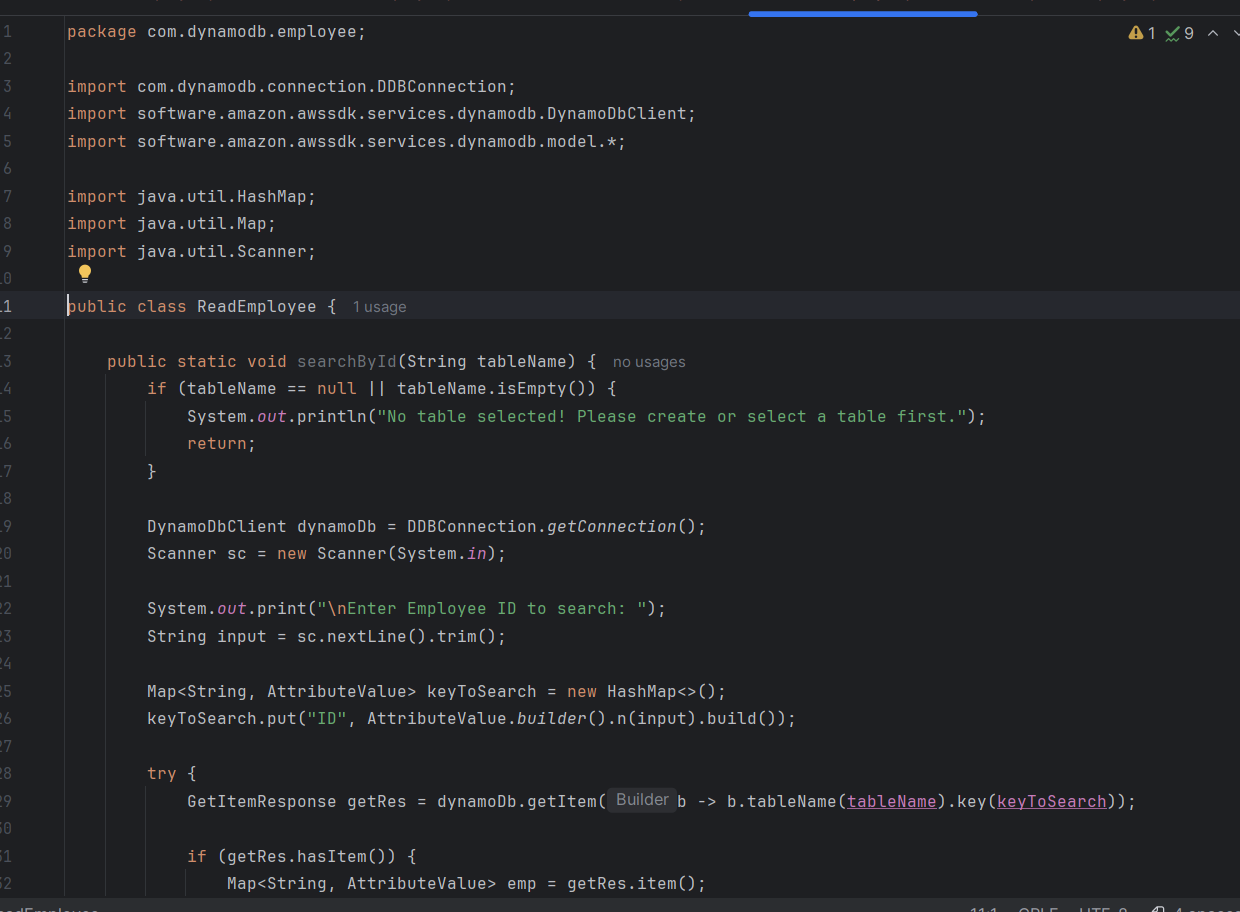
Day 34 - 10th Sept 2025

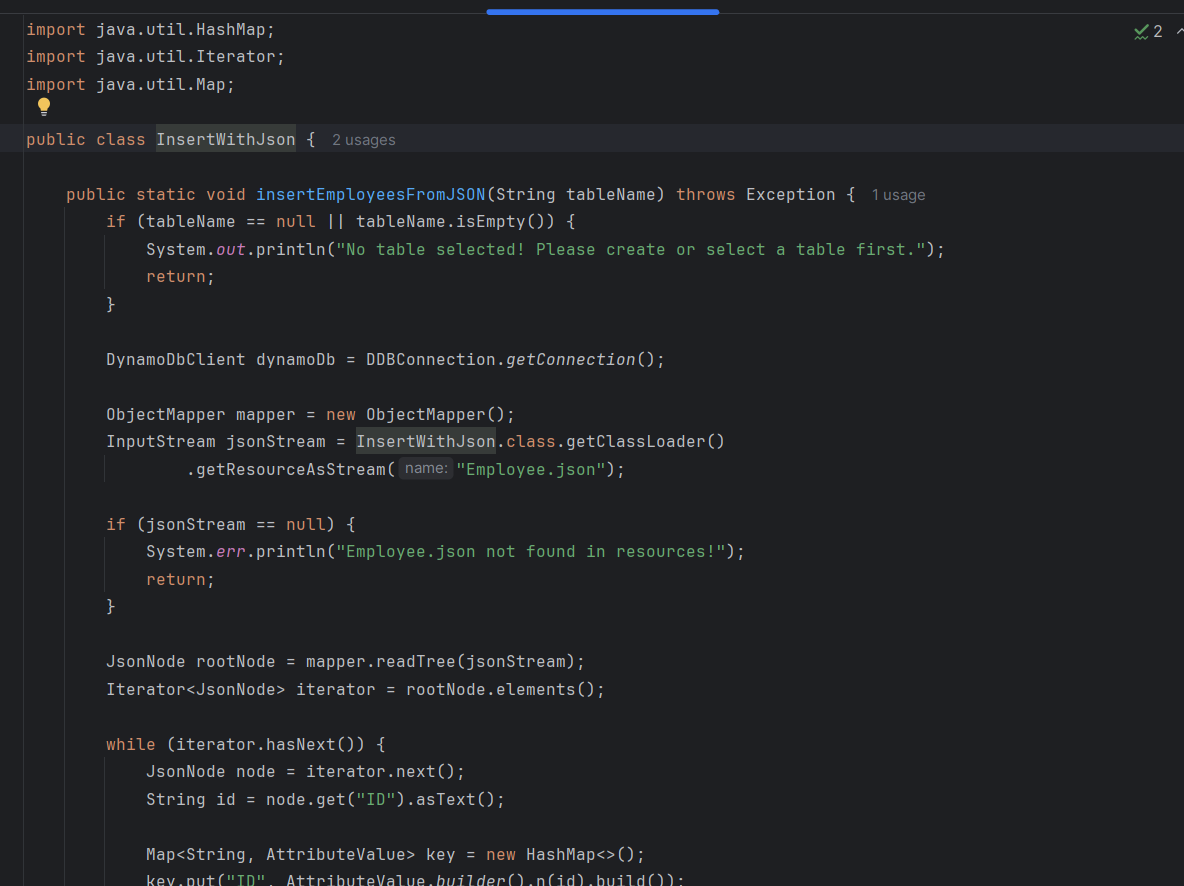
Name: Aravind Kasanagottu

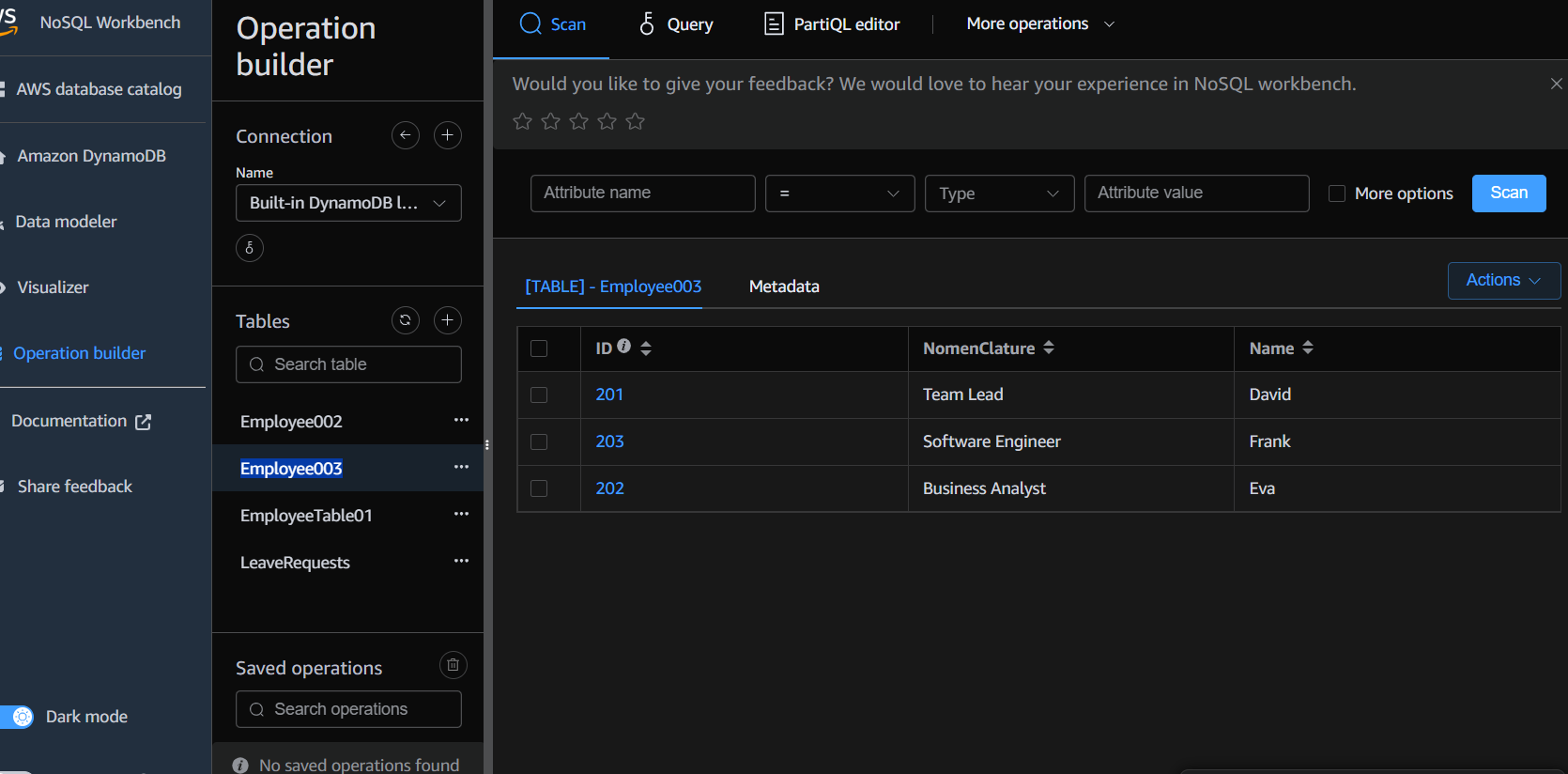
ID :mvsnarav

Task 01:  Create a java program to retrieve dynamodb ttable records and display..



Task 02 : Create a java program to create a table and insert 2 items .. the NOSQl workbench we should be able to see the table updated.





Perfect 👍 I’ll give you **clean Q&A format** (ignoring time & names).

**Task 03: What key features of DynamoDB?**

**Answer:**

* Fully managed NoSQL database (serverless).
* Provides **single-digit millisecond latency** at any scale.
* Supports **key-value and document data models**.
* Built-in **auto scaling** for throughput and storage.
* **Highly available** and **fault-tolerant** with replication across multiple AZs.
* **Event-driven programming** support via DynamoDB Streams.
* **Global tables** for multi-region replication.
* **DAX (DynamoDB Accelerator)** for in-memory caching.

**Task 04: Advantages and Disadvantages of DynamoDB**

**Advantages:**

* Fully managed → no server maintenance.
* High scalability and performance.
* Flexible schema (schema-less).
* Integration with AWS ecosystem.
* Auto-scaling and on-demand pricing.

**Disadvantages:**

* Expensive for very high workloads.
* Querying is limited (no complex joins/aggregates).
* Requires careful design of partition/sort keys.
* Learning curve compared to SQL databases.

