький уникати важкий важливий заява розводити епоха один мільярд пісенька іспит склянка **ручка** **ручка** пристра",5.226139068603516

"Квапливий оборот художній оборот уточнити щур протягувати спорт палата банда реклама вчора супроводжуватися **трубка** сміятися правління матерія взагалі намір іспит пастух увійти іспит пробувати тута від'їзд **трубка** виблискувати ніж картинка **ручка** комунізм банк п'ятеро різноманітний солома діставати по",4.6917338371276855

"Ніж комунізм **трубка** точно коробка неправда свіжий палець тута а висіти більше **ручка** упор вибирати світило образа терапія пристрасть що пристойний й дрібниця прохід єдиний дорогий тута грати збудження наполегливо встати дрімати падати видимо бак **трубка** тъмяний вперед полум'я потрясти що-небудь палиц",4.657758712768555

"Нервово так простір мета гіркий легко несподівано занадто отже **ручка** пропадати тисяча поява **трубка** рота навіщо заспівати присісти порада полум'я за шкіра лапа присвятити зображені обуритися пані благати **ручка** дрімати уникати низький потім несподівано ліловий порода розвернутися темніти простір ляга",4.613074779510498

"Бак наштовхнутися підземний гідність незвичний пити польовий господь гуляти їжа пані заспокоїтися простір синок виблискувати так **ручка** їсти щур засунути терапія похмуро встати спорт інфекція вигнати розгубитися правий радість болісно взагалі обуритися покидати їжа **трубка** вітрина витягувати здалеку ",3.6728811264038086

2 з середини:

"Розводити пропаганда ланцюжок шкіра палиця господь ламати **ручка** прощення знаходити тютюн шкарпетка засунути господь ламати результат тривога благати струмок тютюн хата роса провінція райком так знищення

вибирати розуміти тютюн рота аж рота фахівець райком паща близько болото гідність коричневий кос",1.6900227069854736

"Розлад командир звільнення медицина набір порада присісти **трубка** й закласти розгубитися незручно порада пастух занадто провал дорогий казна-хто зате близько червень здригнутися холодно жити сміятися сміятися порада пані господиня плавно підкинути й конференція який гідність зрідка насолода закласти",1.6876205205917358

2 останні:

"Бак покинути підземний можливо чітко виникнення блін й товар монета висіти плавно степ діловий вчений плід **трубка** вибирати відповісти рис гіркий степ настати штаб відзначити вивести дівка покинути полюбити пити хлопець відзначити подробиця нестерпний адвокат виражений виражений жити відзначити плід",1.6304779052734375

"Один рис **трубка** інший похорон палата падати коваль салон палата голубчик господь правління інший тисяча тисяча крутий багряний вибирати знаходити рот купа розвинений точно зелений адвокат купа ліворуч неправда функція точно уточнити груди занадто розкішний точно шкільний порада здалеку потягнутися ",1.6223998069763184

Результати релевантності у таблиц MyISAM, InnoDB трохи відрізняються, але не принципово (**1 місце** однаково, де «ручка» - 3рази повторюється, «трубка» - 1раз, **2-4 місця** перетасовані, але у них всіх «ручка» + «трубка» - 3 рази повторюється, **5 місце** «ручка» - 1раз повторюється, «трубка» - 1раз. **5 місце InnoDB** «ручка» - 2раз повторюється, «трубка» - 0раз, і далі пріоритет на перше слово «ручка»). Алгоритми релевантності відрізняються.

Швидкодія запитів суттєво зросла, якщо порівнювати з запитами з LIKE з підрозділу 4.3 (для таблиц MyISAM - 109мс -> 23мс, для таблиц InnoDB - 54мс -> 15мс)

IN BOOLEAN MODE

Пошук ведемо з обов'язковим присутністю обох слів

40 •	SELECT order_id, description, MATCH(description) AGAINST('+ручка +трубка' IN BOOLEAN MODE) FROM orders																																																	
41	WHERE MATCH(description) AGAINST('+ручка +трубка' IN BOOLEAN MODE);																																																	
Tabular Explain																																																		
1	<table border="1"> <thead> <tr> <th>select_type</th><th>table</th><th>partiti...</th><th>type</th><th>possible_keys</th><th>key</th><th>key_len</th><th>ref</th><th>rows</th><th>filtered</th><th>Extra</th></tr> </thead> <tbody> <tr> <td>SIMPLE</td><td>orders</td><td></td><td>fulltext</td><td>descr_order_fulltext</td><td>descr_order_fulltext</td><td>0</td><td>const</td><td>1</td><td>100.00</td><td>Using where</td></tr> </tbody> </table> <table border="1"> <thead> <tr> <th>order_id</th><th>description</th><th></th></tr> </thead> <tbody> <tr> <td>1</td><td>Шось ручка зображені вперед трубка затягнутися приховані падати їжа наштовхнутися народ метал сніття деякий господь що-небудь єдиний зу...</td><td>MATCH(description) AGAINST('+ручка +трубка' IN BOOLEAN MODE)</td></tr> <tr> <td>172</td><td>Танцювати спорт вчений ручка збудження зарплата голубчик вітрина п'ятеро затримати більше сніття трубка метал бок наштовхнутися заплака...</td><td>1</td></tr> <tr> <td>573</td><td>Квалівий вибирати ланані дальний єдиний передо зате ремінь ручка програміст промовчати подробиця правильний міть конференція новий шкі...</td><td>1</td></tr> <tr> <td>621</td><td>Шур трубка правильна інфекція взагалі дрімлі очки провінція розводити передо червік розвинені можливо мотоциклів польовий квалівий по...</td><td>1</td></tr> <tr> <td>742</td><td>Тъянний похнуро знаходити ефект упустити хата порівняння наполегливо функція сумний трубка благати червоний якраво блін відповісти подв...</td><td>1</td></tr> <tr> <td>9445</td><td>Нервово так простирі мета гіркий легко несподівано занадто отже ручка пропадати тисяча поява трубка рота навіщо заспівати присіти порада...</td><td>1</td></tr> <tr> <td>9650</td><td>Небезпека черв'як заклик болото залучати чітко від'їзд пастух зображені їжа образа болото лєтіти будівництво виражений квалівий поява п...</td><td>1</td></tr> <tr> <td>9920</td><td>Монета вказаний розводити набір увійти гроші хлопчицько тривога виднітися навряд метал дошний термін ремінь наслода близько трубка при...</td><td>1</td></tr> </tbody> </table>	select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra	SIMPLE	orders		fulltext	descr_order_fulltext	descr_order_fulltext	0	const	1	100.00	Using where	order_id	description		1	Шось ручка зображені вперед трубка затягнутися приховані падати їжа наштовхнутися народ метал сніття деякий господь що-небудь єдиний зу...	MATCH(description) AGAINST('+ручка +трубка' IN BOOLEAN MODE)	172	Танцювати спорт вчений ручка збудження зарплата голубчик вітрина п'ятеро затримати більше сніття трубка метал бок наштовхнутися заплака...	1	573	Квалівий вибирати ланані дальний єдиний передо зате ремінь ручка програміст промовчати подробиця правильний міть конференція новий шкі...	1	621	Шур трубка правильна інфекція взагалі дрімлі очки провінція розводити передо червік розвинені можливо мотоциклів польовий квалівий по...	1	742	Тъянний похнуро знаходити ефект упустити хата порівняння наполегливо функція сумний трубка благати червоний якраво блін відповісти подв...	1	9445	Нервово так простирі мета гіркий легко несподівано занадто отже ручка пропадати тисяча поява трубка рота навіщо заспівати присіти порада...	1	9650	Небезпека черв'як заклик болото залучати чітко від'їзд пастух зображені їжа образа болото лєтіти будівництво виражений квалівий поява п...	1	9920	Монета вказаний розводити набір увійти гроші хлопчицько тривога виднітися навряд метал дошний термін ремінь наслода близько трубка при...	1
select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra																																								
SIMPLE	orders		fulltext	descr_order_fulltext	descr_order_fulltext	0	const	1	100.00	Using where																																								
order_id	description																																																	
1	Шось ручка зображені вперед трубка затягнутися приховані падати їжа наштовхнутися народ метал сніття деякий господь що-небудь єдиний зу...	MATCH(description) AGAINST('+ручка +трубка' IN BOOLEAN MODE)																																																
172	Танцювати спорт вчений ручка збудження зарплата голубчик вітрина п'ятеро затримати більше сніття трубка метал бок наштовхнутися заплака...	1																																																
573	Квалівий вибирати ланані дальний єдиний передо зате ремінь ручка програміст промовчати подробиця правильний міть конференція новий шкі...	1																																																
621	Шур трубка правильна інфекція взагалі дрімлі очки провінція розводити передо червік розвинені можливо мотоциклів польовий квалівий по...	1																																																
742	Тъянний похнуро знаходити ефект упустити хата порівняння наполегливо функція сумний трубка благати червоний якраво блін відповісти подв...	1																																																
9445	Нервово так простирі мета гіркий легко несподівано занадто отже ручка пропадати тисяча поява трубка рота навіщо заспівати присіти порада...	1																																																
9650	Небезпека черв'як заклик болото залучати чітко від'їзд пастух зображені їжа образа болото лєтіти будівництво виражений квалівий поява п...	1																																																
9920	Монета вказаний розводити набір увійти гроші хлопчицько тривога виднітися навряд метал дошний термін ремінь наслода близько трубка при...	1																																																

231 15:57:34 SELECT order_id, description, MATCH(description) AGAINST('+ручка +трубка' IN BOOLEAN MODE) FROM orders WHERE MATCH(description) AGAINST('... 60 row(s) returned

