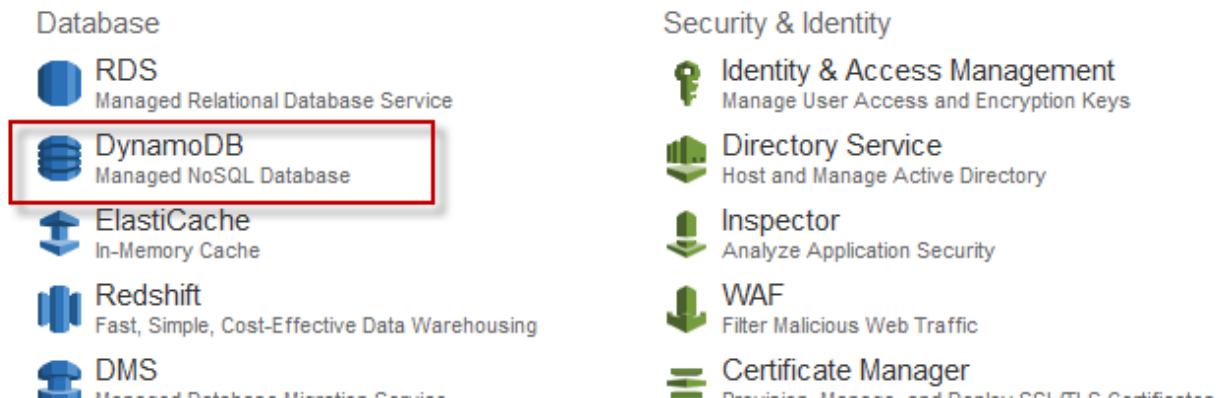


DYNAMODB INTRODUCTION

Amazon DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability. DynamoDB lets you offload the administrative burdens of operating and scaling a distributed database, so that you don't have to worry about hardware provisioning, setup and configuration, replication, software patching, or cluster scaling.

From the Aws Console page choose DynamoDB under Database section.



Choose Create table to create a new one.



On the next page, specify table name as orders, and primary key as order_id and choose type as Number. Then choose Create.

Create DynamoDB table

[Tutorial](#)

DynamoDB is a schema-less database that only requires a table name and primary key. The table's primary key is made up of one or two attributes that uniquely identify items, partition the data, and sort data within each partition.

Table name ⓘ

Primary key* Partition key

ⓘ

☐ Add sort key

Table settings

Default settings provide the fastest way to get started with your table. You can modify these default settings now or after your table has been created.

☒ Use default settings

- No secondary indexes.
- Provisioned capacity set to 5 reads and 5 writes.
- Basic alarms with 80% upper threshold using SNS topic "dynamodb".

Additional charges may apply if you exceed the AWS Free Tier levels for CloudWatch or Simple Notification Service. Advanced alarm settings are available in the CloudWatch management console.

[Cancel](#)[Create](#)

You can see status as table is being created.

[Create table](#) [Actions](#)

Name
orders

orders [Close](#)

[Overview](#) [Items](#) [Metrics](#) [Alarms](#) [Capacity](#) [Indexes](#) [Access control](#)

[Table is being created](#)

Recent alerts

No CloudWatch alarms have been triggered for this table.

Stream details

Stream enabled No

Once created you can see the table status is Active under Overview of Table.

The screenshot shows the AWS Glue console interface. On the left, there's a sidebar with a 'Create table' button and a search filter. The main area displays the 'orders' table details. The 'Overview' tab is selected, showing the table's status as 'Active'. Other tabs include 'Items', 'Metrics', 'Alarms', 'Capacity', 'Indexes', and 'Access control'. The 'Stream details' section shows 'Stream enabled' as 'No'. The 'Table details' section lists the table name, primary partition key, primary sort key, table status, creation date, and provisioned read capacity units.

Table name	orders
Primary partition key	order_id (Number)
Primary sort key	-
Table status	Active
Creation date	May 7, 2016 at 12:55:49 AM UTC+5:30
Provisioned read capacity units	5

Now will try to create a table with both Partition key and sort key, with the following format.

Order ID	Line Number	Product ID	Status
630178	1	99717	unshipped
630178	2	90127	unshipped
607161	1	98379	shipped
672102	1	95326	shipped
672102	2	95671	unshipped
672102	3	94023	shipped
690727	1	95502	shipped

Choose create table to create a new one.

	Name	Status	Partition key	Sort key
<input type="radio"/>	orders	Active	order_id (Number)	-

Then specify table name as order_line_items.

Specify partition key as order_id and sort key as line_number, make both has Number and choose Create.

