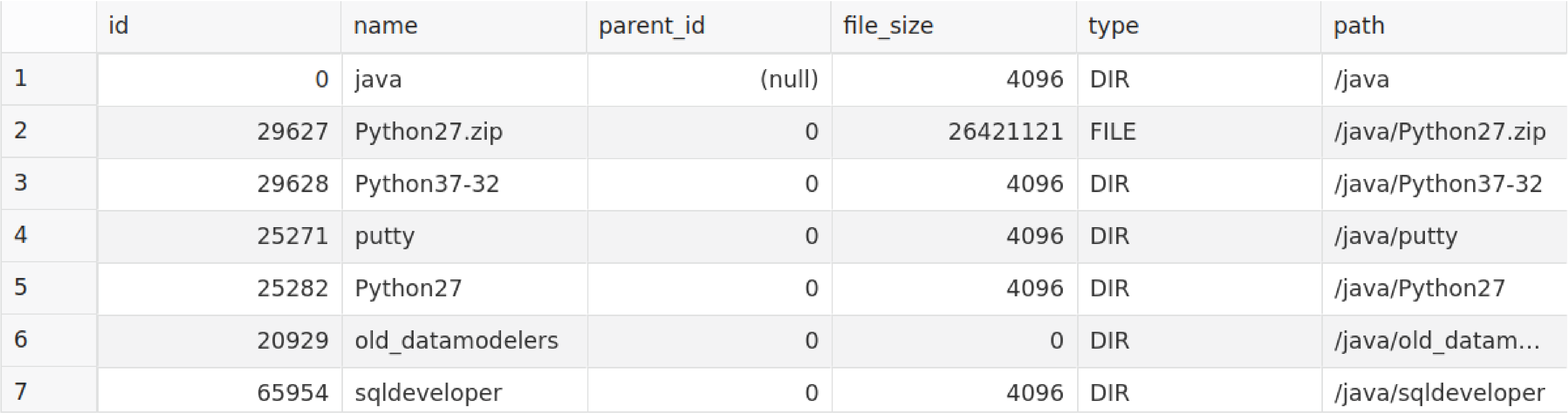
1. Вывести все директории в виде:
ID, Название, Путь до корня

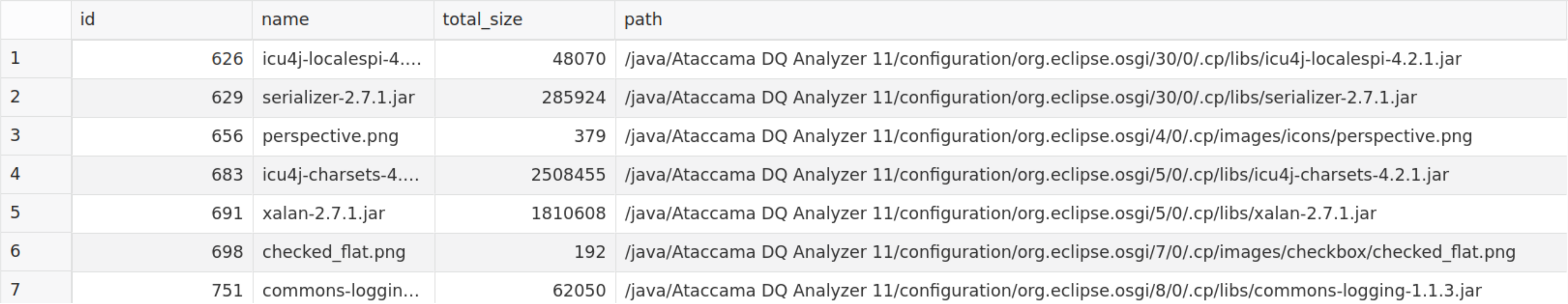
```
with r(id, name, parent_id, file_size, type, path) as (
   select id, name, parent_id, file_size, type, '/' || name from file_system where parent_id is null
   union all
   select f.id, f.name, f.parent_id, f.file_size, f.type, r.path || '/' || f.name
   from file_system f inner join r on (f.parent_id = r.id)
)
select * from r;
```



	id	name	parent_id	file_size	type	path
1	0	java	(null)	4096	DIR	/java
2	29627	Python27.zip	0	26421121	FILE	/java/Python27.zip
3	29628	Python37-32	0	4096	DIR	/java/Python37-32
4	25271	putty	0	4096	DIR	/java/putty
5	25282	Python27	0	4096	DIR	/java/Python27
6	20929	old_datamodelers	0	0	DIR	/java/old_datam...
7	65954	sqldeveloper	0	4096	DIR	/java/sqldeveloper

