# **Day 34 - 19 September 2025**

**Document Name:**Day 34 - hmuvvala@ - Hari Gopal Muvvala

### **Task 01**

**Question:**

In DynamoDB, which factor triggers the creation of new partitions automatically?

1. Every time a user creates a new GSI on the table.
2. When the storage in an existing partition exceeds 10 GB or throughput requirements surpass limits.
3. When global replication is enabled via global tables.
4. Upon inserting items larger than 400 KB individually.

**Answer:** 2

**Notes:**

* Partition split happens automatically when data > 10GB or throughput demand exceeds limit.
* DynamoDB manages partitions internally (not manual).

### **Task 02**

**Question:**

Which statement is true about write operations using AWS SDK in DynamoDB?

1. Write operations always replace the entire item unless conditionally filtered.
2. The SDK enforces write isolation by automatically disabling concurrent access.
3. You can use UpdateItem with expressions to modify specific attributes without affecting the rest.
4. All write operations must be wrapped in transactions when using SDK.

**Answer:** 3

**Notes:**

* UpdateItem allows modifying specific fields.
* PutItem replaces the entire item.
* Transactions are optional, not required for all writes.

### **Task 03**

**Question:**

In the AWS SDK, which method would you use to increase an existing numeric attribute without retrieving its current value?

1. Use PutItem with full item replacement and a computed value.
2. Use UpdateItem with ADD operation on the attribute.
3. Use GetItem followed by UpdateItem with arithmetic expression.
4. Use IncrementItem method under conditional write configuration.

**Answer:** 2

**Notes:**

* UpdateItem + ADD operation is used for **atomic counters**.
* Efficient because no read-modify-write cycle is required.

### **Task 04**

**Question:**

When using the AWS CLI to delete a DynamoDB table, what precaution must be taken?

1. Disable all indexes before delete.
2. Manually drain capacity units.
3. Deleting is immediate, but recovery is only possible if PITR was enabled.
4. Deletes are queued up to 48 hours.

**Answer:** 3

**Notes:**

* Deleting a table is permanent unless **Point-in-Time Recovery** (PITR) is enabled.

### **Task 05**

**Question:**

In the AWS Console, what happens if you try to create a table with a duplicate name in the same region?

1. Console appends timestamp.
2. Creation fails with validation error.
3. Silently overwrites existing table.
4. Creates a versioned table with identical name but different ARN.

**Answer:** 2

**Notes:**

* Table names must be unique per region.

### **Task 06**

**Question:**

What is a key characteristic of DynamoDB partitions that influences table performance?

1. Each partition can contain up to 100 GB and supports 3000 RCUs + 1000 WCUs.
2. Partitions store only indexes.
3. Partition creation must be triggered manually.
4. Throughput quota is shared equally across partitions.

**Answer:** 1

**Notes:**

* Partition limits: **10GB (storage split)** in earlier tasks, but throughput ~ 3000 RCUs & 1000 WCUs per partition.
* Auto-managed by DynamoDB.

### **Task 07**

**Question:**

In DynamoDB data modeling, what does the multi-value sort key pattern allow you to do?

1. Use multiple sort keys across partition keys.
2. Emulate hierarchical schema (time ranges, event types).
3. Store list-type values in sort key.
4. Enable join-like behavior across tables.

**Answer:** 2

**Notes:**

* Multi-value sort keys used to model **hierarchies** or **time series** in one table.

### **Task 08**

**Question:**

What is a recommended method to lower DynamoDB costs while maintaining access performance?

1. Use multiple GSIs for throughput distribution.
2. Store large binary objects directly.
3. Optimize access patterns with compound keys and avoid unnecessary indexes/scans.
4. Enable strong consistency on all reads.

**Answer:** 3

**Notes:**

* Best practice: design **access patterns** properly.
* Store large objects in S3, not DynamoDB.

### **Task 09**

**Question:**

How can hot partitions be avoided in a high-volume DynamoDB design?

1. Use a fixed partition key.
2. Adopt key sharding (prefix/suffix with hash or random).
3. Implement Athena auto-balancing.
4. Enable parallel scans.

**Answer:** 2

**Notes:**

* Sharding distributes load evenly.
* Prevents one partition key being overloaded.

### **Task 10**

**Question:**

How do Local Secondary Indexes (LSIs) affect storage limits per partition key in DynamoDB?

1. No impact.
2. Share same 400 KB item limit, no cumulative limit.
3. Total size per partition key (including LSIs) ≤ 10 GB.
4. Each LSI adds extra 10 GB limit.

**Answer:** 3

**Notes:**

* All items with same partition key (base + LSIs) share **10 GB total limit**.

### **Task 11**

**Question:**

Which AWS CLI command correctly updates an item’s attribute score by adding 5 in a table named Players?

1. aws dynamodb update-item --table-name Players --key '{"PlayerId":{"S":"101"}}' --update-expression "ADD score :val" --expression-attribute-values '{":val":{"N":"5"}}'
2. aws dynamodb modify-item --table-name Players ...
3. aws dynamodb increment-item ...
4. aws dynamodb update-item --table-name Players ... --update-expression "SET score = score + :val"

**Answer:** 1

**Notes:**

* Correct CLI syntax: **UpdateItem + ADD**.
* SET arithmetic is not supported in CLI (only ADD).