

# AWS Project

ระบบแจ้งเตือนราคา Crypto อัตโนมัติ  
(Serverless Crypto Tracker)

# Project

มีการใช้ Service ดังนี้



AWS Lambda



Amazon DynamoDB



AWS KMS



Amazon EventBridge



Amazon SNS

- Search DynamoDB

The screenshot shows the AWS DynamoDB service page. At the top, there's a search bar with 'dynamoDB' typed in. On the left, a sidebar has 'Services' selected. The main area features a large 'DynamoDB' card with a blue icon, the text 'Managed NoSQL Database', and a 'Create table' button.

## Get started

Create a new table to start exploring DynamoDB.

**Create table**

### 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.

PriceTrackingConfig

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

productName

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.

Enter the sort key name

String

1 to 255 characters and case sensitive.

#### Customize settings

Use these advanced features to make DynamoDB work better for your needs.

#### Encryption key management

##### AWS owned key [Learn more ↗](#)

The key is owned and managed by DynamoDB. You are not charged additional fees for using this key.

##### AWS managed key [Learn more ↗](#)

The key is stored in your account and managed by AWS Key Management Service (KMS). AWS KMS charges apply.

##### Customer managed key [Learn more ↗](#)

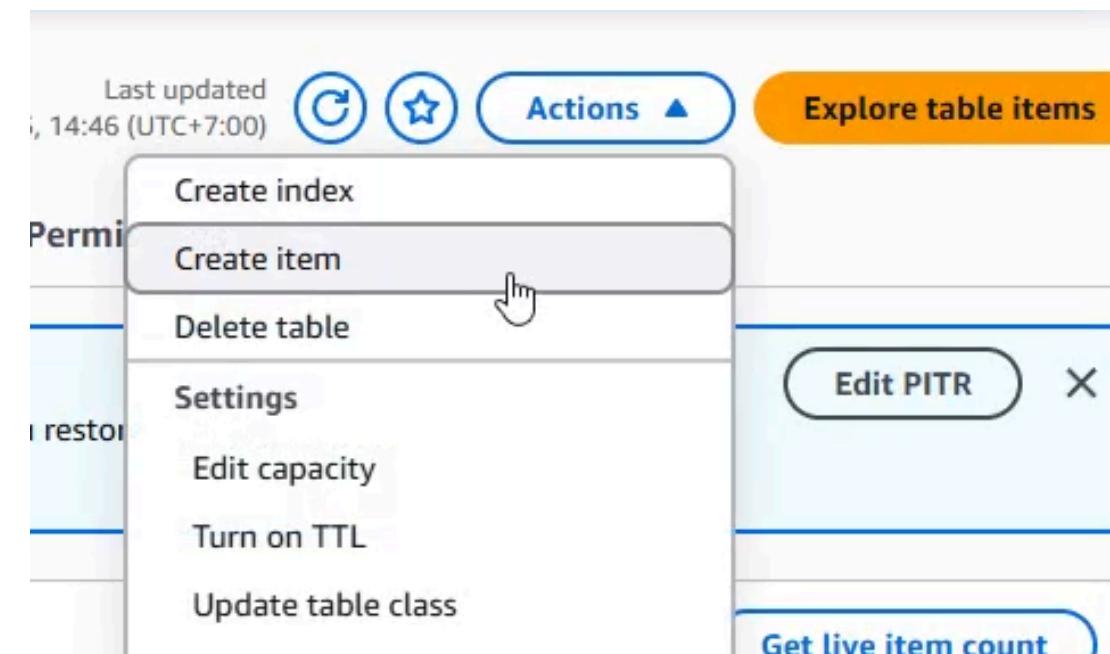
The key is stored in your account and managed by you. AWS KMS charges apply.



คลิกเข้าไป : PriceTrackingConfig

Tables (1) <a href="#">Info</a>						
<input type="checkbox"/>	Name	Status	Partition key	Sort key	Indexes	Replication Regions
<input type="checkbox"/>	<a href="#">PriceTrackingConfig</a>	<span>Active</span>	productName (S)	-	0	0

คลิก Actions > Create item



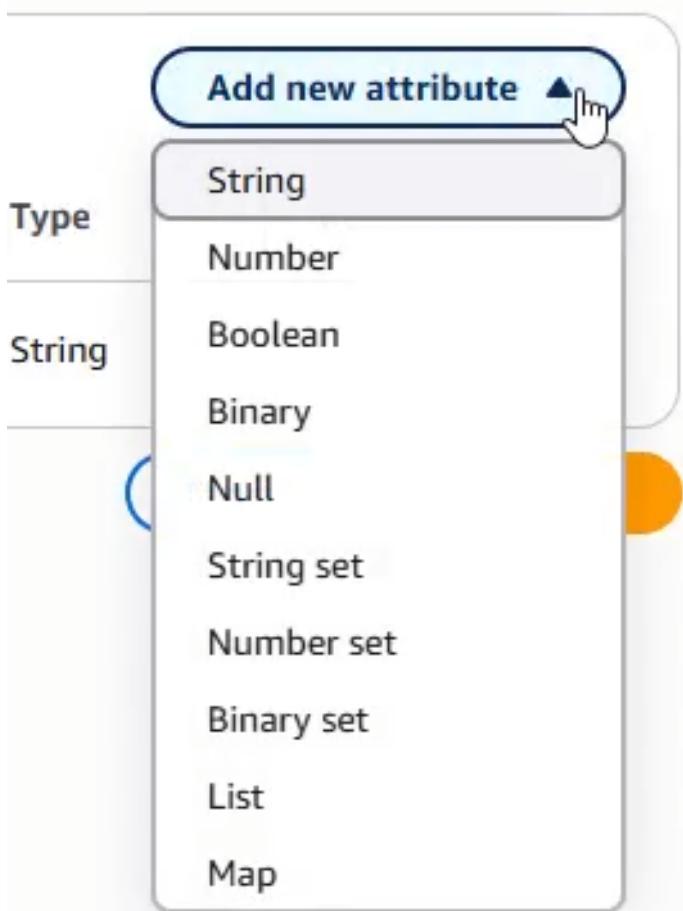
**Create item**

You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. [Learn more](#)

