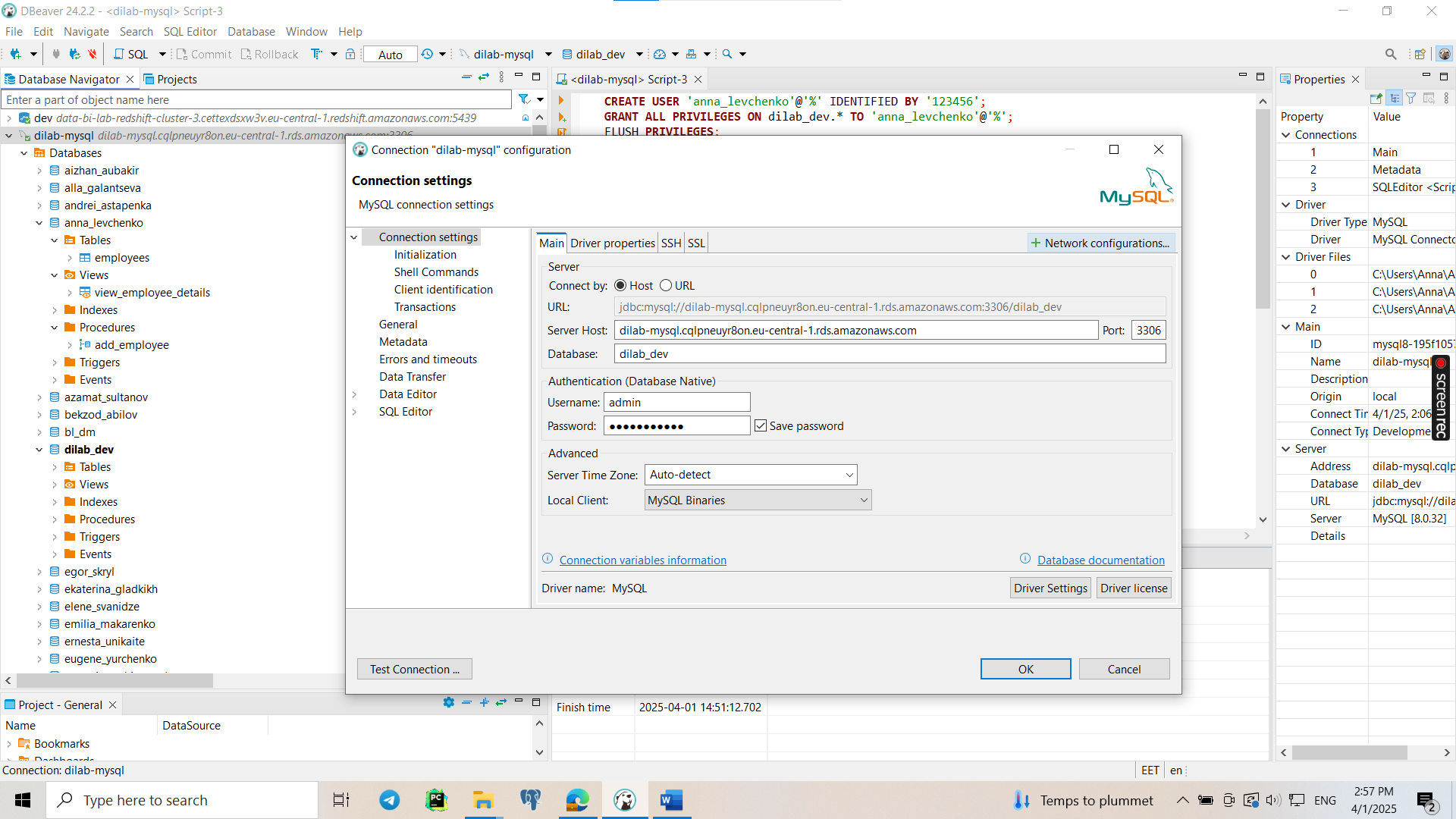
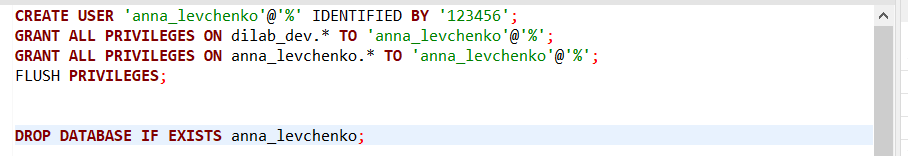
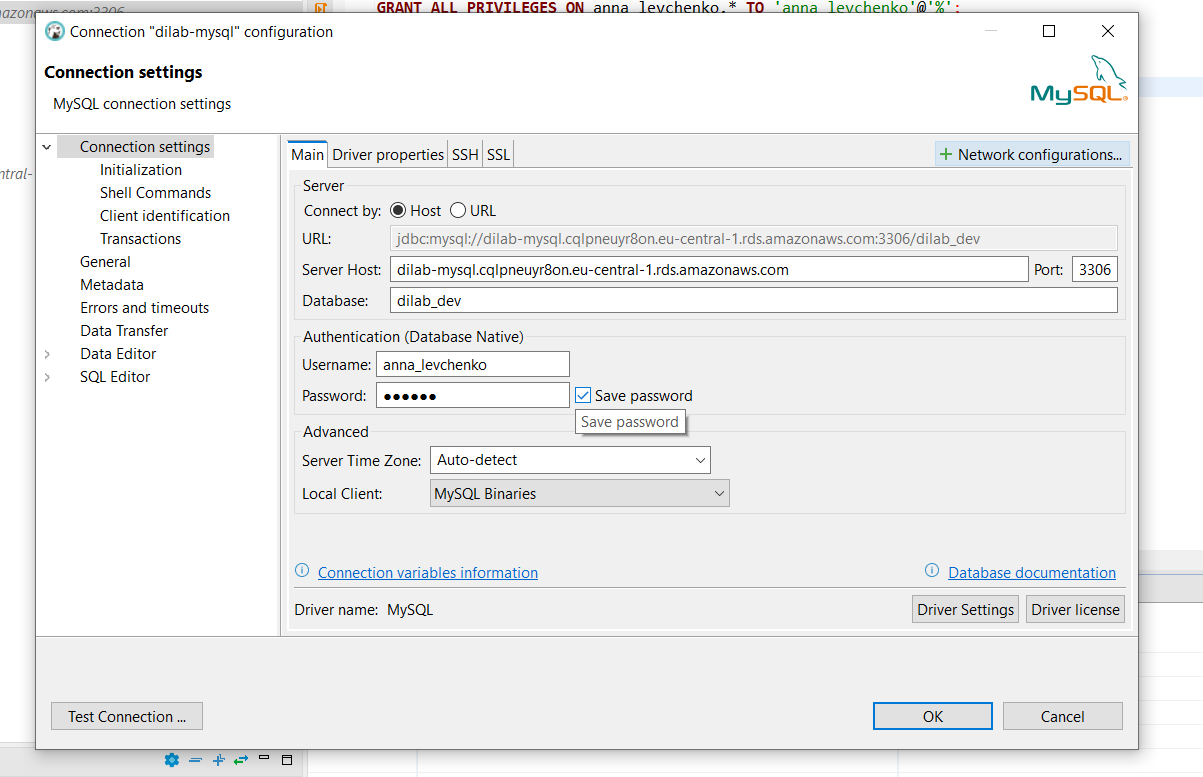
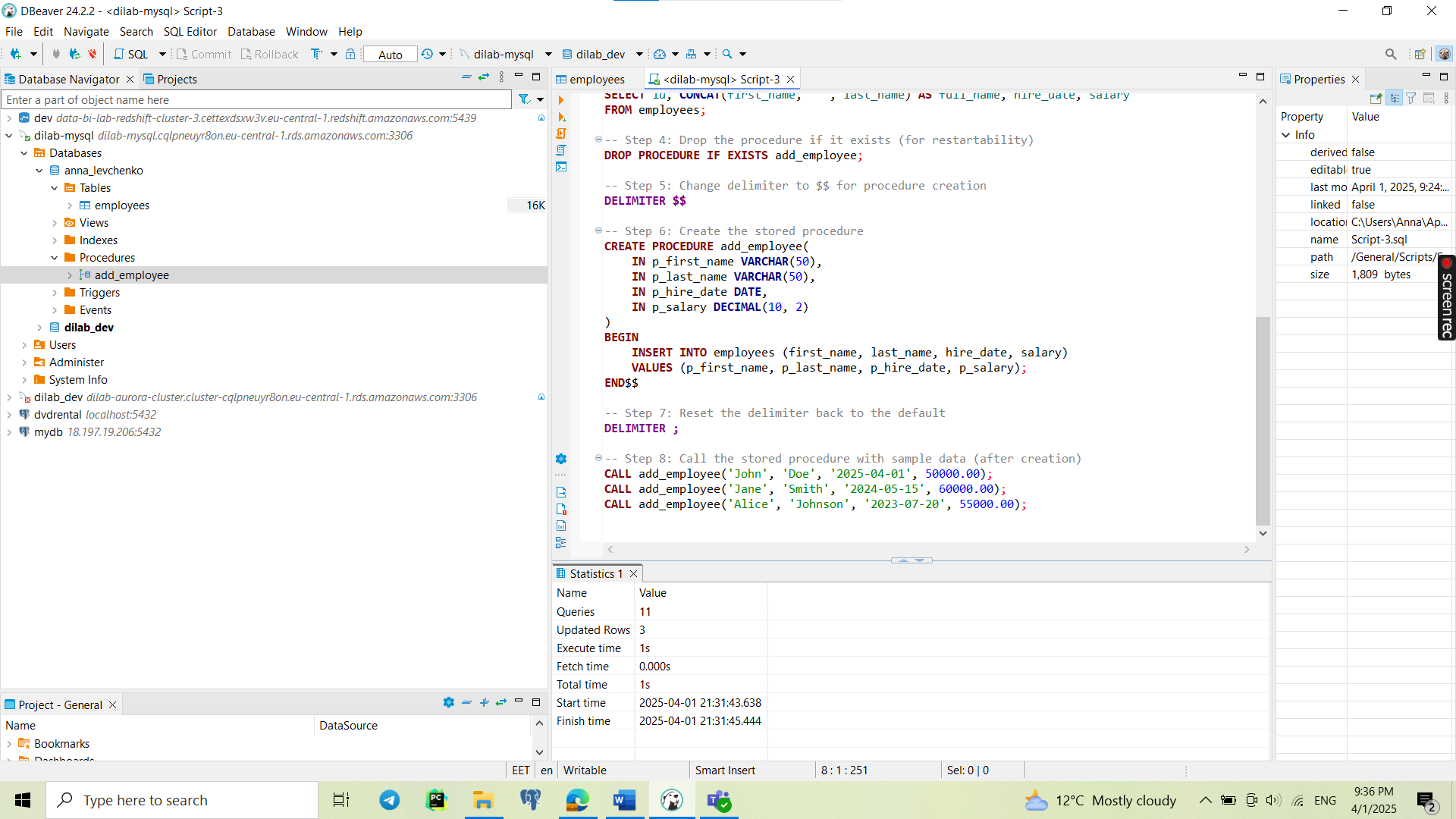
**Task 1**

