



# DynamoDB Table With CloudFormation

01.11.2022

---

Rani Nabwani

AppleSeeds

AWS re/Start BootCamp

## Overview

Creating a DynamoDB Table Using AWS CloudFormation.

I will be using AWS CloudFormation to write the state of the infrastructure using code, which includes a DynamoDB table to hold inventory items.

## Goals

1. Review and Download the CloudFormation Template.
2. Launch the CloudFormation Stack.
3. Verify the DynamoDB Table Was Created.

The screenshot displays the AWS Management Console interface. The top section, 'Recently visited', lists various AWS services including IAM, VPC, EC2, S3, Amazon SageMaker, CloudFormation, Cloud9, CloudWatch, Lambda, DynamoDB, Config, Key Management Service, Amazon Transcribe, and Step Functions. A 'View all services' link is at the bottom of this list. To the right, the 'Welcome to AWS' sidebar offers links for 'Getting started with AWS', 'Training and certification', and 'What's new with AWS?'. Below this, the 'CloudFormation > Stacks' page is shown. It features a 'Stacks (1)' header with a search filter, a 'View nested' toggle, and a status dropdown set to 'Active'. A table lists the stacks, with one stack named 'cfst-3427-373fac50a86bf76782369758c33f5948' in a 'CREATE\_COMPLETE' status, created on '2022-10-31 23:38:10 UTC+0200'. The description for this stack is 'Cloudformation template with 1'.

Stack name	Status	Created time	Description
cfst-3427-373fac50a86bf76782369758c33f5948	CREATE_COMPLETE	2022-10-31 23:38:10 UTC+0200	Cloudformation template with 1

```
35 lines (31 sloc) 859 Bytes
1 AWSTemplateFormatVersion: 2010-09-09
2 Description: CloudFormation Template To Create Alfredo's Inventory DynamoDB Table
3
4 Resources:
5   InventoryTable:
6     Type: AWS::DynamoDB::Table
7     Properties:
8       TableName: Inventory
9       AttributeDefinitions:
10        -
11          AttributeName: !Ref HashKeyElementName
12          AttributeType: !Ref HashKeyElementType
13       KeySchema:
14        -
15          AttributeName: !Ref HashKeyElementName
16          KeyType: HASH
17
18       ProvisionedThroughput:
19         ReadCapacityUnits: 5
20         WriteCapacityUnits: 5
21
22 Parameters:
23   HashKeyElementName:
24     Type: String
25     Default: InventoryId
26     Description: Hash Key Name
27   HashKeyElementType:
28     Type: String
29     Default: S
30     Description: Hash Key Type
31
32 Outputs:
33   Inventory:
34     Description: Alfredo's Inventory Table
35     Value: !Ref InventoryTable
```

[Give feedback](#)

## Create stack

### Prerequisite - Prepare template

#### Prepare template

Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

☒ Template is ready

☐ Use a sample template

☐ Create template in Designer

### Specify template

A template is a JSON or YAML file that describes your stack's resources and properties.

#### Template source

Selecting a template generates an Amazon S3 URL where it will be stored.

☐ Amazon S3 URL

☒ Upload a template file

#### Upload a template file

*acg-dynamodb-template.yaml.txt*

JSON or YAML formatted file

S3 URL: <https://s3.us-east-1.amazonaws.com/cf-templates-jw764bwgwgij-us-east-1/2022-10-31T214906.985Zak3-acg-dynamodb-template.yaml.txt>

[View in Designer](#)

[Cancel](#)

[Next](#)

## Specify stack details

### Stack name

Stack name

dynamodb-stack

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

### Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

HashKeyElementName

Hash Key Name

InventoryId

HashKeyElementType

Hash Key Type

S

Cancel

Previous

Next

## Review dynamodb-stack

### Step 1: Specify template

Edit

### Template

Template URL

https://s3.us-east-1.amazonaws.com/cf-templates-jw764bwgwgij-us-east-1/2022-10-31T214906.985Zak3-acg-dynamodb-template.yaml.txt

Stack description

CloudFormation Template To Create Alfredo's Inventory DynamoDB Table

### Step 2: Specify stack details

Edit

### Parameters

Q Search parameters

Key	Value
HashKeyElementType	S
HashKeyElementName	InventoryId

## dynamodb-stack

Delete Update Stack actions ▼ Create stack ▼

Stack info **Events** Resources Outputs Parameters Template Change sets

### Events (5)

Search events

Timestamp	Logical ID	Status	Status reason
2022-10-31 23:51:06 UTC+0200	dynamodb-stack	✔ CREATE_COMPLETE	-
2022-10-31 23:51:04 UTC+0200	InventoryTable	✔ CREATE_COMPLETE	-
2022-10-31 23:50:51 UTC+0200	InventoryTable	ⓘ CREATE_IN_PROGRESS	Resource creation Initiated
2022-10-31 23:50:49 UTC+0200	InventoryTable	ⓘ CREATE_IN_PROGRESS	-
2022-10-31 23:50:42 UTC+0200	dynamodb-stack	ⓘ CREATE_IN_PROGRESS	User Initiated

## DynamoDB > Tables

### Tables (1)

Find tables by table name Any table tag

< 1 > ⚙

	Name	Status	Partition key	Sort key	Indexes	Read capacity mode	Write capacity mode	Size	Table class
<input type="checkbox"/>	Inventory	✔ Active	InventoryId (S)	-	0	Provisioned (5)	Provisioned (5)	0 bytes	DynamoDB Standard

## DynamoDB > Tables > Inventory

### Inventory

Overview **Indexes** Monitor Global tables Backups Exports and streams Additional settings

General information

Partition key InventoryId (String)	Sort key -	Capacity mode Provisioned	Table status ✔ Active ✔ No active alarms
---------------------------------------	---------------	------------------------------	------------------------------------------------

► Additional info

Items summary

DynamoDB updates the following information approximately every six hours.

Get live item count

Item count 0	Table size 0 bytes	Average item size 0 bytes
-----------------	-----------------------	------------------------------