2. Для каждой директории посчитать объем занимаемого места на диске (с учетом всех вложенных папок)
ID, Название, Путь до корня, total_size

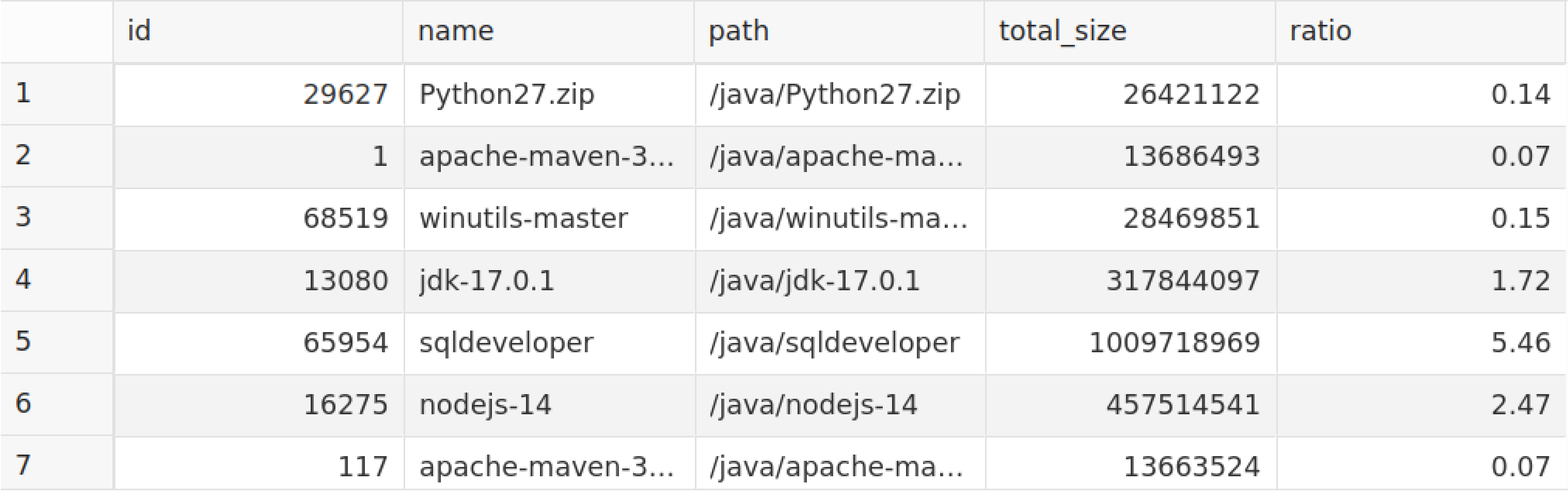
```
with r1(id, name, parent_id, file_size, path) as (
   select id, name, parent_id, file_size, '/' || name from file_system where parent_id is null
   union all
   select f.id, f.name, f.parent_id, f.file_size, r1.path || '/' || f.name
   from file_system f inner join r1 on (f.parent_id = r1.id)
),
r2(id, name, parent_id, file_size, type, size_) as (
   select id, name, parent_id, file_size, type, 0 + file_size from file_system where type = 'FILE'
   union all
   select f.id, f.name, f.parent_id, f.file_size, f.type, r2.size_ + f.file_size
   from file_system f inner join r2 on (f.id = r2.parent_id)
),
step3 as (
   select id, name, sum(size_) total_size
   from r2
   group by id, name
)
select s3.id, s3.name, s3.total_size, r1.path
from step3 s3 inner join r1 on (s3.id = r1.id);
```