****

****

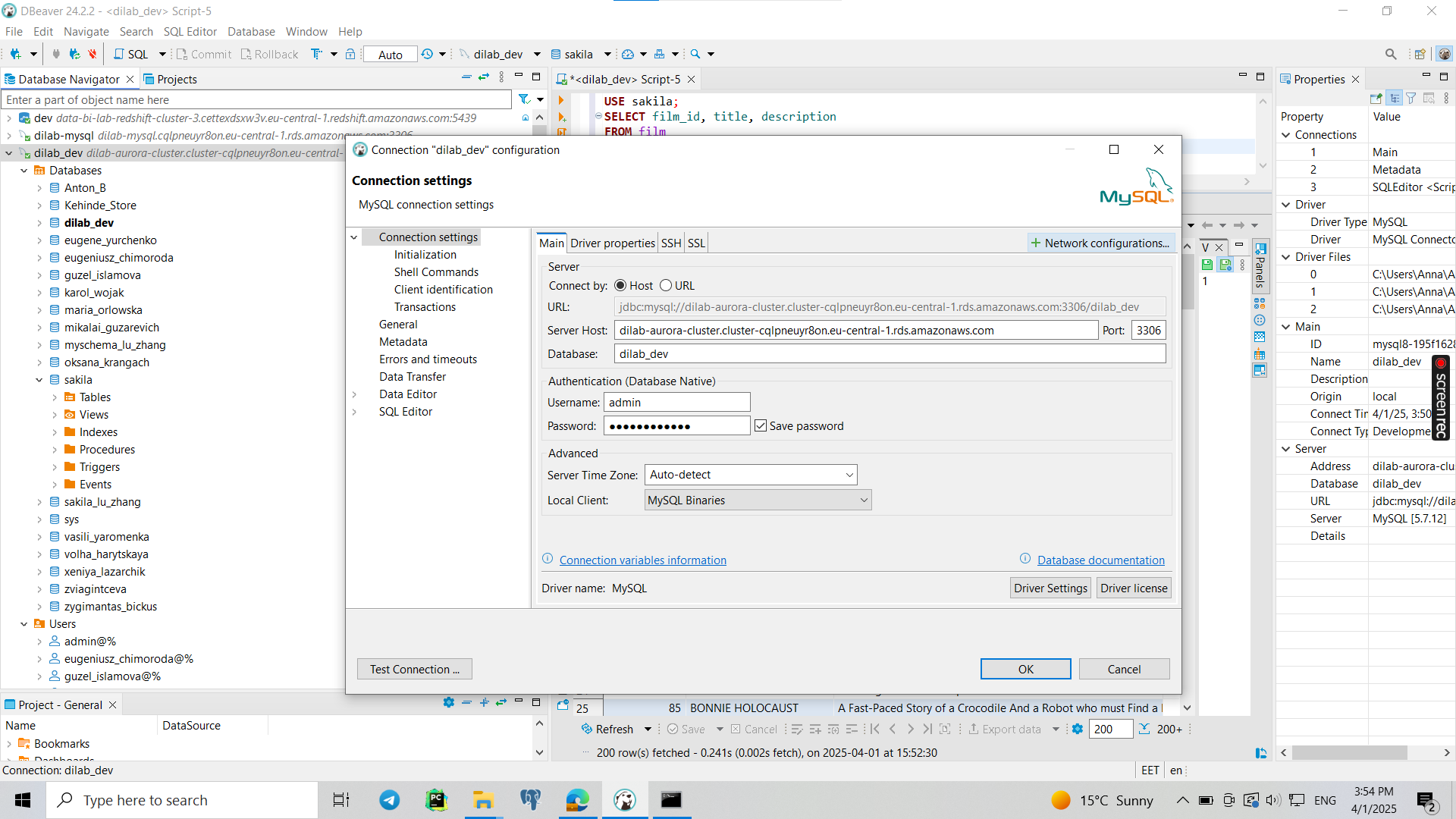
****

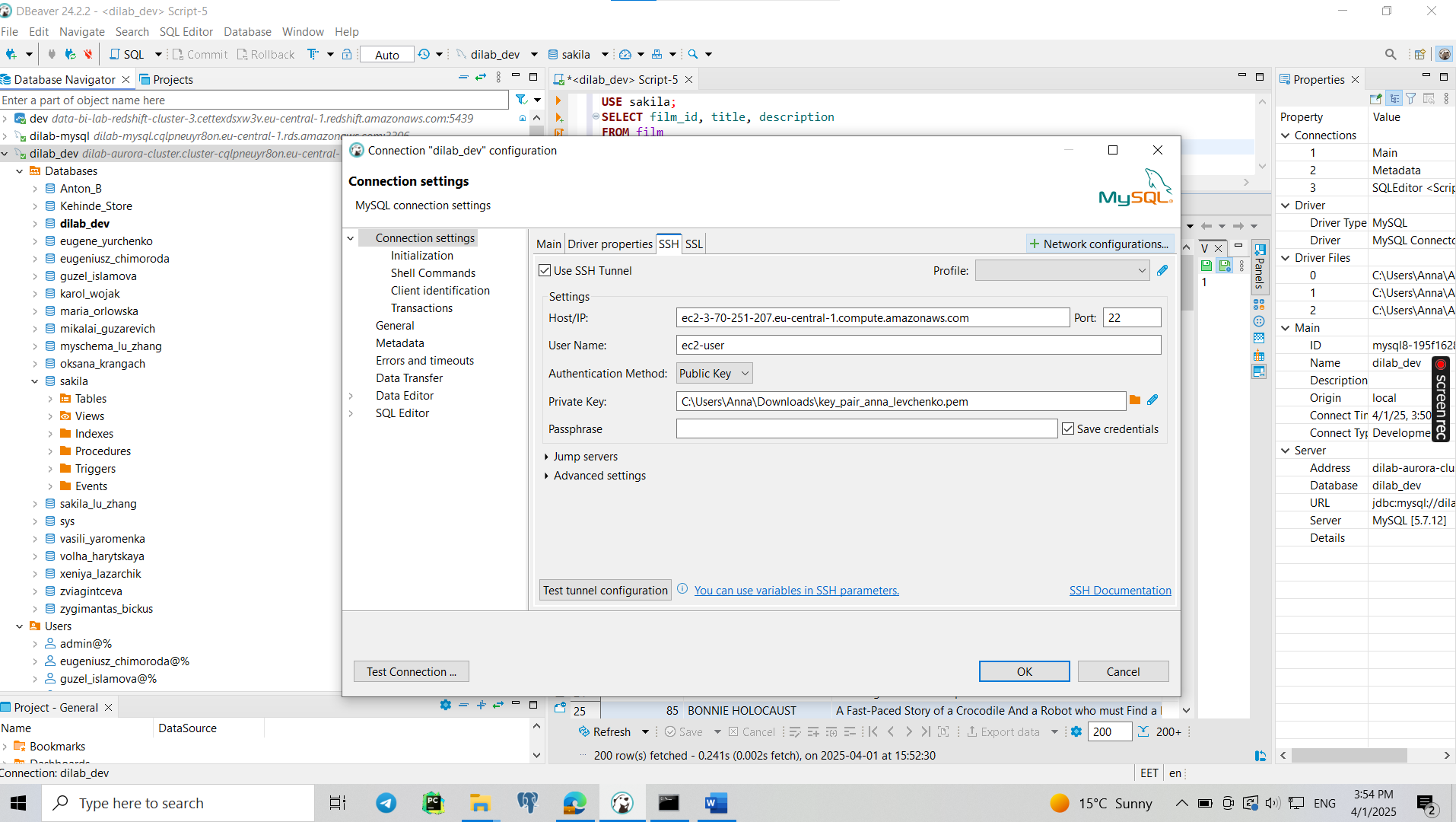
****

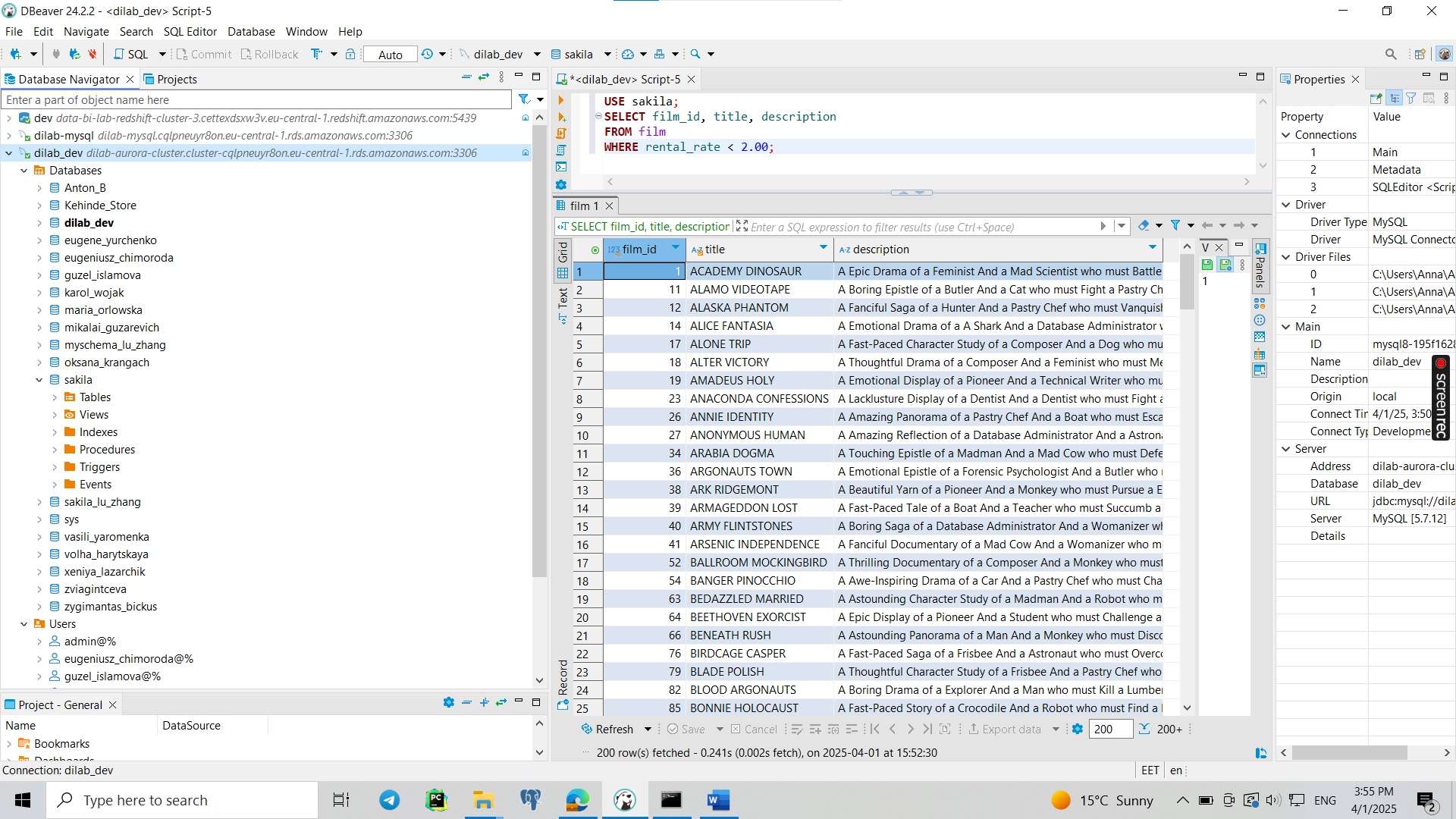
Script\_1 with query is attached

**Task 2**

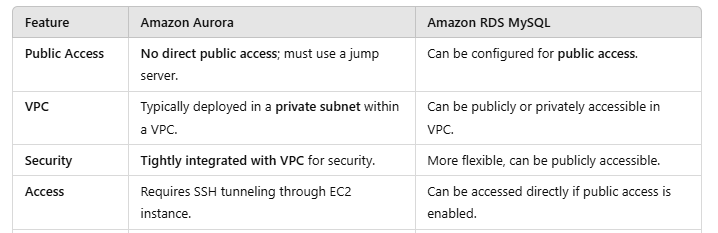
Query editor is not enabled, and this task can be skipped (written in Teams), so I did part with using an SQL Client (Dbever)



