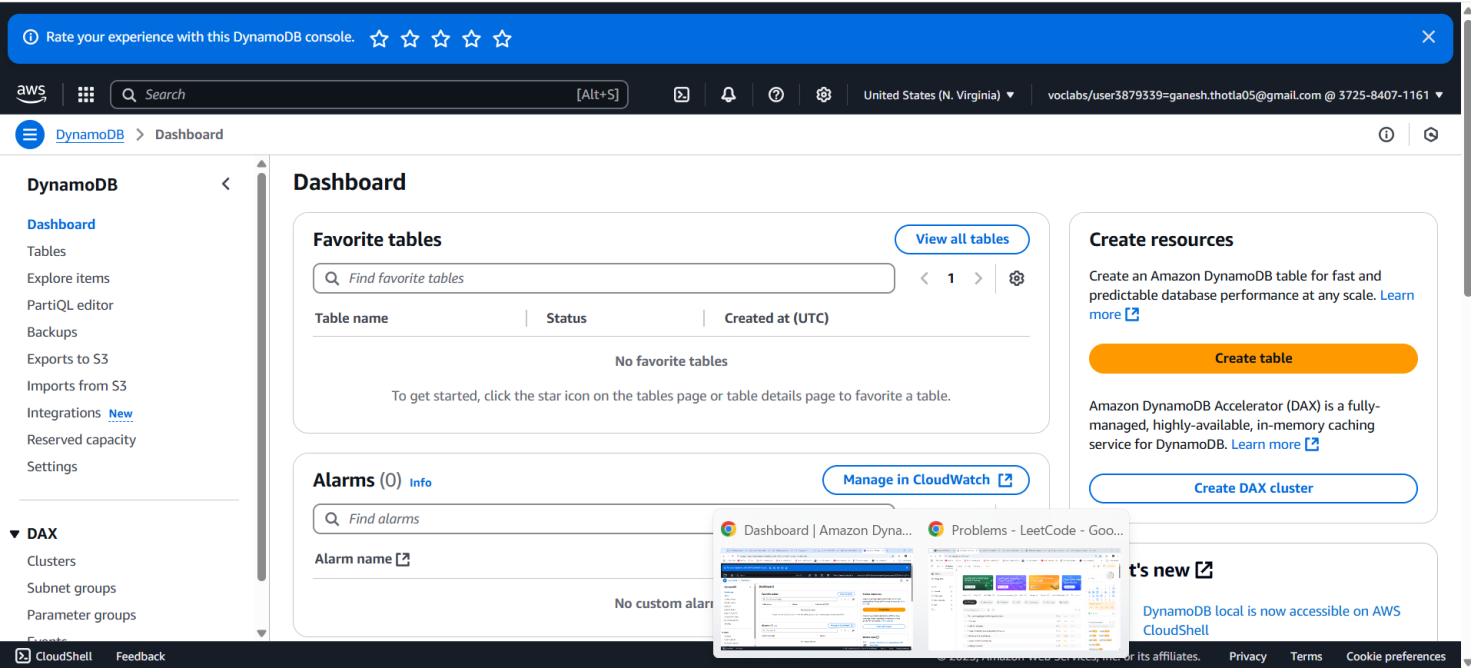
|  |
| --- |
| **Experiment-12:**  **Dynamo DB:**  **Create a Dynamo DB name as rollNo\_exp912:**  **a, Insert a table contains field as student roll No, student name, fav subject, rating**  **b, Insert any 15 values in table**  **c, Print alphabetical order by reference of student name**  **d, Display the count of whose rating is 5.**  **e, Display what is the average rating of each subject**  **f, Update student name**  **g, Delete if any duplicates available** |

**Step 1: Start Your AWS Learner Lab**

1. **Go to https://learn.qwiklabs.com or your assigned Learner Lab platform (e.g., AWS Academy).**
2. **Start the lab assigned to you /**
3. **Click Start Lab**

