

# 9주차 बैं킹 봇 구축

- <https://github.com/aws-samples/aws-ai-ml-workshop-kr/tree/master/aisservices/lex-korean-workshop>

[aws-ai-ml-workshop-kr](#) / [aisservices](#) / [lex-korean-workshop](#) / [BankingBot.md](#) 

 gonsoomoon-ml edit lex readme file 

Preview

Code

Blame

206 lines (117 loc) · 7.97 KB

## 뱅크 봇 구축 워크숍

### 개요

앞서서는 Amazon Lex 콘솔만을 사용하여 간단한 [HelloWorldBot](#)을 만들어 보았습니다. 이번에는 좀 더 복잡한 작업을 하는 봇을 구현해 보겠습니다. 이번 섹션에서는 AWS CloudFormation도 함께 살펴 보겠습니다.

### 워크숍 리소스 생성

# 봇생성

← → 🏠 📌 ↺ https://ap-northeast-2.console.aws.amazon.com/lexv2/home?region=ap-northeast-2#createBot

🔖 북마크 가져오기... ⚙ 자주 방문 순 🔒 Firefox 시작하기

aws 서비스 🔍 검색 [알트+S] 🗪

☰ 2단계 언어 추가

### 생성 방법

- ☒ 빈 봇을 생성합니다.  
사전 구성된 언어, 인텐트 및 슬롯 유형이 없는 기본 봇을 생성합니다.
- ☐ 예제로 시작  
예제 봇에는 미리 구성된 언어, 의도 및 슬롯 유형이 있습니다. 이러한 설정을 변경할 수 있습니다.
- ☐ 트랜스크립트로 시작  
업로드한 대화 트랜스크립트에서 의도를 자동으로 생성합니다. 트랜스크립트로 시작하는 경우 영어(미국) 변형만 사용할 수 있습니다.

### 봇 구성

봇 이름

BookTrip 또는 OrderFlowers

최대 100자입니다. 유효한 문자는 A~Z, a~z, 0~9, -, \_, .입니다.

설명 - 선택 사항

이 설명은 봇 목록 페이지에 나타납니다. 봇의 용도를 식별하는 데 도움이 될 수 있습니다.


북미 사무소의 직원들을 위한 IT HelpDesk 봇입니다.

최대 200자.

### 런타임 역할

봇에 대한 권한을 정의하는 역할을 선택합니다. 사용자 지정 역할을 생성하려면 IAM 콘솔을 사용합니다.

- ☒ 기본 Amazon Lex 권한으로 역할을 생성합니다.
- ☐ 기존 역할을 사용합니다.

 역할을 생성하는 데 몇 분 정도 걸립니다. 역할 생성을 마칠 때까지 역할을 삭제하거나 이 역할의 신뢰 또는 권한 정책을 편집하지 마세요.

### 새 역할

Amazon Lex는 Amazon CloudWatch Logs에 업로드할 수 있는 권한이 있는 런타임 역할을 생성합니다.

AWSServiceRoleForLexV2Bots\_PL4L698162

### COPPA(Children's Online Privacy Protection Act) [Info](#)

봇 사용에 다음 적용 여부 [COPPA\(Children's Online Privacy Protection Act\)](#) ?

- ☐ 예
- ☒ 아니요

# 언어선택

[알트+S]

서울 ▼

봇에 언어 추가

Info

▼ 언어: 한국어(KR)

언어 선택

한국어(KR)

설명 - 선택 사항

최대 200자.

음성 상호 작용

봇이 사용자와 상호 작용하는 데 사용하는 텍스트 투 스피치 음성입니다.

Seoyeon

음성 샘플

안녕하세요 제 이름은 Seoyeon 입니다. 무엇을 도와드릴까요?

