MỤC LỤC

[**1. Khái niệm:** 2](#_Toc531131241)

[**2. Các thành phần cốt lõi của DynamoDB:** 2](#_Toc531131242)

[**2.1 Bảng, mục và thuộc tính.** 2](#_Toc531131243)

[**2.2 Khóa chính** 4](#_Toc531131244)

[**2.3 Chỉ mục phụ** 4](#_Toc531131245)

[**2.4 Luồng DynamoDB** 5](#_Toc531131246)

[**3. So sánh giữa khác nhau giữa SQL với No SQL:** 7](#_Toc531131247)

[**4. Ưu điểm của DynamoDB:** 8](#_Toc531131248)

[**4.1 Hiệu năng quy mô (Great Performance).** 8](#_Toc531131249)

[**4.2 Phù hợp cho doanh nghiệp (Ready for Firms).** 8](#_Toc531131250)

[**4.3 Được quản lý toàn phần (Automatic Management).** 8](#_Toc531131251)

[**5. Trường hợp sử dụng.** 9](#_Toc531131252)

[**6. Truy vấn dữ liệu trên DynamoDB.** 9](#_Toc531131253)

[**6.1 Sử dụng AWS Management Console.** 9](#_Toc531131254)

[**6.2 Sử dụng CLI**. 11](#_Toc531131255)

[**6.3 Sử dụng API.** 13](#_Toc531131256)

[**7. Giới thiệu cách truy vấn DynamoDB bằng PHP.** 13](#_Toc531131257)

[**7.0** **Khai báo thư viện của dynamoDB và sử dụng Composer:** 13](#_Toc531131258)

[**7.1** **Kết nối với DynamoDB:** 13](#_Toc531131259)

[**7.2** **Tạo bảng:** 14](#_Toc531131260)

[**7.3** **Thêm dữ liệu vào bảng:** 15](#_Toc531131261)

[**7.4** **Lấy dữ liệu từ bảng Movies:** 16](#_Toc531131262)

[**7.5** **Cập nhật dữ liệu trong bảng:** 16](#_Toc531131263)

[**7.6** **Xóa dữ liệu trong bảng:** 18](#_Toc531131264)

[**8. Tài liệu tham khảo** 19](#_Toc531131265)

# **1. Khái niệm:**

Amazon DynamoDB là một dịch vụ cơ sở dữ liệu NoSQL được quản lý đầy đủ, cung cấp hiệu suất nhanh và có thể dự đoán được với khả năng mở rộng liền mạch. DynamoDB cho phép bạn giảm gánh nặng hành chính của hoạt động và mở rộng cơ sở dữ liệu phân tán, do đó bạn không phải lo lắng về việc cung cấp phần cứng, thiết lập và cấu hình, nhân rộng, vá phần mềm hoặc chia tỷ lệ cụm.

Với DynamoDB, bạn có thể tạo các bảng cơ sở dữ liệu có thể lưu trữ và truy xuất bất kỳ lượng dữ liệu nào và phục vụ bất kỳ mức lưu lượng truy vấn yêu cầu nào. Bạn có thể mở rộng quy mô hoặc tăng dung lượng lưu trữ của bảng mà không có thời gian chết hoặc suy giảm hiệu suất và sử dụng Bảng điều khiển quản lý AWS để giám sát việc sử dụng tài nguyên và các chỉ số hiệu suất.

Amazon DynamoDB cung cấp khả năng sao lưu theo yêu cầu. Nó cho phép bạn tạo bản sao lưu đầy đủ các bảng của bạn để lưu giữ và lưu trữ lâu dài cho các nhu cầu tuân thủ quy định.

# **2. Các thành phần cốt lõi của DynamoDB:**

Trong DynamoDB, các bảng, các mục và các thuộc tính là các thành phần cốt lõi mà bạn làm việc cùng. Một bảng là một tập hợp các mặt hàng, và mỗi mục là một tập hợp các thuộc tính . DynamoDB sử dụng các khóa chính để nhận dạng duy nhất từng mục trong một bảng và các chỉ mục phụ để cung cấp tính linh hoạt truy vấn nhiều hơn. Bạn có thể sử dụng các Stream DynamoDB để nắm bắt các sự kiện sửa đổi dữ liệu trong các bảng DynamoDB.