**Step 2: Open DynamoDB Service**

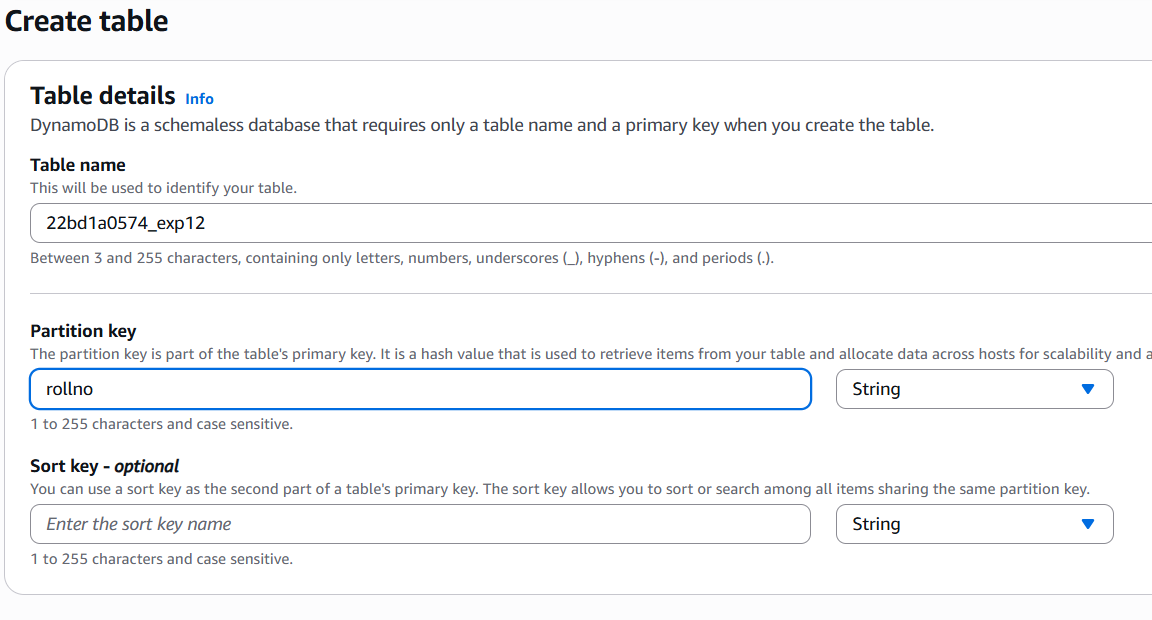
1. **In the AWS Console top search bar, type DynamoDB.**
2. **Click DynamoDB from the search results.**



