1Z0-117Q&As

Oracle Database 11g Release 2: SQL Tuning Exam

Pass Oracle 1Z0-117 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.geekcert.com/1z0-117.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Oracle Official Exam Center

QUESTION 1

Examine Exhibit 1 to view the query and its execution plan.

```
SQL> select
        First_name, e.department_id, d.department_id, d,department_name from employees e, departments d
        Where e.department_id = d.department_id and last_name like '%a%'

106 rows selected.
```

Execution plan

Plan hash value: 1473400139

Id	Operation	Name	Rows	Bytes	Cost	(%CPU)	Time
0	SELECT STATEMENT		106	2576	6	(34)	00:00:01
1	MERGE JOIN		106	2756	6	(34)	00:00:01
2	TABLE ACCESS BY INDEX ROWID	DEPARTMENTS	27	432	2	(0)	00:00:01
3	INDEX FULL SCAN	DEPT_IDPK	27		1	(0)	00:00:01
*4	SORT JOIN		107	1070	4	(50)	00:00:01
5	VIEW	indes$_join$_001	107	7070	3	(34)	00:00:01
*6	HASH JOIN						
7	INDEX FAST FULL SCAN	EMP_DEPARTMENT_IX	107	1070	1	(0)	00:00:01
*8	INDEX FAST FULL SCAN	EMP_NAME_IX	107	1070	1	(0)	00:00:01

Predicate information (Identified by Operation id):

```
4 – access ("E", "DEPARTMENT_ID" = "D", DEPARTMENT_ID)
  – filter ("E", "DEPARTMENT_ID" = "D", DEPARTMENT_ID)
6 – access (ROWID=ROWID)
8 – filter ("LAST_NAME" LIKE '%A%')
```

Statistics

```
1       recursive calls
0       db block gets
20      consistent gets
0       physical reads
0       redo size
4034    bytes sent via SQL*NET to client
596     bytes received via SQL*NET from client
9       SQL*NET roundtrips to/from client
1       sorts (memory)
0       sorts (disk)
106     rows proceed
```

Examine Exhibit 2 to view the structure and indexes for the EMPLOYEES and DEPARTMENTS tables. Examine Exhibit 3 to view the initialization parameters for the instance.

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR(20)
LAST_NAME	NOT NULL	VARCHAR(25)
EMAIL	NOT NULL	VARCHAR (25)
PHONE_NUMBER		VARCHAR(20)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR (10)
SALARY		NUMBER (8, 2)
COMISSION_PCT		NUMBER (2, 2)
MANAGER_ID		NUMBER (6)
DEPARTMENT_ID		NUMBER (4)

INDEX_NAME	INDEX_TYPE	COLUMN_NAME
EMP_NAME_IX	NORMAL	LAST_NAME
EMP_MANAGER_IX	NORMAL	MANAGER_ID
EMP_JOB_IX	NORMAL	JOB_ID
EMP_DEPARTMENT_IX	NORMAL	DEPARTMENT_ID
EMP_EMP_ID_PK	NORMAL	EMPLOYEE_ID
EMP_EMAIL_UK	NORMAL	EMAIL

Departments

Name	Null?	Type
DEPARTMENT	NOT NULL	NUMBER (4)
DEPARTMENT_NAME	NOT NULL	VARCHAR (30)
MANAGER_ID		NUMBER (6)
LOCATION_ID		NUMBER (4)

INDEX_NAME	INDEX_TYPE	COLUMN_NAME
DEPT_LOCATION_IX	NORMAL	LOCATION_ID
DEPT_ID_PK	NORMAL	DEPARTMENT_ID

NAME	TYPE	VALUE
optimizer_capture_sql_plan_baselines	boolean	FALSE
optimizer_dynamic_sampling	integer	2
optimizer_features_sampling	string	11.2.0.1
optimizer_index_catching	integer	0
optimizer_index_cost_adj	integer	100
optimizer_mode	string	ALL_ROWS
optimizer_secure_view_merging	boolean	TRUE
optimizer_use_invisible_indexes	boolean	FALSE
optimizer_use_pending_statistics	boolean	FALSE
optimizer_use_sql_plan_baselines	boolean	TRUE

Why is sort-merge join chosen as the access method?

A. Because the OPTIMIZER_MODE parameter is set to ALL_ROWS.

B. Because of an inequality condition.

C. Because the data is not sorted in the LAST_NAME column of the EMPLOYEES table

D. Because of the LIKE operator used in the query to filter out records

Correct Answer: A

Incorrect:

B: There is not an inequality condition in the statement.