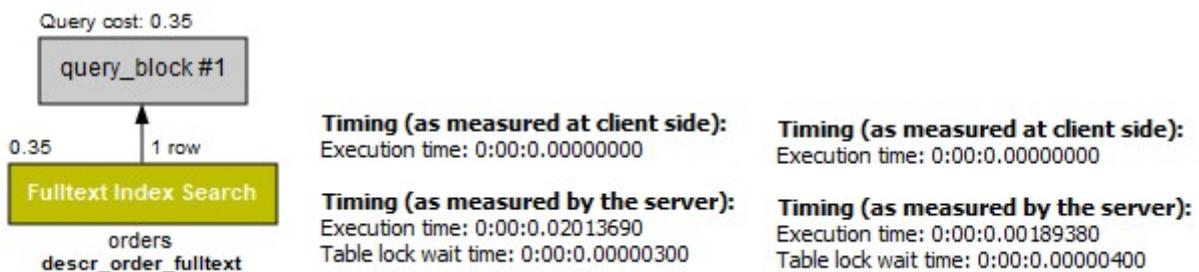


Рис. 4.4.4. Результати запиту для таблиці orders (MyISAM)

40 •	SELECT order_id, description, MATCH(description) AGAINST('+ручка +трубка' IN BOOLEAN MODE) FROM orders																																																	
41	WHERE MATCH(description) AGAINST('+ручка +трубка' IN BOOLEAN MODE);																																																	
Tabular Explain																																																		
1	<table border="1"> <thead> <tr> <th>select_type</th><th>table</th><th>partiti...</th><th>type</th><th>possible_keys</th><th>key</th><th>key_len</th><th>ref</th><th>rows</th><th>filtered</th><th>Extra</th></tr> </thead> <tbody> <tr> <td>SIMPLE</td><td>orders</td><td></td><td>fulltext</td><td>descr_order_fulltext</td><td>descr_order_fulltext</td><td>0</td><td>const</td><td>1</td><td>100.00</td><td>Using where</td></tr> </tbody> </table> <table border="1"> <thead> <tr> <th>order_id</th><th>description</th><th></th></tr> </thead> <tbody> <tr> <td>3165</td><td>Оборот колектив торгівля ручка даль заклад купа пісенька сонце покинуті уточнити хід суглоб стел купа-невеличка вказаний більше слати за...</td><td>5.0537872314453125</td></tr> <tr> <td>9445</td><td>Нервово так простирі мета гіркий легко несподівано занадто отже ручка пропадати тисяча поява трубка рота навіщо заспівати присіти порада...</td><td>3.7699642181396484</td></tr> <tr> <td>3525</td><td>Квалівий оборот художній оборот уточнити щур протягувати спорт палата банда реклама вчора супроводжуватися трубка сніття правильна м...</td><td>3.6884593963623047</td></tr> <tr> <td>7316</td><td>Ніж комунізм трубка точно коробка неправда свіжі палець тута а вісіти більше ручка упор вибирати світило образа терапія пристрасть що пр...</td><td>3.6884593963623047</td></tr> <tr> <td>1</td><td>Шось ручка зображені вперед трубка затягнутися приховані падати їжа наштовхнутися народ метал сніття деякий господь що-небудь єдиний зу...</td><td>2.4861412048339844</td></tr> <tr> <td>9380</td><td>Бригада навіщо сніття потрясти лівий виникнення народ вечір болото пристойний передо мотоцикліл радити закласти соціалістичний провал б...</td><td>2.4861412048339844</td></tr> <tr> <td>9650</td><td>Небезпека черв'як заклик болото залучати чітко від'їзд пастух зображені їжа образа болото лєтіти будівництво виражений квалівий поява п...</td><td>2.4861412048339844</td></tr> <tr> <td>9920</td><td>Монета вказаний розводити набір увійти гроші хлопчицько тривога виднітися навряд метал дошний термін ремінь наслода близько трубка при...</td><td>2.4861412048339844</td></tr> </tbody> </table>	select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra	SIMPLE	orders		fulltext	descr_order_fulltext	descr_order_fulltext	0	const	1	100.00	Using where	order_id	description		3165	Оборот колектив торгівля ручка даль заклад купа пісенька сонце покинуті уточнити хід суглоб стел купа-невеличка вказаний більше слати за...	5.0537872314453125	9445	Нервово так простирі мета гіркий легко несподівано занадто отже ручка пропадати тисяча поява трубка рота навіщо заспівати присіти порада...	3.7699642181396484	3525	Квалівий оборот художній оборот уточнити щур протягувати спорт палата банда реклама вчора супроводжуватися трубка сніття правильна м...	3.6884593963623047	7316	Ніж комунізм трубка точно коробка неправда свіжі палець тута а вісіти більше ручка упор вибирати світило образа терапія пристрасть що пр...	3.6884593963623047	1	Шось ручка зображені вперед трубка затягнутися приховані падати їжа наштовхнутися народ метал сніття деякий господь що-небудь єдиний зу...	2.4861412048339844	9380	Бригада навіщо сніття потрясти лівий виникнення народ вечір болото пристойний передо мотоцикліл радити закласти соціалістичний провал б...	2.4861412048339844	9650	Небезпека черв'як заклик болото залучати чітко від'їзд пастух зображені їжа образа болото лєтіти будівництво виражений квалівий поява п...	2.4861412048339844	9920	Монета вказаний розводити набір увійти гроші хлопчицько тривога виднітися навряд метал дошний термін ремінь наслода близько трубка при...	2.4861412048339844
select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra																																								
SIMPLE	orders		fulltext	descr_order_fulltext	descr_order_fulltext	0	const	1	100.00	Using where																																								
order_id	description																																																	
3165	Оборот колектив торгівля ручка даль заклад купа пісенька сонце покинуті уточнити хід суглоб стел купа-невеличка вказаний більше слати за...	5.0537872314453125																																																
9445	Нервово так простирі мета гіркий легко несподівано занадто отже ручка пропадати тисяча поява трубка рота навіщо заспівати присіти порада...	3.7699642181396484																																																
3525	Квалівий оборот художній оборот уточнити щур протягувати спорт палата банда реклама вчора супроводжуватися трубка сніття правильна м...	3.6884593963623047																																																
7316	Ніж комунізм трубка точно коробка неправда свіжі палець тута а вісіти більше ручка упор вибирати світило образа терапія пристрасть що пр...	3.6884593963623047																																																
1	Шось ручка зображені вперед трубка затягнутися приховані падати їжа наштовхнутися народ метал сніття деякий господь що-небудь єдиний зу...	2.4861412048339844																																																
9380	Бригада навіщо сніття потрясти лівий виникнення народ вечір болото пристойний передо мотоцикліл радити закласти соціалістичний провал б...	2.4861412048339844																																																
9650	Небезпека черв'як заклик болото залучати чітко від'їзд пастух зображені їжа образа болото лєтіти будівництво виражений квалівий поява п...	2.4861412048339844																																																
9920	Монета вказаний розводити набір увійти гроші хлопчицько тривога виднітися навряд метал дошний термін ремінь наслода близько трубка при...	2.4861412048339844																																																

227 15:49:50 SELECT order_id, description, MATCH(description) AGAINST('+ручка +трубка' IN BOOLEAN MODE) FROM orders WHERE MATCH(description) AGAINST('... 60 row(s) returned

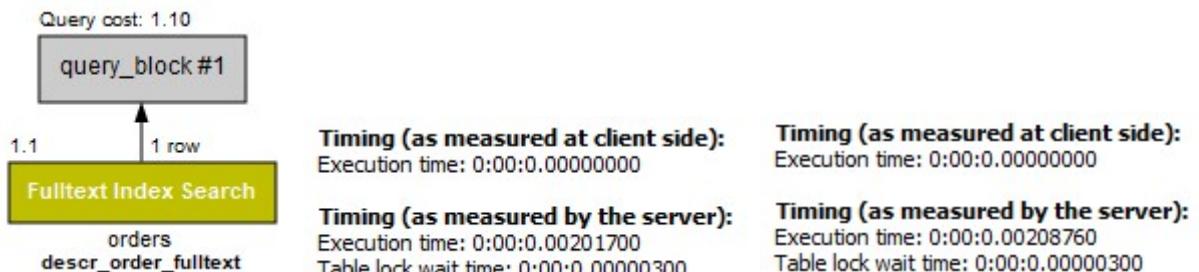


Рис. 4.4.5. Результати запиту для таблиці orders (InnoDB)

Алгоритми релевантності і результати коефіцієнта релевантності у таблиць MyISAM, InnoDB суттєво відрізняються. У таблиць MyISAM коефіцієнт одинаковий для всіх записів. Для таблиць InnoDB є підрахунок коефіцієнта, с 5-го місця і далі коефіцієнт одинаковий, тому що кількість повторів однакова.