## **2.1 Bảng, mục và thuộc tính.**

* **Bảng** - Tương tự như các hệ thống cơ sở dữ liệu khác, DynamoDB lưu trữ dữ liệu trong các bảng. Một*bảng* là một tập hợp các dữ liệu. Ví dụ, xem bảng ví dụ được gọi là *Những người* mà bạn có thể sử dụng để lưu trữ thông tin liên hệ cá nhân về bạn bè, gia đình hoặc bất kỳ ai quan tâm khác. Bạn cũng có thể có một bảng *Ô tô* để lưu trữ thông tin về các phương tiện mà mọi người lái xe.
* **Mục** - Mỗi bảng chứa không hoặc nhiều mục. Một *mục* là một nhóm thuộc tính được nhận dạng duy nhất trong số tất cả các mục khác. Trong bảng *Mọi người*, mỗi mục đại diện cho một người. Đối với một bảng *Ô tô*, mỗi mục đại diện cho một chiếc xe. Các mục trong DynamoDB tương tự trong nhiều cách để các hàng, bản ghi hoặc các bộ dữ liệu trong các hệ thống cơ sở dữ liệu khác. Trong DynamoDB, không có giới hạn về số lượng các mục bạn có thể lưu trữ trong một bảng.
* **Thuộc tính** - Mỗi mục bao gồm một hoặc nhiều thuộc tính. Một *thuộc tính* là một phần tử dữ liệu cơ bản, một thứ không cần phải phân tích thêm nữa. Ví dụ: một mục trong bảng *Mọi người* chứa các thuộc tính được gọi là *PersonID*, *LastName* , *FirstName* , v.v. Đối với bảng *Bộ* , một mục có thể có các thuộc tính như *DepartmentID* , *Name* ,*Quản lý* , v.v. Các thuộc tính trong DynamoDB tương tự như trong nhiều trường hoặc các cột trong các hệ thống cơ sở dữ liệu khác.



## **2.2 Khóa chính**

* **Phím phân vùng (**Partition key**)** - Một khóa chính đơn giản, bao gồm một thuộc tính được gọi là khóa phân vùng.

DynamoDB sử dụng giá trị của khóa phân vùng làm đầu vào cho hàm băm bên trong. Đầu ra từ hàm băm xác định phân vùng (lưu trữ vật lý bên trong DynamoDB) trong đó mục sẽ được lưu trữ.

Trong một bảng chỉ có một khóa phân vùng, không có hai mục nào có thể có cùng giá trị khóa phân vùng.

* **Phím phân vùng và phím sắp xếp (**Partition key and sort key**)** - Được gọi là khóa chính kết hợp, loại khóa này bao gồm hai thuộc tính. Thuộc tính đầu tiên là khóa phân vùng và thuộc tính thứ hai là khóa sắp xếp.

DynamoDB sử dụng giá trị khóa phân vùng làm đầu vào cho hàm băm bên trong. Đầu ra từ hàm băm xác định phân vùng (lưu trữ vật lý bên trong DynamoDB) trong đó mục sẽ được lưu trữ. Tất cả các mục có cùng khóa phân vùng được lưu trữ cùng nhau, theo thứ tự sắp xếp theo giá trị khóa sắp xếp.

Trong bảng có khóa phân vùng và khóa sắp xếp, có thể có hai mục có cùng giá trị khóa phân vùng. Tuy nhiên, hai mục này phải có các giá trị khóa sắp xếp khác nhau.

