# **Day 32 - 12 September 2025**

**Document Name:**Day 32 - hmuvvala@ - Hari Gopal Muvvala

### **1. Features of DynamoDB**

* Fully managed NoSQL database service.
* Provides predictable performance with seamless scaling.
* Offers flexible data models: key–value and document.
* Automatic replication across multiple AZs (high availability).
* Fine-grained access control with IAM.
* Supports ACID transactions for mission-critical workloads.

### **2. Advantages of DynamoDB**

* Highly scalable (automatic partitioning).
* Low latency reads/writes.
* Integration with other AWS services (Lambda, API Gateway, etc.).
* Flexible consistency models (eventual and strong consistency).
* Supports both provisioned and on-demand capacity modes.

### **3. Disadvantages of DynamoDB**

* Costs can increase with heavy read/write workloads.
* Querying is limited compared to relational databases.
* Indexes must be designed carefully at the start.
* Complex joins or aggregations are not natively supported.

### **4. DynamoDBMapper**

* An object persistence model in the AWS SDK for Java.
* Maps Java classes to DynamoDB tables.
* Simplifies CRUD operations without manually building requests.
* Example: annotate Java class with @DynamoDBTable, use DynamoDBMapper to save, load, or delete objects.

### **5. DynamoDB Streams**

* Captures a time-ordered sequence of item-level changes in a table.
* Can be used to trigger AWS Lambda functions for real-time processing.
* Use cases:
  + Replicating data across regions.
  + Building event-driven architectures.
  + Auditing and logging changes.

### **6. Pricing Models**

* **Provisioned mode** → specify Read Capacity Units (RCU) and Write Capacity Units (WCU).
* **On-demand mode** → pay per request, no need to specify capacity.
* Additional charges for storage, Streams, backups, and DAX usage.

### **7. Hot Keys Issue**

* A hot key is when too many read/write requests are routed to the same partition key.
* Leads to throttling and uneven distribution of workload.
* Best practice: choose partition keys with high cardinality to spread requests evenly.

### **8. DynamoDB Accelerator (DAX)**

* In-memory caching service for DynamoDB.
* Reduces read latency from milliseconds to microseconds.
* Fully managed and highly available.
* Drop-in replacement (works with DynamoDB API calls).