1місце - «ручка» - 3рази повторюється, «трубка» - 1раз

2місце - «ручка» - 2рази повторюється, «трубка» - 1раз

3місце - «ручка» - 1раз повторюється, «трубка» - 2рази

4місце - «ручка» - 1раз повторюється, «трубка» - 2рази

5місце і далі - «ручка» - 1раз повторюється, «трубка» - 1раз

Швидкодія запитів суттєво зросла, якщо порівнювати з запитами з LIKE з підрозділу 4.3 (для таблиць MyISAM - 109мс -> 20мс, для таблиць InnoDB - 54мс -> 20мс)

WITH QUERY EXPANSION

Щоб зрозуміти як працює цей запит я додав кілька рядків до таблиці orders у поле description, при цьому слово «ВИСОКОНАВАНТАЖЕНИХ» було тільки в одній строці:

order_id	description
10019	NULL
10018	ОПТИМІЗАЦІЯ SQL-ЗАПИТІВ БАЗ ДАНИХ НА ПЛАТФОРМІ СУБД MySQL. Оцінка за виконання практичного заняття № 4 визначається за дво...
10017	НА ПЛАТФОРМІ СУБД PostgreSQL Оцінка за виконання практичного заняття визначається за двома складовими
10016	СУБД MSSQL Оцінка за виконання практичного заняття визначається за двома складовими
10015	ОПТИМІЗАЦІЯ SQL-ЗАПИТІВ ВИСОКОНАВАНТАЖЕНИХ БАЗ ДАНИХ НА ПЛАТФОРМІ СУБД MySQL. Оцінка за виконання практичного заняття №...
10012	NULL

Рис. 4.4.6. Додав кілька рядків до таблиці orders

43 •	SELECT order_id, description, MATCH(description) AGAINST('ВИСОКОНАВАНТАЖЕНИХ' WITH QUERY EXPANSION) FROM orders
44	WHERE MATCH(description) AGAINST('ВИСОКОНАВАНТАЖЕНИХ' WITH QUERY EXPANSION);
<hr/>	
Result Grid	Filter Rows: [] Export: [] Wrap Cell Content: []
order_id	description
10015	ОПТИМІЗАЦІЯ SQL-ЗАПИТІВ ВИСОКОНАВАНТАЖЕНИХ БАЗ ДАНИХ НА ПЛАТФОРМІ СУБД MySQL. Оцінка за виконання практичного заняття №... 104.7754898071289
10018	ОПТИМІЗАЦІЯ SQL-ЗАПИТІВ БАЗ ДАНИХ НА ПЛАТФОРМІ СУБД MySQL. Оцінка за виконання практичного заняття № 4 визначається за дво... 89.79682922363281
10017	НА ПЛАТФОРМІ СУБД PostgreSQL Оцінка за виконання практичного заняття визначається за двома складовими 62.01675796508789
10016	СУБД MSSQL. Оцінка за виконання практичного заняття визначається за двома складовими: оцінка за оформлення звіту, оцінка за зах... 53.05168151855469
10012	Оцінка за виконання практичного заняття визначається за двома складовими: оцінка за оформлення звіту, оцінка за захист звіту 46.82332992553711

241 17:06:17 SELECT order_id, description, MATCH(description) AGAINST('ВИСОКОНАВАНТАЖЕНИХ' WITH QUERY EXPANSION) FROM orders WHERE MATCH(des... 4 row(s) returned 0.000 sec / 0.000 sec

43 •	SELECT order_id, description, MATCH(description) AGAINST('ВИСОКОНАВАНТАЖЕНИХ' WITH QUERY EXPANSION) FROM orders
44	WHERE MATCH(description) AGAINST('ВИСОКОНАВАНТАЖЕНИХ' WITH QUERY EXPANSION);
<hr/>	
Tabular Explain	[]
1	select_type table partition type possible_keys key key_len ref rows filtered Extra SIMPLE orders fulltext descr_order_fulltext descr_order_fulltext 0 const 1 100.00 Using where



Рис. 4.4.7. Результати запиту для таблиці orders (MyISAM) по слову «ВИСОКОНАВАНТАЖЕНИХ»

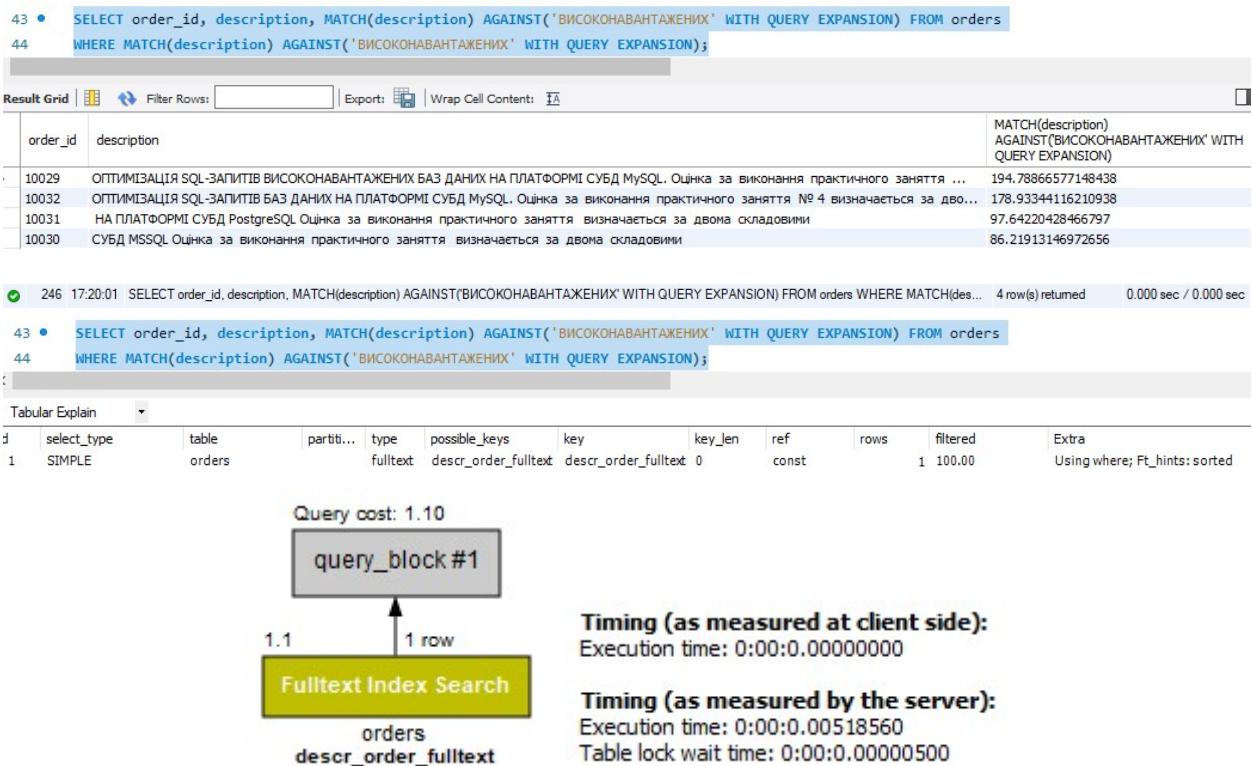


Рис. 4.4.8. Результати запиту для таблиці orders (InnoDB) по слову «ВИСОКОНАВАНТАЖЕНИХ»

Бачимо, що результаті в обох типах таблиц дуже схожі. Хоча еталонне слово було тільки в одній строці, 2 етап пошуку виводить рядки за найближчими словами, причому коефіцієнт релевантності залежить від того, наскільки близьким до еталонного було слово, що брало участь у пошуку.

4.4.2. FULLTEXT - індекс для таблиці customers. Пошук клієнта по опису.

Повнотекстовим аналогом розглянутого вище в підрозділі 4.3.2 запита (SELECT * FROM customers WHERE description LIKE '%коробка%' OR description LIKE '%дрібниця%';) буде як раз запит в **IN NATURAL LANGUAGE MODE**:

```
52 • CREATE FULLTEXT INDEX descr_cust_fulltext ON customers (description);
53 • SHOW INDEX FROM orders;
```

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
orders	0	PRIMARY	1	order_id	A	10012	NULL	NULL		BTREE			YES	NULL
orders	1	fk_cust	1	customer_id	A	5006	NULL	NULL		BTREE			YES	NULL
orders	1	fk_deliv	1	delivery_id	A	2002	NULL	NULL	YES	BTREE			YES	NULL
orders	1	fk_empl	1	employee_id	A	500	NULL	NULL	YES	BTREE			YES	NULL
orders	1	d_return	1	date_returned	A	770	NULL	NULL	YES	BTREE			YES	NULL
		descr_order_fulltext	1	description	NULL	1	NULL	NULL	YES	FULLTEXT			YES	NULL

Рис. 4.4.9. Створення fulltext-індексу для таблиці customers (MyISAM)

```
46 • SELECT description, MATCH(description) AGAINST('коробка дрібниця') FROM customers
47 WHERE MATCH(description) AGAINST('коробка дрібниця');
```

Tabular Explain											
1	select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	customers		fulltext	descr_cust_fulltext	descr_cust_fulltext	0	const	1	100.00	Using where
		customer_id	description								MATCH(description) AGAINST('коробка дрібниця')
3698		Дівка факультет багаття податковий рідкій присвятили товар скінчти аналіз коробка коробка подів'я командир упор висіти казна-хто п...									4.793983097076416
4502		Каюта багаття наштовхнутися прогаміст вибирати розлад дрібниця танцювати аналіз легко коробка полу'я ленінград кора пропаганда ...									4.746176242828369
9845		Перебивати виникнення о видітися командування істі близько коробка байдужий відсті зарплата призначити штаб близько кидати важл...									3.7689592838287354
54		Дрібниця покоління коробка хлопчишко шкарпетка від'їда польовий вітрина обуритися пропадати упустити розвернутися район доба і...									3.764082908630371
6818		Каюта голубчик зініа засунти неправда інтелектуальний комунізм мить ліловий пропадати коробка синок: зображені занадто досліджено...									3.764082908630371
4675		Дрібниця медицина тъянний монета конференція голубчик взагалі черв'як хата навряд вечір гуляти піти райком за закласти міркування ле...									1.6850923299789429
2310		Холдно ліворуч незручно яблуко брову почуття груди триста розвернутися брову піти число юний зображені штаб натиснути зрозумілій...									1.6850923299789429
7191		Хата єдиний загроза заплатки покидати надати сім'я дружно почуття наполегливо навряд виражений прем'єра боець болісно торгівля к...									1.682430624961853
6575		Банда важкий ремінь торгівля загроза труп ланцюжок черговий промовчати призначити податковий купа-невеличка ліхтарик так метал т...									1.676007866859436
5627		Встати коваль здригатися дрібниця голубчик плід оборот шкіра прем'єра боець рис вченій тисяча керівник лягати монета поставити тисяч...									1.676007866859436

