1 Постановка задачи

Цели лабораторной работы:

- 1. Оптимизация запросов. Индексы. Типы индексов. В-Тree.
- 2. Оптимизация запросов. Explain. Profiling.

2 Типы индексов

- 1. Primary, Unique, Index (B-Tree)
- 2. Fulltext (InvertedList)
- 3. Clustered/Non clustered

3 Инструменты

3.1 Explain

EXPLAIN **SELECT** * **FROM** title;

id select_type	table	partitions	type	Ì	possible_keys	ĺ	key	I	key_len	l	ref	ĺ	rows	l	filtered	Ext	ra
1 SIMPLE																	
++++++																	

EXPLAIN SELECT * FROM title WHERE titleID = 'tt0468569';

id	select_type	table	partitions	l type	possible_keys	l key	key_len	ref	rows	filtered Extra
1 100.00	SIMPLE NULL	title	NULL	const	PRIMARY	PRIMARY	1 40	const	1	İ
	in set, 1 warı			+	+	+	+	+	+	++

EXPLAIN

SELECT * **FROM** title

INNER JOIN title_rating ON title.titleID = title_rating.titleID ORDER BY $numVotes\ DESC$

id r	select_type	table	partitions	l type	possible_keys	key	key_len	+
1		title_rating		ALL	PRIMARY	NULL	NULL	NULL
1		title	NULL	. 1-	PRIMARY	PRIMARY		imdb_full.title_rating.titleID
		rning (0 00 sec		r	T	T	r	T

```
EXPLAIN
SELECT * FROM title
WHERE titleID IN (
    SELECT titleID FROM title genre
   WHERE title genre.genre IN ("Comedy", "Action")
)
1 | SIMPLE | title_genre | NULL | index | PRIMARY | PRIMARY | 41 | NULL
911
1 | SIMPLE | title | NULL | eq_ref | PRIMARY | PRIMARY | 40 | imdb_full.title_genre.titleID |
2 rows in set, 1 warning (0.00 sec)
EXPLAIN
SELECT * FROM title
WHERE EXISTS(
    SELECT 1 FROM title genre
    WHERE title genre genre IN ("Comedy", "Action")
    AND title.titleID = title_genre.titleID
)
2 rows in set, 2 warnings (0.00 sec)
3.2
    Profile
SET PROFILING=1;
SET profiling history size = 1;
SHOW PROFILES;
| Query_ID | Duration | Query
 1 | 10.16626100 | SELECT * FROM title |
1 row in set, 1 warning (0.00 sec)
SHOW PROFILE FOR QUERY 1;
            0.000857
```

```
0.000005
| System lock
  optimizing
 statistics
                        1 0.000007
 preparing
                        0.000006
  executing
                           0.000002
 Sending data
                       0.000011
 end
 query end
closing tables
freeing items
                        0.000007
                      0.000008
0.000059
0.000016
 free<sub>116</sub>.
cleaning up
15 rows in set, 1 warning (0.00 sec)
```

4 Использование индексов

4.1 Scan

Общее количество фильмов:

SELECT COUNT(*) FROM title;

```
+----+
| COUNT(*) |
+------+
| 6243651 |
+-----+
1 row in set (1.82 sec)
```

4.2 Key lookup

Без индекса по полю startYear:

SELECT SQL_NO_CACHE * FROM title WHERE startYear = 2019 LIMIT 1000;

EXPLAIN SELECT SQL_NO_CACHE * FROM title WHERE startYear = 2019 LIMIT

Добавление индекса:

ALTER TABLE 'title 'ADD INDEX 'startYearIndex' ('startYear' ASC);

С индексом по полю startYear:

После добавления индекса по полю averageRating:

++	.		4				_	_	_		
id select_type table rows filtered Extra	partitions	type	possible_keys	lkey	key_len	ref					
1 SIMPLE title_rating 100.00 Using index condition	NULL	range	averageRating	averageRating	8	NULL	72116	I			
1 row in set, 2 warnings (0.00 sec)											

4.4 Group by

До добавления индекса по полю genre:

SELECT SQL NO CACHE genre, COUNT(titleID) FROM imdb full.title genre G

İ	Query_ID	I	Duration	I	Query										
İ	8	l	5.94673000	I	SELECT	sQL_	NO_CACHE	genre,	COUNT(titleID)	FROM	imdb_full	title_genre	GROUP	ВЧ	genre
			1 warning												

EXPLAIN SELECT SQL_NO_CACHE genre, COUNT(titleID) FROM imdb_full.title_

После добавления индекса по полю genre: Query_ID | Duration | Query 9 | 1.78642400 | SELECT SQL_NO_CACHE genre, COUNT(titleID) FROM imdb_full.title_genre GROUP BY genre | 1 row in set, 1 warning (0.00 sec | key | key_len | ref id | select_uype | taule | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | real | rea 1 row in set, 2 warnings (0.00 sec) 4.5 Order by До добавления индекса по полю numVotes. SELECT SQL_NO_CACHE titleID, numVotes FROM title_rating ORDER BY numVo 21 | 0.30404600 | SELECT SQL_NO_CACHE titleID, numVotes FROM title_rating ORDER BY numVotes DESC LIMIT 1000 | 1 row in set, 1 warning (0.00 sec) EXPLAIN SELECT SQL NO CACHE titleID, numVotes FROM title rating ORDER I | id | select_type | table | partitions | type | possible_keys | key | key_len | ref | rows | filtered | Extra | | title_rating | NULL | ALL | NULL | NULL | NULL | NULL | 981120 | 1 | SIMPLE 100.00 | Using filesort | 1 row in set, 2 warnings (0.00 sec) ALTER TABLE 'title_rating' ADD INDEX 'numVotesIndex' ('numVotes' ASC); После добавления индекса по полю numVotes. 22 | 0.00075700 | SELECT SQL_NO_CACHE titleID, numVotes FROM title_rating ORDER BY numVotes DESC LIMIT 1000 | 1 row in set, 1 warning (0.00 sec) 100.00 | Using index | 1 row in set, 2 warnings (0.00 sec)

ALTER TABLE 'imdb full'. 'title genre' ADD INDEX 'genreIndex' ('genre'.

4.6 Join

```
EXPLAIN SELECT * FROM title
INNER JOIN title rating ON title.titleID = title rating.titleID
```

4.7 Subqueries

```
EXPLAIN

SELECT * FROM title

WHERE titleID IN (

SELECT titleID FROM title_genre

WHERE title_genre.genre IN ("Comedy", "Action")
)

**The state of the partitions | type | possible_keys | key | key_len | ref |
| rows | filtered | Extra | | | | | | |
| 1 | SIMPLE | title_genre | NULL | index | PRIMARY | PRIMARY | 41 | NULL |
| 9118175 | 12.50 | Using where; Using index; LooseScan |
| 1 | SIMPLE | title | NULL | eq_ref | PRIMARY | PRIMARY | 40 | imdb_full.title_genre.titleID |
| 1 | 100.00 | NULL |
| 2 rows in set, 1 warning (0.00 sec)
```

5 Примеры оптимизации запросов

Вывести список людей и количество людей с такой же датой рождения.

EXPLAIN

```
SELECT F. primaryName, COUNT(F. nameID) FROM name F, name S WHERE F. nameID != S. nameID AND F. birthYear = S. birthYear GROUP BY F. nameID
```

ALTER TABLE 'name' ADD INDEX 'birthYearIndex' ('birthYear' ASC);

```
1 | SIMPLE | F | NULL
9371346 | 100.00 | Using where
1 | SIMPLE | S | NULL
                                                                | imdb_full.F.birthYear |
2 rows in set, 1 warning (0.00 sec)
| Query_ID | Duration | Query
     38 | 2.14287300 | SELECT F.primaryName, COUNT(F.nameID) FROM name F, name S WHERE F.nameID != S.nameID AND F.birthYear
1 row in set, 1 warning (0.00 sec)
   Получить список фильмов отсортированных по рейтингу:
EXPLAIN SELECT SQL NO CACHE * FROM title
JOIN title rating ON title.titleID = title rating.titleID
ORDER BY title rating.numVotes DESC
LIMIT 1000
| Query_ID | Duration
     43 | 0.57302600 | SELECT title.titleID, title.primaryTitle, numVotes FROM title INNER JOIN title_rating ON title.titleID
1 row in set, 1 warning (0.00 sec)
ALTER TABLE 'title rating 'ADD INDEX 'numVotes' ('numVotes' DESC);
EXPLAIN SELECT SQL NO CACHE * FROM title
JOIN title rating ON title.titleID = title rating.titleID
ORDER BY title rating numVotes DESC
LIMIT 1000
| Query_ID | Duration | Query
    44 | 0.00314300 | SELECT title.titleID, title.primaryTitle, numVotes FROM title INNER JOIN title_rating ON title.titleID
```

| key_len | ref

6 Используемые источники

1 row in set, 1 warning (0.00 sec)

| id | select_type | table | partitions | type | possible_keys | key

- 1. https://dev.mysql.com/doc/refman/8.0/en/execution-plan-information.html
- 2. https://dev.mysql.com/doc/refman/8.0/en/explain-output.htmlv
- 3. https://dev.mysql.com/doc/refman/8.0/en/optimization.html
- 4. https://dev.mysql.com/doc/refman/8.0/en/optimization-indexes.html