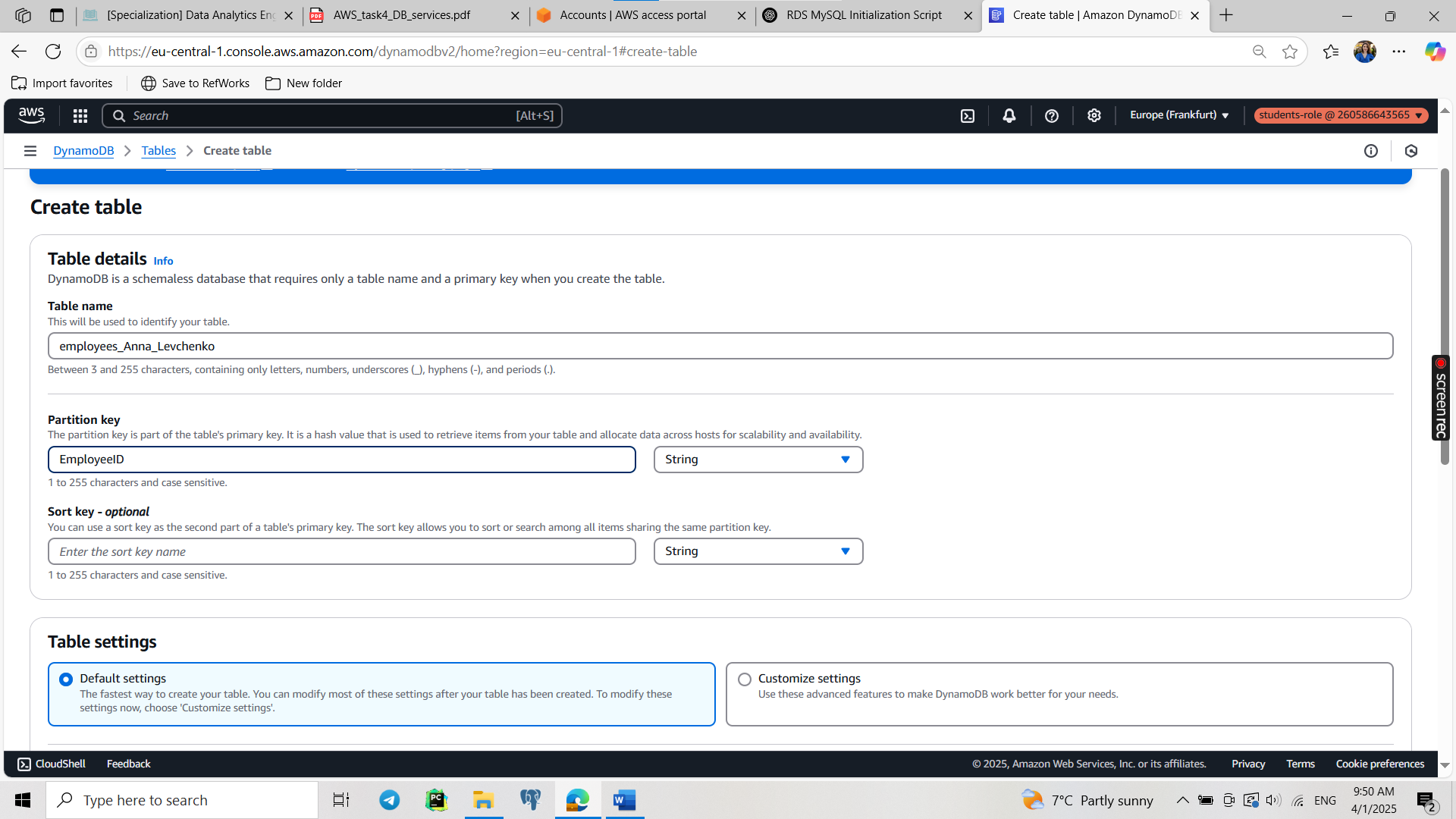


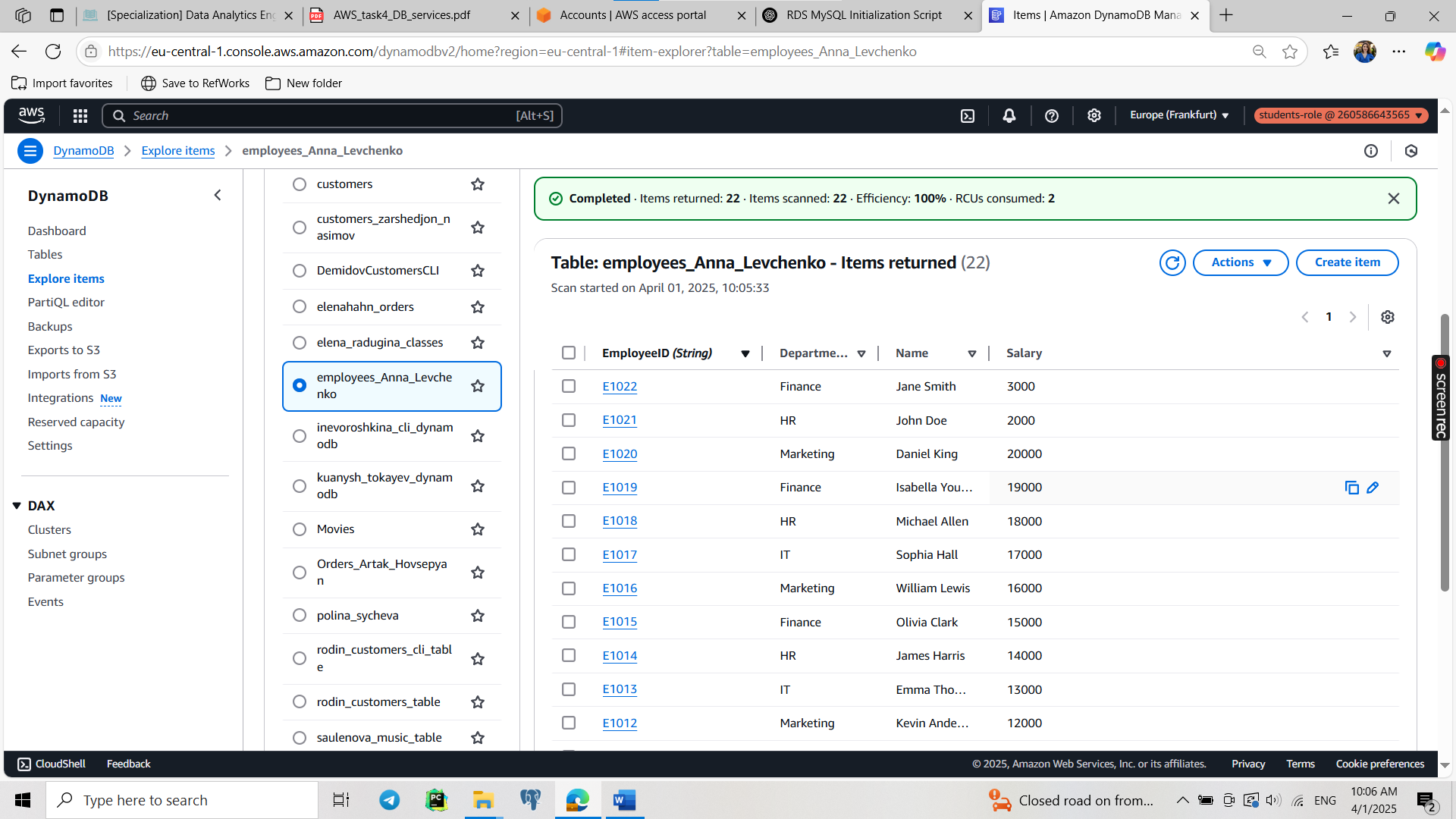


Connecting to **Amazon Aurora** from your **local PC** using a **jump server (EC2 instance)** and **SSH tunnel** is the only way to access Aurora in most configurations due to its **private networking model** and **security considerations**. Aurora is designed to be deployed within a VPC for **security and high availability**, and direct public access is not typically allowed. This contrasts with **Amazon RDS MySQL**, which can be configured for direct access via the public internet. The use of an EC2 jump server ensures that your database remains secure while allowing you to connect from a local machine.