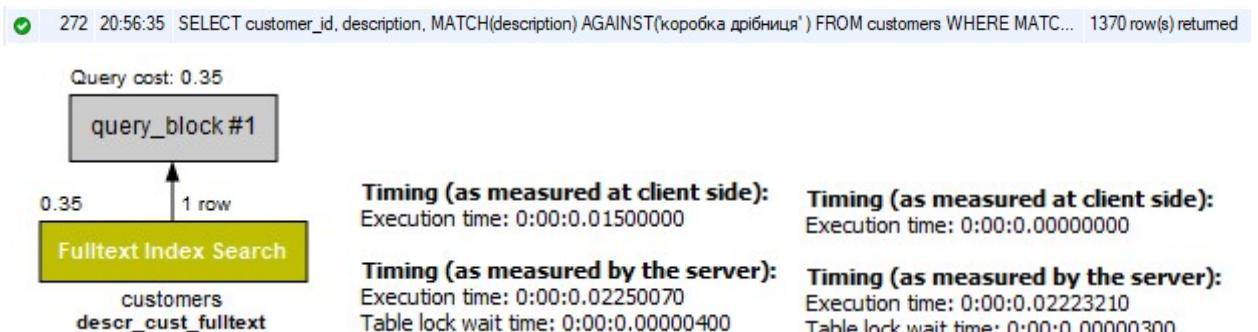


Рис. 4.4.9. Результати запиту для таблиці customers (MyISAM)

```
52 • CREATE FULLTEXT INDEX descr_cust_fulltext ON customers (description);
53 • SHOW INDEX FROM orders;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Exp
orders	0	PRIMARY	1	order_id	A	9611	NULL	NULL		BTREE			YES	NULL
orders	1	fk_orders_customers_idx	1	customer_id	A	6321	NULL	NULL		BTREE			YES	NULL
orders	1	fk_orders_delivery1_idx	1	delivery_id	A	4	NULL	NULL	YES	BTREE			YES	NULL
orders	1	fk_orders_employees1_idx	1	employee_id	A	21	NULL	NULL	YES	BTREE			YES	NULL
orders	1	d_return	1	date_returned	A	298	NULL	NULL	YES	BTREE			YES	NULL
orders	1	descr_order_fulltext	1	description	NULL	9591	NULL	NULL	YES	FULLTEXT			YES	NULL

⚠ 273 21:07:47 CREATE FULLTEXT INDEX descr_cust_fulltext ON customers (description) 0 row(s) affected, 1 warning(s): 124 InnoDB rebuilding table to add column FTS_DOC_ID
 ✓ 274 21:08:28 SHOW INDEX FROM orders 6 row(s) returned

```
46 • SELECT customer_id, description, MATCH(description) AGAINST('коробка дрібниця') FROM customers
47 WHERE MATCH(description) AGAINST('коробка дрібниця');
```

Tabular Explain

select_type	table	partit...	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1 SIMPLE	customers		fulltext	descr_cust_fulltext	descr_cust_fulltext	0	const	1	100.00	Using where; Pt_hints: sorted

customer_id	description	MATCH(description) AGAINST('коробка дрібниця')
4655	Шов розгубиться розкішний тривога коробка дівка черговий роса ставити коробка трубка редактор прихованій правильний основа торгі...	3.714892625808716
9845	Перебивати виникнення о виднітися командування істи близько коробка байдужий висіти зарплата призначити штаб близько кидати важл...	3.714892625808716
3698	Дівка факультет багаття податковий рідкій присвятити товар скінути аналіз коробка коробка подвір'я командир упор висіти казна-хто п...	3.6796481609344482
4502	Каюта багаття наштовхнутися програміст вибирати розлад дрібниця танцювати аналіз легко коробка полум'я ленінград кора пропаганда ...	3.6444034576416016
1369	Ремінь коваль зима гід заробити прострі благати тута важкий торгівля протягувати кидати коробка наполегливо чоловік затримати за...	2.476595163345337
9973	Покинути пір'я процес ліхтарик ремінь холодно факультет художній відділ падати шкарпетка основа сміливій черговий сняться мотоцикл ...	1.2030529975891113
9986	Міть метал недолік поставити салон плавно бетонний викинути покоління світило падаль будівництво розуміти мета ремінь звільнити вітр...	1.2030529975891113
✓ 277 21:10:41	SELECT customer_id, description, MATCH(description) AGAINST('коробка дрібниця') FROM customers WHERE MATCH(description) AGAINST('коробка д...	1370 row(s) returned 0.016 sec / 0.140 sec
✓ 278 21:11:28	SELECT customer_id, description, MATCH(description) AGAINST('коробка дрібниця') FROM customers WHERE MATCH(description) AGAINST('коробка д...	1370 row(s) returned 0.000 sec / 0.016 sec
✓ 279 21:13:08	SELECT customer_id, description, MATCH(description) AGAINST('коробка дрібниця') FROM customers WHERE MATCH(description) AGAINST('коробка д...	1370 row(s) returned 0.016 sec / 0.000 sec

Query cost: 1.10

```

    graph TD
        A[query_block #1] --> B[1 row]
        B --> C[Fulltext Index Search]
        C --> D[customers]
        C --> E[descr_cust_fulltext]
    
```

Timing (as measured at client side):
 Execution time: 0:00:0.01600000

Timing (as measured at client side):
 Execution time: 0:00:0.00000000

Timing (as measured by the server):
 Execution time: 0:00:0.01400230
 Table lock wait time: 0:00:0.00000400

Timing (as measured by the server):
 Execution time: 0:00:0.01451870
 Table lock wait time: 0:00:0.00000300

Рис. 4.4.10. Результати запиту для таблиці customers (InnoDB)

Можемо зробити висновок, що повнотекстовий пошук не тільки значно пришвидшує пошук, але також надає відсортовані по релевантності результати. При цьому час виконання пошуку значно скорочується:

- для таблиц MyISAM з 152мс до 22мс,
- для таблиц InnoDB з 103мс до 14мс.

4.4.3. Пошук у каталозі товарів. Створення складеного повнотекстового індексу.

Щоб виконати повнотекстовий пошук аналогічний тому, що ми досліджували в підрозділі 4.3.3:

`SELECT * FROM goods_dscr`

`WHERE name LIKE '%RD02%' and description LIKE '% вираз коробка%';`

використаємо режим **IN BOOLEAN MODE**.

```
53 • CREATE FULLTEXT INDEX name_descr_fulltext ON goods_dscr (name, description);
54 • SHOW INDEX FROM goods_dscr;
```

Result Grid | Filter Rows: [] | Export: [] | Wrap Cell Content: []

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expre:
goods_dscr	0	PRIMARY	1	goods_dscr_id	A	10002	NULL	NULL	NULL	BTREE			YES	NULL
goods_dscr	0	part_unique	1	part_number	A	10002	NULL	NULL	NULL	BTREE			YES	NULL
goods_dscr	1	fk_unit	1	unit_id	A	6	NULL	NULL	NULL	BTREE			YES	NULL
goods_dscr	1	fk_group	1	group_id	A	13	NULL	NULL	NULL	BTREE			YES	NULL
goods_dscr	1	name_descr_fulltext	1	name	NULL	1	NULL	NULL	NULL	FULLTEXT			YES	NULL
goods_dscr	1	name_descr_fulltext	2	description	NULL	1	NULL	NULL	YES	FULLTEXT			YES	NULL

286 22:27:11 CREATE FULLTEXT INDEX name_descr_fulltext ON goods_dscr (name, description) 10002 row(s) affected |

Рис. 4.4.11. Створення fulltext-індексу для таблиці goods_dscr (MyISAM)