재생

의도 분류 신뢰도 점수 임계값

0.40

# 의도 이름

The screenshot displays the Amazon Lex console interface. At the top, the AWS logo and '서비스' (Services) are visible. A search bar contains the text '검색' (Search). The main header shows 'Amazon Lex' with a close button. Below this, a green banner indicates '봇 BankingDemo 생성 성공' (Bot BankingDemo creation successful). The breadcrumb navigation path is 'Lex > 봇 > 봇: BankingDemo > 버전 > 버전: Draft > 모든 언어 > 언어: 한국어(KR) > 의도 > 의도: NewIntent'. The left sidebar contains a search bar, a '검색' (Search) button, and a '마지막 업데이트 날짜순으로...' (Sort by last update date...) dropdown. The main content area shows the '의도: NewIntent' (Intent: NewIntent) configuration page. It includes a 'Draft version' dropdown, a '한국어(KR)' language selector, and a '구축되지 않음' (Not built) button. A blue information box states '시작을 위한 의도를 추가했습니다.' (Added intent for start). Below this, there are sections for '대화 흐름' (Conversation flow) and '의도 세부 정보' (Intent details). The '의도 세부 정보' section shows the '의도 이름' (Intent name) field with the value 'NewIntent' and a note: '최대 100자입니다. 유효한 문자는 A~Z, a~z, 0~9, -, \_입니다.' (Maximum 100 characters. Valid characters are A~Z, a~z, 0~9, -, \_). At the bottom, there are tabs for 'Editor' and 'Visual builder', with a 'New' button next to the 'Visual builder' tab.

aws 서비스 검색 [알트+S]

Amazon Lex ×

✓ 봇 BankingDemo 생성 성공

< 의도 목록으로 돌아가기 (2)

검색

마지막 업데이트 날짜순으로... ▼

NewIntent 저장되지 않음

FallbackIntent

Lex > 봇 > 봇: BankingDemo > 버전 > 버전: Draft > 모든 언어 > 언어: 한국어(KR) > 의도 > 의도: NewIntent

Draft version ▼ 한국어(KR) ▼ 구축되지 않음 ⓘ 한국어(KR) ha

## 의도: NewIntent Info

의도는 사용자의 요청을 이행하는 작업을 나타냅니다. 의도는 변수 정보를 나타내는 슬롯이라는 인수를 가질 수 있습니다.

ⓘ 시작을 위한 의도를 추가했습니다. ×

▶ 대화 흐름 Info

▼ 의도 세부 정보 Info

의도 이름

NewIntent

최대 100자입니다. 유효한 문자는 A~Z, a~z, 0~9, -, \_입니다.

Editor Visual builder New

# 봇 구성



# 의도 이름 설정

✔ 봇 BankingDemo 생성 성공

Draft version ▼

한국어(KR) ▼

구축되지 않음

▶ 대화 흐름 [Info](#)

▼ 의도 세부 정보 [Info](#)

의도 이름

✓ Checkbalance

최대 100자입니다. 유효한 문자는 A~Z, a~z, 0~9, -, \_입니다.

설명 - 선택 사항

최대 200자.

ID: RZPPKWHYOW

# 샘플표현, 빌드

The screenshot shows the Amazon Lex console interface for a bot named 'BankingDemo'. At the top, a green banner indicates '봇 BankingDemo 생성 성공' (Bot BankingDemo creation successful). Below this, there are controls for the draft version, language (Korean), and a 'Build' button. The 'Build' button is circled in red, with a red arrow pointing to it from the text '한국어(KR) has not built changes.' To the right of the 'Build' button is a 'Test' button. The main content area shows '샘플 표현 (1)' (Sample Expression (1)) with an 'Info' link. Below this is a search bar with the placeholder '필터' and a dropdown menu for sorting by '추가된 날짜순으로 정렬(오름차순)'. There are also tabs for '평가판' (Preview) and '일반 텍스트' (General Text). At the bottom, a blue message bubble with a red checkmark says '계좌를 조회하고 싶습니다.' (I want to check my account).