Attributes		Type	Remove
Attribute name	Value		
productName - Partition key	Bitcoin	String	
emailAddress	mintada.anek@bumail.net	String	<a href="#">Remove</a>
targetPrice	100000	Number	<a href="#">Remove</a>
lastCheckPrice	0	Number	<a href="#">Remove</a>

[Add new attribute ▾](#)

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



## Add new attribute

productName: [ Bitcoin ] String

emailAddress: [ Emailของคุณ ] String

targetPrice: [ 100000 ] Number

lastCheckPrice :[ 0 ] Number

คลิก : Create item

Number

[Remove](#)

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

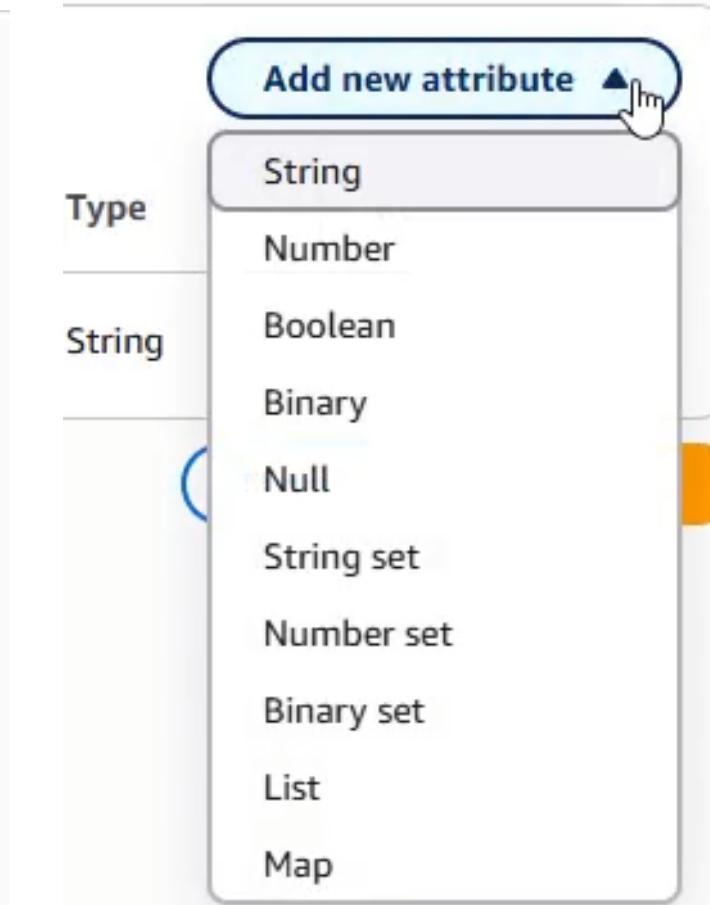
The item has been saved successfully.

### Create item

You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. [Learn more ↗](#)

Attributes		Type	Remove
+ Attribute name	Value		Add new attribute ▾
productName - Partition key	Ethereum	String	
emailAddress	mintada.anek@bumail.net	String	<a href="#">Remove</a>
targetPrice	4000	Number	<a href="#">Remove</a>
lastCheckPrice	0	Number	<a href="#">Remove</a>

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



## Add new attribute

productName:[ Ethereum ] String

emailAddress:[ Emailของคุณ ] String

targetPrice:[ 4000 ] Number

lastCheckPrice :[ 0 ] Number

คลิก : Create item

Number [Remove](#)

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

The item has been saved successfully.

## Create item

You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. [Learn more ↗](#)

Attributes		Type	Add new attrib
Attribute name	Value	Type	Remove
productName - Partition key	Dogecoin	String	
emailAddress	mintada.anek@bumail.net	String	
targetPrice	1	Number	
lastCheckPrice	0	Number	

## Add new attribute

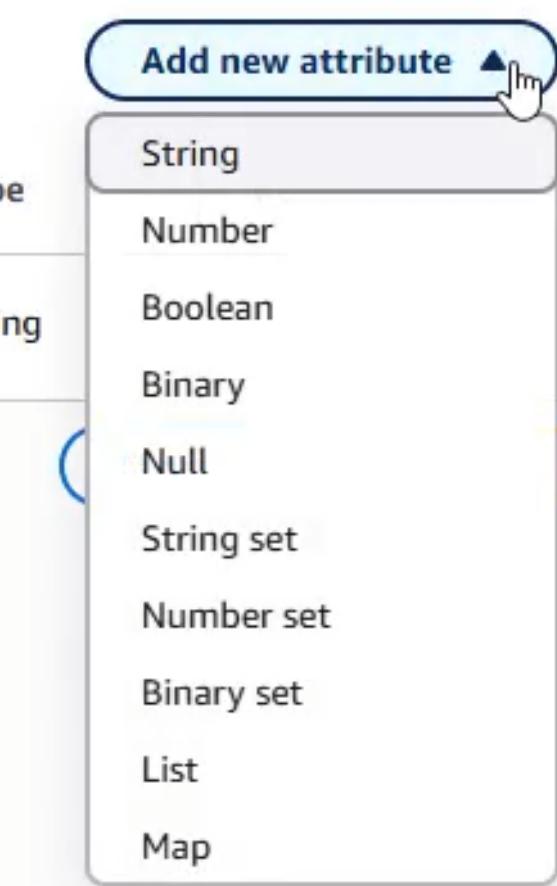
productName: [ Dogecoin ] String

emailAddress:[ Emailของคุณ ] String

targetPrice: [ 1 ] Number

lastCheckPrice : [ 0 ] Number

คลิก : Create item



Number

The item has been saved successfully.