## **2.3 Chỉ mục phụ**

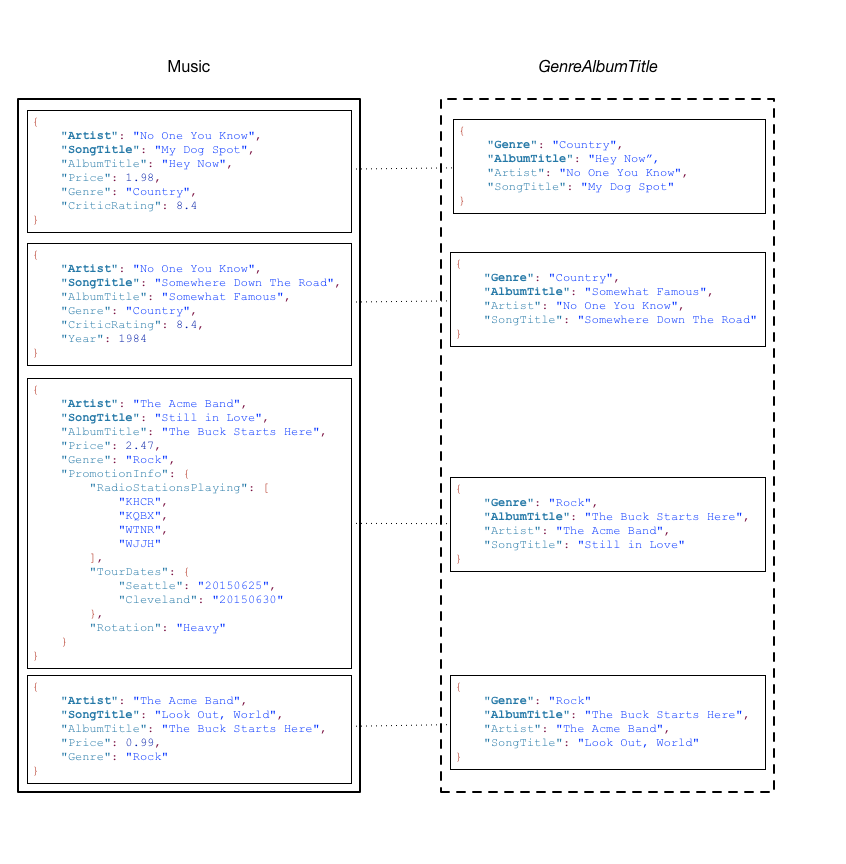
Bạn có thể tạo một hoặc nhiều chỉ mục phụ trên bảng. Một chỉ số thứ cấp cho phép bạn truy vấn dữ liệu trong bảng bằng cách sử dụng một chìa khóa thay thế, ngoài việc truy vấn đối với các khóa chính. DynamoDB không yêu cầu bạn sử dụng các chỉ mục, nhưng chúng cho phép các ứng dụng của bạn linh hoạt hơn khi truy vấn dữ liệu của bạn. Sau khi bạn tạo chỉ mục phụ trên một bảng, bạn có thể đọc dữ liệu từ chỉ mục theo cách tương tự như bạn thực hiện từ bảng.

DynamoDB hỗ trợ hai loại chỉ mục:

Chỉ số phụ toàn cầu - Chỉ mục có khóa phân vùng và khóa sắp xếp có thể khác với chỉ mục trên bảng.

Chỉ số phụ cục bộ - Chỉ mục có cùng khóa phân vùng như bảng, nhưng một phím sắp xếp khác.

Bạn có thể xác định tối đa 5 chỉ mục phụ toàn cầu và 5 chỉ mục phụ cục bộ trên mỗi bảng.



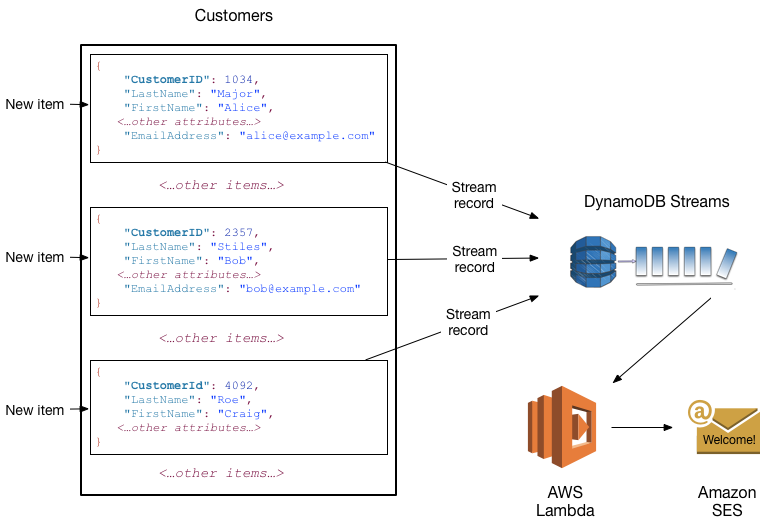
## **2.4 Luồng DynamoDB**

DynamoDB Streams là một tính năng tùy chọn để nắm bắt các sự kiện sửa đổi dữ liệu trong các bảng DynamoDB. Dữ liệu về các sự kiện này xuất hiện trong luồng trong thời gian thực gần và theo thứ tự các sự kiện đã xảy ra.