DynamoDB is a schema-less database that only requires a table name and primary key. The table's primary key is made up of one or two attributes that uniquely identify items, partition the data, and sort data within each partition.

Table name* order_line_items ⓘ

Primary key*

Partition key: order_id Number ⓘ

☒ Add sort key

Sort key: line_number Number ⓘ

Table settings

Default settings provide the fastest way to get started with your table. You can modify these default settings now or after your table has been created.

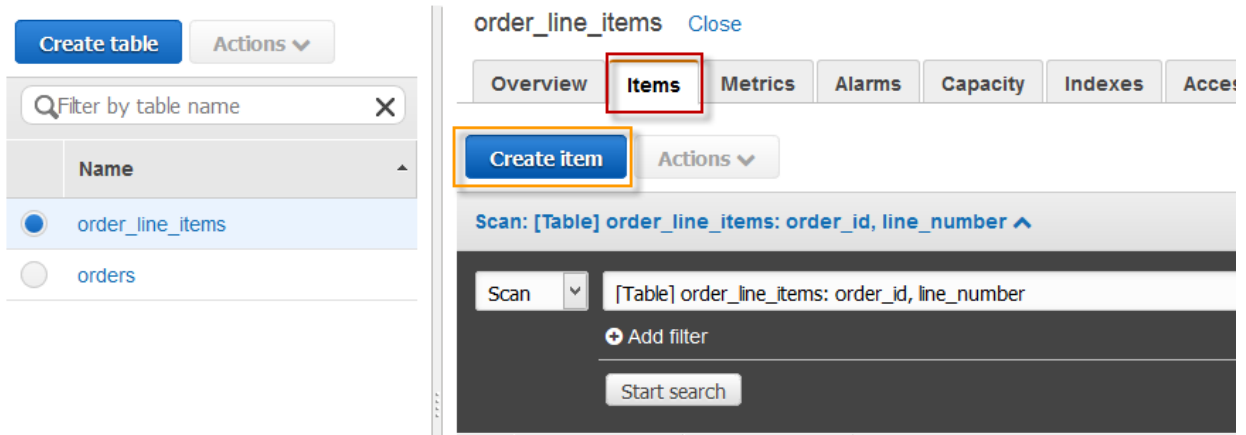
☒ Use default settings

- No secondary indexes.
- Provisioned capacity set to 5 reads and 5 writes.
- Basic alarms with 80% upper threshold using SNS topic "dynamodb".

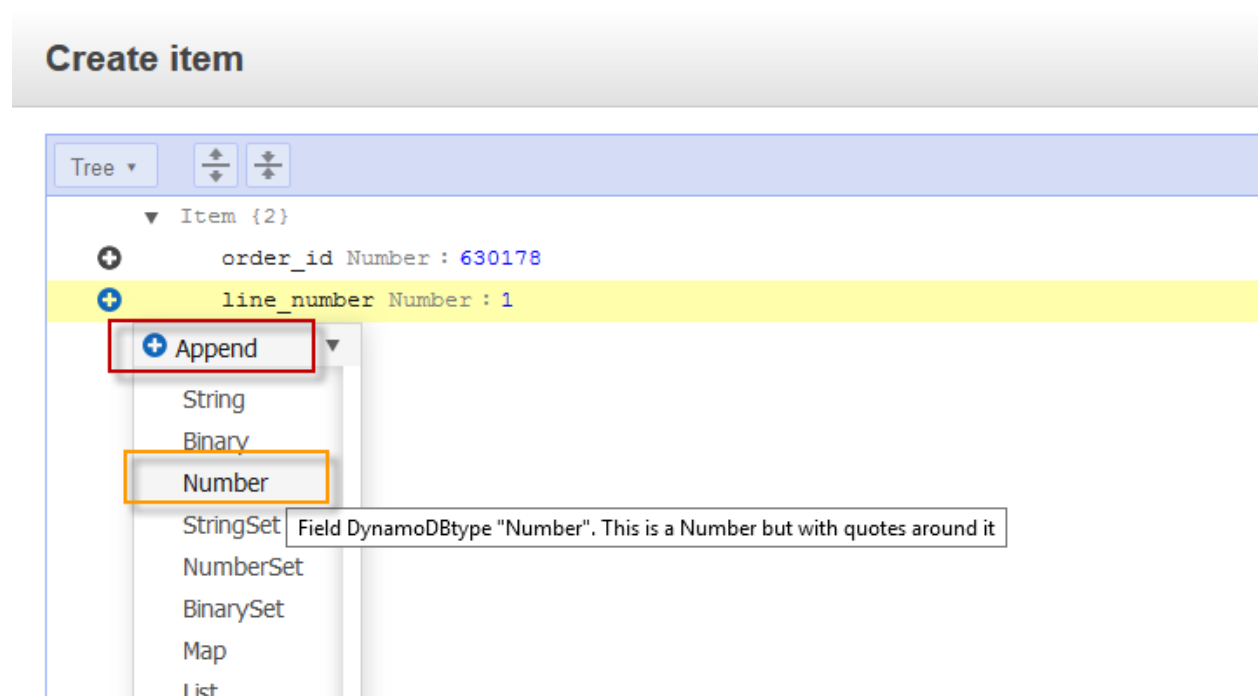
Additional charges may apply if you exceed the AWS Free Tier levels for CloudWatch or Simple Notification Service. Advanced alarm settings are available in the CloudWatch management console.