X

## Create item

You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. [Learn more ↗](#)

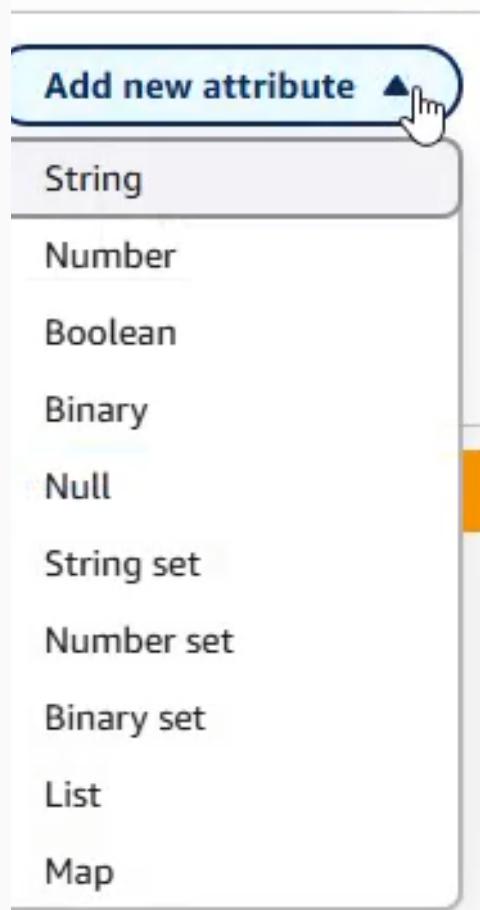
Form | JSON view

**Attributes**

Attribute name	Value	Type	Remove
productName - Partition key	Solana	String	
emailAddress	mintada.anek@bumail.net	String	<a href="#">Remove</a>
targetPrice	300	Number	<a href="#">Remove</a>
lastCheckPrice	0	Number	<a href="#">Remove</a>

[Add new attribute ▾](#)

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



## Add new attribute

productName: [ Solana ] String

emailAddress: [ Emailของคุณ ] String

targetPrice: [ 300 ] Number

lastCheckPrice : [ 0 ] Number

คลิก : Create item

Number [Remove](#)

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

ผลลัพธ์

Table: PriceTrackingConfig - Items returned (4)

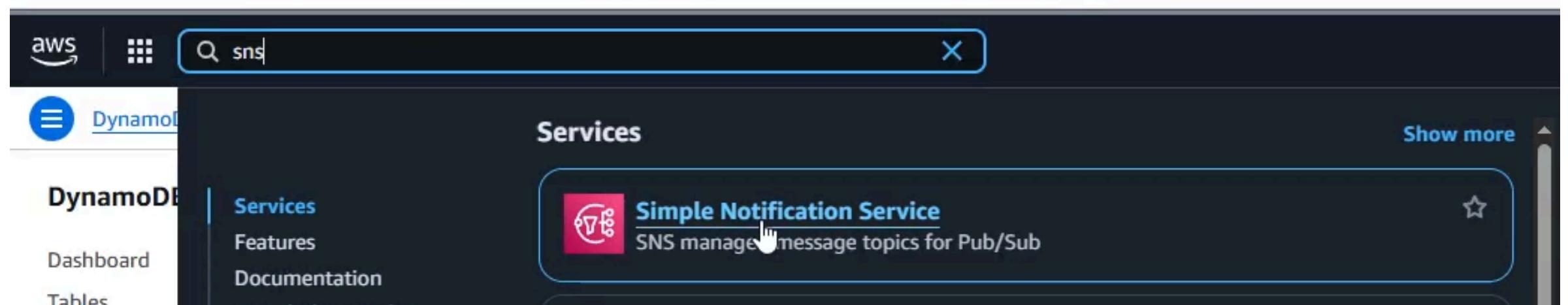
Scan started on November 22, 2025, 14:49:56

C Actions ▾ Create item

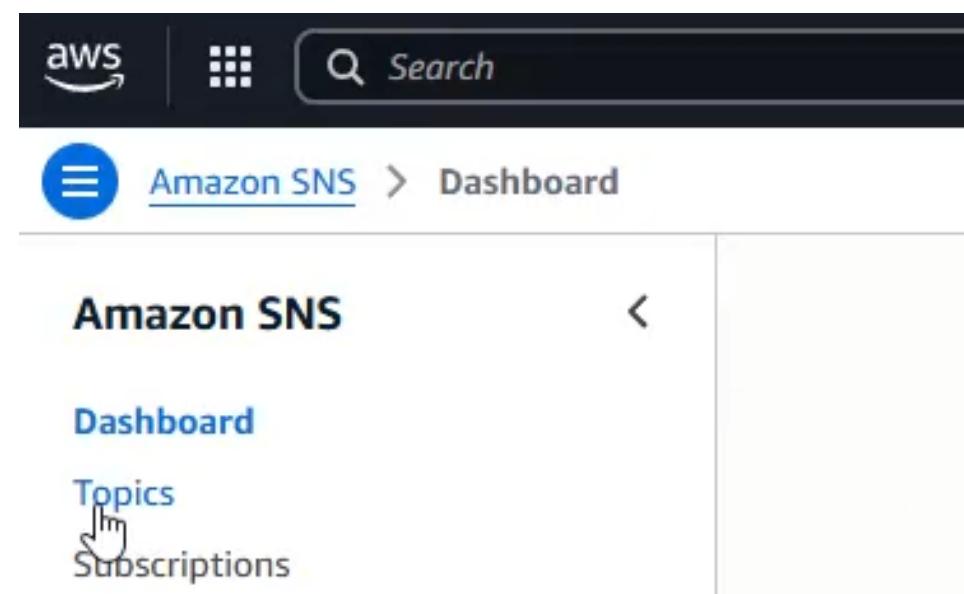
< 1 > | ⚙

	productName (String)	emailAddress	lastCheckPrice	targetPrice
	Solana	mintada.anek...	0	300
	Dogecoin	mintada.anek...	0	1
	Ethereum	mintada.anek...	0	4000
	Bitcoin	mintada.anek...	0	100000