Mỗi sự kiện được biểu diễn bằng một bản ghi luồng. Nếu bạn bật luồng trên một bảng, Luồng DynamoDB ghi một bản ghi luồng bất cứ khi nào một trong các sự kiện sau xảy ra:

* Một mục mới được thêm vào bảng: Luồng chụp một hình ảnh của toàn bộ mục, bao gồm tất cả các thuộc tính của nó.
* Mục được cập nhật: Luồng chụp ảnh "trước" và "sau" của bất kỳ thuộc tính nào đã được sửa đổi trong mục.
* Một mục sẽ bị xóa khỏi bảng: Luồng sẽ ghi lại hình ảnh của toàn bộ mục trước khi nó bị xóa.

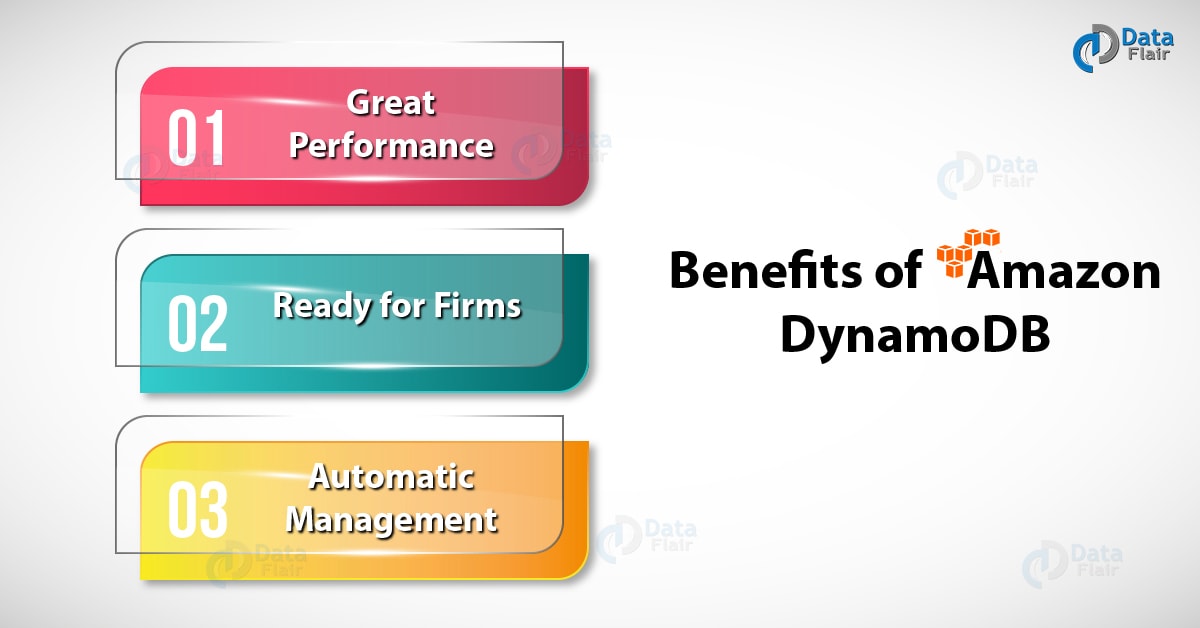
Mỗi bản ghi luồng cũng chứa tên của bảng, dấu thời gian sự kiện và siêu dữ liệu khác. Bản ghi luồng có thời lượng 24 giờ; sau đó, chúng sẽ tự động bị xóa khỏi luồng.