**Task 3**

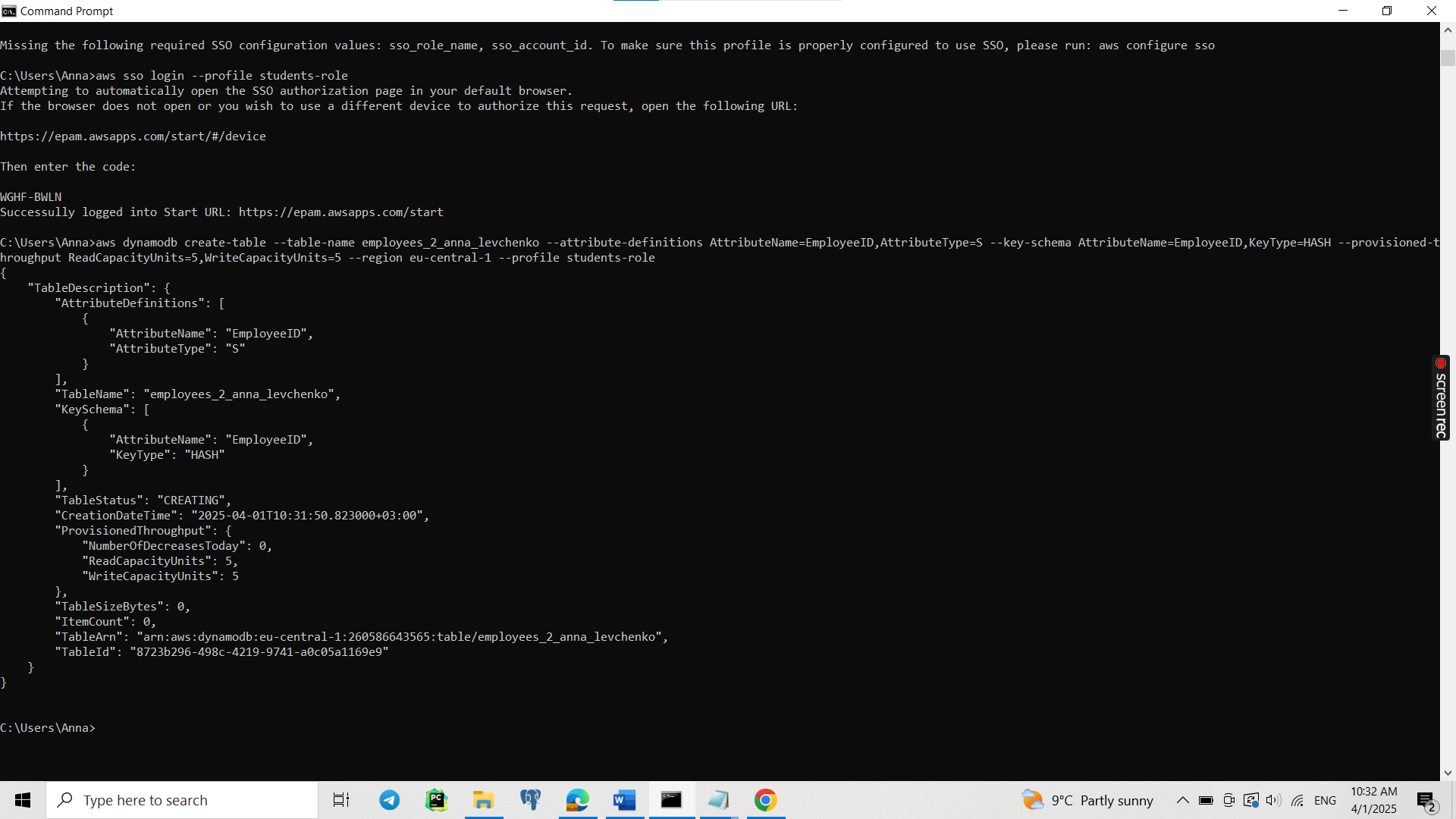


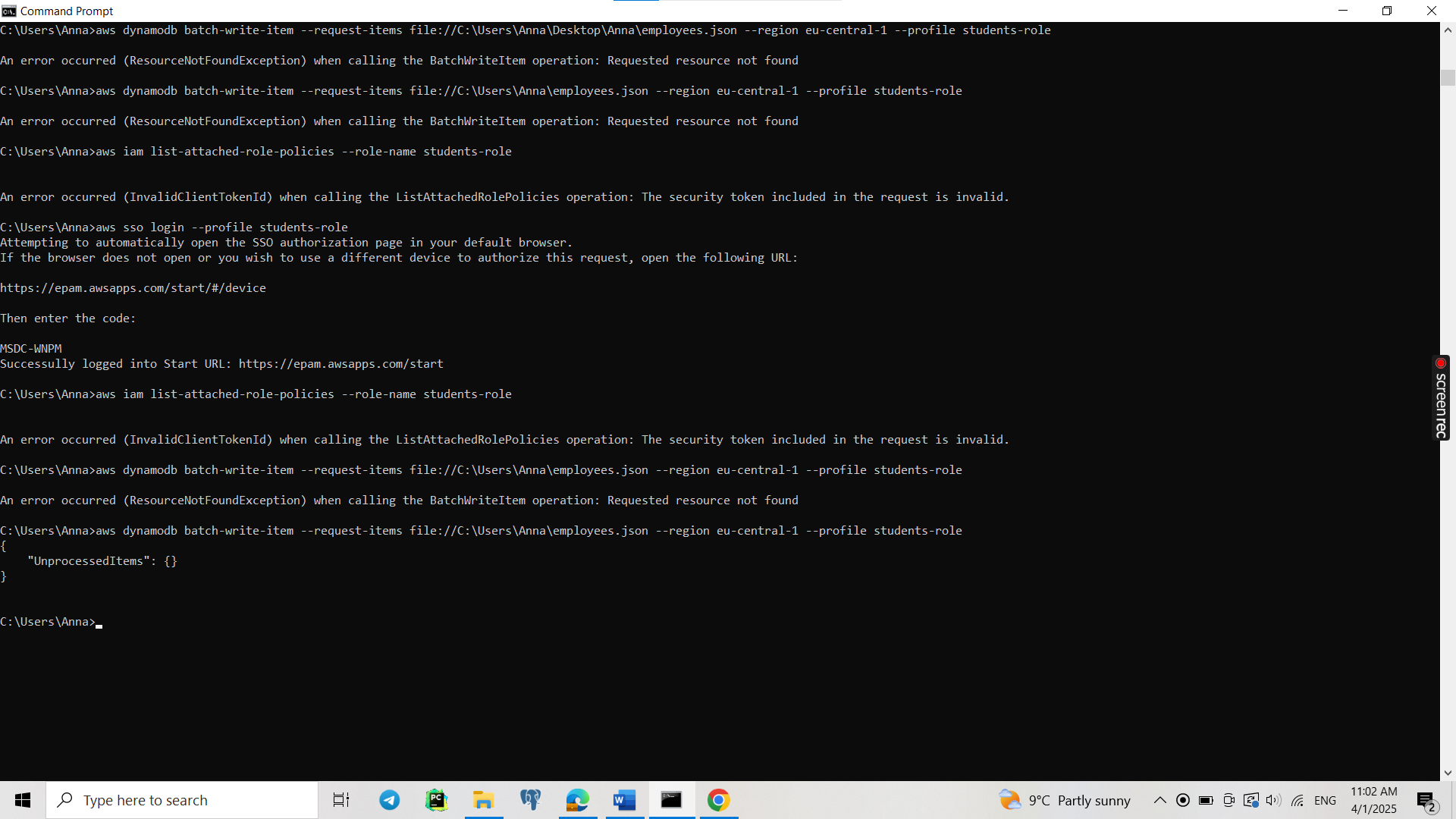