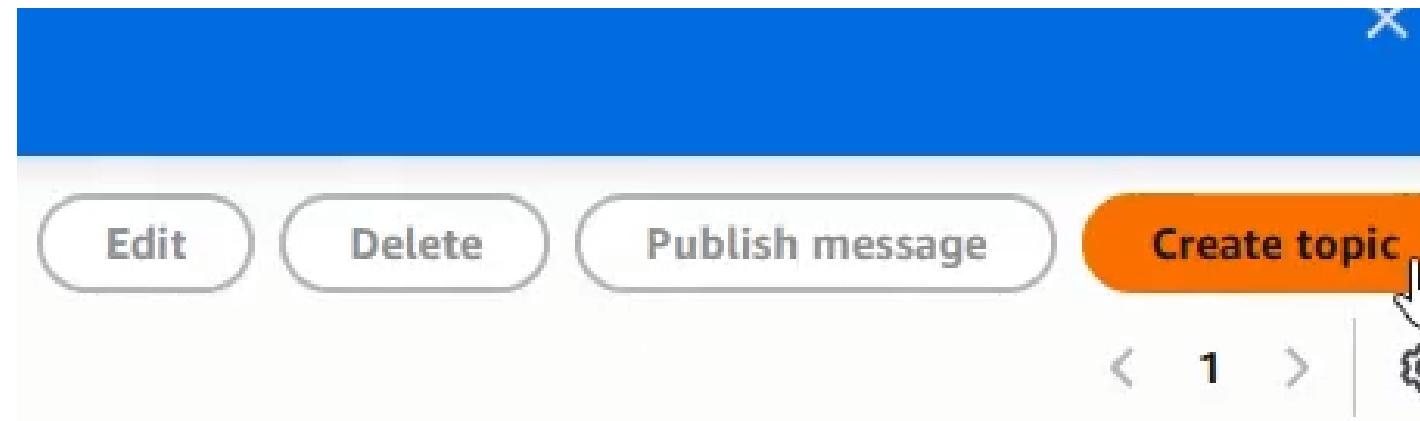
Search Amazon SNS  
Simple Notification Service



ไปที่ Topics



คลิก Create topic



Details type เลือก : Standard  
Name: PriceAlertTopic  
คลิก Create topic

**Create topic**

**Details**

Type | Info Topic type cannot be modified after topic is created

FIFO (first-in, first-out)

- Strictly-preserved message ordering
- Exactly-once message delivery
- Subscription protocols: SQS

Standard

- Best-effort message ordering
- At-least once message delivery
- Subscription protocols: SQS, Lambda, Data Firehose, HTTP, SMS, email, mobile application endpoints

**Name**

PriceAlertTopic

Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (\_).

**Display name - optional** | Info To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message.

My Topic

Maximum 100 characters.

คัดลอก ARN เก็บไว้

Protocol: เลือก Email

Endpoint: Email ของคุณ

คลิก : Create subscription

เปิดEmailที่กรอกไว้

The screenshot shows the AWS SNS console with the topic 'PriceAlertTopic' selected. In the top right corner, there is a small blue circular icon with a white 'E' and the number '11'. The main interface displays the 'Details' section of the topic, including its ARN (arn:aws:sns:us-east-1:381492142099:PriceAlertTopic), name, topic owner, and display name. Below the details, there are tabs for 'Encryption', 'Tags', and 'Integrations'. A prominent orange button labeled 'Create subscription' is highlighted with a mouse cursor. Below this button, a modal window titled 'Create subscription' is open, showing fields for 'Topic ARN' (set to the topic's ARN), 'Protocol' (set to 'Email'), and 'Endpoint' (set to 'mintada.anek@bumail.net'). A note at the bottom of the modal says, 'After your subscription is created, you must confirm it.' At the bottom of the modal, there are two optional sections: 'Subscription filter policy - optional' and 'Redrive policy (dead-letter queue) - optional'. To the right of the 'Protocol' field, a dropdown menu lists other protocol options: 'Amazon Kinesis Data Firehose', 'Amazon SQS', 'AWS Lambda', 'Email' (which is selected and highlighted with a mouse cursor), and 'Email-JSON'.

# คลิก Confirm subscription

AWS Notification - Subscription Confirmation External Inbox x

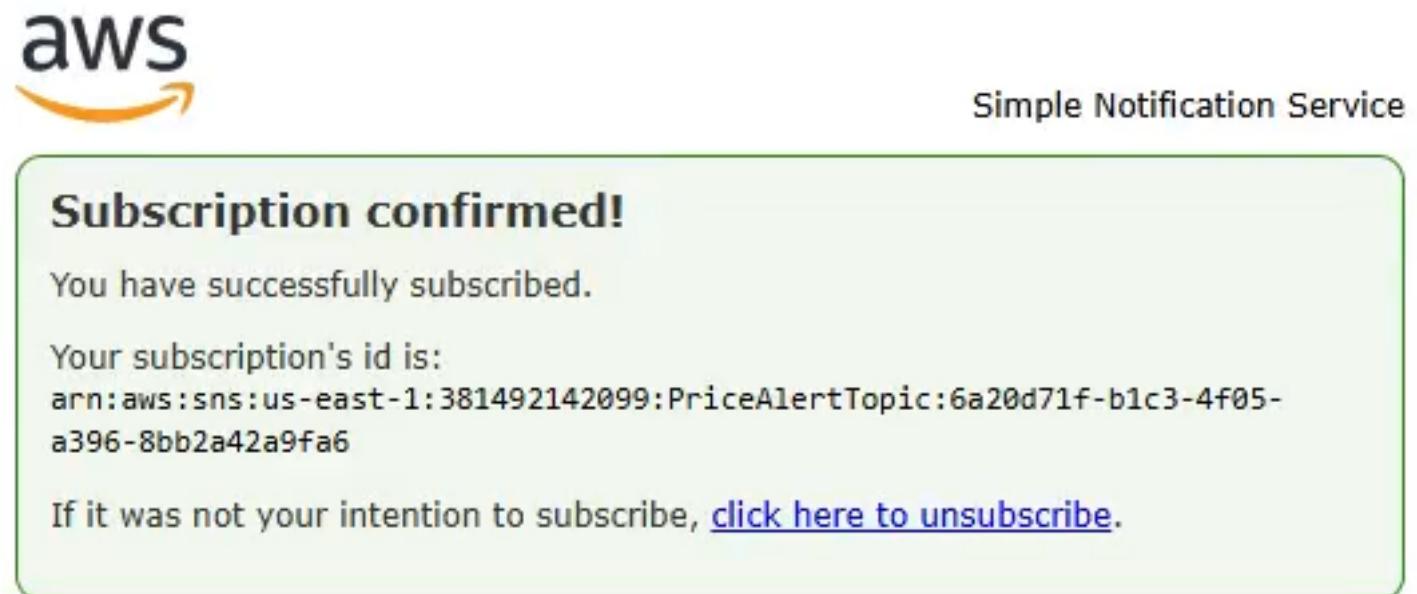
 AWS Notifications <no-reply@sns.amazonaws.com>  
to me ▾

