Google Cloud

Partner Certification Academy

Professional Cloud Developer

v2309

Quiz questions*

Cloud Bigtable

** These are for practice only and are not actual exam questions*

Question: In Cloud Bigtable, what is the primary driver for schema design?

- A. Storage capacity
- B. Write requests
- C. Queries or read requests
- D. Column families

Question: Which of the following is NOT a general concept of Bigtable schema design?

- A. Bigtable supports multiple indices.
- B. Bigtable is a key/value store, not a relational store.
- C. Each table has only one index, the row key.

- D. Rows are sorted lexicographically by row key.

Question: What is the most efficient way to read your Bigtable data?

- A. Reading a specific column
- B. Reading a row range
- C. Reading based on column families
- D. Reading based on timestamps

Question: What is the maximum size of a row key in Bigtable?

- A. 1 KB
- B. 4 KB
- C. 16 KB
- D. 256 MB

Question: How are rows sorted in Bigtable by default?

- A. Numerically
- B. Alphabetically
- C. Lexicographically by row key
- D. Based on timestamps

Question: Why is it recommended to use human-readable string values in your row keys for Bigtable?

- A. To reduce storage costs.
- B. To improve write performance.
- C. To make it easier to use the Key Visualizer tool.
- D. To increase data encryption.

Question: Which of the following row key patterns can cause a hotspot in Bigtable?

- A. Row keys that start with a user ID.
- B. Row keys that start with a timestamp.
- C. Row keys that end with a timestamp.
- D. Row keys that start with a domain name.

Question: For values that change frequently, such as a counter updated hundreds of times each minute, what is the recommended approach in Bigtable?

- A. Update the same row key repeatedly.
- B. Store each new reading in a new row.
- C. Use a timestamp as the row key.
- D. Store the data in multiple tables.

Question: Why is it advised to avoid using hashed values as row keys in Bigtable?

- A. Hashed values increase storage costs.
- B. Hashed values make it challenging to use the Key Visualizer tool.
- C. Hashed values reduce read performance.
- D. Hashed values are not supported in Bigtable.

Question: Why is it important to pad integers with leading zeroes when they are part of a row key in Bigtable?

- A. To ensure they are stored as strings.
- B. To ensure they are sorted numerically.
- C. To reduce storage costs.
- D. To improve write performance.

Question: When designing a row key for tracking mobile device data, which of the following row key designs allows you to retrieve data with a single request for a combination of device type and device ID?

- A. deviceID#20200501#phone
- B. 20200501#phone#deviceID
- C. phone#deviceID#20200501
- D. deviceID#phone#20200502

Answers to Quiz questions

Cloud Bigtable

Question: In Cloud Bigtable, what is the primary driver for schema design?

- A. Storage capacity
- B. Write requests
- C. Queries or read requests
- D. Column families

Correct Answer: C. Queries or read requests

Explanation: In Bigtable, schema design is driven primarily by the queries, or read requests, that you plan to send to the table.
Resource: Cloud Bigtable Schema Design Best Practices

Question: Which of the following is NOT a general concept of Bigtable schema design?

- A. Bigtable supports multiple indices.
- B. Bigtable is a key/value store, not a relational store.
- C. Each table has only one index, the row key.
- D. Rows are sorted lexicographically by row key.

Correct Answer: A. Bigtable supports multiple indices.

Explanation: Bigtable does not support multiple indices. Each table has only one index, which is the row key.
Resource: Cloud Bigtable Schema Design Best Practices

Question: What is the most efficient way to read your Bigtable data?

- A. Reading a specific column
- B. Reading a row range
- C. Reading based on column families
- D. Reading based on timestamps

Correct Answer: B. Reading a row range

Explanation: Reading a row range is the fastest way to read your Bigtable data.
Resource: Cloud Bigtable Schema Design Best Practices

Question: What is the maximum size of a row key in Bigtable?

- A. 1 KB
- B. 4 KB
- C. 16 KB
- D. 256 MB

Correct Answer: B. 4 KB

Explanation: A row key in Bigtable must be 4 KB or less.
Resource: Cloud Bigtable Schema Design Best Practices

Question: How are rows sorted in Bigtable by default?

- A. Numerically
- B. Alphabetically
- C. Lexicographically by row key
- D. Based on timestamps

Correct Answer: C. Lexicographically by row key

Explanation: Rows are sorted lexicographically by row key, from the lowest to the highest byte string.
Resource: Cloud Bigtable Schema Design Best Practices

Question: Why is it recommended to use human-readable string values in your row keys for Bigtable?

- A. To reduce storage costs.
- B. To improve write performance.
- C. To make it easier to use the Key Visualizer tool.
- D. To increase data encryption.

Correct Answer: C. To make it easier to use the Key Visualizer tool.

Explanation: Using human-readable string values in row keys makes it easier to troubleshoot issues with Bigtable using the Key Visualizer tool.
Resource: Cloud Bigtable Schema Design Best Practices

Question: Which of the following row key patterns can cause a hotspot in Bigtable?

- A. Row keys that start with a user ID.
- B. Row keys that start with a timestamp.
- C. Row keys that end with a timestamp.
- D. Row keys that start with a domain name.

Correct Answer: B. Row keys that start with a timestamp.

Explanation: Row keys that start with a timestamp cause sequential writes to be pushed onto a single node, creating a hotspot.
Resource: Cloud Bigtable Schema Design Best Practices

Question: For values that change frequently, such as a counter updated hundreds of times each minute, what is the recommended approach in Bigtable?

- A. Update the same row key repeatedly.
- B. Store each new reading in a new row.
- C. Use a timestamp as the row key.
- D. Store the data in multiple tables.

Correct Answer: B. Store each new reading in a new row.

Explanation: For frequently changing values, it's best to store each new reading in a new row. This avoids overloading a tablet that stores a frequently used row and ensures efficient data retrieval.
Resource: Cloud Bigtable Schema Design Best Practices

Question: Why is it advised to avoid using hashed values as row keys in Bigtable?

- A. Hashed values increase storage costs.
- B. Hashed values make it challenging to use the Key Visualizer tool.
- C. Hashed values reduce read performance.
- D. Hashed values are not supported in Bigtable.

Correct Answer: B. Hashed values make it challenging to use the Key Visualizer tool.

Explanation: Hashing a row key removes the ability to take advantage of Bigtable's natural sorting order, making it challenging to use tools like the Key Visualizer for troubleshooting.
Resource: Cloud Bigtable Schema Design Best Practices

Question: Why is it important to pad integers with leading zeroes when they are part of a row key in Bigtable?

- A. To ensure they are stored as strings.
- B. To ensure they are sorted numerically.
- C. To reduce storage costs.
- D. To improve write performance.

Correct Answer: B. To ensure they are sorted numerically.

Explanation: Bigtable stores data lexicographically. For example, lexicographically, 3 > 20 but 20 > 03. Padding the 3 with a leading zero ensures that the numbers are sorted numerically. This tactic is especially important for timestamps where range-based queries are used.
Resource: Schema design best practices

Question: When designing a row key for tracking mobile device data, which of the following row key designs allows you to retrieve data with a single request for a combination of device type and device ID?

- A. deviceID#20200501#phone
- B. 20200501#phone#deviceID
- C. phone#deviceID#20200501
- D. deviceID#phone#20200502

Correct Answer: C. phone#deviceID#20200501

Explanation: A row key design that starts with the device type, followed by the device ID, and then the day the data is recorded, such as "phone#deviceID#20200501", allows efficient retrieval of data based on a combination of device type and device ID.
Resource: Schema design best practices