

**Міністерство освіти і науки України
Національний технічний університет України
"Київський політехнічний інститут імені Ігоря Сікорського"
Фізико-технічний інститут**

«Харні технології»

Лабораторна робота №3

Виконала:
студентка групи ФБ-95
Гурджия Валерія Вахтангівна

ЗАВДАННЯ

- Спроекувати структуру даних (таблицю)
 - Вивчити способи роботи з даними засобами DynamoDB
 - Виконати завдання відповідно до варіанту (в GUI, засобами AWS CLI та Python – останнє за бажанням)
-
- Для даних, отриманих в роботі №2 спроекувати таблицю DynamoDB (засобами NOSQL Workbench, AWS CLI та AWS Management Console)
 - Відпрацювати додавання, видалення та пошук даних засобами AWS Management Console та AWS CLI
-
- За додаткові бали імплементувати ці операції засобами Python SDK (приклади для boto3 доступні за посиланням <https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/GettingStarted.Python.html>)

Спроектуємо таблицю DynamoDB за допомогою **NOSQL Workbench**

Ввели данні користувача, щоб можна було завантажити таблицю у Amazon DynamoDB

Remote

DynamoDB local

i

To use DynamoDB, you must have an AWS account. You can use AWS Identity and Access Management (IAM) roles and temporary security credentials to grant users in your AWS account access to your DynamoDB resources from NoSQL Workbench.

[Setting up DynamoDB](#)

* Connection name

lera

* Default AWS Region

us-west-1

* Access key ID

AKIA2QDR7ZVUCPLEFVOP

* Secret access key

.....

Session token

AWS session token

IAM role ARN

IAM role ARN

Persist connection

☐

Create data model for Amazon DynamoDB

* Name

Data

Author

Enter author name

Description

Describe this data model

Cancel

Create

Додамо всі атрибути

Table name: Currency

Primary key attributes:

- Partition key: CurrencyCode (String)
- Sort key: CurrencyCodeL (String)

Other attributes:

- Attribute name: StartDate (String)
- Attribute name: TimeSign (String)
- Attribute name: Units (Number)
- Attribute name: Amount (Number)

Завантажимо дані з файлу data.csv, який ми отримали у попередній роботі, натиснувши на “Visualize data model”, “Update” та Import CSV file

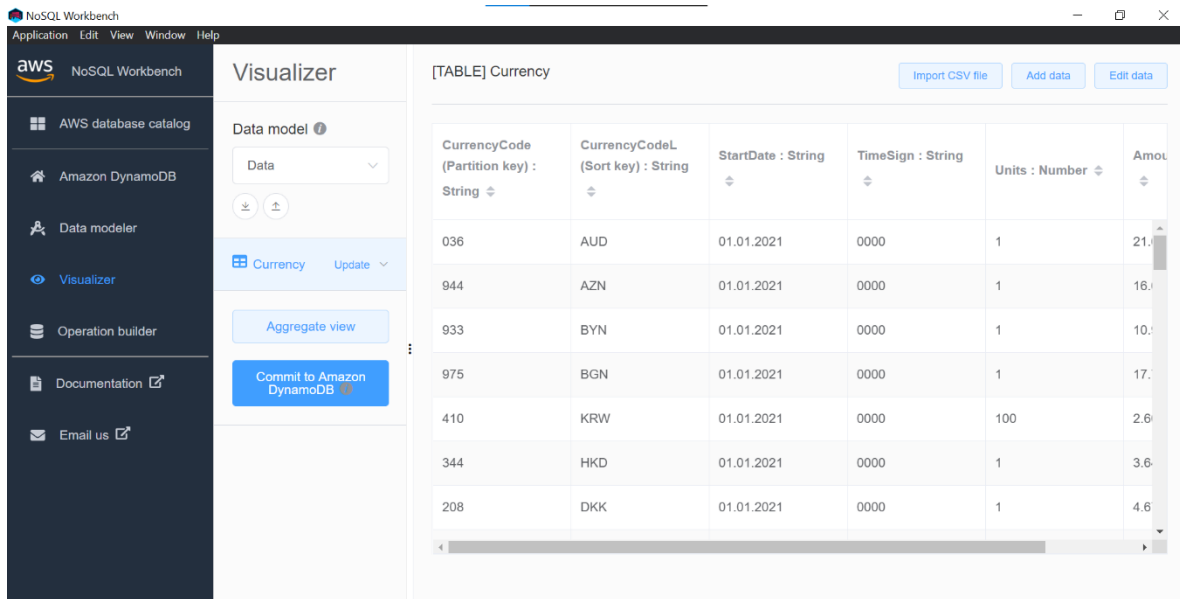
Як бачимо, дані успішно завантажилися

[TABLE] Currency

| CurrencyCode (Partition key) : String | CurrencyCodeL (Sort key) : String | StartDate : String | TimeSign : String | Units : Number |
|---------------------------------------|-----------------------------------|--------------------|-------------------|----------------|
| 036 | AUD | 01.01.2021 | 0000 | 1 |
| 944 | AZN | 01.01.2021 | 0000 | 1 |
| 933 | BYN | 01.01.2021 | 0000 | 1 |
| 975 | BGN | 01.01.2021 | 0000 | 1 |
| 410 | KRW | 01.01.2021 | 0000 | 100 |

Success
The CSV data has been imported.

Зробили коміт до DynamoDB



The screenshot shows the AWS NoSQL Workbench Visualizer interface. On the left is a sidebar with navigation options: AWS database catalog, Amazon DynamoDB, Data modeler, Visualizer (selected), Operation builder, Documentation, and Email us. The main area is titled 'Visualizer' and shows a data model for a table named '[TABLE] Currency'. The data model includes columns: CurrencyCode (Partition key) : String, CurrencyCodeL (Sort key) : String, StartDate : String, TimeSign : String, Units : Number, and Amount. Below the data model, there are buttons for 'Import CSV file', 'Add data', 'Edit data', 'Aggregate view', and 'Commit to Amazon DynamoDB'. The table data is displayed in a grid:

| CurrencyCode (Partition key) : String | CurrencyCodeL (Sort key) : String | StartDate : String | TimeSign : String | Units : Number | Amount |
|---------------------------------------|-----------------------------------|--------------------|-------------------|----------------|--------|
| 036 | AUD | 01.01.2021 | 0000 | 1 | 21.0 |
| 944 | AZN | 01.01.2021 | 0000 | 1 | 16.0 |
| 933 | BYN | 01.01.2021 | 0000 | 1 | 10.0 |
| 975 | BGN | 01.01.2021 | 0000 | 1 | 17.0 |
| 410 | KRW | 01.01.2021 | 0000 | 100 | 2.6 |
| 344 | HKD | 01.01.2021 | 0000 | 1 | 3.6 |
| 208 | DKK | 01.01.2021 | 0000 | 1 | 4.6 |

Commit to Amazon DynamoDB

On this page, you create server-side resources such as tables and global secondary indexes for the selected data model. If a table that uses auto scaling is being committed and a failure occurs while enabling auto scaling, table creation and item insertion will still be attempted and can be completed successfully. If the table has been created but auto scaling has not been enabled, use the DynamoDB console to enable auto scaling manually for the created table. If a table with auto scaling is being committed to DynamoDB local, auto scaling will not be enabled because it is not supported.

< [Use saved connections](#) Add a new remote connection Add a new DynamoDB I >

Saved connections

lera

Cancel

Reset

Commit

Таблиця з'явилася у списку

DynamoDB > Tables

Tables (1) Info

↻

Actions ▾

Delete

Create table

🔍 Find tables by table name

Any table tag ▾

< 1 > ⚙️

| <input type="checkbox"/> | Name ▲ | Status | Partition key | Sort key | Indexes | Read capacity mode | Write capacity mode |
|--------------------------|----------|--------|------------------|-------------------|---------|-----------------------------------|-----------------------------------|
| <input type="checkbox"/> | Currency | Active | CurrencyCode (S) | CurrencyCodeL (S) | 0 | Provisioned with auto scaling (1) | Provisioned with auto scaling (1) |

Tables (1) ×

Any table tag ▾

🔍 Find tables by table name

< 1 > ⚙️

● Currency

Currency

Autopreview

↻

Actions ▾

Create item

Update table settings

▶ Scan/Query items

Expand to query or scan items.

Items returned (50)

< 1 ... > ⚙️ 🗖️

| <input type="checkbox"/> | Currency... ▾ | Currency... ▾ | Amount ▾ | StartDate ▾ | TimeSign ▾ |
|--------------------------|---------------|---------------|----------|-------------|------------|
| <input type="checkbox"/> | 972 | TJS | 2.5022 | 01.01.2021 | 0000 |
| <input type="checkbox"/> | 949 | TRY | 3.8448 | 01.01.2021 | 0000 |
| <input type="checkbox"/> | 348 | HUF | 9.5181 | 01.01.2021 | 0000 |
| <input type="checkbox"/> | 756 | CHF | 32.0156 | 01.01.2021 | 0000 |

Створимо таблицю через AWS Management Console

Create table

Table details [Info](#)

DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

Table name

This will be used to identify your table.

Files

Between 3 and 255 characters, containing only letters, numbers, underscores (`_`), hyphens (`-`), and periods (`.`).

Partition key

The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.

FileOwner

String ▼

1 to 255 characters and case sensitive.

Sort key - *optional*

You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.

FileName

String ▼

1 to 255 characters and case sensitive.

✔ The Files table was created successfully.
✕

DynamoDB > Tables

Tables (3) [Info](#)

🔄

Actions ▾

Delete

Create table

🔍 Find tables by table name

Any table tag ▾

< 1 >

⚙️

| | Name ▲ | Status | Partition key | Sort key | Inde... | Read capacity mode | Write capacity mode | Size |
|--------------------------|----------|----------|------------------|-------------------|---------|-----------------------------------|-----------------------------------|---------------|
| <input type="checkbox"/> | Currency | ✔ Active | CurrencyCode (S) | CurrencyCodeL (S) | 0 | Provisioned with auto scaling (1) | Provisioned with auto scaling (1) | 4.8 kilobytes |
| <input type="checkbox"/> | Files | ✔ Active | FileOwner (S) | FileName (S) | 0 | Provisioned with auto scaling (5) | Provisioned with auto scaling (5) | 0 bytes |

Додамо дані натиснувши на Explore items, а потім на Create Item

DynamoDB > Items: Files > Item editor

Item editor

FormJSON

Attributes

Add new attribute ▼

| Attribute name | Value | Type |
|--|--|---------------------------|
| FileOwner - Partition key | <input type="text" value="admin"/> <div>New</div> | String |
| FileName - Sort key | <input type="text" value="important_file"/> <div>New</div> | String |
| <input type="text" value="FileExtension"/> | <input type="text" value=".txt"/> | String <div>Remove</div> |
| <input type="text" value="IsFileEmpty"/> | <div><input type="radio"/> True <input checked="" type="radio"/> False</div> | Boolean <div>Remove</div> |

CancelSave changes

Додамо ще дані через json

Create item Form JSON

Attributes View DynamoDB JSON

```
1 {
2   "FileOwner": {
3     "S": "Iera"
4   },
5   "FileName": {
6     "S": "labka"
7   },
8   "FileExtension": {
9     "S": ".docx"
10  },
11  "DateOfCreating": {
12    "S": "11.08.2022"
13  }
14 }
```

Cancel Create item

Items returned (2) < 1 > ⚙️

| <input type="checkbox"/> | FileOwner ▾ | FileName ▾ | DateOfC... ▾ | FileExt... ▾ | IsFileEmpty ▾ |
|--------------------------|-------------|---------------|--------------|--------------|---------------|
| <input type="checkbox"/> | lera | labka | 11.08.2022 | .docx | |
| <input type="checkbox"/> | admin | impotrant_... | | .txt | false |

Видалимо строку з partition key 'lera'

Files Autopreview ↻ Actions ▴ Create item Update table settings

► Scan/Query items
Expand to query or scan items.

Items returned (2) < 1 > ⚙️

| <input type="checkbox"/> | FileOwner ▾ | FileName ▾ | DateOfC... ▾ | FileExt... ▾ | IsFileEmpty ▾ |
|-------------------------------------|-------------|---------------|--------------|--------------|---------------|
| <input checked="" type="checkbox"/> | lera | labka | 11.08.2022 | .docx | |
| <input type="checkbox"/> | admin | impotrant_... | | .txt | false |

Edit item

Duplicate item

Delete items

Download selected items to CSV

Download results to CSV

Selected items have been deleted successfully.

DynamoDB > Items > Files

Tables (3)

Any table tag

Find tables by table name

< 1 >

Currency

Files

Students

Files

Autopreview

Actions

Create item

Update table settings

Scan/Query items

Expand to query or scan items.

Items returned (1)

< 1 >

| | FileOwner | FileName | FileExt... | IsFileEmpty |
|--------------------------|-----------|---------------|------------|-------------|
| <input type="checkbox"/> | admin | impotrant_... | .txt | false |

Пошук будемо здійснювати у таблиці Currency, так як там більше даних

▼ Scan/Query items

Scan

Query

Table or index

Currency

▼

▼ Filters

| Attribute name | Type | Condition | Value | |
|----------------|--------|-----------|-------|--------|
| Amount | Number | Between | 2 | Remove |
| | | | and | |
| | | | 4 | |
| Units | Number | Equal to | 1 | Remove |
| Add filter | | | | |

Run

Reset

Результати пошуку

✔ Completed

Read capacity units consumed: 1

Items returned (6)

< 1 > ⚙️ 🔍

| <input type="checkbox"/> | Currency... ▾ | Currency... ▾ | Amount ▾ | StartDate ▾ | TimeSign ▾ |
|--------------------------|---------------|---------------|----------|-------------|------------|
| <input type="checkbox"/> | 972 | TJS | 2.5022 | 01.01.2021 | 0000 |
| <input type="checkbox"/> | 949 | TRY | 3.8448 | 01.01.2021 | 0000 |
| <input type="checkbox"/> | 344 | HKD | 3.6472 | 01.01.2021 | 0000 |
| <input type="checkbox"/> | 752 | SEK | 3.453 | 01.01.2021 | 0000 |
| <input type="checkbox"/> | 578 | NOK | 3.2957 | 01.01.2021 | 0000 |
| <input type="checkbox"/> | 504 | MAD | 3.1763 | 01.01.2021 | 0000 |

Створимо таблицю через AWS CLI

```
[ec2-user@ip-172-31-11-110 ~]$ aws dynamodb create-table --table-name Students --attribute-definitions AttributeName=Name,AttributeType=S AttributeName=Score,AttributeType=N --key-schema AttributeName=Name,KeyType=HASH AttributeName=Score,KeyType=RANGE --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5
{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "Name",
        "AttributeType": "S"
      },
      {
        "AttributeName": "Score",
        "AttributeType": "N"
      }
    ],
    "TableName": "Students",
    "KeySchema": [
      {
        "AttributeName": "Name",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "Score",
        "KeyType": "RANGE"
      }
    ],
    "TableStatus": "CREATING",
  }
}
```

Таблиця створилася

```
[ec2-user@ip-172-31-11-110 ~]$ aws dynamodb list-tables
{
  "TableNames": [
    "Currency",
    "Files",
    "Students"
  ]
}
[ec2-user@ip-172-31-11-110 ~]$
```

Додамо item

```
[ec2-user@ip-172-31-11-110 ~]$ aws dynamodb put-item --table-name Students --item '{"Name": {"S": "Lera"}, "Score": {"N": "80"}}'
[ec2-user@ip-172-31-11-110 ~]$ aws dynamodb scan --table-name Students
{
  "Items": [
    {
      "Score": {
        "N": "80"
      },
      "Name": {
        "S": "Lera"
      }
    }
  ],
  "Count": 1,
  "ScannedCount": 1,
  "ConsumedCapacity": null
}
```

Додамо дані через json файл

```
[ec2-user@ip-172-31-11-110 ~]$ aws dynamodb batch-write-item --request-items file:/students.json
{
  "UnprocessedItems": {}
}
```