C: Merge joins are beneficial if the columns are sorted.

D: All regular joins should be able to use Hash or Sort Merge, except LIKE, !=, and NOT ... joins.

Note:

*

A sort merge join is a join optimization method where two tables are sorted and then joined.

*

A "sort merge" join is performed by sorting the two data sets to be joined according to the join keys and then merging them together. The merge is very cheap, but the sort can be prohibitively expensive especially if the sort spills to disk. The cost of the sort can be lowered if one of the data sets can be accessed in sorted order via an index, although accessing a high proportion of blocks of a table via an index scan can also be very expensive in comparison to a full table scan.

*

Sort merge joins are useful when the join condition between two tables is an inequality condition (but not a nonequality) like =. Sort merge joins

perform better than nested loop joins for large data sets. You cannot use hash joins unless there is an equality

condition.

*

When the Optimizer Uses Sort Merge Joins

The optimizer can choose a sort merge join over a hash join for joining large amounts of data if any of the following conditions are true:

/ The join condition between two tables is not an equi-join.

/ Because of sorts already required by other operations, the optimizer finds it is cheaper to use a sort merge than a hash join. Reference: Oracle Database Performance Tuning Guide , Sort Merge Joins

QUESTION 2

One of your databases supports a mixed workload.

When monitoring SQL performance, you detect many direct paths reads full table scans.

What are the two possible causes?

A. Histograms statistics not available

B. Highly selective filter on indexed columns

C. Too many sort operations performed by queries

D. Indexes not built on filter columns

E. Too many similar type of queries getting executed with cursor sharing disabled

Correct Answer: BD

Note:

* The direct path read Oracle metric occurs during Direct Path operations when the data is asynchronously read from the database files into the PGA instead of

into the SGA data buffer.

Direct reads occur under these conditions:

-

When reading from the TEMP tablespace (a sort operation)

-

When reading a parallel full-table scan (parallel query factotum (slave) processes)

-Reading a LOB segment

* The optimizer uses a full table scan in any of the following cases:

-Lack of Index

-Large Amount of Data

-Small Table

-High Degree of Parallelism

QUESTION 3

Which three statements are true the Automatic Tuning Optimizer (ATO)?

A. It identifies the objects with stale or missing statistics and gathers statistics automatically.

B. It investigates the effect of new or modified indexes on the access paths for a workload and recommends running that statistics through the SQL Access Advisor.

C. It recommends a SQL profile to help create a better execution plan.

D. It picks up resource-intensive SQL statements from the ADDM and recommends the use of materialized views to improve query performance.

E. It identifies the syntactic, semantic, or design problems with structure of SQL statements leading to poor performance and suggests restricting the statements.

F. It identifies resource-intensive SQL statements, runs them through the SQL Tuning Advisor, and implements the recommendations automatically.

Correct Answer: ADF

Under tuning mode, the optimizer can take several minutes to tune a single statement. It is both time and resource intensive to invoke Automatic Tuning Optimizer every time a query must be hard-parsed. Automatic Tuning Optimizer is meant for complex and high-load SQL statements that have nontrivial impact on the database.

Automatic Database Diagnostic Monitor (ADDM) proactively identifies high-load SQL state- ments that are good candidates for SQL tuning. The automatic SQL tuning feature also automati- cally identifies problematic SQL statements and implements tuning recommendations during system maintenance windows as an automated maintenance task.

The Automatic Tuning Optimizer performs the following types of tuning analysis:

Statistics Analysis SQL Profiling Access Path Analysis SQL Structure Analysis Alternative Plan Analysis

Note:

* Oracle Database uses the optimizer to generate the execution plans for submitted SQL state- ments. The optimizer operates in the following modes:

Normal mode The optimizer compiles the SQL and generates an execution plan. The normal mode generates a reasonable plan for the vast majority of SQL statements. Under normal mode, the optimizer op- erates with very strict time constraints, usually a fraction of a second.

Tuning mode

The optimizer performs additional analysis to check whether it can further improve the plan pro- duced in normal mode. The optimizer output is not an execution plan, but a series of actions, along with their rationale and expected benefit for

producing a significantly better plan. When running in tuning mode, the optimizer is known as the Automatic Tuning Optimizer.