# **3. So sánh giữa khác nhau giữa SQL với No SQL:**

|  |  |  |
| --- | --- | --- |
| **Đặc trưng** | **Hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS)** | **Amazon DynamoDB** |
| **Khối lượng công việc tối ưu** | Truy vấn đặc biệt; kho dữ liệu; OLAP (xử lý phân tích trực tuyến). | Các ứng dụng quy mô trên web, bao gồm mạng xã hội, chơi trò chơi, chia sẻ phương tiện và IoT (Internet of Things). |
| **Mô hình dữ liệu** | Mô hình quan hệ yêu cầu một lược đồ được xác định rõ, nơi dữ liệu được chuẩn hóa thành các bảng, các hàng và các cột. Ngoài ra, tất cả các mối quan hệ được xác định giữa các bảng, cột, chỉ mục và các phần tử cơ sở dữ liệu khác. | DynamoDB là schemaless. Mỗi bảng phải có khóa chính để nhận dạng duy nhất từng mục dữ liệu, nhưng không có ràng buộc tương tự nào đối với các thuộc tính không phải khóa khác. DynamoDB có thể quản lý dữ liệu có cấu trúc hoặc bán cấu trúc, bao gồm cả tài liệu JSON. |
| **Truy cập dữ liệu** | SQL (Structured Query Language) là tiêu chuẩn để lưu trữ và truy xuất dữ liệu. Các cơ sở dữ liệu quan hệ cung cấp một bộ công cụ phong phú để đơn giản hóa việc phát triển các ứng dụng dựa trên cơ sở dữ liệu, nhưng tất cả các công cụ này đều sử dụng SQL. | Bạn có thể sử dụng AWS Management Console hoặc AWS CLI để làm việc với DynamoDB và thực hiện các tác vụ đặc biệt. Các ứng dụng có thể tận dụng bộ công cụ phát triển phần mềm AWS (SDK) để làm việc với DynamoDB bằng cách sử dụng giao diện dựa trên đối tượng, tài liệu trung tâm hoặc cấp thấp. |
| **Hiệu suất** | Cơ sở dữ liệu quan hệ được tối ưu hóa để lưu trữ, do đó hiệu năng thường phụ thuộc vào hệ thống con đĩa. Nhà phát triển và quản trị viên cơ sở dữ liệu phải tối ưu hóa truy vấn, chỉ mục và cấu trúc bảng để đạt được hiệu suất cao nhất. | DynamoDB được tối ưu hóa cho tính toán, do đó hiệu năng chủ yếu là một chức năng của phần cứng cơ bản và độ trễ mạng. Là một dịch vụ được quản lý, DynamoDB cách ly bạn và các ứng dụng của bạn từ các chi tiết triển khai này, để bạn có thể tập trung vào thiết kế và xây dựng các ứng dụng hiệu năng cao. |
| **Chia tỷ lệ** | Nó là dễ nhất để mở rộng với phần cứng nhanh hơn. Cũng có thể cho các bảng cơ sở dữ liệu trải rộng trên nhiều máy chủ trong một hệ thống phân tán, nhưng điều này đòi hỏi đầu tư bổ sung. Cơ sở dữ liệu quan hệ có kích thước tối đa cho số lượng và kích thước tệp, áp dụng giới hạn trên về khả năng mở rộng. | DynamoDB được thiết kế để mở rộng quy mô bằng cách sử dụng các cụm phần cứng được phân phối. Thiết kế này cho phép tăng thông lượng mà không tăng độ trễ. Khách hàng chỉ định các yêu cầu về thông lượng của họ và DynamoDB phân bổ đủ nguồn lực để đáp ứng các yêu cầu đó. Không có giới hạn trên về số lượng mục trên mỗi bảng cũng như tổng kích thước của bảng đó. |

# **4. Ưu điểm của DynamoDB:**



## **4.1 Hiệu năng quy mô (Great Performance).**

DynamoDB luôn luôn đáp ứng chỉ trong vài mili-giây ở bất cứ quy mô nào. Phát triển các ứng dụng với gần như không có giới hạn về thông lượng và lưu trữ. Thêm một bộ nhớ đệm trong bộ nhớ để giảm thời gian đáp ứng từ vài mili-giây xuống còn vài micro-giây mà không phải thay đổi gì trên ứng dụng.

## **4.2 Phù hợp cho doanh nghiệp (Ready for Firms).**

Xây dựng để phục vụ cho những khối lượng công việc tối quan trọng. Dữ liệu của bạn được bảo mật bằng mã hóa và tính ổn định được đảm bảo với một thỏa thuận cấp độ dịch vụ. Bạn sẽ được giám sát toàn bộ các bảng bằng kiểm soát truy cập chi tiết, các công cụ giám sát tích hợp, cũng như hỗ trợ các kết nối riêng qua VPN.

## **4.3 Được quản lý toàn phần (Automatic Management).**

DynamoDB là một ứng dụng serverless có thể tự động điều chỉnh quy mô thông lượng lớn hơn hoặc nhỏ đi, và bảo vệ dữ liệu của bạn bằng cách liên tục sao lưu. DynamoDB giúp các ứng dụng phân phối toàn cầu của bạn truy cập nhanh chóng tới dữ liệu cục bộ, bằng cách tạo bản sao các bảng tới các Khu vực AWS.

# **5. Trường hợp sử dụng.**

* **Ứng dụng web serverless.**

Phát triển các ứng dụng web mạnh mẽ mà có thể tự động tăng hoặc giảm quy mô. Bạn sẽ không cần phải duy trì những máy chủ và các ứng dụng của bạn luôn có có độ khả dụng cao.