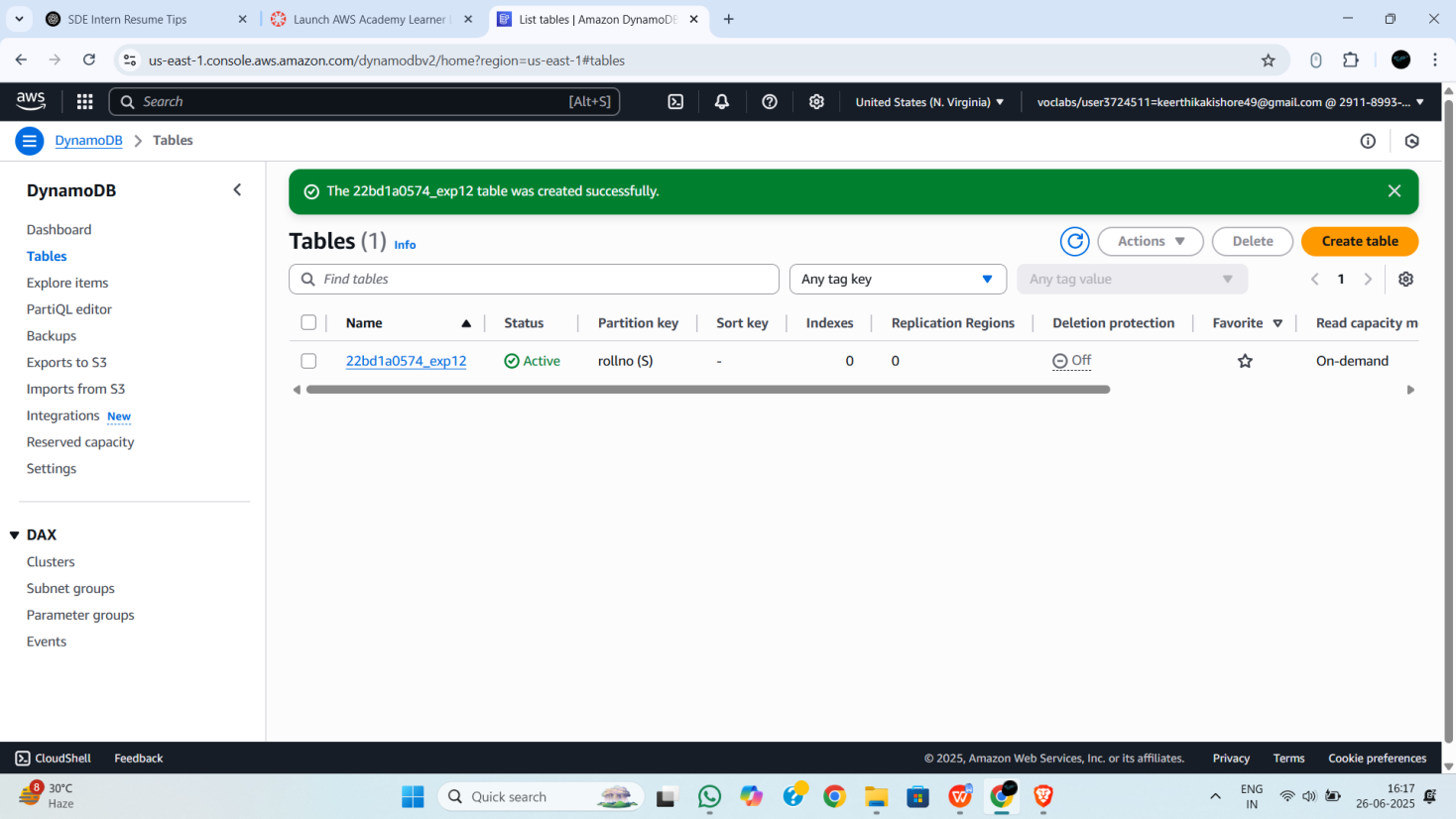
**Figure-1**

**Step 3: Create a Table – rollNo\_exp12**

1. **Click “Create table”**
2. **Fill in:**
   * **Table name: 22bd1a0574\_exp12**
   * **Partition key: rollNo (type: String)**
3. **Leave everything else default (No sort key needed).**
4. **Click Create Table  
   - Wait for table status to become “Active”**



**Figure-2**



**Figure-3**

**Step-4:You can copy-paste this script in AWS CloudShell or upload it, and it will:**

1. **Insert all 15 items**
2. **Print alphabetical order of students**
3. **Count how many students have rating = 5**
4. **Compute average rating per subject**
5. **Update a specific student’s name**
6. **Delete duplicates based on name**

|  |
| --- |
| **CODE:**  **import boto3**  **from collections import defaultdict**  **# Connect to DynamoDB**  **dynamodb = boto3.resource('dynamodb',region\_name=’us-east-1’)**  **table = dynamodb.Table('22bd1a0574\_exp12')**  **# Student data**  **students = [**  **{"rollNo": "001", "name": "Aarav", "subject": "Math", "rating": 5},**  **{"rollNo": "002", "name": "Riya", "subject": "Science", "rating": 4},**  **{"rollNo": "003", "name": "Dev", "subject": "Math", "rating": 3},**  **{"rollNo": "004", "name": "Anika", "subject": "English", "rating": 5},**  **{"rollNo": "005", "name": "Karan", "subject": "Science", "rating": 2},**  **{"rollNo": "006", "name": "Meera", "subject": "English", "rating": 5},**  **{"rollNo": "007", "name": "Ishaan", "subject": "Math", "rating": 4},**  **{"rollNo": "008", "name": "Tara", "subject": "Science", "rating": 5},**  **{"rollNo": "009", "name": "Aarav", "subject": "English", "rating": 3},**  **{"rollNo": "010", "name": "Sneha", "subject": "Math", "rating": 5},**  **{"rollNo": "011", "name": "Zoya", "subject": "Science", "rating": 5},**  **{"rollNo": "012", "name": "Yash", "subject": "Math", "rating": 4},**  **{"rollNo": "013", "name": "Lakshmi", "subject": "English", "rating": 5},**  **{"rollNo": "014", "name": "Aarav", "subject": "Science", "rating": 2},**  **{"rollNo": "015", "name": "Rohan", "subject": "Math", "rating": 1}**  **]**  **# Step b: Insert records**  **for student in students:**  **table.put\_item(Item=student)**  **print("\n✅ Inserted 15 student records.")**  **# Step c: Alphabetical order by student name**  **response = table.scan()**  **items = response['Items']**  **sorted\_items = sorted(items, key=lambda x: x['name'])**  **print("\n📚 Students in alphabetical order:")**  **for item in sorted\_items:**  **print(item['rollNo'], item['name'], item['subject'], item['rating'])**  **# Step d: Count students with rating = 5**  **rating\_5\_count = sum(1 for item in items if item['rating'] == 5)**  **print(f"\n⭐ Count of students with rating = 5: {rating\_5\_count}")**  **# Step e: Average rating per subject**  **subject\_ratings = defaultdict(list)**  **for item in items:**  **subject\_ratings[item['subject']].append(item['rating'])**  **print("\n📊 Average rating per subject:")**  **for subject, ratings in subject\_ratings.items():**  **avg\_rating = sum(ratings) / len(ratings)**  **print(f"{subject}: {avg\_rating:.2f}")**  **# Step f: Update student name (rollNo = 014) from Aarav to Arjun**  **table.update\_item(**  **Key={'rollNo': '014'},**  **UpdateExpression='SET #n = :newname',**  **ExpressionAttributeNames={'#n': 'name'},**  **ExpressionAttributeValues={':newname': 'Arjun'}**  **)**  **print("\n✏️ Updated student name for rollNo 014 to Arjun")**  **# Step g: Delete duplicate names (keep first only)**  **seen\_names = set()**  **for item in sorted\_items:**  **if item['name'] in seen\_names:**  **table.delete\_item(Key={'rollNo': item['rollNo']})**  **print(f"🗑️ Deleted duplicate: {item['rollNo']} - {item['name']}")**  **else:**  **seen\_names.add(item['name'])**  **print("\n✅ Cleanup complete. Duplicates removed.")** |
|  |