	id	name	total_size	path
1	626	icu4j-localespi-4....	48070	/java/Ataccama DQ Analyzer 11/configuration/org.eclipse.osgi/30/0/.cp/libs/icu4j-localespi-4.2.1.jar
2	629	serializer-2.7.1.jar	285924	/java/Ataccama DQ Analyzer 11/configuration/org.eclipse.osgi/30/0/.cp/libs/serializer-2.7.1.jar
3	656	perspective.png	379	/java/Ataccama DQ Analyzer 11/configuration/org.eclipse.osgi/4/0/.cp/images/icons/perspective.png
4	683	icu4j-charsets-4....	2508455	/java/Ataccama DQ Analyzer 11/configuration/org.eclipse.osgi/5/0/.cp/libs/icu4j-charsets-4.2.1.jar
5	691	xalan-2.7.1.jar	1810608	/java/Ataccama DQ Analyzer 11/configuration/org.eclipse.osgi/5/0/.cp/libs/xalan-2.7.1.jar
6	698	checked_flat.png	192	/java/Ataccama DQ Analyzer 11/configuration/org.eclipse.osgi/7/0/.cp/images/checkbox/checked_flat.png
7	751	commons-loggin...	62050	/java/Ataccama DQ Analyzer 11/configuration/org.eclipse.osgi/8/0/.cp/libs/commons-logging-1.1.3.jar

3. Добавить в запрос: сколько процентов директория занимает места относительно всех среди своих соседей (siblings)
ID, Название, Путь до корня, total_size, ratio

```
with r1(id, name, parent_id, type, path) as (
   select id, name, parent_id, type, '/' || name from file_system where parent_id is null
   union all
   select f.id, f.name, f.parent_id, f.type, r1.path || '/' || f.name
   from file_system f inner join r1 on (f.parent_id = r1.id)
),
r2(id, parent_id, file_size, type, size_) as (
   select id, parent_id, file_size, type, 1 + file_size from file_system where type = 'FILE'
   union all
   select f.id, f.parent_id, f.file_size, f.type, r2.size_ + f.file_size
   from file_system f inner join r2 on (f.id = r2.parent_id)
),
step3 as (
   select id, sum(size_) total_size
   from r2
   group by id
),
step4 as (
   select s3.id, r1.parent_id, r1.name, s3.total_size, r1.path, r1.type
   from step3 s3 inner join r1 on (s3.id = r1.id)
),
step5 as (
   select id, parent_id, name, path, type, total_size, sum(total_size) over (partition by parent_id) parent_size
   from step4
)
select id, name, path, total_size, round(total_size / parent_size * 100, 2) ratio
from step5;
```



	id	name	path	total_size	ratio
1	29627	Python27.zip	/java/Python27.zip	26421122	0.14
2	1	apache-maven-3...	/java/apache-ma...	13686493	0.07
3	68519	winutils-master	/java/winutils-ma...	28469851	0.15
4	13080	jdk-17.0.1	/java/jdk-17.0.1	317844097	1.72
5	65954	sqldeveloper	/java/sqldeveloper	1009718969	5.46
6	16275	nodejs-14	/java/nodejs-14	457514541	2.47
7	117	apache-maven-3...	/java/apache-ma...	13663524	0.07

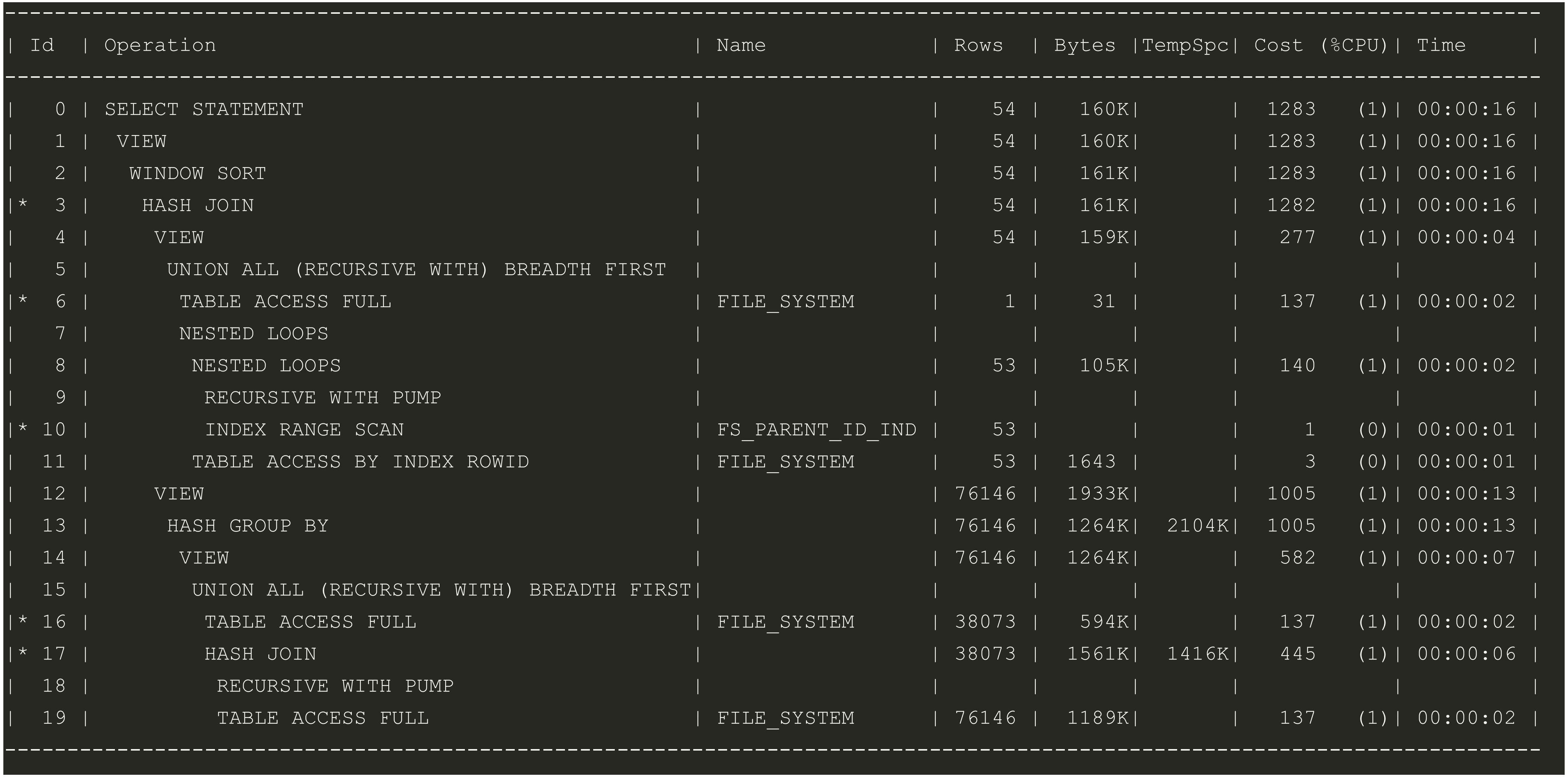
4. Проанализировать план выполнения последнего запроса и предложить вариант оптимизации.

```
-----------------------------------------------------------------------------------------------------------------------
| Id  | Operation                                    | Name          | Rows  | Bytes |TempSpc| Cost (%CPU)| Time     |
-----------------------------------------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT                             |               |    54 |  160K|       |  1418   (1)| 00:00:18 |
|   1 |  VIEW                                        |               |    54 |  160K|       |  1418   (1)| 00:00:18 |
|   2 |   WINDOW SORT                                |               |    54 |  161K|       |  1418   (1)| 00:00:18 |
|*  3 |    HASH JOIN                                 |               |    54 |  161K|       |  1417   (1)| 00:00:18 |
|   4 |     VIEW                                     |               |    54 |  159K|       |   411   (1)| 00:00:05 |
|   5 |      UNION ALL (RECURSIVE WITH) BREADTH FIRST |               |       |       |       |            |          |
|*  6 |       TABLE ACCESS FULL                      | FILE_SYSTEM |     1 |    31 |       |   137   (1)| 00:00:02 |
|*  7 |       HASH JOIN                              |               |    53 |  105K|       |   274   (1)| 00:00:04 |
|   8 |        RECURSIVE WITH PUMP                   |               |       |       |       |            |          |
|   9 |        TABLE ACCESS FULL                     | FILE_SYSTEM | 76146 |  2305K|       |   137   (1)| 00:00:02 |
|  10 |     VIEW                                     |               | 76146 |  1933K|       |  1005   (1)| 00:00:13 |
|  11 |      HASH GROUP BY                           |               | 76146 |  1264K|  2104K|  1005   (1)| 00:00:13 |
|  12 |       VIEW                                   |               | 76146 |  1264K|       |   582   (1)| 00:00:07 |
|  13 |        UNION ALL (RECURSIVE WITH) BREADTH FIRST|               |       |       |       |            |          |
|* 14 |         TABLE ACCESS FULL                    | FILE_SYSTEM | 38073 |   594K|       |   137   (1)| 00:00:02 |
|* 15 |         HASH JOIN                            |               | 38073 |  1561K|  1416K|   445   (1)| 00:00:06 |
|  16 |          RECURSIVE WITH PUMP                 |               |       |       |       |            |          |
|  17 |          TABLE ACCESS FULL                   | FILE_SYSTEM | 76146 |  1189K|       |   137   (1)| 00:00:02 |
-----------------------------------------------------------------------------------------------------------------------
```

Видим четыре штуки table access full. Первое, что приходит в голову, повесить индекс на parent_id, да там может быть очень много одинаковых значений, но всё же:

```
CREATE INDEX fs_parent_id_ind ON file_system (parent_id);
```

Помогло: один из table access full превратился в table access by index rowid и показатель Rows стал 53, вместо 76 тысяч



```
---------------------------------------------------------------------------------------------------------------------------
| Id  | Operation                                      | Name             | Rows  | Bytes |TempSpc| Cost (%CPU)| Time     |
---------------------------------------------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT                               |                  |    54 |   160K|       |  1283   (1)| 00:00:16 |
|   1 |  VIEW                                          |                  |    54 |   160K|       |  1283   (1)| 00:00:16 |
|   2 |   WINDOW SORT                                  |                  |    54 |   161K|       |  1283   (1)| 00:00:16 |
|*  3 |    HASH JOIN                                   |                  |    54 |   161K|       |  1282   (1)| 00:00:16 |
|   4 |     VIEW                                       |                  |    54 |   159K|       |   277   (1)| 00:00:04 |
|   5 |      UNION ALL (RECURSIVE WITH) BREADTH FIRST  |                  |       |       |       |            |          |
|*  6 |       TABLE ACCESS FULL                        | FILE_SYSTEM      |     1 |    31 |       |   137   (1)| 00:00:02 |
|   7 |       NESTED LOOPS                             |                  |       |       |       |            |          |
|   8 |        NESTED LOOPS                            |                  |    53 |   105K|       |   140   (1)| 00:00:02 |
|   9 |         RECURSIVE WITH PUMP                    |                  |       |       |       |            |          |
|* 10 |         INDEX RANGE SCAN                       | FS_PARENT_ID_IND |    53 |       |       |     1   (0)| 00:00:01 |
|  11 |        TABLE ACCESS BY INDEX ROWID             | FILE_SYSTEM      |    53 |  1643 |       |     3   (0)| 00:00:01 |
|  12 |     VIEW                                       |                  | 76146 |  1933K|       |  1005   (1)| 00:00:13 |
|  13 |      HASH GROUP BY                             |                  | 76146 |  1264K|  2104K|  1005   (1)| 00:00:13 |
|  14 |       VIEW                                     |                  | 76146 |  1264K|       |   582   (1)| 00:00:07 |
|  15 |        UNION ALL (RECURSIVE WITH) BREADTH FIRST|                  |       |       |       |            |          |
|* 16 |         TABLE ACCESS FULL                      | FILE_SYSTEM      | 38073 |   594K|       |   137   (1)| 00:00:02 |
|* 17 |         HASH JOIN                              |                  | 38073 |  1561K|  1416K|   445   (1)| 00:00:06 |
|  18 |          RECURSIVE WITH PUMP                   |                  |       |       |       |            |          |
|  19 |          TABLE ACCESS FULL                     | FILE_SYSTEM      | 76146 |  1189K|       |   137   (1)| 00:00:02 |
---------------------------------------------------------------------------------------------------------------------------
```