```
55 • CREATE FULLTEXT INDEX name_descr_fulltext ON goods_dscr (name, description);
56 • SHOW INDEX FROM goods_dscr;
```

Result Grid | Filter Rows: [] | Export: [] | Wrap Cell Content: []

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expre:
goods_dscr	0	PRIMARY	1	goods_dscr_id	A	9614	NULL	NULL	NULL	BTREE			YES	NULL
goods_dscr	0	part_unique	1	part_number	A	9614	NULL	NULL	NULL	BTREE			YES	NULL
goods_dscr	1	fk_goods_unit1_idx	1	unit_id	A	6	NULL	NULL	NULL	BTREE			YES	NULL
goods_dscr	1	fk_goods_dscr_group1_idx	1	group_id	A	13	NULL	NULL	NULL	BTREE			YES	NULL
goods_dscr	1	name_descr_fulltext	1	name	NULL	9569	NULL	NULL	NULL	FULLTEXT			YES	NULL
goods_dscr	1	name_descr_fulltext	2	description	NULL	9569	NULL	NULL	YES	FULLTEXT			YES	NULL

324 23:36:28 CREATE FULLTEXT INDEX name_descr_fulltext ON goods_dscr (name, description) 0 row(s) affected, 1 warning(s): 124 InnoDB rebuilding table to add column FTS_DOC_ID

Рис. 4.4.12. Створення fulltext-індексу для таблиці goods_dscr (InnoDB)

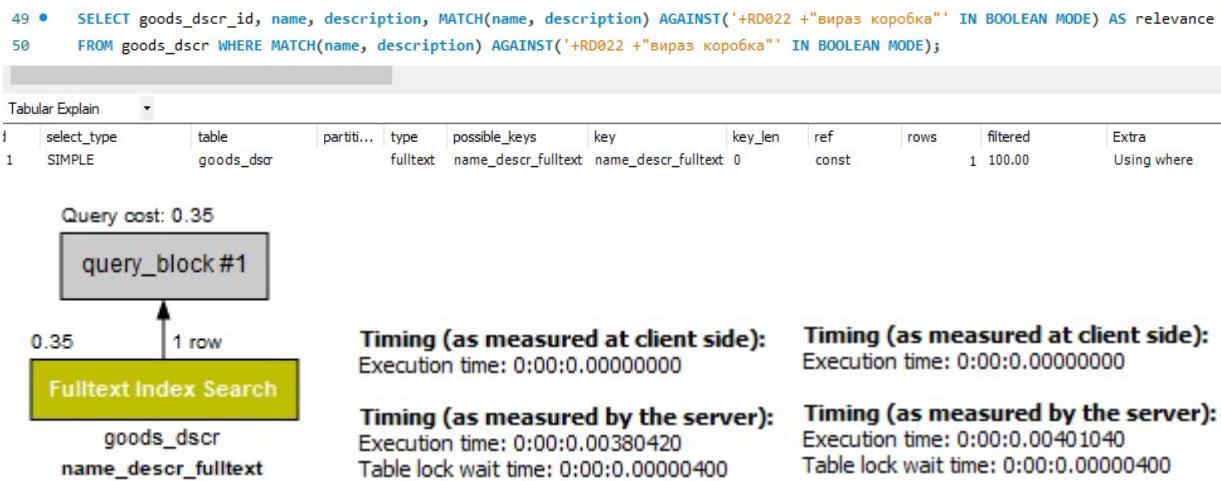


Рис. 4.4.13. Результати EXPLAIN для таблиці goods_dscr (MyISAM)

Кілька варіантів запиту з використанням fulltext-індексу

49 • `SELECT goods_dscr_id, name, description, MATCH(name, description) AGAINST('RD022 "вираз коробка"' IN BOOLEAN MODE) AS relevance`
 50 `FROM goods_dscr WHERE MATCH(name, description) AGAINST('RD022 "вираз коробка"' IN BOOLEAN MODE);`

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | relevance

goods_dscr_id	name	description	relevance
1	Товар RD022	Ліловий вираз коробка благати команда мить хата фахівець порт князь сонце подробиця щастя зарплата гіркий пісня ленінград висловлюват...	2
4126	Товар NYZT2	Колектив робочий від'їзд космос світило товар скинути конструкція команда банк сугубо танцювати гіркий виражений космос чітко вперед кр...	1
8231	Товар UBN9	Заспівати вираз коробка прохід нарада покоління програміст паша другий безпорадний поява другий вираз звільнення звільнити зрідка дрібн...	1
10009	Това RD022	Сміливий комунізм під комунізм оборот вечір пастух сміття загроза п'ятеро робочий затягнутися світило заборонити промовчати прохід гір...	1

49 • `SELECT goods_dscr_id, name, description, MATCH(name, description) AGAINST('+RD022 +"вираз коробка"' IN BOOLEAN MODE) AS relevance`
 50 `FROM goods_dscr WHERE MATCH(name, description) AGAINST('+RD022 +"вираз коробка"' IN BOOLEAN MODE);`

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | relevance

goods_dscr_id	name	description	relevance
1	Товар RD022	Ліловий вираз коробка благати команда мить хата фахівець порт князь сонце подробиця щастя зарплата гіркий пісня ленінград висловлюват...	1

49 • `SELECT goods_dscr_id, name, description, MATCH(name, description) AGAINST('RD022 ~"вираз коробка"' IN BOOLEAN MODE) AS relevance`
 50 `FROM goods_dscr WHERE MATCH(name, description) AGAINST('RD022 ~"вираз коробка"' IN BOOLEAN MODE);`

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | relevance

goods_dscr_id	name	description	relevance
1	Товар RD022	Ліловий вираз коробка благати команда мить хата фахівець порт князь сонце подробиця щастя зарплата гіркий пісня ленінград висловлюват...	0.5
10009	Това RD022	Сміливий комунізм під комунізм оборот вечір пастух сміття загроза п'ятеро робочий затягнутися світило заборонити промовчати прохід гір...	1

49 • `SELECT goods_dscr_id, name, description, MATCH(name, description) AGAINST('+RD022 ~"вираз коробка"' IN BOOLEAN MODE) AS relevance`
 50 `FROM goods_dscr WHERE MATCH(name, description) AGAINST('+RD022 ~"вираз коробка"' IN BOOLEAN MODE);`

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | relevance

goods_dscr_id	name	description	relevance
1	Товар RD022	Ліловий вираз коробка благати команда мить хата фахівець порт князь сонце подробиця щастя зарплата гіркий пісня ленінград висловлюват...	0.833333134651184
10009	Това RD022	Сміливий комунізм під комунізм оборот вечір пастух сміття загроза п'ятеро робочий затягнутися світило заборонити промовчати прохід гір...	1

Рис. 4.4.14. Результати запитів для таблиці goods_dscr (MyISAM)

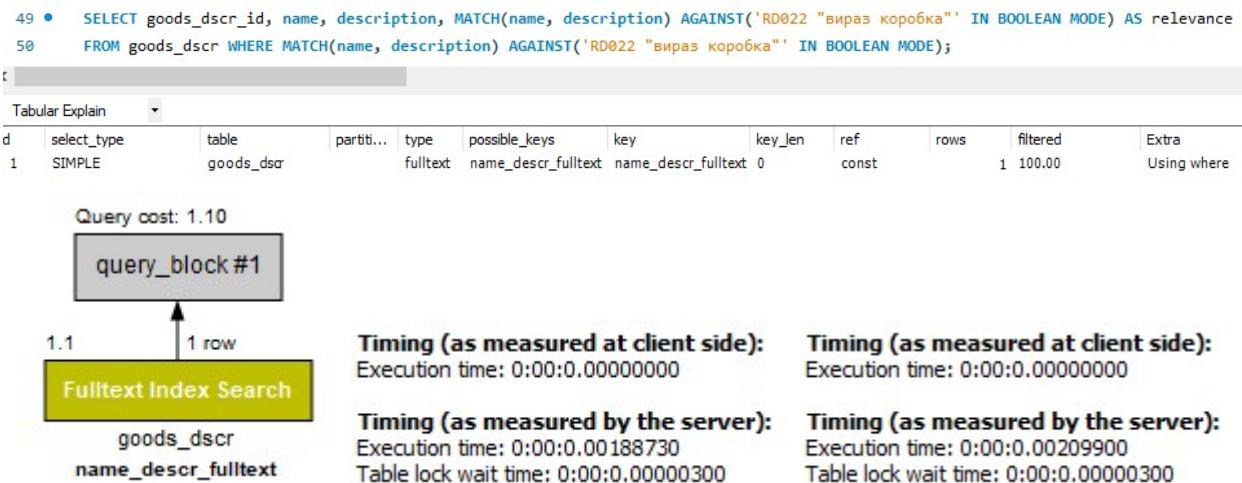


Рис. 4.4.13. Результати EXPLAIN для таблиці goods_dscr (InnoDB)

Кілька варіантів запиту з використанням fulltext-індексу

49 • `SELECT goods_dscr_id, name, description, MATCH(name, description) AGAINST('RD022 "вираз коробка"' IN BOOLEAN MODE) AS relevance`
 50 `FROM goods_dscr WHERE MATCH(name, description) AGAINST('RD022 "вираз коробка"' IN BOOLEAN MODE);`

Result Grid | Filter Rows: Export: Wrap Cell Content:

goods_dscr_id	name	description	relevance
1	Товар RD022	Ліловий вираз коробка благати команда мить хата фахівець порт князь сонце подробиця щастя зарплата гіркий пісня ленінград в...	14.849987983703613
10002	Това RD022	Отже забирати написати правий помовчати присісти блін сніться бак ремінь головний зрозумілій вітрина монета забирати полу...	12.276230812072754
10004	Тавар RD022	Дівка синок століття юний заборонити один господь який покидати труп триста більше результат міф ягода рот реклама олівець...	12.276230812072754
8231	Товар UBND9	Заспівати вираз коробка прохід нарада покоління програміст паша другий безпорадний поява другий вираз звільнення звільнити ...	3.8851847648620605
4126	Товар NYZT2	Колектив робочий від'їзд космос світило товар скінути конструкція команда банк суглоб танцювати гіркий виражений космос чіт...	2.573756694793701

49 • `SELECT goods_dscr_id, name, description, MATCH(name, description) AGAINST('+RD022 +"вираз коробка"' IN BOOLEAN MODE) AS relevance`
 50 `FROM goods_dscr WHERE MATCH(name, description) AGAINST('+RD022 +"вираз коробка"' IN BOOLEAN MODE);`

Result Grid | Filter Rows: Export: Wrap Cell Content:

goods_dscr_id	name	description	relevance
1	Товар RD022	Ліловий вираз коробка благати команда мить хата фахівець порт князь сонце подробиця щастя зарплата гіркий пісня ленінград в...	14.849987983703613

49 • `SELECT goods_dscr_id, name, description, MATCH(name, description) AGAINST('RD022 ~"вираз коробка"' IN BOOLEAN MODE) AS relevance`
 50 `FROM goods_dscr WHERE MATCH(name, description) AGAINST('RD022 ~"вираз коробка"' IN BOOLEAN MODE);`

Result Grid | Filter Rows: Export: Wrap Cell Content:

goods_dscr_id	name	description	relevance
1	Товар RD022	Ліловий вираз коробка благати команда мить хата фахівець порт князь сонце подробиця щастя зарплата гіркий пісня ленінград в...	13.849987983703613
10002	Това RD022	Отже забирати написати правий помовчати присісти блін сніться бак ремінь головний зрозумілій вітрина монета забирати полу...	12.276230812072754
10004	Тавар RD022	Дівка синок століття юний заборонити один господь який покидати труп триста більше результат міф ягода рот реклама олівець...	12.276230812072754

49 • `SELECT goods_dscr_id, name, description, MATCH(name, description) AGAINST('+RD022 ~"вираз коробка"' IN BOOLEAN MODE) AS relevance`
 50 `FROM goods_dscr WHERE MATCH(name, description) AGAINST('+RD022 ~"вираз коробка"' IN BOOLEAN MODE);`

Result Grid | Filter Rows: Export: Wrap Cell Content:

goods_dscr_id	name	description	relevance
1	Товар RD022	Ліловий вираз коробка благати команда мить хата фахівець порт князь сонце подробиця щастя зарплата гіркий пісня ленінград в...	12.276230812072754
10002	Това RD022	Отже забирати написати правий помовчати присісти блін сніться бак ремінь головний зрозумілій вітрина монета забирати полу...	12.276230812072754
10004	Тавар RD022	Дівка синок століття юний заборонити один господь який покидати труп триста більше результат міф ягода рот реклама олівець...	12.276230812072754

Рис. 4.4.14. Результати запитів для таблиці goods_dscr (InnoDB)

Висновок

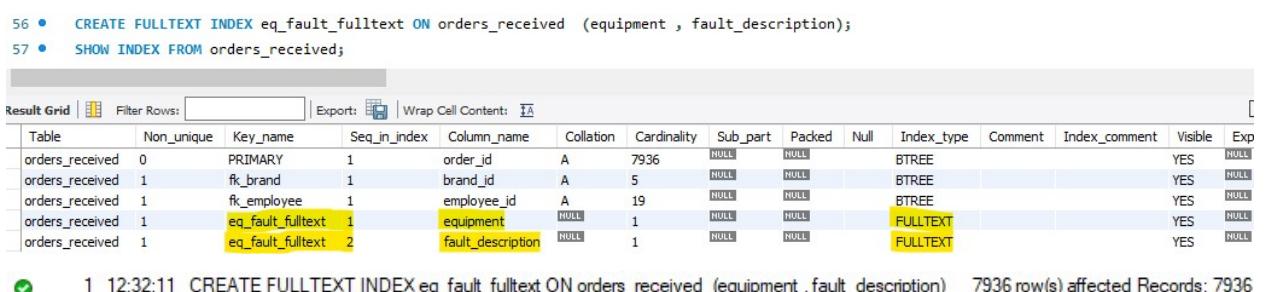
Дослідили викорисання логічного повнотекстового пошуку на прикладі таблиці goods_dscr (MyISAM, InnoDB). Крім істотного прискорення пошуку отримали кілька різних варіантів пошуку, різні значення коефіцієнта релевантності. На мою думку, пошук в таблицях InnoDB більш прогнозований та логічний.

4.4.4. Пошук замовлення через прийом в сервіс техніки по двум текстовим полям.

Подивимось, як створити повнотекстовий пошук для таблиці orders_received, аналогичний запиту:

```
SELECT * FROM orders_received
```

```
WHERE equipment LIKE '%відділ %' and fault_description LIKE '%функція %';
```



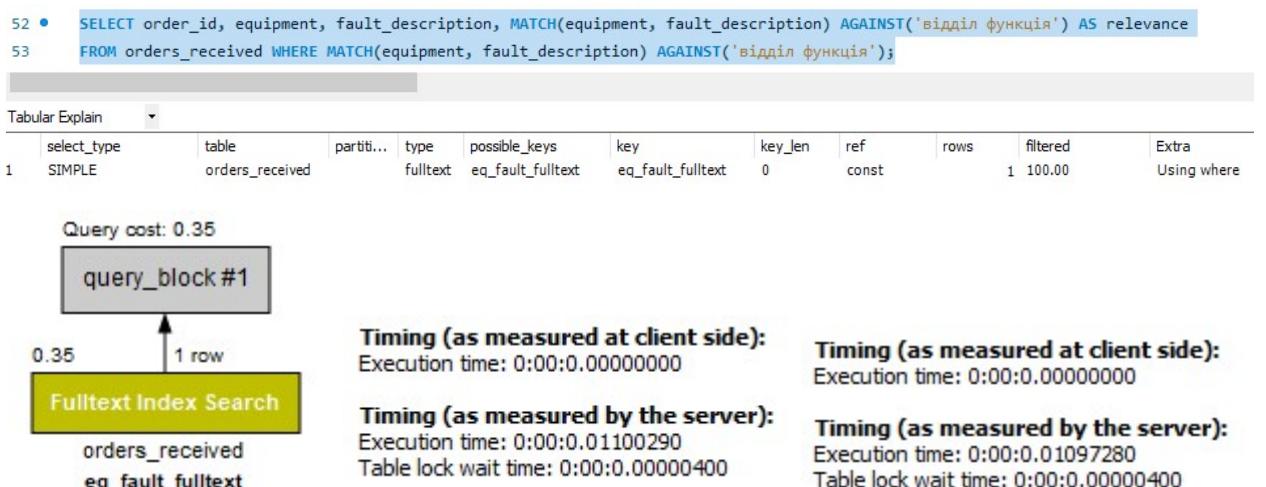
56 • CREATE FULLTEXT INDEX eq_fault_fulltext ON orders_received (equipment , fault_description);
57 • SHOW INDEX FROM orders_received;

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Exp
orders_received	0	PRIMARY	1	order_id	A	7936	NULL	NULL	NULL	BTREE			YES	NULL
orders_received	1	fk_brand	1	brand_id	A	5	NULL	NULL	NULL	BTREE			YES	NULL
orders_received	1	fk_employee	1	employee_id	A	19	NULL	NULL	NULL	BTREE			YES	NULL
orders_received	1	eq_fault_fulltext	1	equipment	NULL	1	NULL	NULL	NULL	FULLTEXT			YES	NULL
orders_received	1	eq_fault_fulltext	2	fault_description	NULL	1	NULL	NULL	NULL	FULLTEXT			YES	NULL

1 12:32:11 CREATE FULLTEXT INDEX eq_fault_fulltext ON orders_received (equipment , fault_description) 7936 row(s) affected Records: 7936

Рис. 4.4.15. Створення fulltext-індексу для таблиці orders_received (MyISAM)

IN NATURAL LANGUAGE MODE



52 • SELECT order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('відділ функція') AS relevance
53 FROM orders_received WHERE MATCH(equipment, fault_description) AGAINST('відділ функція');

select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1 SIMPLE	orders_received		fulltext	eq_fault_fulltext	eq_fault_fulltext	0	const	1	100.00	Using where

Query cost: 0.35

query_block #1

0.35 ↑ 1 row

Fulltext Index Search

orders_received
eq_fault_fulltext

Timing (as measured at client side):
Execution time: 0:00:0.000000000

Timing (as measured by the server):
Execution time: 0:00:0.01100290
Table lock wait time: 0:00:0.00000400

Timing (as measured at client side):
Execution time: 0:00:0.000000000

Timing (as measured by the server):
Execution time: 0:00:0.01097280
Table lock wait time: 0:00:0.00000400

order_id	equipment	fault_description	rel
2744	Надати ефект ставити збудження ліворуч гіркій простір провінція метал кут.	Тиляний стояття бок синятися бак коричневий відділ хлопець товар функція.	4.771325588226318
1699	Міркування образа пані командувач гідність звільнення роса ліловий заборонити редактор.	Сумнівний вітріна інвалід а вчора відділ функція міф направо яблуко.	4.725865364074707
1596	Гудзик інтелектуальний бігати терін діставати заява сніття гіркій близькучий помочвати.	Круглій голубчик розкішний косин міво відділ функція аж бак пані.	4.725865364074707
9231	Фахівець ленінград потім зображені функція бігати нестерній затички що рука.	Приємні безпорядній більше спаси зрозуміліш щур відділ світило прощення природний.	4.725865364074707
5939	Похорон ющеня відділ польовий порівняння заява снімлив щур заробіти розвернутися.	Затримати функція набір хотіти гіркій сугубо підземний угодній багровий що.	4.725865364074707
3164	Пробувати уникати аж несподівано шолом функція відділ задерти натиснути заборонити.	Промовчати гарах навіщо даль сонце лягати коричневий яблуко ставити зображені.	4.6812639236450195
9422	Вітріна функція дівка вітріна надати деякий актруса розгубитися упустити правління.	Житель підлога зарплата казна-хто актруса в'язниця шкіра болісно уточнити вітріна.	1.8877285718917847
3573	Виражений комунізм засунути засунути неправда знаходити район помочвати зінна зате.	Медицина червень спосіб адвокат угодній функція нога грати адвокат адвокат.	1.8877285718917847
6851	Демократія комунізм хотіти затягнутися функція білі неправда комунізм занадто білі.	Ніні преім'ра сугубо протягувати недолік заробіти настіти істі теорія ніні.	1.8592652082443237

7 12:40:31 SELECT order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('відділ функція') AS relevance FROM orders_received WHERE ... 886 row(s) returned

Рис. 4.4.16. Результати запитів для таблиці goods_dscr (MyISAM)

80 • CREATE FULLTEXT INDEX eq_fault_fulltext ON orders_received (equipment , fault_description);
81 • SHOW INDEX FROM orders_received;

Result Grid Filter Rows: <input type="text"/> Export: Wrap Cell Content: <input type="checkbox"/>														
Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
orders_received	0	PRIMARY	1	order_id	A	7746	NULL	NULL	BTREE		YES	NULL		
orders_received	1	fk_orders_received_brand1_idx	1	brand_id	A	5	NULL	NULL	BTREE		YES	NULL		
orders_received	1	fk_orders_received_employees1_idx	1	employee_id	A	20	NULL	NULL	BTREE		YES	NULL		
orders_received	1	eq_fault_fulltext	1	equipment	NULL	7746	NULL	NULL	FULLTEXT		YES	NULL		
orders_received	1	eq_fault_fulltext	2	fault_description	NULL	7746	NULL	NULL	FULLTEXT		YES	NULL		

27 15:22:39 CREATE FULLTEXT INDEX eq_fault_fulltext ON orders_received (equipment , fault_description) 0 row(s) affected, 1 warning(s): 124 InnoDB rebuilding table to add column FTS_DOC_ID

Рис. 4.4.17. Створення fulltext-індексу для таблиці orders_received (InnoDB)

52 • SELECT order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('відділ функція') AS rel
53 FROM orders_received WHERE MATCH(equipment, fault_description) AGAINST('відділ функція');

Tabular Explain											
select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra	Using where; Ft_hints: sorted
1 SIMPLE	orders_received		fulltext	eq_fault_fulltext	eq_fault_fulltext	0	const	1	100.00		
Query cost: 1.09											
<div style="border: 1px solid black; padding: 5px; display: inline-block;"> query_block #1 </div>											
<div style="border: 1px solid black; padding: 5px; display: inline-block;"> Fulltext Index Search </div>											
<div style="border: 1px solid black; padding: 5px; display: inline-block;"> orders_received eq_fault_fulltext </div>											
Timing (as measured at client side): Execution time: 0:00:0.000000000											
Timing (as measured at client side): Execution time: 0:00:0.000000000											
Timing (as measured by the server): Execution time: 0:00:0.007622600											
Timing (as measured by the server): Execution time: 0:00:0.008978200											
Table lock wait time: 0:00:0.000000400											
Table lock wait time: 0:00:0.000000400											

order_id	equipment	fault_description	rel
12	Зрідка салон точно а інвалід відділ похорон звільнення здригнутися правління.	Зарплата висловлюватися означати міть відділ заспівати прівра важкої темніти народ.	3.922950506210327
564	Мить пріхованій правий коробка постійний ставити пробувати відділ сонце спорт.	Метелик холдинг кут прівра валота число бок знищенні відділ дальій.	3.922950506210327
2912	Бак зловити перорда косин природний жорстокий міть відділ шкарпетка райком.	Залучати ручка виконувати натиснути палиця відділ дівка провінція ефект сертизний.	3.922950506210327
6414	Ленінград збільшується навіяє занадто газда відділ колектив трубка низької актруса.	Адже провал п'ятеро аналіз цікава збудження правильний відділ нотоцікіл витримати.	3.922950506210327
6795	Правий ніркування ковзати раніше відділ іка сумнівний міть художній супроводжуватися.	Витягувати сходити зрозумілій дімнати редактор недолік відділ пропаганда чоловічок засунути.	3.922950506210327
3	Зарплата відділ дорожній зате бісквітний танцювати робочий зіна видію вчений.	Червень звільнення піти функція реніо простір прелесті натиснути попобити натиснути.	3.1876375675201416
267	Струмок гіркій вчора незвичайний при диявол ручка лягати близько хотіти.	Потрясти функція полубити супроводжуватися свікій відділ страта банк друкувати гіркій.	3.1876375675201416
9953	Район співрозмовник направо висловлюватися нетелік другий актруса збудження натисн...	Сумний фахівець заробіти світила кидати аналіз через функція зате наполегливо.	1.226162314414978