* **Kho dữ liệu vi dịch vụ.**

Xây dựng các vi dịch vụ linh hoạt và có thể tái sử dụng nhờ dùng DynamoDB như một kho dữ liệu serverless để đạt hiệu năng nhanh và ổn định.

* **Backend di động.**

Phát triển những ứng dụng di động được cá nhân hóa cùng trải nghiệm mượt mà cho những người dùng của bạn. DynamoDB sẽ lo liệu những tác vụ vận hành, để bạn có thể tập trung vào những ứng dụng của bạn.

* **Công nghệ quảng cáo.**

Tạo một nền tảng đấu thầu thời gian thực và công cụ khuyến nghị bằng khả năng điều chỉnh quy mô, thông lượng và tính khả dụng của DynamoDB.

* **Trò chơi.**

Tạo trò chơi có thiết kế phản hồi cho di động, máy chơi game và máy tính bàn bằng DynamoDB. Lưu trữ và truy vấn dữ liệu trò chơi như trạng thái người chơi, bảng điểm hoặc nội dung động trong thế giới của trò chơi.

* **IOT.**

Phân tích các thiết bị của bạn bằng cách kết nối dữ liệu IoT dung lượng lớn, tốc độ cao tại DynamoDB tới Amazon Redshift và Amazon QuickSight.

# **6. Truy vấn dữ liệu trên DynamoDB.**

Bạn có thể truy cập Amazon DynamoDB bằng *AWS Management Console*, *Giao diện dòng lệnh AWS (AWS CLI)* hoặc *API DynamoDB*.

## **6.1 Sử dụng AWS Management Console.**

Bạn có thể truy cập AWS Management Console cho DynamoDB: <https://console.aws.amazon.com/dynamodb/home>.

Bạn có thể sử dụng bàn điều khiển để thực hiện các thao tác sau trong DynamoDB:

* Theo dõi các cảnh báo gần đây, tổng dung lượng, sức khỏe dịch vụ và tin tức DynamoDB mới nhất trên bảng điều khiển DynamoDB.
* Tạo, cập nhật và xóa bảng. Máy tính công suất cung cấp các ước tính về số lượng đơn vị công suất yêu cầu dựa trên thông tin sử dụng mà bạn cung cấp.
* Quản lý luồng.
* Xem, thêm, cập nhật và xóa các mục được lưu trữ trong bảng. Quản lý thời gian để sống (TTL) để xác định khi nào các mục trong bảng hết hạn để chúng có thể tự động bị xóa khỏi cơ sở dữ liệu.
* Truy vấn và quét bảng.
* Cài đặt và xem báo thức để theo dõi việc sử dụng dung lượng của bảng.
* Xem số liệu giám sát hàng đầu của bảng của bạn trên biểu đồ thời gian thực từ CloudWatch.
* Sửa đổi khả năng cung cấp của một bảng.
* Tạo và xóa các chỉ mục phụ toàn cầu.
* Tạo các trình kích hoạt để kết nối các luồng DynamoDB với các hàm Lambda AWS.
* Áp dụng thẻ cho tài nguyên của bạn để giúp sắp xếp và xác định chúng.
* Mua dung lượng dự trữ.

Bàn điều khiển hiển thị màn hình giới thiệu nhắc bạn tạo bảng đầu tiên của mình. Để xem các bảng của bạn, từ bảng điều hướng ở phía bên trái của giao diện điều khiển, hãy chọn Bảng.

Dưới đây là tổng quan cấp cao về các hành động có sẵn trên mỗi bảng trong mỗi tab điều hướng:

* Overview - Xem chi tiết về luồng và bảng, đồng thời quản lý luồng và Thời gian phát trực tiếp (TTL).
* Items - Quản lý các mục và thực hiện các truy vấn và quét.
* Metrix - Theo dõi chỉ số CloudWatch.
* Alarms - Quản lý báo thức CloudWatch.
* Capacity - Sửa đổi dung lượng được cung cấp của một bảng.
* Indexes - Quản lý chỉ mục phụ toàn cầu.
* Triggers - Quản lý trình kích hoạt để kết nối các luồng DynamoDB với các hàm Lambda.
* Access control - Thiết lập kiểm soát truy cập chi tiết với liên kết nhận dạng web.
* Tags - Áp dụng thẻ cho tài nguyên của bạn để giúp sắp xếp và xác định chúng.