You have chosen to subscribe to the topic:  
`arn:aws:sns:us-east-1:381492142099:PriceAlertTopic`

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):  
[Confirm subscription](#)

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to [sns-opt-out](#)

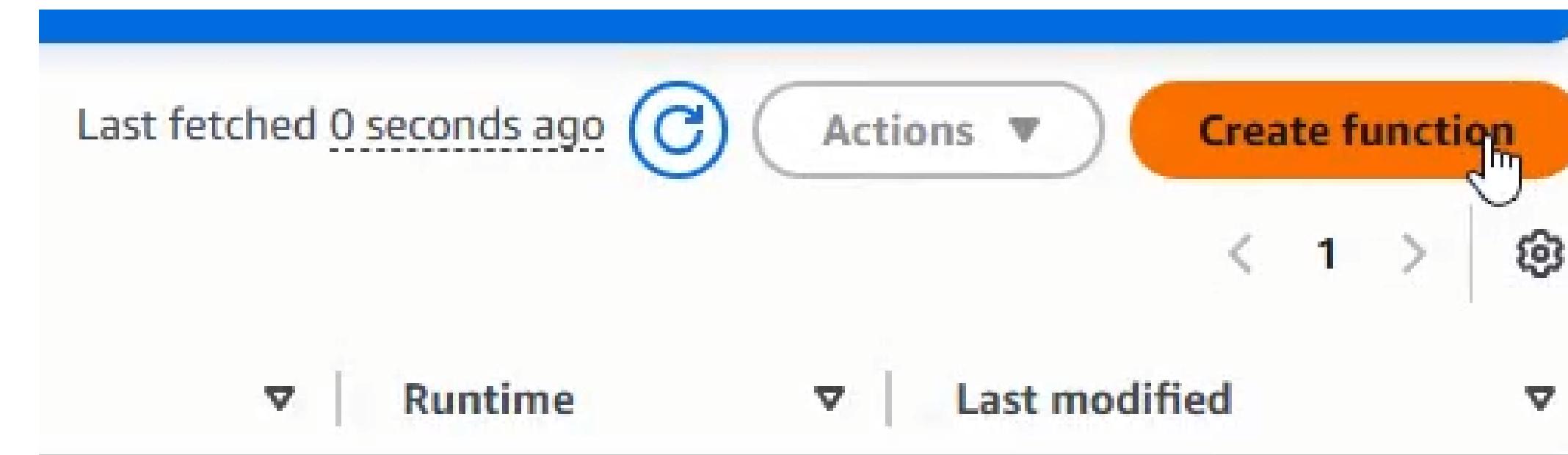
[Reply](#) [Forward](#)



- Search Lambda



คลิก Create function



เลือก: Author from scratch

Function name: PriceCheckerFunction

Runtime: Python 3.12

Architecture: x86\_64

กด **Change default execution role**

เลือก : Use an existing role

Existing role เลือก : LabRole

คลิก : Create function

PriceCheckerFunction

Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, !

Runtime | [Info](#)  
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.

Python 3.12

Architecture | [Info](#)  
Choose the instruction set architecture you want for your function code.

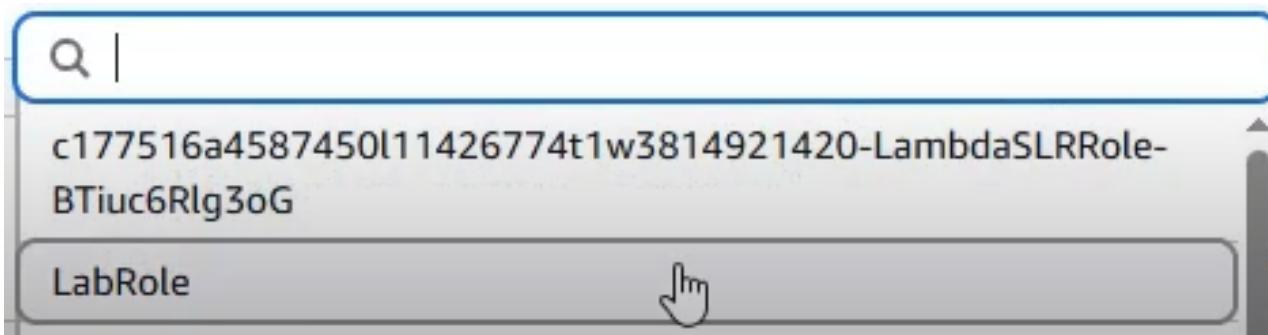
arm64  
 x86\_64

Permissions [Info](#)  
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. [IAM console](#)

▼ Change default execution role

Execution role  
Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#).