Cancel Create

Once created, select table go to Items on right pane.
Then choose Create Item to create a new one.



Specify order_id and line_number.
Then choose plus (+) symbol on line_number and select append choose Number.



Then specify Field as product_id and specify value as 99717

Create item

Tree ▾

↕

⊞

▼ Item {3}

⊕

order_id

Number : 630178

⊕

line_number

Number : 1

⊕

product_id

Number : 99717

Then try to append a new value, but this time choose as string instead of number.
Then choose save button save.

Create item ×

Tree ▾

↕

⊞

🔍

▼

▲

▼ Item {4}

⊕

order_id

Number : 630178

⊕

line_number

Number : 1

⊕

product_id

Number : 99717

⊕

status

String : unshipped

Cancel

Save

Then create all items as specified on the above screen shot.

If you want to delete an item, choose the item, select Delete under Actions drop down menu.

The screenshot shows the 'order_line_items' table interface. On the left, there is a 'Create table' button and an 'Actions' dropdown. Below this is a search bar 'Filter by table name' and a list of tables: 'order_line_items' (selected) and 'orders'. On the right, the 'order_line_items' table is displayed with tabs for 'Overview', 'Items', 'Metrics', 'Alarms', 'Capacity', 'Indexes', and 'Access control'. The 'Items' tab is active, showing a 'Create item' button and an 'Actions' dropdown. The 'Actions' dropdown is open, showing options: 'Duplicate', 'Edit', 'Delete' (highlighted with a red box), and 'Export to .csv'. Below the dropdown is a 'Start search' button. The table itself has columns: 'order_id', 'line_number', 'product_id', and 'status'. It contains three rows of data:

order_id	line_number	product_id	status
607161	1	98379	shipped
630178	2	90127	unshipped
630178	1	99717	unshipped

To edit the item, select the item, then choose pencil symbol which is hovering near item value.

The screenshot shows the 'order_line_items' table interface. At the top, there is a 'Create item' button and an 'Actions' dropdown. Below this is a search bar 'Scan: [Table] order_line_items: order_id, line_number' and a 'Start search' button. The table itself has columns: 'order_id', 'line_number', 'product_id', and 'status'. It contains three rows of data:

order_id	line_number	product_id	status
607161	1	98379	shipped
630178	2	90127	unshipped
630178	1	99717	unshipped

The third row is selected, and a pencil icon is visible next to the 'unshipped' status, indicating the edit function.

Then change the item value and choose save.