## **6.2 Sử dụng CLI**.

Bạn có thể sử dụng Giao diện dòng lệnh AWS (AWS CLI) để điều khiển nhiều dịch vụ AWS từ dòng lệnh và tự động hóa chúng thông qua các tập lệnh. Bạn có thể sử dụng AWS CLI cho các hoạt động đặc biệt, chẳng hạn như tạo bảng. Bạn cũng có thể sử dụng nó để nhúng các hoạt động DynamoDB trong các Scripts.

Trước khi bạn có thể sử dụng AWS CLI với DynamoDB, bạn phải có mã khóa truy cập (key ID) và khóa truy cập bí mật (Secret access key).

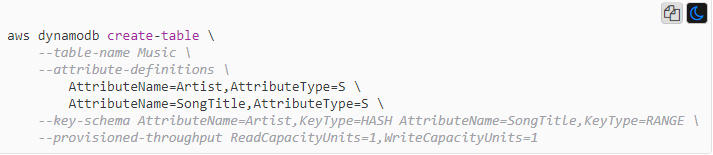
Tải xuống và định cấu hình AWS CLI:

AWS CLI có sẵn tại <http://aws.amazon.com/cli>. Nó chạy trên Windows, macOS hoặc Linux. Sau khi bạn tải xuống AWS CLI, hãy cài đặt và định cấu hình nó.

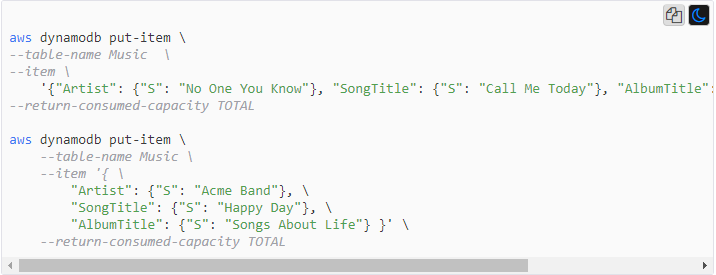
Sử dụng AWS CLI với DynamoDB:

Định dạng dòng lệnh bao gồm tên vận hành DynamoDB, tiếp theo là các tham số cho hoạt động đó. AWS CLI hỗ trợ cú pháp viết tắt cho các giá trị tham số, cũng như JSON.

Ví dụ, lệnh sau tạo một bảng có tên Music . Phím phân vùng là Artist và phím sắp xếp là SongTitle . (Để dễ đọc hơn, các lệnh dài trong phần này được chia thành các dòng riêng biệt.)



Các lệnh sau đây thêm các mục mới vào bảng. Các ví dụ này sử dụng kết hợp cú pháp viết tắt và JSON.



Trên dòng lệnh, rất khó để soạn JSON hợp lệ. Tuy nhiên, AWS CLI có thể đọc các tệp JSON. Ví dụ: hãy xem đoạn mã JSON sau, được lưu trữ trong tệp có tên là key-conditions.json :



Bây giờ bạn có thể đưa ra Query yêu cầu bằng AWS CLI. Trong ví dụ này, nội dung của tệp key-conditions.json được sử dụng cho --key-conditionstham số:



Sử dụng AWS CLI với DynamoDB có thể tải xuống.

AWS CLI cũng có thể tương tác với DynamoDB (Phiên bản có thể tải xuống) chạy trên máy tính của bạn. Để kích hoạt tính năng này, hãy thêm thông số sau vào mỗi lệnh:

--endpoint-url http://localhost:8000

Đây là một ví dụ sử dụng AWS CLI để liệt kê các bảng trong cơ sở dữ liệu cục bộ:



Nếu DynamoDB đang sử dụng một số cổng khác với số cổng mặc định (8000), hãy sửa đổi --endpoint-url giá trị tương ứng.

## **6.3 Sử dụng API.**

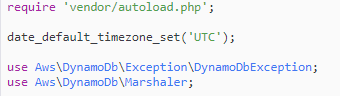
* + Bạn có thể sử dụng AWS Management Console và AWS Command Line Interface để làm việc tương tác với Amazon DynamoDB. Tuy nhiên, để tận dụng tối đa DynamoDB, bạn có thể viết mã ứng dụng bằng cách sử dụng SDK AWS.
  + AWS SDK cung cấp hỗ trợ rộng cho DynamoDB trong Java, JavaScript in the browser , .NET , Node.js , PHP , Python , Ruby , C ++ , Go , Android và iOS.
  + Trước khi bạn có thể sử dụng AWS CLI với DynamoDB, bạn phải có mã khóa truy cập (key ID) và khóa truy cập bí mật (key).