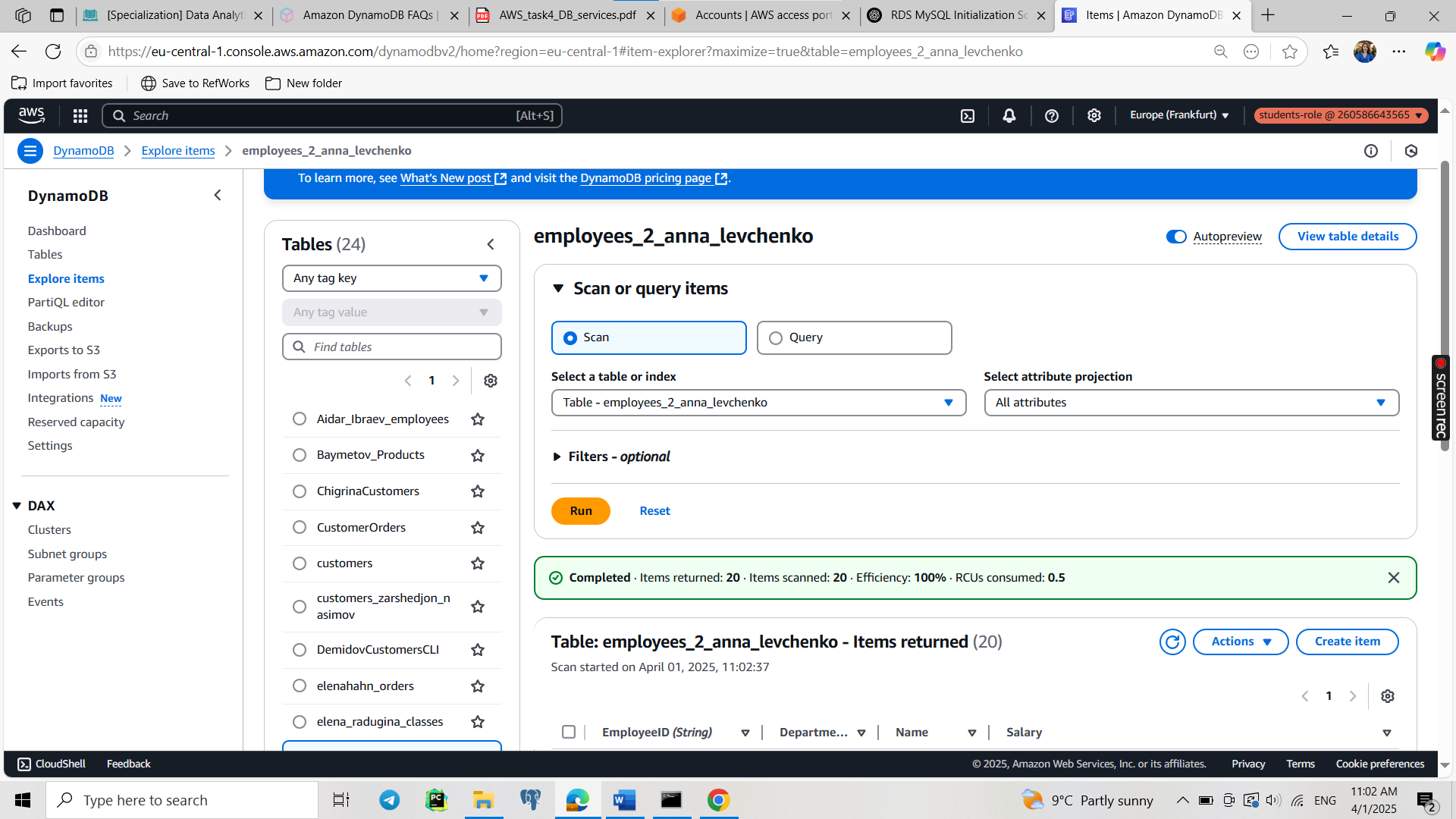
Choosing EmployeeID as the Partition Key:

* Uniqueness: Each employee has a unique ID.
* Efficient Queries: DynamoDB queries are optimized for primary keys, ensuring quick lookups.
* Scalability: A partition key evenly distributes records across storage nodes, avoiding performance bottlenecks.

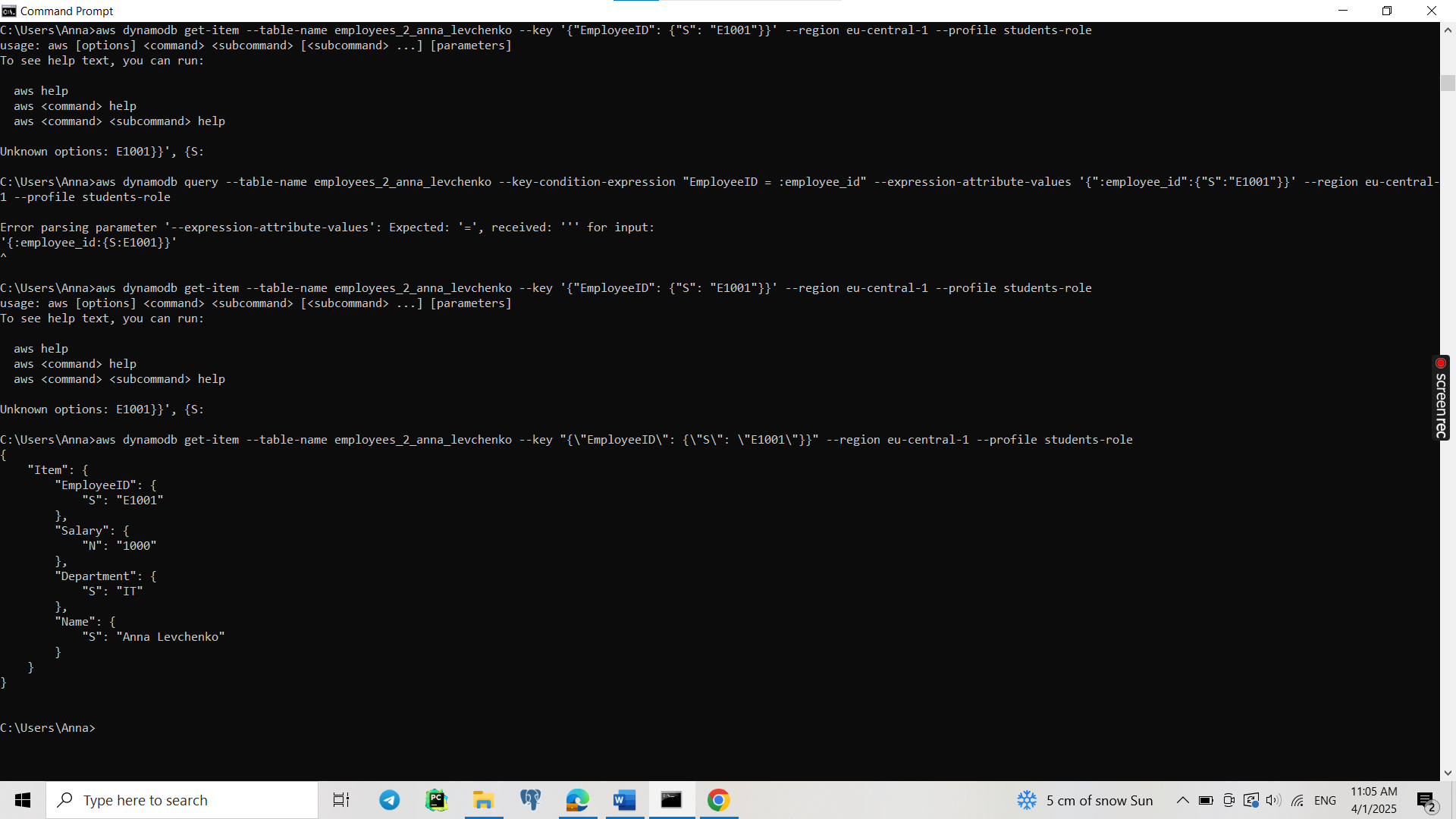
2. Using AWS CLI:



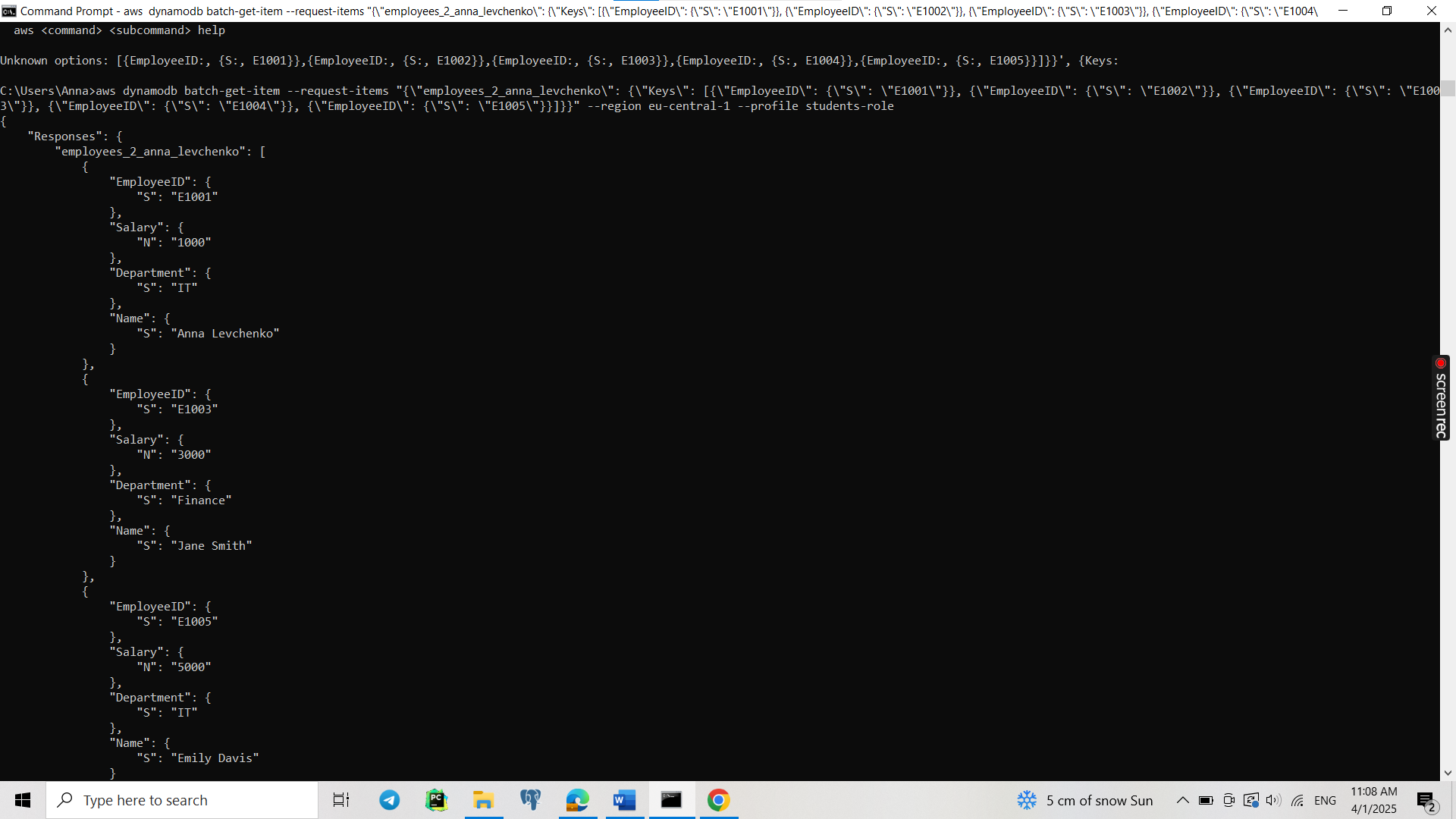




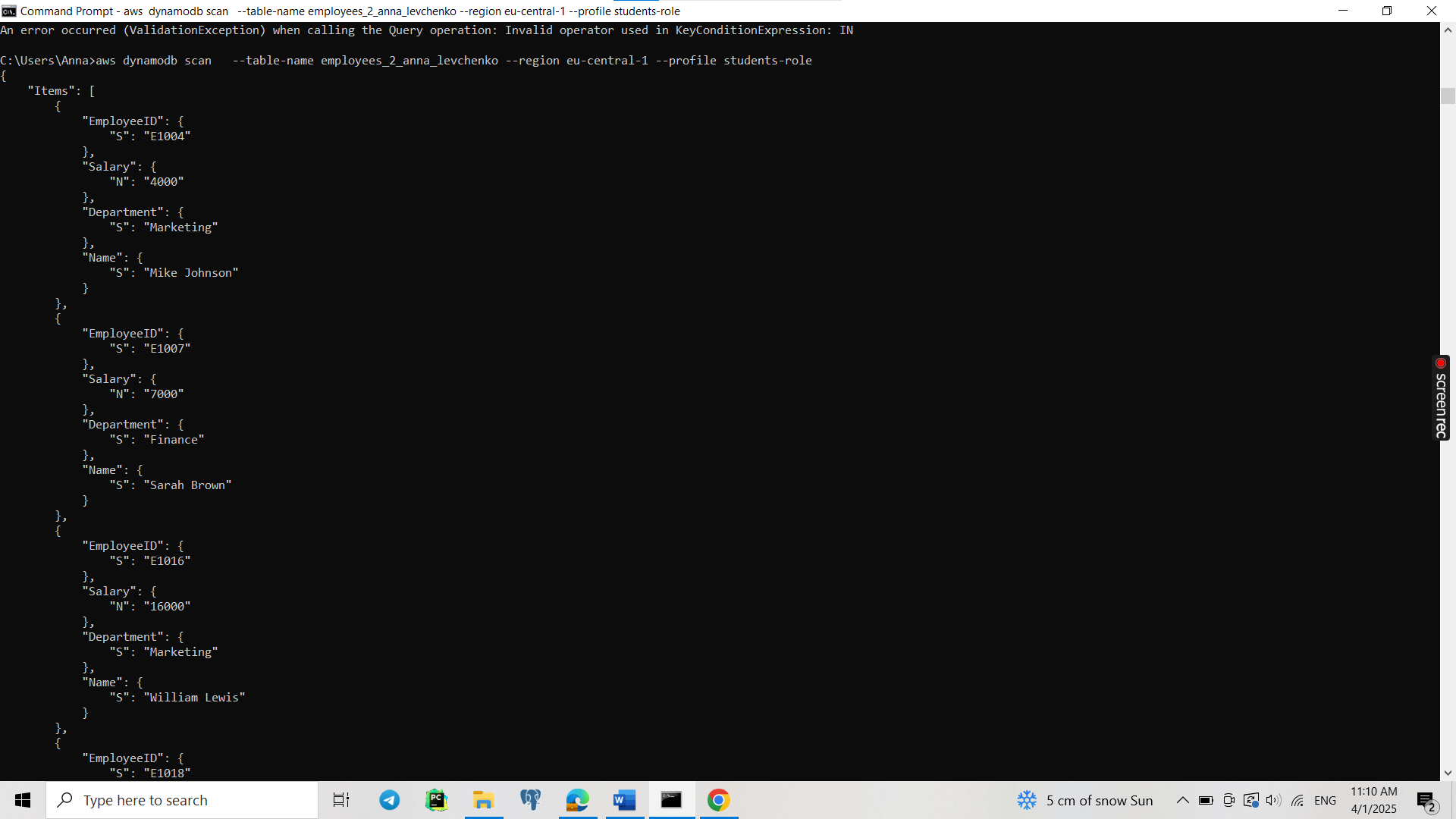
Select 1 employee:



Select 5 employees:



Select all:



Delete 2:

