PassQueen
a.k.a dumpsqueen.com

Oracle AI Vector Search Professional

Oracle 1z0-184-25

Version Demo

Total Demo Questions: 7

Total Premium Questions: 72

Buy Premium PDF

https://passqueen.com

support@passqueen.com

QUESTION NO: 1

Which parameter is used to define the number of closest vector candidates considered during HNSW index creation?

A. EFCONSTRUCTION

B. VECTOR_MEMORY_SIZE

C. NEIGHBOURS

D. TARGET_ACCURACY

ANSWER: A

Explanation:

`EFCONSTRUCTION` is the HNSW index-creation parameter that specifies the size of the dynamic candidate list used while building the graph. During construction, Oracle evaluates candidate vectors to identify suitable graph connections; this setting determines how many closest candidates are retained and considered in that process. Consequently, it is a key build-time quality-versus-resource control: a larger value can produce a better-connected HNSW graph and improve the likelihood of accurate nearest-neighbor results, while also requiring more index-build work, time, and memory. The parameter applies specifically to creation of an HNSW vector index, rather than being a general database memory setting or a query-only accuracy preference. Oracle documents `EFCONSTRUCTION` among the HNSW index parameters and describes it as controlling the number of closest vector candidates considered at each construction step. See Oracle's HNSW vector index creation documentation and the CREATE VECTOR INDEX SQL reference for the supported index syntax and parameter definitions.

QUESTION NO: 2

An administrator creates an HNSW vector index and receives an error indicating that the database cannot allocate sufficient vector memory. The table and embedding model are otherwise valid. What is the most appropriate corrective action?

A. Increase VECTOR_MEMORY_SIZE and make the static parameter change effective

B. Create a B-tree index on the VECTOR column

C. Increase the text chunk overlap used before embedding generation

D. Replace the VECTOR column with a CLOB column

ANSWER: A

Explanation:

Increase `VECTOR_MEMORY_SIZE` to provide sufficient Vector Pool memory and restart as required for the static parameter change. HNSW indexes use an in-memory proximity graph supported by the Vector Pool. Adding an ordinary B-tree index or changing the source-document chunk size does not provide memory for the HNSW graph. Changing to an IVF index is an architectural choice, not the direct correction for insufficient HNSW vector memory.

QUESTION NO: 3

You need to prioritize accuracy over speed in a similarity search for a dataset of images. Which should you use?

A. Approximate similarity search with HNSW indexing and target accuracy of 70%

B. Multivector similarity search with partitioning

C. Exact similarity search using a full table scan

D. Approximate similarity search with IVF indexing and target accuracy of 70%

ANSWER: C

Explanation:

Exact similarity search using a full table scan is the appropriate choice when accuracy is more important than response time. In an exact vector search, Oracle evaluates the distance between the query vector and every candidate vector in the searched dataset, then returns the vectors with the best calculated distances according to the selected distance metric. Because all applicable vectors are considered, this method provides exact nearest-neighbor results rather than results bounded by an approximate-search recall target. That behavior is especially suitable for image datasets when missing a truly closest image would be unacceptable, such as in validation, quality-sensitive retrieval, or benchmark scenarios. The tradeoff is that evaluating every vector can require substantially more database work and therefore take longer as the dataset grows. Oracle documents exact similarity search as the non-indexed approach and explains that a vector index is used for approximate similarity search, which prioritizes faster retrieval. See Oracle's Perform Exact Similarity Search documentation and its Perform Approximate Similarity Search documentation.

QUESTION NO: 4

What is the purpose of the VECTOR_DISTANCE function in Oracle Database 23ai similarity search?

A. To fetch rows that match exact vector embeddings

B. To create vector indexes for efficient searches

C. To group vectors by their exact scores

D. To calculate the distance between vectors using a specified metric

ANSWER: D

Explanation:

The correct answer is **"To calculate the distance between vectors using a specified metric."** Oracle Database's `VECTOR_DISTANCE` SQL function evaluates the distance, or dissimilarity, between two vector expressions. Its optional metric argument selects the calculation method, such as Euclidean distance, cosine distance, dot product, Manhattan distance, Hamming distance, or Jaccard distance, subject to the vector formats and dimensions involved. A smaller distance generally indicates that vectors are more similar under the selected metric.

This function is fundamental to vector similarity search because it provides a score that SQL can use to rank data, commonly through an `ORDER BY VECTOR_DISTANCE(...)` clause together with a row limit to return the nearest results. For example, an application can compare a query embedding generated from a user's text with embedding vectors stored for documents, products, or images, then retrieve the records with the lowest calculated distances. The chosen distance metric must align with the metric used by the embedding model and, where applicable, the vector index configuration, so that similarity results are meaningful and can benefit from optimized vector search execution. See Oracle's VECTOR_DISTANCE SQL Language Reference and the Oracle AI Vector Search similarity-search documentation.

QUESTION NO: 5

Which PL/SQL package is primarily used for interacting with Generative AI services in Oracle Database 23ai?

A. DBMS_AI

B. DBMS_ML

C. DBMS_VECTOR_CHAIN

D. DBMS_GENAI

ANSWER: A

Explanation:

`DBMS_AI` is the primary Oracle Database PL/SQL package for configuring and invoking generative AI capabilities. It provides the database-side interface used to create and manage AI profiles, which contain the credentials and provider settings needed to connect Oracle Database to supported external AI services, including OCI Generative AI. These profiles enable SQL and PL/SQL workloads to submit prompts and use large language model functionality without requiring application code to implement the service integration directly. A central use case is Select AI: after an AI profile is configured, developers can use natural-language prompts to generate, explain, or summarize SQL-related results through database features backed by generative AI models. The package also supports programmatic administration of those profiles, making it the appropriate package when the task is interaction with generative AI services rather than solely vector transformation or traditional in-database machine-learning model operations. Oracle documents `DBMS_AI` as the package for managing AI profiles and accessing generative AI services from Oracle Database. See the DBMS_AI package reference and Oracle's SELECT AI documentation.

QUESTION NO: 6

What is the primary function of AI Smart Scan in Exadata System Software 24ai?

A. To provide real-time monitoring and diagnostics for AI applications

B. To accelerate AI workloads by leveraging Exadata RDMA Memory (XRMEM), Exadata Smart Cache, and on-storage processing

C. To automatically optimize database queries for improved performance

ANSWER: B

Explanation:

AI Smart Scan is designed to accelerate AI and vector-search workloads by extending Exadata's storage offload model to vector operations. Rather than requiring all candidate vector data to be transferred from storage to database servers for processing, the feature enables relevant work to be performed closer to the data on Exadata storage servers. This substantially reduces data movement and database-server CPU demand, which is especially valuable when searching large vector collections.

The feature integrates Exadata capabilities such as RDMA Memory (XRMEM), Exadata Smart Cache, and on-storage processing. These capabilities help make vector similarity queries more efficient by serving frequently accessed data quickly and by offloading eligible processing. The result is lower latency and higher throughput for AI applications that use Oracle AI Vector Search, while retaining the database's SQL, security, and transactional capabilities. Oracle describes Exadata's AI capabilities as optimized for accelerating vector search and generative-AI workloads, and its Exadata documentation covers the storage-server offload architecture underpinning Smart Scan functionality. See Oracle Exadata and Oracle AI Vector Search documentation.

QUESTION NO: 7

You want to quickly retrieve the top-10 matches for a query vector from a dataset of billions of vectors, prioritizing speed over exact accuracy. What is the best approach?

A. Exact similarity search using flat search

B. Approximate similarity search with a low target accuracy setting

C. Relational filtering combined with an exact search

D. Exact similarity search with a high target accuracy setting

ANSWER: B

Explanation:

Approximate similarity search with a low target accuracy setting is the appropriate approach when response time is more important than returning the mathematically exact nearest neighbors. Oracle AI Vector Search supports approximate nearest-neighbor search through vector indexes, including Hierarchical Navigable Small World indexes and Inverted File Flat indexes. These indexes avoid comparing the query vector against every vector in a multi-billion-row data set, substantially reducing query latency and making top-k retrieval practical at large scale.

The target accuracy controls the recall objective for an approximate search: a lower value permits the optimizer and index search to examine less of the search space, generally improving speed while accepting that some exact nearest neighbors may not be returned. Requesting only the top 10 results further aligns with indexed approximate retrieval, because the database can efficiently identify a small set of highly similar candidates rather than performing an exhaustive scan. Oracle documents approximate similarity search as the scalable choice for vector workloads where a controlled accuracy-versus-performance trade-off is acceptable. See the Oracle documentation on approximate similarity search and Oracle guidance for HNSW vector indexes.