Create a new role with basic Lambda permissions  
 Use an existing role  
 Create a new role from AWS policy templates

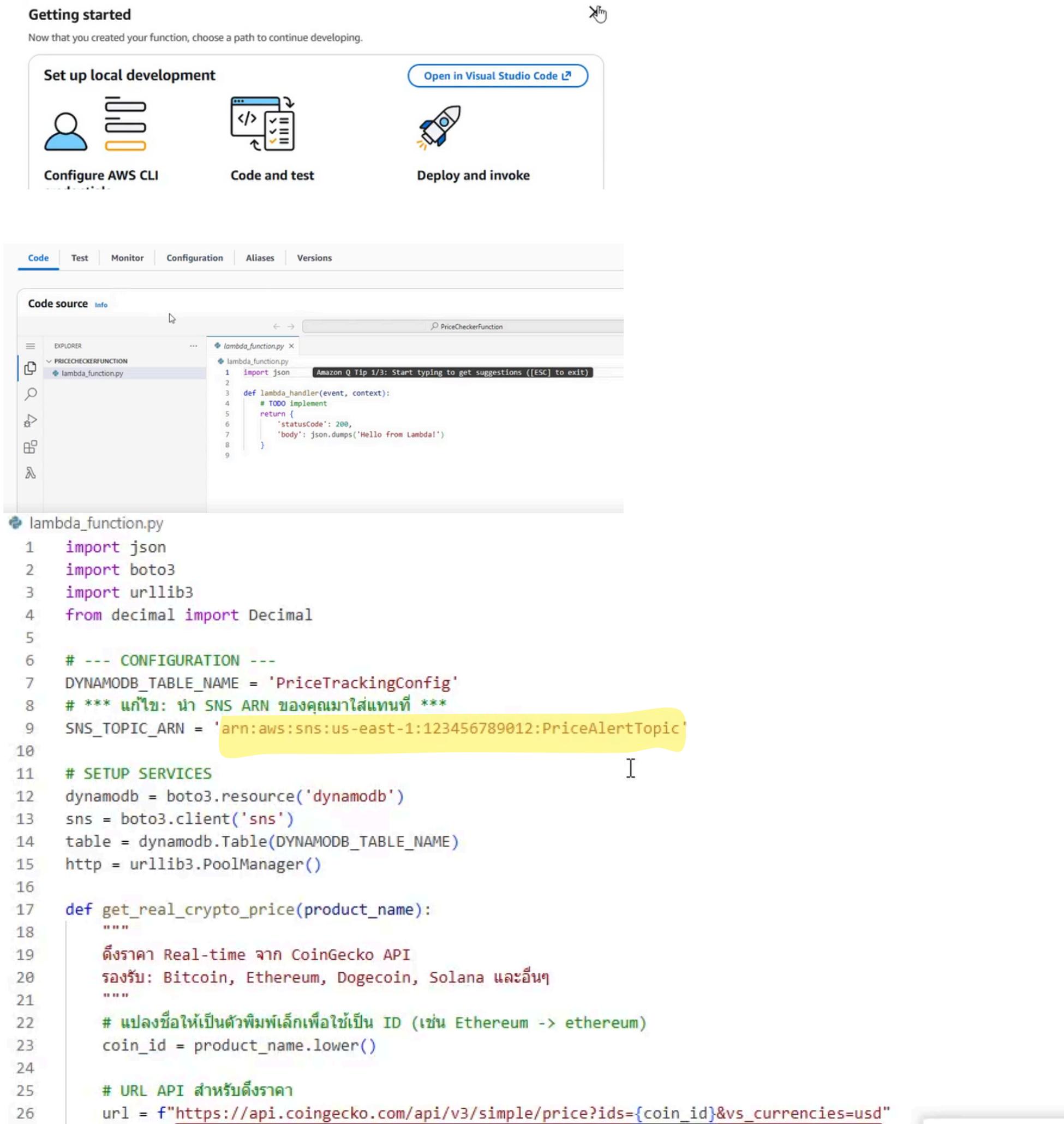


คลิก ปุ่ม กากบาท

ดาวน์โหลดไฟล์

Lamda1.py

วาง ARN ที่คัดลอก  
เก็บไว้



Getting started

Now that you created your function, choose a path to continue developing.

