DUMPSARENA

Oracle Database 19c: Performance Management and Tuning

Oracle 1z0-084

Version Demo

Total Demo Questions: 7

Total Premium Questions: 73

Buy Premium PDF

https://dumpsarena.co

sales@dumpsarena.co

sales@dumpsarena.co
dumpsarena.co

Topic Break Down

Topic	No. of Questions
Topic 1, General Database and Network Administration	5
Topic 2, Using Oracle Database Resource Manager	1
Topic 3, Managing Database Performance	28
Topic 4, SQL Tuning	16
Topic 5, Optimizing Instance Performance	3
Topic 6, Managing Optimizer Statistics	10
Topic 7, Managing Memory	8
Topic 8, Mix Questions	2
Total	73

QUESTION NO: 1

Which two options are part of a Soft Parse operation?

A. Syntax Check

B. SQL Row Source Generation

C. SQL Optimization

D. Shared Pool Memory Allocation

E. Semantic Check

F. Syntax Check and Semantic Check

ANSWER: F

Explanation:

Syntax Check and Semantic Check is correct. When Oracle receives a SQL statement, parsing includes validating that the statement is structurally valid SQL and that its referenced database objects and names can be resolved in the current context. Syntax checking verifies that the SQL text follows Oracle SQL language rules. Semantic checking validates the meaning of the statement, including object existence and accessibility, and is associated with checking the statement in the context of the submitting user's schema and privileges.

A soft parse occurs when Oracle can find a matching, reusable cursor in the shared SQL area rather than having to fully optimize the statement and generate a new executable representation. Reuse still depends on Oracle being able to validate the statement and cursor for the current execution context. Consequently, syntax and semantic validation are the two parsing-related checks identified by this question. Combining them into one answer is necessary because the item is designated Single Choice even though its wording asks for two operations.

Oracle describes SQL parsing, shared SQL areas, and cursor reuse in the Oracle Database Concepts 19c SQL processing documentation. Additional parsing and cursor-sharing context is available in the Oracle Database SQL Tuning Guide 19c.

QUESTION NO: 2

A serial query runs for eight seconds of elapsed time, but its database CPU time plus I/O wait time is only one second.

You want Oracle Database to generate a SQL Monitor report for this statement on every execution.

Which action should you take?

A. Add the `MONITOR` hint to the statement.

B. Add the `GATHER_PLAN_STATISTICS` hint to the statement.

C. Enable SQL Trace for the session executing the statement.

D. Set `STATISTICS_LEVEL` to `ALL` for the session.

ANSWER: A

Explanation:

Adding the `MONITOR` hint is correct. Automatic SQL monitoring normally begins when a statement is executed in parallel or when its combined database CPU and I/O time exceeds the monitoring threshold. Elapsed time that is attributable to client processing, network delay, or other nonqualifying time does not make this serial statement eligible automatically.

The `MONITOR` hint explicitly requests SQL monitoring for the statement. `GATHER_PLAN_STATISTICS` collects row-source statistics for plan analysis, whereas SQL Trace and `STATISTICS_LEVEL=ALL` do not force Real-Time SQL Monitoring.

QUESTION NO: 3

A database instance is suffering poor I/O performance on two frequently accessed large tables.

No Big Table caching occurs in the database.

Examine these parameter settings:

Which are two actions either one of which will allow Big Table caching to occur?

A. Increasing DB_BIG_TABLE_CACHE_PERCENT_TARGET to at least 50

B. Setting DB_KEEP_CACHE_SIZE to at least 50M

C. Increasing DB_BIG_TABLE_CACHE_PERCENT_TARGET to at least 25

D. Increasing DB_CACHE_SIZE to 1 G

E. Setting PARALLEL_DEGREE_POLICY to ADAPTIVE

F. Setting PARALLEL_DEGREE_POLICY to AUTO

ANSWER: E F

Explanation:

Setting `PARALLEL_DEGREE_POLICY` to `ADAPTIVE` or setting `PARALLEL_DEGREE_POLICY` to `AUTO` will allow Big Table caching to occur. Oracle Big Table caching uses a specified portion of the buffer cache for caching blocks read during large-table scans. The existing nonzero `DB_BIG_TABLE_CACHE_PERCENT_TARGET` setting makes a portion of the cache eligible for this feature, but eligibility alone does not activate Big Table caching for the relevant scans.

Automatic degree of parallelism must be enabled for Big Table caching. Both `AUTO` and `ADAPTIVE` enable automatic DOP. With either setting, Oracle can determine that a statement should use parallel execution and can use the Big Table cache for qualifying large-table scans. `ADAPTIVE` additionally provides adaptive parallelism behavior, whereas `AUTO` enables automatic DOP without that adaptive behavior; both satisfy the Big Table caching prerequisite.