봇 BankingDemo 생성 성공

Draft version ▼ 한국어(KR) ▼ 구축되지 않음

한국어(KR) has not built changes. Build Test

샘플 표현 (1) Info

이 의도를 호출하기 위해 사용자가 말하거나 입력할 것으로 예상되는 대표적인 구문입니다. Amazon Lex는 샘플과 다를 수 있는 사용자 입력을 해석하기 위해 샘플 표현을 기반으로 추정합니다. 샘플 표현의 우선순위는 의도 분류 출력을 결정하는 데 사용되지 않습니다.

Q 필터 추가된 날짜순으로 정렬(오름차순) ▼

평가판 일반 텍스트

✓ 계좌를 조회하고 싶습니다.



# 슬롯유형, 프롬프트

요청을

슬롯 추가

×

슬롯은 의도를 이행하기 위해 사용자로부터 정보를 캡처하는 데 사용됩니다.

☒ 이 의도에 필요함

사용자가 값을 제공하지 않으면 대화 중 봇이 이 슬롯을 묻는 프롬프트를 표시합니다.

이름

슬롯 유형

✓ BankAccount

✓ AMAZON.Number ▼

프롬프트

✓ 계좌번호 알려주세요.

취소

추가

# 슬롯추가

를 표시합니다.

슬롯 추가

×

슬롯은 의도를 이행하기 위해 사용자로부터 정보를 캡처하는 데 사용됩니다.

☒ 이 의도에 필요함

사용자가 값을 제공하지 않으면 대화 중 봇이 이 슬롯을 묻는 프롬프트를 표시합니다.

이름

슬롯 유형

BankName

슬롯 유형 선택 ▼

프롬프트

어떤 은행 계좌인가요?

취소

추가

ation

사용자가 의

확인하리

메시지

티브

...

슬롯을 묻는 프롬프트: BankName

슬롯 유형

✕

메시지: 어떤 은행 계좌인가요?

☒ 이 의도에 필요함

사용자가 값을 제공하지 않으면 대화 중 봇이 이 슬롯을 묻는 프롬프트를 표시합니다.

이름

BankName

슬롯 유형

슬롯 유형 선택

프롬프트

어떤 은행 계좌인가요?

고급 옵션 설정을 사용하여 사용자 지정 페이로드, 카드 그룹, SSML 등의 서식 있는 메시지를 구성할 수 있습니다.

☒ 고급 옵션

## 슬롯 유형 값

슬롯 값을 인식하도록 기계 학습 모델을 훈련하는 데 사용되는 값 목록을 수정합니다.

Q 슬롯 유형 값 검색

아마존은행

×

AWS은행

×

한국은행

×

값

✓ 값 추가

최대 140자입니다. 유효한 문자는 A~Z, a~z, 0~9, @, #, \$입니다.

✓ 슬롯 유형 저장

# 슬롯유형 설정

Draft version ▼

한국어(KR) ▼

성공적으로 구축됨

한국어(K

Filter

슬롯을 묻는 프롬프트: BankName

슬롯 유형

BankName

✕

☒ 이 의도에 필요함

사용자가 값을 제공하지 않으면 대화 중 봇이 이 슬롯을 묻는 프롬프트를 표시합니다.

이름

BankName

프롬프트

어떤 은행 계좌인가요?