Create item

Actions ▾

Scan: [Table] order_line_items: order_id, line_number ^ Viewing 1

Scan ▾ [Table] order_line_items: order_id, line_number ▾ ^

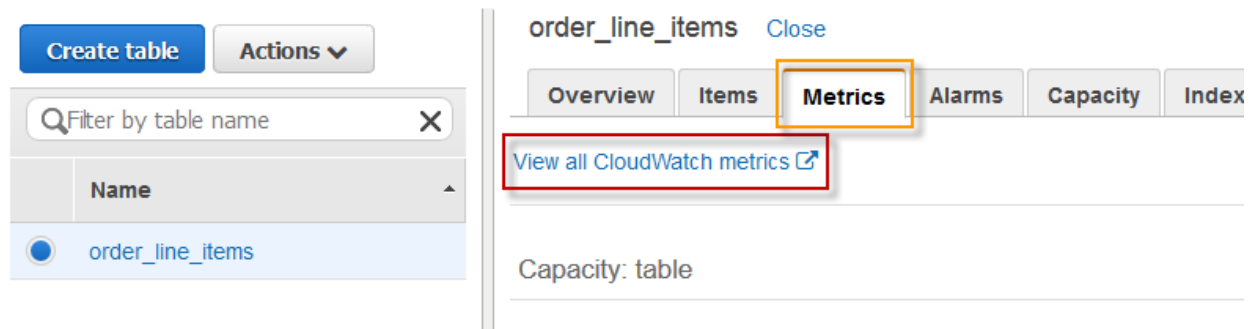
+ Add filter

Start search

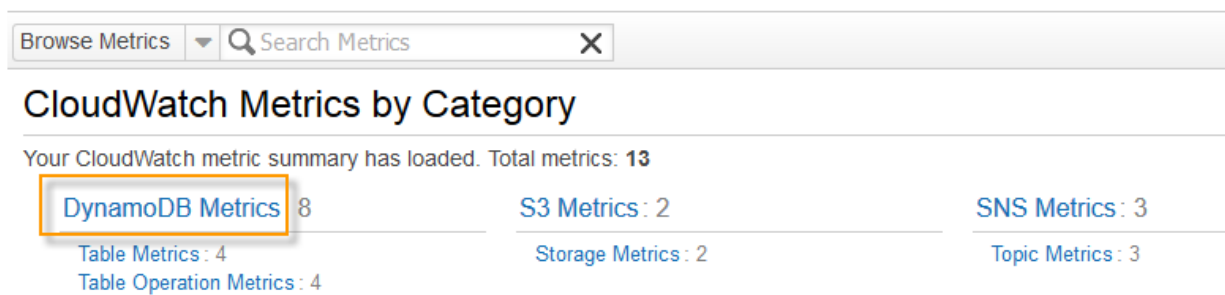
<input type="checkbox"/>	order_id	line_number	product_id	status	
<input type="checkbox"/>	607161	1	98379	shipped	
<input type="checkbox"/>	630178	2	90127	unshipped	
<input checked="" type="checkbox"/>	630178	1	99717	unshipped	<div><div>shipped</div><div>Cancel Save</div></div>

CONFIGURING ALARMS FOR DYNAMODB

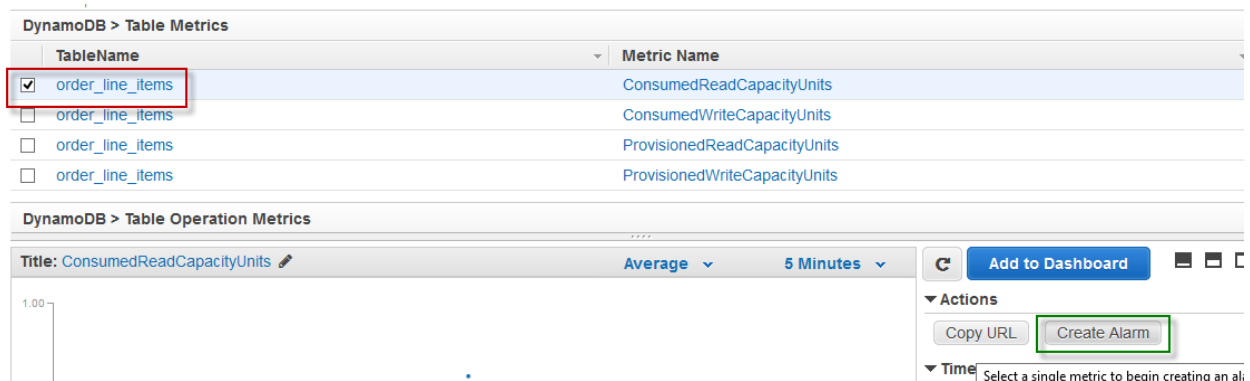
Select your table, choose metrics from right pane, and then choose View all cloudwatch metrics.



Under CloudWatch, choose DynamoDB Metrics.



Choose an Alarm, choose Create Alarm from below right pane.



Specify a name and add a description, then configure RCU as per your requirements.

Choose SNS topic, then click on Create Alarm.

Create Alarm

1. Select Metric2. Define Alarm

appropriate threshold.

Name:

Description:

Whenever: ConsumedReadCapacityUnits

is

for: consecutive period(s)

Actions

Define what actions are taken when your alarm changes state.

Notification Delete

Whenever this alarm:

Send notification to: New list Enter list

Email list:

Namespace: AWS/DynamoDB

TableName:

Metric Name:

Period:

Statistic:

to or above the red line for a duration of 5 minutes

ConsumedReadCapacityUnits >= 200

Time	ConsumedReadCapacityUnits
5/07 12:00	~100
5/07 13:00	~100
5/07 14:00	~100

CancelPreviousNextCreate Alarm