QUESTION 4

Examine the parallelism parameter for your instance:

NAME	TYPE	VALUE
Parallel_degree_limit	string	CPU
Parallel_degree_policy	string	AUTO
Parallel_max_servers	integer	128
Parallel_min_percent	integer	0
Parallel_min_servers	integer	0
Parallel_min_time_threshold	string	AUTO
Parallel_servers_target	integer	64

parallel_servers_target

Now examine the resource plan containing parallel statement directives:

Consumer Group resource plan containing parallel statement directives:

Consumer Group	: URGENT_GROUP
MGMT_P1	: 100%
PARELLEL_DEGREE_LIMIT_P1	: 12
PARALLEL_TARGET_PERCENTAGE	:
PARELLEL_TARGET	:

Consumer Group	: ETL_GROUP
MGMT_P1	: 100%
PARALLEL_DEGREE_LIMIT_P1	: 8
PARELLEL_QUIEZ_TIMEOUT	:

Consumer Group	: OTHER_GROUPS
MGMT_P3	: 100%
PARELLEL_DEGREE_LIMIT_P1	: 2
PARALLEL_TARGET_PERCENTAGE	:50%
PAALLEL_QUIT_TIMEOUT	: 360

Which two are true about parallel statement queuing when this plan is active?

A. Urgent_group sessions collectively can consume up to 64 parallel execution servers before queuing starts for this consumer group.

B. ETL_GROUP sessions can collectively consume up to 64 parallel execution servers before the queuing starts for this consumer.

C. A single OTHER_GROUPS session will execute serially once it is queued for six minutes.

D. A single ETL_GROUP session can consume up to eight parallel execution servers.

E. A single ETL_GROUP session can consume up to 32 parallel execution servers.

F. A single OTHER_GROUPS session will execute in parallel once it is queued for six minutes.

Correct Answer: AD

(http://docs.oracle.com/cd/E11882_01/server.112/e25494/dbrm.htm#ADMIN13466)

QUESTION 5

Examine the Exhibit.

```
CREATE TABLE dept AS SELECT * FROM departments;
ALTER TABLE dept PARALLEL 2;

CREATE TABLE emp_range_did PARTITION BY RANGE (department_id)
            (PARTITION emp_p1 VALUES LESS THAN (150),
            PARTITION emp_p5 VALUES LESS THAN (MAXVALUE) )
    AS SELECT * FROM employees;

ALTER TABLE emp_range_did PARALLEL 2;

EXPLAIN PLAN FRO
SELECT /*PQ_DISTRIBUTE (d NONE PARTITION) ORDERED */ e.last_name, d.department_name
            FROM emp_range_did e, dept d
            WHERE e.department_id = d.department_id;

------------------------------------------------------------------------------------------------

|    Id |    Operations                      Name            Rows    Bytes    Cost    Pstart    Pstop
TQ   |IN-OUT | |PQ  DISTRIB|
----------------------------------------------------------------------------------------

|  0|       SELECT STATEMENT                                  284     16188    6
      |      |
|  1|       PX COORDINATOR
      |      |
|  2|       PX SEND QC (RANDOM) :        TQ10001             284     16188    6
Q1, 01     |P->S| QC (RAND)|
|*  3|      HASH JOIN                                         284     16188    6
Q1, 01     |PCWP|
|  4|       PX PARTITION RANGE ALL                             284     7668     2        1         2
Q1, 01     |PCWC|
|  5|       TABLE ACCESS FULL            EMP_RANGE_DID       284     7668     2        1         2
Q1, 0      |PCWP|
|  6|       BUFFER SORT
Q1, 01     |PCWC|
|  7|       PX RECEIVE                                        21      630      2
Q1, 01     |PCWP|
|   8|      PX SEND PARTITION (KEY) :    TQ10000              21      630      2
           |S->P|PART (KEY)|
|   9|      TABLE ACCESS FULL            DEPT                 21      630      2
           |            |            |
-----------------------------------------------------------------------------------
```

Which two options are true about the execution plan and the set of statements?

A. The query uses a partial partition-wise join.

B. The degree of parallelism is limited to the number of partitions in the EMP_RANGE_DID table.

C. The DEPT table id dynamically distributed based on the partition keys of the EMP_RANGE_DID table.

D. The server process serially scans the entire DEPT table for each range partition on the EMP_RANGE_DID table.

E. The query uses a full partition-wise join.

Correct Answer: AD