Set up local development

Configure AWS CLI

Code and test

Deploy and invoke

Open in Visual Studio Code

Code | Test | Monitor | Configuration | Aliases | Versions

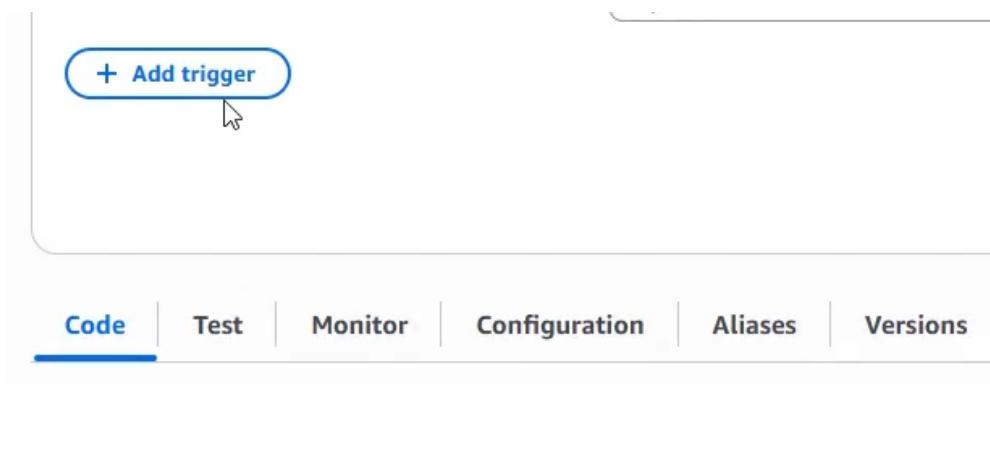
Code source Info

EXPLORER PRICECHECKERFUNCTION lambda\_function.py

```
lambda_function.py
1 import json
2
3 def lambda_handler(event, context):
4     # TODO implement
5     return {
6         'statusCode': 200,
7         'body': json.dumps('Hello from Lambda!')
8     }
```

```
lambda_function.py
1 import json
2 import boto3
3 import urllib3
4 from decimal import Decimal
5
6 # --- CONFIGURATION ---
7 DYNAMODB_TABLE_NAME = 'PriceTrackingConfig'
8 # *** แก้ไข: นำ SNS ARN ของคุณมาใส่แทนที่ ***
9 SNS_TOPIC_ARN = 'arn:aws:sns:us-east-1:123456789012:PriceAlertTopic'
10
11 # SETUP SERVICES
12 dynamodb = boto3.resource('dynamodb')
13 sns = boto3.client('sns')
14 table = dynamodb.Table(DYNAMODB_TABLE_NAME)
15 http = urllib3.PoolManager()
16
17 def get_real_crypto_price(product_name):
18     """
19         ดึงราคา Real-time จาก CoinGecko API
20         รองรับ: Bitcoin, Ethereum, Dogecoin, Solana และอื่นๆ
21     """
22     # แปลงชื่อให้เป็นตัวพิมพ์เล็กเพื่อใช้เป็น ID (เช่น Ethereum -> ethereum)
23     coin_id = product_name.lower()
24
25     # URL API สำหรับดึงราคา
26     url = f"https://api.coingecko.com/api/v3/simple/price?ids={coin_id}&vs_currencies=usd"
```

## กด Add trigger



## Add trigger

### Trigger configuration Info

Select a source

event

Batch/bulk data processing

EventBridge (CloudWatch Events)

aws asynchronous schedule management-tools

Lambda > Add triggers

### Add trigger

**Trigger configuration** Info

EventBridge (CloudWatch Events)  
aws asynchronous schedule management-tools

**Rule**  
Pick an existing rule, or create a new one.  
 Create a new rule  
 Existing rules

**Rule name**  
Enter a name to uniquely identify your rule.  
PriceCheckerSchedule

**Rule description**  
Provide an optional description for your rule.  
Run every 1 hour

**Rule type**  
Trigger your target based on an event pattern, or based on an automated schedule.  
 Event pattern  
 Schedule expression

**Schedule expression**  
Self-trigger your target on an automated schedule using [Cron or rate expressions](#). Cron expressions are in UTC.  
rate(1 hour)

e.g. rate(1 day), cron(0 17 ? \* MON-FRI \*)

Lambda will add the necessary permissions for Amazon EventBridge (CloudWatch Events) to invoke your Lambda function from this trigger. [Learn more](#) about the Lambda permissions model.

[Cancel](#) [Add](#)

Select : EventBridge (CloudWatch Events)

Rule เลือก : Create a new rule

Rule name : PriceCheckerSchedule

Rule description : Run every 1 hour

Rule type : Schedule expression

Schedule expression : rate(1 hour)

คลิก : Add

คลิก : Deploy

Code source [Info](#)

EXPLORER

PRICECHECKERFUNCTION

lambda\_function.py

DEPLOY Undeployed

Deploy (Ctrl+Shift+U) 

λ PriceCheckerFunction

```
lambda_function.py x
lambda_function.py
1 import json
2 import boto3
3 import urllib3
4 from decimal import Decimal
5
6 # --- CONFIGURATION ---
7 DYNAMODB_TABLE_NAME = 'PriceTrackingConfig'
8 # *** แก้ไข: นำ SNS ARN ของคุณมาใส่แทนที่ ***
9 SNS_TOPIC_ARN = 'arn:aws:sns:us-east-1:381492142099:PriceAlertTopic'
10
11 # SETUP SERVICES
```