Вміст файлу students.json:

```
1 {
2   "Students": [
3     {
4       "PutRequest": {
5         "Item": {
6           "Name": {
7             "S": "Ira"
8           },
9           "Score": {
10            "N": "79"
11          },
12           "Age": {
13            "N": "20"
14          },
15           "DateOfEnrolment": {
16            "S": "01.09.2019"
17          },
18           "Subjects": {
19            "SS": [
20              "Algebra",
21              "English",
22              "Physics"
23            ]
24          }
25        }
26      }
27    },
28    {
29      "PutRequest": {
30        "Item": {
31          "Name": {
32            "S": "Igor"
33          },
34          "Score": {
35            "N": "81"
36          },
37          "Age": {
38            "N": "21"
39          },
40          "DateOfEnrolment": {
41            "S": "01.09.2018"
42          },
43          "Subjects": {
44            "SS": [
45              "Geometry",
46              "English",
47              "Physics"
48            ]
49          }
50        }
51      }
52    },
53    {
54      "PutRequest": {
55        "Item": {
56          "Name": {
57            "S": "Maryna"
58          },
59          "Score": {
60            "N": "86"
61          },
62          "Age": {
63            "N": "20"
64          },
65          "DateOfEnrolment": {
66            "S": "01.09.2018"
67          },
68          "Subjects": {
69            "SS": [
70              "Algebra",
71              "German",
72              "IT"
73            ]
74          }
75        }
76      }
77    },
78    {
79      "PutRequest": {
80        "Item": {
81          "Name": {
82            "S": "Anton"
83          },
84          "Score": {
85            "N": "95"
86          },
87          "Age": {
88            "N": "19"
89          },
90          "DateOfEnrolment": {
91            "S": "01.09.2020"
92          },
93          "Subjects": {
94            "SS": [
95              "Chemistry",
96              "English",
97              "Physics"
98            ]
99          }
100        }
101      }
102    },
103    {
104      "PutRequest": {
105        "Item": {
106          "Name": {
107            "S": "Ana"
108          },
109          "Score": {
110            "N": "77"
111          },
112          "Age": {
113            "N": "18"
114          },
115          "DateOfEnrolment": {
116            "S": "01.09.2020"
117          },
118          "Subjects": {
119            "SS": [
120              "Algebra",
121              "Geometry",
122              "Physics"
123            ]
124          }
125        }
126      }
127    }
128  ]
129 }
```

Дані успішно завантажилися

| | | | | | | |
|---|--------|---------|-------|--------------|---------------------------------------|--|
| <div><div>Completed</div><div>Read capacity units consumed: 0.5</div></div> | | | | | | |
| Items returned (6) | | | | | | |
| <div><div><</div><div>1</div><div>></div><div>⚙</div><div>✖</div></div> | | | | | | |
| <input type="checkbox"/> | Name ▾ | Score ▾ | Age ▾ | DateOfE... ▾ | Subjects ▾ | |
| <input type="checkbox"/> | Igor | 81 | 21 | 01.09.2018 | { "English", "Geometry", "Physics" } | |
| <input type="checkbox"/> | Lera | 80 | | | | |
| <input type="checkbox"/> | Ira | 79 | 20 | 01.09.2019 | { "Algebra", "English", "Physics" } | |
| <input type="checkbox"/> | Maryna | 86 | 20 | 01.09.2018 | { "Algebra", "German", "IT" } | |
| <input type="checkbox"/> | Anton | 95 | 19 | 01.09.2020 | { "Chemistry", "English", "Physics" } | |
| <input type="checkbox"/> | Ana | 77 | 18 | 01.09.2020 | { "Algebra", "Geometry", "Physics" } | |

Видалимо строку з partition key 'Anton' та sort key '95'

```
[ec2-user@ip-172-31-11-110 ~]$ aws dynamodb delete-item --table-name Students
--key '{"Name":{"S":"Anton"},"Score":{"N":"95"}}' --return-value ALL_OLD
{
  "Attributes": {
    "DateOfEnrolment": {
      "S": "01.09.2020"
    },
    "Score": {
      "N": "95"
    },
    "Subjects": {
      "SS": [
        "Chemistry",
        "English",
        "Physics"
      ]
    },
    "Age": {
      "N": "19"
    },
    "Name": {
      "S": "Anton"
    }
  }
}
```

Пошук даних здійснюється командою query

```
[ec2-user@ip-172-31-11-110 ~]$ aws dynamodb query --table-name Currency --key-condition-expression "CurrencyCode = :v1" --expression-attribute-values file://conf.json
{
  "Items": [
    {
      "CurrencyCodeL": {
        "S": "EUR"
      },
      "StartDate": {
        "S": "01.01.2021"
      },
      "TimeSign": {
        "S": "0000"
      },
      "Units": {
        "N": "1"
      },
      "Amount": {
        "N": "34.7396"
      },
      "CurrencyCode": {
        "S": "978"
      }
    }
  ],
  "Count": 1,
  "ScannedCount": 1,
  "ConsumedCapacity": null
}
[ec2-user@ip-172-31-11-110 ~]$
[ec2-user@ip-172-31-11-110 ~]$ cat conf.json
{
  ":v1": {"S": "978"}
}
```

Висновок: В ході виконання лабораторної роботи я ознайомилася з DynamoDB та як керувати даними через Management Console та AWS CLI. Виникли труднощі з завантаженням даних у форматі json, які ми отримали у попередній роботі, через Management Console та AWS CLI, так як не підходив синтаксис json для DynamoDB.