1Z0-117Q&As

Oracle Database 11g Release 2: SQL Tuning Exam

Pass home 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 home Official Exam Center

QUESTION 1

Which two statements are true about the trcsess utility?

A. It merges multiple trace files and produces a formatted output file.

B. It merges multiple trace files from a particular session into one single trace file.

C. It produces multiple files only for DBA sessions, which can be consolidated into one formatted file using the tkprof utility.

D. It produces multiple files for a service, which can be consolidated into one formatted file using the tkprof utility.

E. It merges files pertaining to a user session scattered across different processes in a shared server configuration.

Correct Answer: AB

The trcsess utility consolidates trace output from selected trace files based on several criteria:

Session ID Client ID Service name Action name Module name After trcsess merges the trace information into a single output file, the output file could be processed by TKPROF.

Note:

*

trcsess is useful for consolidating the tracing of a particular session for performance or debugging purposes. Tracing a specific session is usually not a problem in the dedicated server model as a single dedicated process serves a session during its lifetime. You can see the trace information for the session from the trace file belonging to the dedicated server serving it. However, in a shared server configuration a user session is serviced by different processes from time to time. The trace pertaining to the user session is scattered across different trace files belonging to different processes. This makes it difficult to get a complete picture of the life cycle of a session.

Reference: Oracle Database Performance Tuning Guide 11g , Using the trcsess Utility

*

Now there is a new tool, a command line utility called trcsess to help read the trace files. The trcsess command-line utility consolidates trace information from selected trace files, based on specified criteria. The criteria include session id, client id, service name, action name and module name.

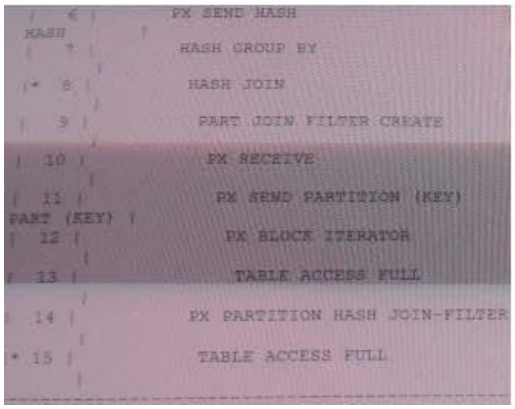
QUESTION 2

Examine the execution plan:

Id	Operation	Name	Pstart	Pstop	IN-OUT	PQ	Distrib
0	SELECT STATEMENT						
1	PX COORDINATOR						
2	PX SEND QC (RANDOM)	:TQ10002			P->S	QC	RAND
*3	FILTER				PCWC		
4	HASH GROUP BY				PCWP		
5	PX RECEIVE				PCWP		
6	PX SEND HASH	:TQ10001			P->P		HASH
7	HASH GROUP BY				PCWP		
*8	HASH JOIN				PCWP		
9	PART JOIN FILTER CREATE	:BF0000			PCWP		
10	PX RECEIVE				PCWP		
11	PX SEND PARTITION (KEY)	:TQ10000			P->P		PART(KEY)
12	PX BLOCK ITERATOR				PCWC		
13	TABLE ACCESS FULL	CUSTOMERS			PCWP		
14	PX PARTITION HASH JOIN-FILTER		:BF0000	:BF0000	PCWC		
*15	TABLE ACCESS FULL	SALES	:BF0000	:BF0000	PCWP		

Predicate Information (identified by operation id):

3 – filter (COUNT(SYS_OP_CSR(SYS_OP_MSR(COUNT(0)), 0))>100)
8 – access (“S”. “CUST_ID” = “C”. CUST_ID)
15 – filter (“S”. “TIME_ID” <=TO_DATE (‘ 1999 – 10 – 01 00:00:00’, ‘syyy-mm-dd ff24:mi:ss’)
AND
“S”. “TIME_ID”>=TO_DATE(‘1999-07-01’)
00:00:00’, ‘syyy-mm-dd hh24:mi:ss’))



Which two are true regarding the execution plan?

A. The CUSTOMERS table is hash partitioned.

B. The SALES table is hash partitioned.

C. The CUSTOMERS table is scanned first and selected partitions from the SALES table are scanned based on the BLOOM Filter created during the scan of the CUSTOMERS table.

D. The SALES table is scanned first and selected partitions from the CUSTOMERS table are scanned based on the Bloom Filter created during the scan of the SALES table.

E. Both the CUSTOMERS and SALES tables are scanned simultaneously and rows from the CUSTOMERS table are joined to row of the SALES table.

F. The CUSTOMERS table is range partitioned.

Correct Answer: BC

B: As per line 14 and 15.

C: As per exhibit line 13 is execute before line 15.

Incorrect:

Not E: As per line 13 and 15 they are not executed simultaneously.

QUESTION 3

You want to run SQL Tuning Advisor statements that are not captured by ADDM, AWR, and are not in the library cache. What is the prerequisite?

A. Enable SQL plan management

B. Create a SQL plan baseline for each query

C. Create a SQL Tuning Set (STS) containing the SQL statements

D. Gather statistics for objects used in the application

Correct Answer: C

You can use an STS as input to SQL Tuning Advisor, which performs automatic tuning of the SQL statements based on other user-specified input parameters.

Note:

A SQL tuning set (STS) is a database object that includes one or more SQL statements along with their execution statistics and execution context, and could

include a user priority ranking. You can load SQL statements into a SQL tuning set from different SQL sources, such as AWR, the shared SQL area, or

customized SQL provided by the user. An STS includes:

A set of SQL statements

Associated execution context, such as user schema, application module name and action, list of bind values, and the cursor compilation environment

Associated basic execution statistics, such as elapsed time, CPU time, buffer gets, disk reads, rows processed, cursor fetches, the number of executions, the

number of complete executions, optimizer cost, and the command type Associated execution plans and row source statistics for each SQL statement (optional). Reference: Oracle Database Performance Tuning Guide, Managing SQL Tuning Sets

QUESTION 4

Which statement is true about the usage of the STAR_TRANSFORMATION hint in a query?

A. The optimizer always uses a plan in which the transformation is used.

B. The optimizer uses transformation only if the cost is less than a query executing without transformation.

C. The optimizer always generates subqueries to transform a query.

D. The optimizer always uses bitmap indexes on the primary key column for any dimension table to transform a query.

Correct Answer: B

(http://docs.oracle.com/cd/E11882_01/server.112/e41084/sql_elements006.htm#SQLRF50508) says that "Even if the hint is specified, there is no guarantee that the transformation will take place".

QUESTION 5

Identify two situations in which full table scans will be faster than index range scans.

A. A query with a highly selective filter fetching less than 5 percent of the rows from a table.

B. A highly selective query on a table having high clustering factor for an index.

C. A query fetching less number of blocks than value specified by DB_FILE_MULTIBLOCK_READ_COUNT.

D. A query executing in parallel on a partitioned table with partitioned indexes.

E. A query on a table with sparsely populated table blocks.

Correct Answer: CD

D: DB_FILE_MULTIBLOCK_READ_COUNT is one of the parameters you can use to minimize I/O during table scans. It specifies the maximum number of blocks read in one I/O operation during a sequential scan. The total number of I/Os needed to perform a full table scan depends on such factors as the size of the table, the multiblock read count, and whether parallel execution is being utilized for the operation.

Online transaction processing (OLTP) and batch environments typically have values in the range of 4 to 16 for this parameter. DSS and data warehouse environments tend to benefit most from maximizing the value of this parameter. The optimizer is more likely to choose a full table scan over an index if the value of this parameter is high.

Note:

* See 6) and 7) below.

The oracle optimizer choose the best plan and execute the query according the plan. It is common to hear that my table has indexes but why oracle does not use

indexes rather it is using full table scan.

There are several reasons behind choosing optimizer full table scans.

1)The table has no indexes within it.

2)Table has indexes but they are not appropriate to queries. For example in the table there is normal B- tree indexes but in the query the column used in the

WHERE clause contains function.

3)Query access large amount of data. The table has indexes but query against it select almost all of the rows. In that case optimizer might choose to full access of

table.

4)Index creation order may not appropriate. You have composite indexes on a table but in the where clause the leading column inside indexes are not used rather

trailing columns are used.

5)The table is skewed. For example column gender contains value \\'M\\' 10,000 times but value \\'F\\' only 10 times.

6)The table is small. If a table can read in a single I/O call, then a full table scan might be cheaper than an index range scan. Single I/O call is defined by

DB_FILE_MULTIBLOCK_READ_COUNT parameter and value defined by blocks.

Check it by,

SQL> show parameter DB_FILE_MULTIBLOCK_READ_COUNT

NAME TYPE VALUE

---------- ---------- ---------- db_file_multiblock_read_count integer 16

7)High degree of parallelism. High degree of parallelism skews the optimizer toward full table scans.

8)In the query if there is no filtering then full table scan is the choice.

If an index has poor cardinality (ie. more than 4% rows with the same index key) *

then it will perform poorly. It will usually be faster to perform a full table scan. eg. Table SALES has an index on the column PAYMENT_METHOD which can

contain values such as COD, CREDIT, CHEQUE, CASH. The statement

SELECT *

FROM sales

WHERE payment_method = \\'CASH\\'

will probably perform so badly that you are better off without the index.

* Oracle uses the full table scan as it assumes that it will have to read a certain part of the table.

Reference: Oracle Database Reference, DB_FILE_MULTIBLOCK_READ_COUNT