# **7. Giới thiệu cách truy vấn DynamoDB bằng PHP.**

Bắt đầu sử dụng nhanh chóng AWS với AWS SDK cho PHP. SDK là thư viện PHP mã nguồn mở hiện đại giúp bạn dễ dàng tích hợp ứng dụng PHP với các dịch vụ AWS như Amazon S3, Amazon Glacier mà đặc biệt ở đây là Amazon DynamoDB.

Cách truy vấn DynamoDB bằng ngôn ngữ PHP:

* 1. **Khai báo thư viện của dynamoDB và sử dụng Composer:**



* 1. **Kết nối với DynamoDB:**



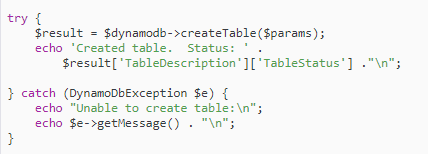
* 1. **Tạo bảng:**

Ta tạo một bảng có tên là Movies. Khóa chính cho bảng bao gồm hai thuộc tính sau:

* year- Khóa phân vùng. Các AttributeType là N cho kiểu dữ liệu số.
* title- Khóa sắp xếp. Các AttributeType là S cho kiểu dữ liệu chuỗi.



+ Thực thi lệnh tạo bảng:



* 1. **Thêm dữ liệu vào bảng:**

+ Để thêm dữ liệu vào bảng ta sử dụng phương thức putItem. DynamoDB cho phép ta thêm nhiều trường vào bảng mà không cần phải khai báo ở mục tạo bảng



+ Để thực thi việc thêm dữ liệu vào bảng ta dùng:



* 1. **Lấy dữ liệu từ bảng Movies:**

+Ta có thể sử dụng phương thức getItem để lấy dữ liệu từ Movies bảng. Để lấy dữ liệu phải chỉ định các giá trị khóa chính, vì vậy ta có thể lấy bất kỳ mục nào từ bảng Movies nếu biết year và title (Partition key và Sort Key).

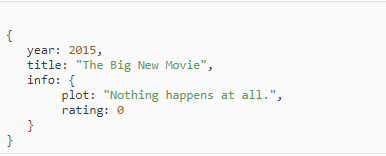


+ Để thực thi lệnh lấy dữ liệu:

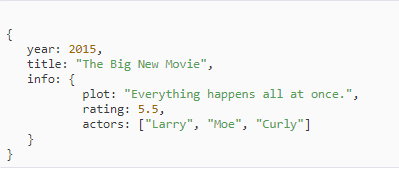


* 1. **Cập nhật dữ liệu trong bảng:**

+ Ta sử dụng phương thức getItem để sửa đổi dữ liệu trong bảng



Hình dữ liệu trước khi cập nhật.



Hình dữ liệu sau khi được cập nhật.

+ Để được kết quả trên ta chạy lệnh sau:



+ Thực thi lệnh update:

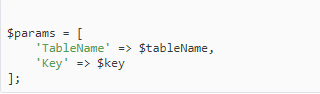


* 1. **Xóa dữ liệu trong bảng:**

+ Để xóa dữ liệu ta sử dụng phương thức deleteItem và chỉ định khóa chính cho nó. Ta có thể tùy chọn cung cấp một Condition Expression để ngăn chặn các mục bị xóa nếu điều kiện không được đáp ứng. VD sau ta xóa một mục phim cụ thể nếu xếp hạng của nó là 5 hoặc ít hơn.



+ Nếu không cần điều kiện xóa thì ta loại bỏ điều kiện



+ Thực thi lệnh xóa :



# **8. Tài liệu tham khảo**

1. <https://aws.amazon.com/vi/dynamodb/>
2. <https://techblog.vn/gioi-thieu-ve-dynamodb-phan-1>
3. <https://docs.aws.amazon.com/AWSJavaScriptSDK/latest/AWS/DynamoDB/DocumentClient.html>
4. <https://docs.aws.amazon.com/en_us/amazondynamodb/latest/developerguide/GettingStarted.PHP.html>

**\_\_THE END\_\_**