**Task 05: Use Cases of DynamoDB**

* **E-commerce apps** (catalogs, orders, shopping carts).
* **Gaming leaderboards** and session tracking.
* **IoT data storage** with fast writes/reads.
* **Banking/Financial transactions** (atomic operations).
* **Content management systems**.
* **Mobile and web apps** requiring low-latency data access.

**Task 06: What is DynamoDBMapper?**

**Answer:**

* A **Java object persistence model** provided by AWS SDK.
* Maps Java classes (POJOs) to DynamoDB tables.
* Helps developers use **annotations (@DynamoDBTable, @DynamoDBHashKey, @DynamoDBAttribute)** instead of writing raw queries.
* Provides simple **save(), load(), query(), scan()** methods.

**Task 07: What are DynamoDB Projections?**

**Answer:**

* **Projections** define which attributes are copied to a secondary index.
* Types of Projections:
  + **KEYS\_ONLY** → Only partition & sort keys.
  + **INCLUDE** → Keys + selected attributes.
  + **ALL** → All attributes from the table.

**Task 08: How does DynamoDB prevent data loss?**

**Answer:**

* Stores data **across 3 Availability Zones** automatically.
* **Point-in-Time Recovery (PITR)** for backup/restore.
* **On-demand backups**.
* **Streams** help recover from accidental changes.
* **Strong consistency option** prevents stale reads.

**Task 09: What are In-place Atomic Updates? Does DynamoDB support them?**

**Answer:**

* **In-place atomic update** → Modify only part of an item (not rewrite whole item).
* DynamoDB **supports atomic updates** using:
  + **UpdateItem API**.
  + Atomic counters (ADD operation).
  + Conditional writes to ensure no conflicts.

**Task 10: What are Streams in DynamoDB?**

**Answer:**

* DynamoDB **Streams capture real-time changes** in a table (insert, update, delete).
* Each change is recorded as an **event**.
* Used for **replication, event-driven triggers (via Lambda), and analytics pipelines**.

**Task 11: What are DynamoDB Pricing Tiers?**

**Answer:**

* **On-Demand Capacity** → Pay per request (no need to specify throughput).
* **Provisioned Capacity** → Pre-allocate Read/Write Capacity Units (RCUs/WCUs).
* **Free Tier** → 25 GB storage + 25 RCU/WCU for 12 months.
* Pricing also for:
  + Storage (per GB).
  + Streams.
  + Global tables.
  + DAX caching.
  + Backup & restore.

**Task 12: Max Item Size in DynamoDB**

**Answer:**

* **400 KB per item** (including attribute names + values).

**Task 13: Max Global Secondary Indexes (GSI)**

**Answer:**

* You can create **up to 20 GSIs per table**.

**Task 14: What is DynamoDB Accelerator (DAX)?**

**Answer:**

* **In-memory caching layer** for DynamoDB.
* Provides **microsecond response times**.
* Reduces read latency for high-traffic apps.
* Fully managed and compatible with DynamoDB APIs.

**Task 15: What are DynamoDB Global Tables?**

**Answer:**

* **Multi-region, multi-master replication**.
* Automatically replicates tables across AWS regions.
* Useful for **disaster recovery and global apps**.
* All regions can accept **read/write operations**.

**Task 16: What are Indexes and Secondary Indexes?**

**Answer:**

* **Index** = An alternate way to query a table.
* **Local Secondary Index (LSI):**
  + Same partition key as table but different sort key.
  + Up to **5 LSIs per table**.
* **Global Secondary Index (GSI):**
  + Different partition and sort key from the table.
  + Up to **20 GSIs per table**.

**Task 17: What are Hot Keys and Hot Partitions?**

**Answer:**

* **Hot Key** → A single partition key that receives a **disproportionate number of requests**.
* **Hot Partition** → A partition storing multiple hot keys, causing throttling.
* Best Practice → Distribute workload evenly with randomization techniques.

**Task 18: Table-level vs Item-level Operations**

**Table-level operations:**

* CreateTable
* DescribeTable
* UpdateTable
* DeleteTable
* ListTables

**Item-level operations:**

* PutItem
* GetItem
* UpdateItem
* DeleteItem
* Query
* Scan