Oracle documents that Big Table caching requires a nonzero `DB_BIG_TABLE_CACHE_PERCENT_TARGET` and automatic DOP. See the DB_BIG_TABLE_CACHE_PERCENT_TARGET reference and the PARALLEL_DEGREE_POLICY reference.

QUESTION NO: 4

Which two statements are true about Automatic Workload Repository (AWR) snapshots and baselines?

A. A manual snapshot can be created by using `DBMS_WORKLOAD_REPOSITORY.CREATE_SNAPSHOT`.

B. A baseline preserves its included snapshots beyond normal AWR retention.

C. Changing the AWR snapshot interval automatically creates a fixed baseline.

D. Creating a baseline enables SQL monitoring for all SQL statements.

E. A snapshot can be included in only one AWR baseline.

ANSWER: A B

Explanation:

A manual AWR snapshot can be created by using `DBMS_WORKLOAD_REPOSITORY.CREATE_SNAPSHOT`. This is useful when a DBA wants snapshots immediately before and after a short or anticipated workload event rather than waiting for the normal snapshot interval.

An AWR baseline preserves the snapshots included in the baseline beyond the normal AWR retention period. A baseline is therefore suitable for retaining representative workload data for comparison with a later period. Changing the snapshot

interval does not itself create a baseline, and a baseline does not cause every SQL statement in the database to be monitored.

QUESTION NO: 5

A SQL statement has two enabled, accepted, and reproducible plans in its SQL Plan Baseline.

One plan is marked as fixed and has a higher optimizer cost. The other plan is not fixed and has a lower optimizer cost.

Which plan will Oracle Database select?

A. The enabled fixed plan.

B. The enabled nonfixed plan because it has the lower optimizer cost.

C. Both plans alternately, because both plans are accepted.

D. Neither plan, because fixed plans cannot coexist with nonfixed plans.

ANSWER: A

Explanation:

Oracle Database gives preference to fixed SQL plan baseline plans. When more than one fixed plan is enabled, accepted, and reproducible, the optimizer chooses the fixed plan with the lowest cost among those fixed plans. A nonfixed accepted plan is not selected while an eligible fixed plan exists, even if the nonfixed plan has a lower calculated cost.

Marking a plan as fixed is therefore an explicit plan-stability decision. It should be used only when administrators intend to preserve that plan in preference to ordinary baseline plan evolution and cost-based selection.

QUESTION NO: 6

A table has deliberately locked optimizer statistics because its application provider controls all statistics changes.

Which two statements are true?

A. Automatic optimizer statistics collection does not gather statistics for the locked table.

B. `DBMS_STATS.GATHER_TABLE_STATS` can gather statistics despite the lock when `FORCE` is set to `TRUE`.

C. Locking statistics prevents inserts, updates, and deletes against the table.

D. Locking statistics causes the optimizer to ignore the statistics already stored for the table.

E. Statistics can be locked only for objects in the `SYS` schema.

ANSWER: A B

Explanation:

Locked statistics protect the existing statistics from normal collection activity. Automatic optimizer statistics collection does not gather statistics for an object whose statistics are locked, so the approved statistics remain in effect.

A DBA can deliberately override the lock when necessary by using a `DBMS_STATS` gathering procedure with the `FORCE` argument set to `TRUE`. This is an explicit override and should be used only when replacing the locked statistics has been approved. Locking statistics does not prevent DML and does not prevent the optimizer from using the existing statistics.

QUESTION NO: 7

Multiple sessions are inserting data concurrently into a table that has an LOB column.

At some point in time, one of the sessions cannot find available space in the LOB segment and needs to allocate a new extent.

Which wait event will be raised in the other sessions that need space in the LOB column?

A. enq: SQ - contention

B. enq: TM - contention

C. enq: HW - contention

D. enq: TX - allocate ITL entry

ANSWER: C

Explanation:

`enq: HW - contention` is the appropriate wait event because concurrent inserts into the same LOB segment can require Oracle to advance the segment's high water mark or extend the segment by allocating an additional extent. The high water mark enqueue serializes this space-management activity so that only one session performs the relevant allocation and high-water-mark update at a time. While that operation is in progress, other sessions that also need newly available space in that LOB segment wait on the high water mark enqueue, reported as `enq: HW - contention`. This is particularly visible when concurrent activity rapidly consumes free space and causes repeated segment growth. The wait therefore identifies contention associated with extending a segment and making that newly allocated space available for use, rather than contention over application data values. Oracle documents enqueue waits, including high water mark contention, in its wait-event reference. Further context on segment space allocation and diagnosing contention is available in the Oracle performance tuning documentation. See Oracle Database Reference: Descriptions of Wait Events and Oracle Database Performance Tuning Guide.