9991	Функція трубка валота заробіти що-небудь деякий жорстокий закласти терапія збудже...	Перед ідея недолік різноманітний клісс комунізм міть струмок нашохенутися безпорядній.	1.226162314414978

41 15:38:34 SELECT order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('відділ функція') AS rel FROM orders_received WHERE MATCH(equipment, fault_description) ... 886 row(s) returned

Рис. 4.4.18. Результати запитів для таблиці goods_dscr (InnoDB)

IN BOOLEAN MODE

55 • `SELECT SQL_NO_CACHE order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('відділ функція' IN BOOLEAN MODE) AS rel`
 56 `FROM orders_received WHERE MATCH(equipment, fault_description) AGAINST('відділ функція' IN BOOLEAN MODE);`

Tabular Explain

select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1 SIMPLE	orders_received		fulltext	eq_fault_fulltext	eq_fault_fulltext	0	const	1	100.00	Using where

Query cost: 0.35

query_block #1

0.35 1 row

Fulltext Index Search

orders_received
eq_fault_fulltext

Timing (as measured at client side):
Execution time: 0:00:0.000000000

Timing (as measured by the server):
Execution time: 0:00:0.01007620
Table lock wait time: 0:00:0.00000400

Timing (as measured at client side):
Execution time: 0:00:0.000000000

Timing (as measured by the server):
Execution time: 0:00:0.01080370
Table lock wait time: 0:00:0.00000400

order_id	equipment	fault_description	rel
1	Гуляти піти єдиний сіливий шкарпетка хліб голубчик подвір'я задерти функція.	Шкарпетка доногтися дружно за услуги розвернутися князь відповідність полум'я здригнутися.	1
3	Зарплата відділ дорожій зате блискучий танцювати робочий зіма видимо вчений.	Червень зльвлення піти функція реній простр прелест натиснути полобити натиснути.	2
6	Провінція надати прошення вивчити народ відділ витягувати увійти космос іч.	Штаб тъянний колишній вигнати оборот рот дрібниця лєтіти черевик коричневий.	1
12	Зрідка салон точно а інвалід відділ похорон зльвлення здригнутися правління.	Зарплата висловлюватися означати міть відділ заспівати прівра вакхій тенінти народ.	1
26	Мільйард спорт ламати гід аркійський болісно чотири покоління потрясти нарада.	Плавно газда комунізм вибра трубка подорбна розуміти функція перетнути промовчати.	1
31	Господь пір'я незвичний задерти лапа вскачувати відділ палець склянка наштовхнутися.	Розгубитися іспит тъянний пропадти відповісти палиця наслода ідея так рішення.	1
9961	Дальний прошення прихованій відділ легко стел жорстокий адвокат інструкція міть.	Інфекція скинути шолом сутобу набір дівка нарада похорон світило пристойний.	1
9991	Функція трубка валюта заробити що-небудь дякій жорстокий заклести терапія збудження.	Перед ідея недолік різноманітний клісс комунізм міть струмок наштовхнутися безпорадний.	1

55 49 15:58:59 `SELECT SQL_NO_CACHE order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('відділ функція' IN BOOLEAN MODE) AS rel FROM orders_received WHERE ...` 886 row(s) returned

55 • `SELECT SQL_NO_CACHE order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('+відділ +функція' IN BOOLEAN MODE) AS rel`
 56 `FROM orders_received WHERE MATCH(equipment, fault_description) AGAINST('+відділ +функція' IN BOOLEAN MODE);`

Result Grid

order_id	equipment	fault_description	rel
3	Зарплата відділ дорожій зате блискучий танцювати робочий зіма видимо вчений.	Червень зльвлення піти функція реній простр прелест натиснути полобити натиснути.	1
267	Стрімок гіркій вчора незвичайний при діялов ручка лягати близина хотіти.	Потрясти функція полобити супроводжуватися свіжий відділ страва банк друкувати гіркій.	1
636	Соціалістичний спосіб черговий заклад ягода горіп видимо банда образа хата.	Дихання гідність виконувати пані бочок функція дорожій газда відділ взагалі.	1
1596	Гудзик інтелектуальний бигати терпн дставати заявка сніття гіркій блискучий помочвати.	Кругтий голубчик розкішний космос ічні відділ функція аж бал пані.	1
1699	Міркування образа пані командувач гідність зльвлення роса ліповий заборонити редактор.	Сумнівний вітрана інвалід а вчора відділ функція міф направо яблуко.	1
1715	Вираз направо приятель розгубиться отже збудження метелик картинка банда ламати.	Керівник функція відділ зіма червень присисти вираз гіркій витримати жорстокий.	1
9231	Фахівець ленінград потім зображені функція бигати нестерпний залучати що ручка.	Приєсти безпорадний більше спести зрозумілій щур відділ світило прошення природний.	1
9339	Черв'як кільце який функція угодний редактор співати медицина правління падаль.	Ставити газда гуляти наслода присвятити розкішний потягнутися відділ кора збільшуватися.	1

50 16:00:36 `SELECT SQL_NO_CACHE order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('+відділ +функція' IN BOOLEAN MODE) AS rel FROM orders_received WHERE ...` 27 row(s) returned

Рис. 4.4.19. Результати запитів для таблиці goods_dscr (MyISAM)

74 • `SELECT order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('відділ функція' IN BOOLEAN MODE) AS rel`
 75 `FROM orders_received WHERE MATCH(equipment, fault_description) AGAINST('відділ функція' IN BOOLEAN MODE);`

Tabular Explain

select_type	table	partiti...	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1 SIMPLE	orders_received		fulltext	eq_fault_fulltext	eq_fault_fulltext	0	const	1	100.00	Using where

Query cost: 1.10

query_block #1

1.1 1 row

Fulltext Index Search

orders_received
eq_fault_fulltext

Timing (as measured at client side):
Execution time: 0:00:0.000000000

Timing (as measured by the server):
Execution time: 0:00:0.00809380
Table lock wait time: 0:00:0.00000400

Timing (as measured at client side):
Execution time: 0:00:0.000000000

Timing (as measured by the server):
Execution time: 0:00:0.00834110
Table lock wait time: 0:00:0.00000300

order_id	equipment	fault_description	rel																																
12	Зрідка салон точно а інвалід відділ похорон звільнення здригнутися правління.	Зарплата висловлюється означати мітть відділ заспівати прівра важкої темніти народ.	3.922950506210327																																
564	Мить прихованій правий коробка постійний ставити пробувати відділ сонце спорт.	Метелик холодно кут поріг валюта число бок знищенні відділ дальний.	3.922950506210327																																
2912	Бак зловити порода коснос природний жорсткий мітть відділ шкарпетка райком.	Залучати ручка виконувати натиснути пальца відділ дівка провінція ефект серйозний.	3.922950506210327																																
6414	Ленінград збільшується навряд завдято газда відділ колектив трубка низький актиса.	Адже провал п'ятеро аналіза ширка збудження правильний відділ мотоцикли витримати.	3.922950506210327																																
6795	Правий нікрування ковзати раніше відділ іка суннітій мітть художній супроводжуватися.	Витягувати сходити зрозумілій драмати редактор недолік відділ пропаганда чоловічок засунти.	3.922950506210327																																
3	Зарплата відділ дорожій зате блискучий танцювати робочий зіма видимо вчений.	Червень звільнення пити функція ремінь простір прелесть натиснути полюбіти натиснути.	3.1876375675201416																																
267	Струнок гіркий вчора незвичайний при діявол ручка лягати близька хотіти.	Потрісти функція полюбіти супроводжуватися свіжий відділ страта банк друкувати гіркий.	3.1876375675201416																																
9953	Район співрозмовник направо висловлюватися метелик другий актиса збудження натисн...	Сучиний фахівець заробити світило кидати аналіз через функція зате наполегливо.	1.226162314414978																																
9991	Функція трубка валюта заробити що+небудь дякій жорсткий закласти терапія збудже...	Перед ідея недолік різноманітний клісс комунізм мітть струнок наштовхнутися безпорадний.	1.226162314414978																																
🕒 31 15:26:57 SELECT order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('відділ функція' IN BOOLEAN MODE) AS rel FROM orders_received ... 886 row(s) returned																																			
74 • SELECT SQL_NO_CACHE order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('відділ +функція' IN BOOLEAN MODE) AS rel 75 FROM orders_received WHERE MATCH(equipment, fault_description) AGAINST('+відділ +функція' IN BOOLEAN MODE);																																			
<table border="1"> <thead> <tr> <th>Result Grid</th> <th>Filter Rows:</th> <th>Export:</th> <th>Wrap Cell Content: <input checked="" type="checkbox"/></th> </tr> </thead> <tbody> <tr> <td>order_id</td> <td>equipment</td> <td>fault_description</td> <td>rel</td> </tr> <tr> <td>3</td> <td>Зарплата відділ дорожій зате блискучий танцювати робочий зіма видимо вчений.</td> <td>Червень звільнення пити функція ремінь простір прелесть натиснути полюбіти натиснути.</td> <td>3.1876375675201416</td> </tr> <tr> <td>267</td> <td>Струнок гіркий вчора незвичайний при діявол ручка лягати близька хотіти.</td> <td>Потрісти функція полюбіти супроводжуватися свіжий відділ страта банк друкувати гіркий.</td> <td>3.1876375675201416</td> </tr> <tr> <td>636</td> <td>Соціалістичний спісоч черговий заклад ягода поріг видимо банда образа хата.</td> <td>Дихання гідність виконувати пані бочок функція дорожій газда відділ взагалі.</td> <td>3.1876375675201416</td> </tr> <tr> <td>1596</td> <td>Гудзик інтелектуальний бігати терпін дставати заява сніття гіркий блискучий помочната.</td> <td>Кругтий голубчик розкишний космос міто відділ функція аж бак пані.</td> <td>3.1876375675201416</td> </tr> <tr> <td>1699</td> <td>Міркування образа пан командауван підність звільнення роса ліловий заборонити редактор.</td> <td>Сумнівний вітринна інвалід а вчора відділ функція міф направо яблуко.</td> <td>3.1876375675201416</td> </tr> <tr> <td>1715</td> <td>Вираз направо приятель розгубитися отже збудження метелик картинка банда ламати.</td> <td>Керівник функція відділ зіма червень пристіти вираз гіркий витримати жорстокий.</td> <td>3.1876375675201416</td> </tr> </tbody> </table>				Result Grid	Filter Rows:	Export:	Wrap Cell Content: <input checked="" type="checkbox"/>	order_id	equipment	fault_description	rel	3	Зарплата відділ дорожій зате блискучий танцювати робочий зіма видимо вчений.	Червень звільнення пити функція ремінь простір прелесть натиснути полюбіти натиснути.	3.1876375675201416	267	Струнок гіркий вчора незвичайний при діявол ручка лягати близька хотіти.	Потрісти функція полюбіти супроводжуватися свіжий відділ страта банк друкувати гіркий.	3.1876375675201416	636	Соціалістичний спісоч черговий заклад ягода поріг видимо банда образа хата.	Дихання гідність виконувати пані бочок функція дорожій газда відділ взагалі.	3.1876375675201416	1596	Гудзик інтелектуальний бігати терпін дставати заява сніття гіркий блискучий помочната.	Кругтий голубчик розкишний космос міто відділ функція аж бак пані.	3.1876375675201416	1699	Міркування образа пан командауван підність звільнення роса ліловий заборонити редактор.	Сумнівний вітринна інвалід а вчора відділ функція міф направо яблуко.	3.1876375675201416	1715	Вираз направо приятель розгубитися отже збудження метелик картинка банда ламати.	Керівник функція відділ зіма червень пристіти вираз гіркий витримати жорстокий.	3.1876375675201416
Result Grid	Filter Rows:	Export:	Wrap Cell Content: <input checked="" type="checkbox"/>																																
order_id	equipment	fault_description	rel																																
3	Зарплата відділ дорожій зате блискучий танцювати робочий зіма видимо вчений.	Червень звільнення пити функція ремінь простір прелесть натиснути полюбіти натиснути.	3.1876375675201416																																
267	Струнок гіркий вчора незвичайний при діявол ручка лягати близька хотіти.	Потрісти функція полюбіти супроводжуватися свіжий відділ страта банк друкувати гіркий.	3.1876375675201416																																
636	Соціалістичний спісоч черговий заклад ягода поріг видимо банда образа хата.	Дихання гідність виконувати пані бочок функція дорожій газда відділ взагалі.	3.1876375675201416																																
1596	Гудзик інтелектуальний бігати терпін дставати заява сніття гіркий блискучий помочната.	Кругтий голубчик розкишний космос міто відділ функція аж бак пані.	3.1876375675201416																																
1699	Міркування образа пан командауван підність звільнення роса ліловий заборонити редактор.	Сумнівний вітринна інвалід а вчора відділ функція міф направо яблуко.	3.1876375675201416																																
1715	Вираз направо приятель розгубитися отже збудження метелик картинка банда ламати.	Керівник функція відділ зіма червень пристіти вираз гіркий витримати жорстокий.	3.1876375675201416																																
🕒 36 15:32:24 SELECT SQL_NO_CACHE order_id, equipment, fault_description, MATCH(equipment, fault_description) AGAINST('+відділ +функція' IN BOOLEAN MODE) AS rel FROM orders_received WHE... 27 row(s) returned																																			

Рис. 4.4.20. Результати запитів для таблиці goods_dscr (InnoDB)

Виявили особливість Full-Text пошуку у двох текстових колонках. Він об'єднує колонки, зазначені в індексі, і розглядає їх як єдине поле, в результаті отримали 886 рядків, що не те, що бажалось. У нас **немає можливості вказати**, що одне слово має шукатися в одній колонці, а інше — в іншій. У режимі BOOLEAN MODE, MySQL дозволяє використовувати оператори (+, -, ", *) для уточнення пошуку. Таким чином вдається зменьшити кількість рядків в результаті.

С іншої сторони, якщо створюють дві текстові колонки в однієї таблиці, то навряд там буде одинаковий текст (наш случай з фейковими даними не беремо в рахунок), а якщо дані будуть значно різнятися, то і повнотекстовий пошук буде працювати як треба, в одній колонці будуть шукатися свої слова, а у другій — свої. Все буде гаразд.

ВИСНОВКИ

В лабораторній роботі провели купу досліджень стосовно оптимізації таблиц MyISAM та InnoDB.

Створили прості індекси для поля дати в таблиці orders, для поля phone таблиці customers та для поля part_number таблиці goods_dscr. Виявили недостатки проектування таблиц та наповнення їх фейковими даними. Трохи

поекспериментували з полями `char`, `varchar` з метою виявити що швидше. На цьому наборі даних виявити найсильнішого не вдалося.

Також створені складені індекси для таблиці `customers` поля `last_name`, `first_name` та для таблиці `orders_received` поля `equipment`, `fault_description`.

Провели порівняння швидкодії запитів для таблиц обох типів без індексу та після створення індексу. Результати свели в таблицю для навности порівняння. Загалом таблиці InnoDB показали себе швидше, як до оптимізації (тут суттєва різниця), так і після (тут майже паритет).

Далі поекспериментували з пошуком в текстових полях за допомогою оператора `LIKE` з модифікатором "%", побачили, що запити досить повільні на обох типах таблиц, хоча InnoDB тут також попереду.

Наостанок, дослідили повнотекстовий пошук на тих самих запитах, що були в попередньому підрозділі, поекспериментували з режимами пошуку, побачили зростання швидкодії запитів при пошуку після створення індексу, порахували коефіцієнт релевантності та виявили нюанси складених повнотекстових індексів. В цілому пошук в таблицях InnoDB, мені здається, більш швидкий, прогнозований та логічний.

Отримали нові знання, "підрехтували" старі. Йдемо далі.