**Figure-4**

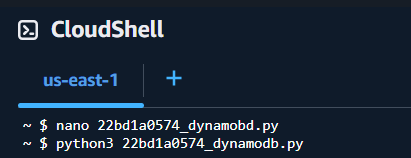
**Step-5: To Run This in AWS CloudShell**

1. **Open CloudShell (top-right icon in AWS Console).**
2. **Paste the script into a file:**

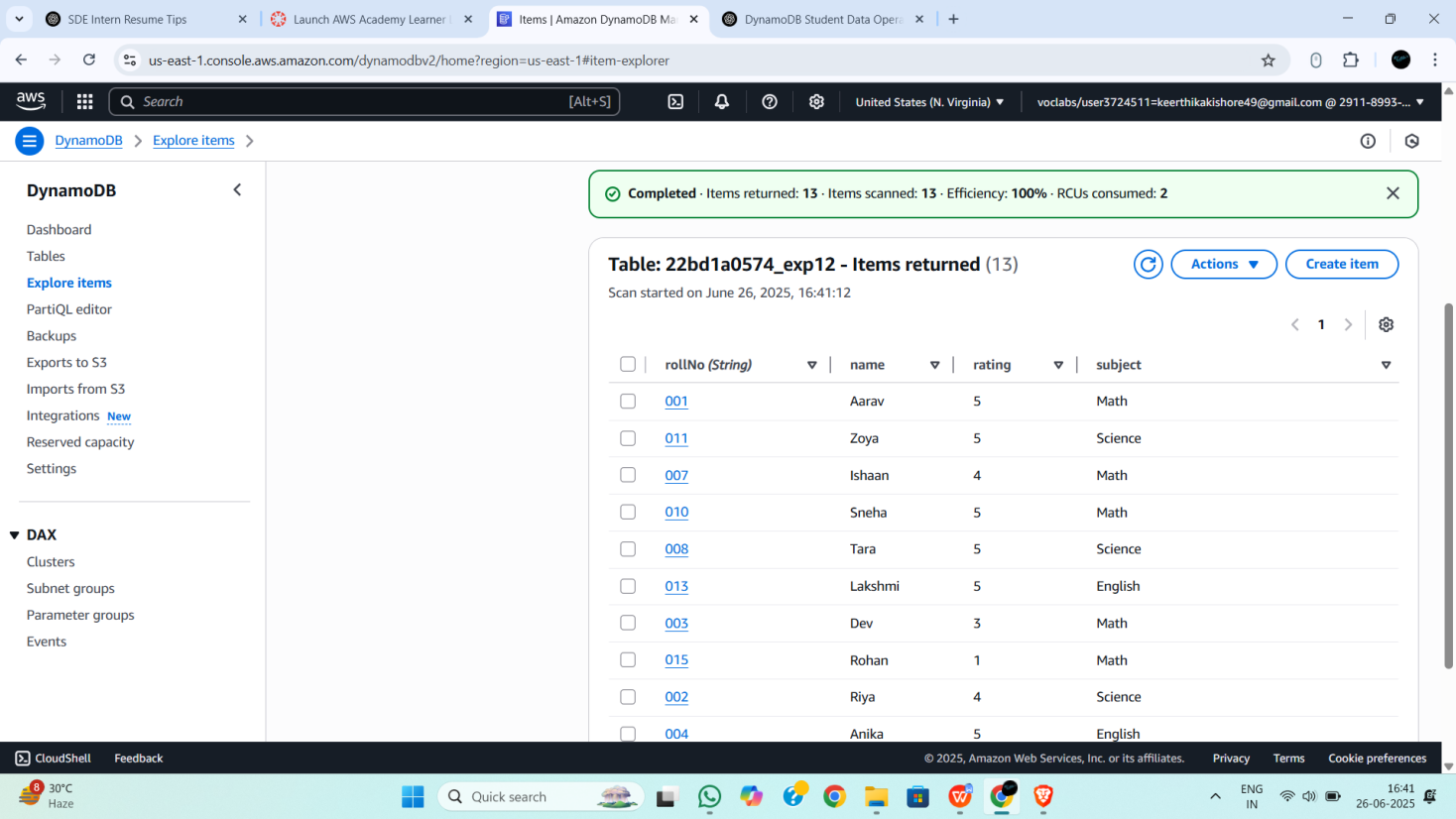
**nano dynamodb\_exp9.py**

1. **Paste the code, press Ctrl+O then Enter to save, then Ctrl+X to exit.**
2. **Run:**

**python3 dynamodb\_exp9.py**







**Figure-5**