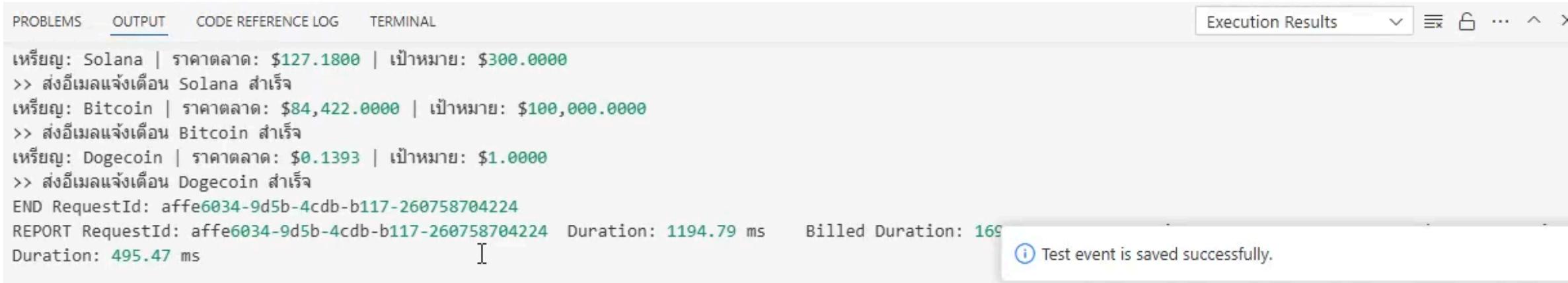
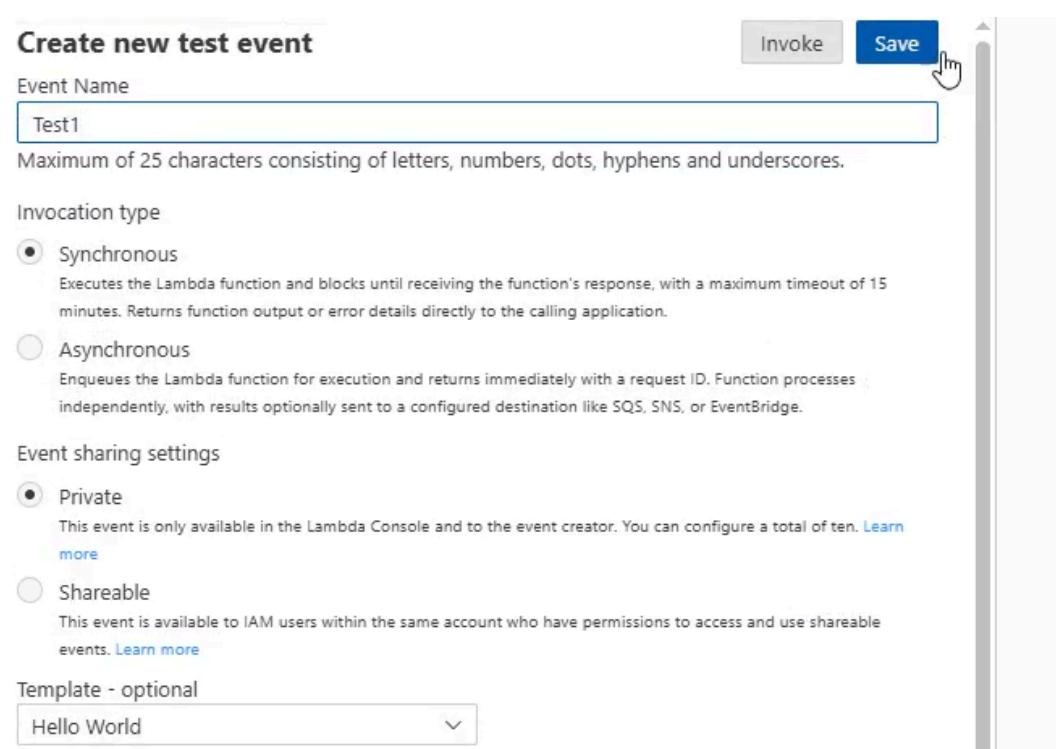
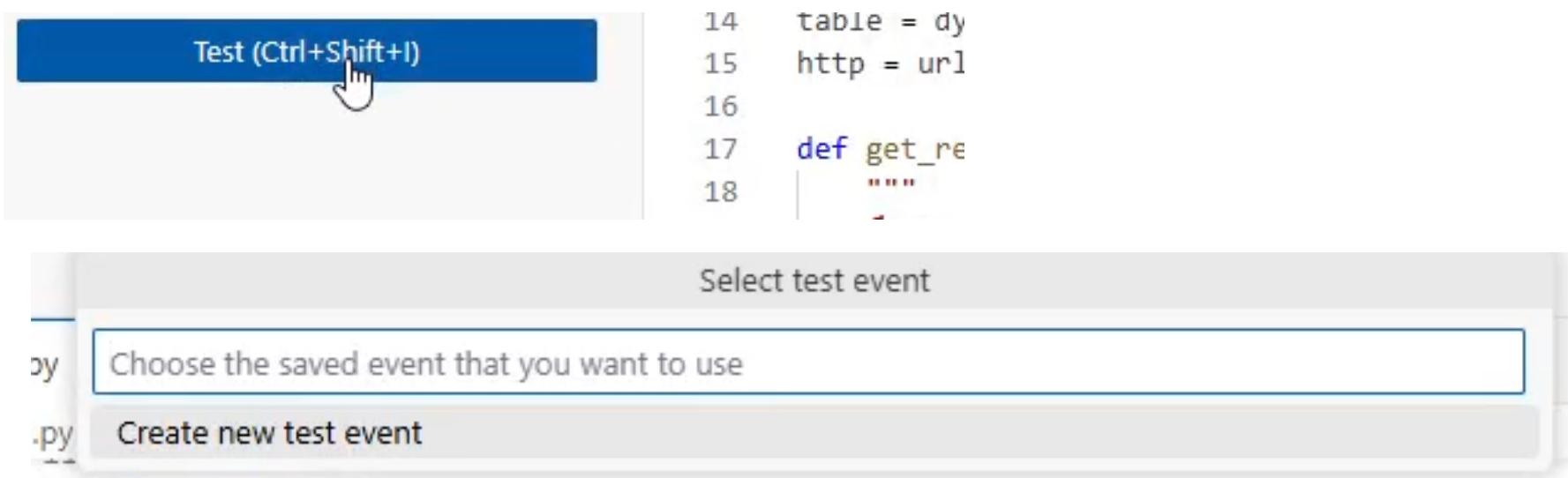
Successfully updated the function PriceCheckerFunction.

## คลิก: Create new test event

# Event Name : Test1

## คลิก : Save

# คลิก : Test



## • ผลลัพธ์

The image displays four separate email notifications from AWS Lambda, each titled with a different cryptocurrency's price alert. Each notification includes a recipient icon labeled "AWS Notifications" and "to me". A yellow "External" button and a grey "Inbox" button are visible at the top of each message header. A prominent callout box in each message states "It looks like this message is in Thai" with a "Translate to English" link. Below the callout, each message contains a "CRYPTO ALERT!" heading followed by the current price and the previous day's price.

**Price Alert: Ethereum** (External, Inbox)

AWS Notifications to me

It looks like this message is in Thai [Translate to English](#)

CRYPTO ALERT! Ethereum ราคาแตะเป้าหมายแล้ว!  
ราคาปัจจุบัน: \$2,746.7000 USD  
ราคาเป้าหมาย: \$4,000.0000 USD

**Price Alert: Dogecoin** (External, Inbox)

AWS Notifications to me

It looks like this message is in Thai [Translate to English](#)

CRYPTO ALERT! Dogecoin ราคาแตะเป้าหมายแล้ว!  
ราคาปัจจุบัน: \$0.1393 USD  
ราคาเป้าหมาย: \$1.0000 USD

**Price Alert: Solana** (External, Inbox)

AWS Notifications to me

It looks like this message is in Thai [Translate to English](#)

CRYPTO ALERT! Solana ราคาแตะเป้าหมายแล้ว!  
ราคาปัจจุบัน: \$127.1800 USD  
ราคาเป้าหมาย: \$300.0000 USD

**Price Alert: Bitcoin** (External, Inbox)

AWS Notifications to me

It looks like this message is in Thai [Translate to English](#)

CRYPTO ALERT! Bitcoin ราคาแตะเป้าหมายแล้ว!  
ราคาปัจจุบัน: \$84,422.0000 USD  
ราคาเป้าหมาย: \$100,000.0000 USD