고급 옵션 설정을 사용하여 사용자 지정 페이로드, 카드 그룹, SSML

고급 옵션

슬롯 유형

BankName ✓ ▲

Q

사용자 지정 슬롯 유형

BankName

기본 제공 슬롯 유형

Confirmation Info

○ 액티브

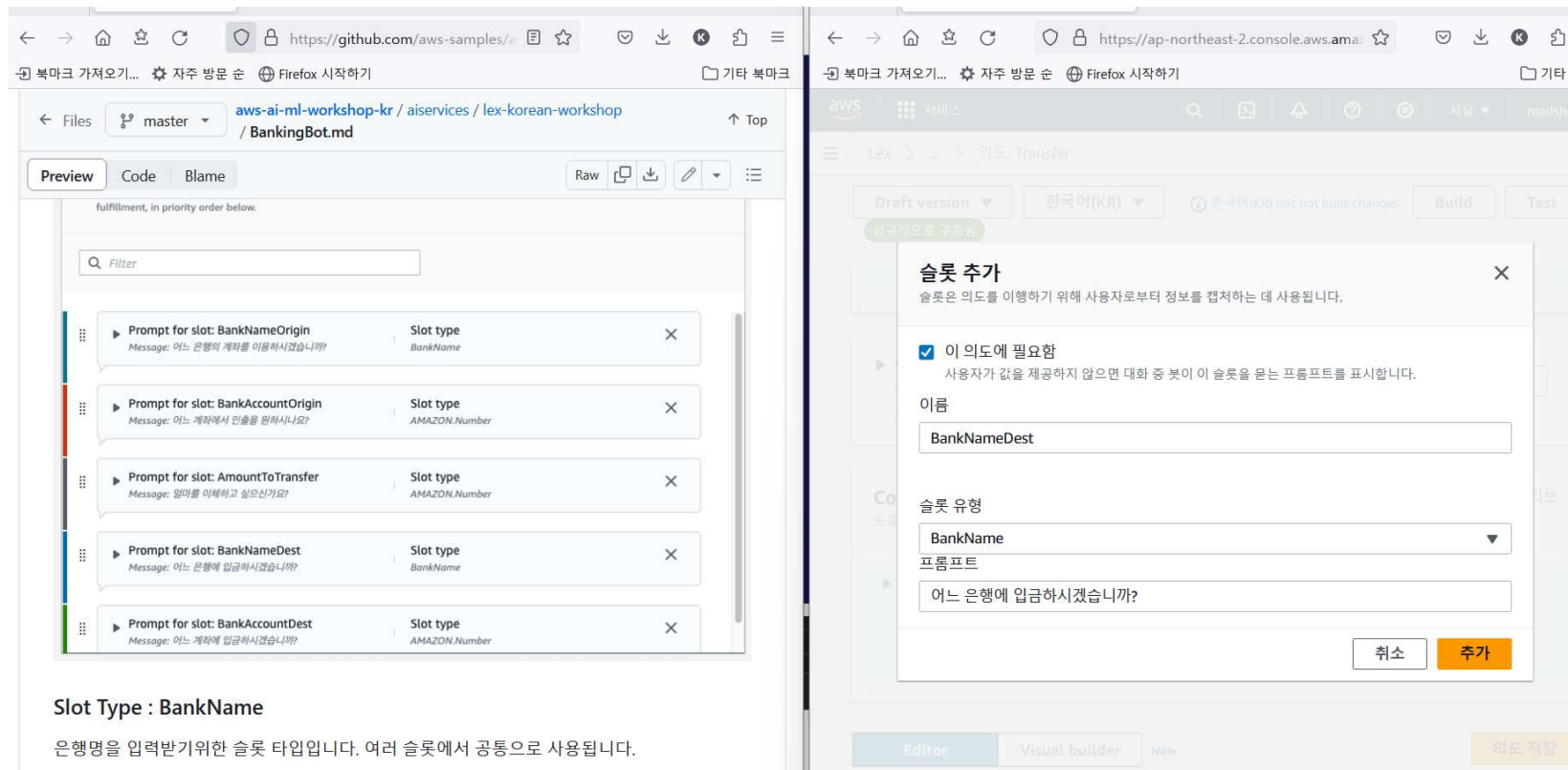
# Build

계좌를 조회하고 싶습니다.

최대 250자입니다.

의도저장 -> 빌드

# Transfer의도 슬롯추가



The left screenshot shows a GitHub repository for `aws-ai-ml-workshop-kr / aiseservices / lex-korean-workshop / BankingBot.md`. It displays a list of prompts for the `BankingBot` intent, each with a slot type and a message. The prompts are:

- Prompt for slot: BankNameOrigin (Slot type: BankName, Message: 어느 은행의 계좌를 이용하시겠습니까?)
- Prompt for slot: BankAccountOrigin (Slot type: AMAZON.Number, Message: 어느 계좌에서 인출을 원하시나요?)
- Prompt for slot: AmountToTransfer (Slot type: AMAZON.Number, Message: 얼마를 이체하고 싶으신가요?)
- Prompt for slot: BankNameDest (Slot type: BankName, Message: 어느 은행에 입금하시겠습니까?)
- Prompt for slot: BankAccountDest (Slot type: AMAZON.Number, Message: 어느 계좌에 입금하시겠습니까?)

The right screenshot shows the AWS Lex console interface for the `Transfer` intent. It displays a modal for adding a slot, with the following details:

- Slot name: `BankNameDest`
- Slot type: `BankName`
- Prompt: `어느 은행에 입금하시겠습니까?`

Below the screenshots, the text reads:

**Slot Type : BankName**  
은행명을 입력받기위한 슬롯 타입입니다. 여러 슬롯에서 공통으로 사용됩니다.

# Labda

- V2에서는 람다함수 1개



# BankingDemo

✔ 봇 한국어(KR)에서 언어 BankingDemo 구축 성공

Lex > 봇 > 봇: BankingDemo

## BankingDemo

배포를 위한 버전 및 별칭 생성 [Info](#)



봇과 별칭의 스냅샷을 생성할 버전을 정의하여 클라이언트 애플리케이션에서 사용하는 버전을 설정합니다. 채널을 추가하여 애플리케이션, 모바일 앱 또는 Facebook Messenger와 같은 메시지 플랫폼에 봇을 배포합니다.

별칭

별칭 보기

✔ 별칭 1개

채널

채널 추가

# 별칭 생성

Lex > 봇 > 봇: BankingDemo > 별칭

## 별칭 (1) [Info](#)

별칭은 특정 버전의 봇을 가리킵니다. 별칭을 사용하면 클라이언트 애플리케이션에서 사용하는 봇 버전을 업데이트.

🔍 Search 별칭 이름

별칭 이름	생성됨
<input type="radio"/> <a href="#">TestBotAlias</a>	55 분 전

별칭 이름

TestBotAlias

연결된 버전

[초안 버전](#)

ID: TSTALIASID

알림

test bot alias

감정 분석-

비활성화됨

## 언어 (1) [Info](#)

테스트

별칭의 언어 관리

별칭에서 활성화할 언어를 선택합니다. 구축된 언어만 활성화할 수 있습니다.

Search 언어

< 1 > 

언어

상태

☐ Korean (South Korea)

성공적으로 구축됨

## 대화 로그 [Info](#)

대화 로그 관리

봇과의 대화 세부 정보를 기록하려면 로깅을 활성화합니다. 오디오 로깅, 텍스트 로깅 또는 둘 다 활성화할 수 있습니다.

# 람다함수 생성

서비스

검색

[알트+S]

[Lambda](#) > [함수](#) > 함수 생성

## 함수 생성 [정보](#)

다음 옵션 중 하나를 선택하여 함수를 생성합니다.

☒ 새로 작성  
간단한 Hello World 예제는 시작하십시오.

☐ 블루프린트 사용  
샘플 코드 및 구축 Lambda 애플리케이션을 위한 구성 사전 설정을 일반적인 사용 사례를 살펴봅니다.

☐ 컨테이너 이미지  
함수에 대해 배포할 컨테이너 이미지를 선택합니다.

### 기본 정보

함수 이름  
함수의 용도를 설명하는 이름을 입력합니다.

bankBot

공백 없이 문자, 숫자, 하이픈 또는 밑줄만 사용합니다.

런타임 [정보](#)  
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

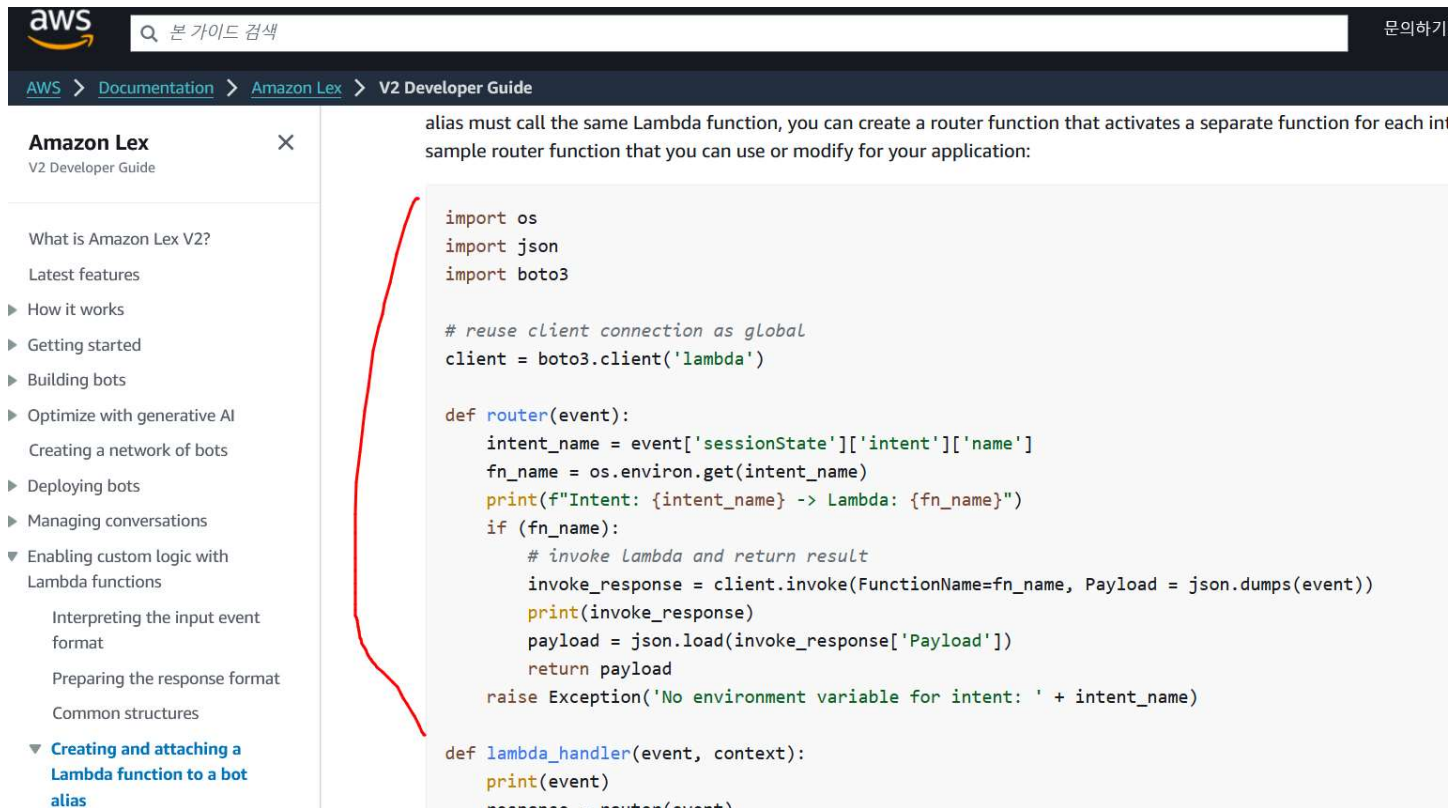
↻

아키텍처 [정보](#)  
함수 코드에 대해 원하는 명령 세트 아키텍처를 선택합니다.

☒ x86\_64

☐ arm64

# <https://docs.aws.amazon.com/lexv2/latest/dg/lambda-attach.html>



The screenshot shows the AWS documentation page for Amazon Lex V2 Developer Guide. The sidebar on the left contains the following navigation links:

- What is Amazon Lex V2?
- Latest features
- ▶ How it works
- ▶ Getting started
- ▶ Building bots
- ▶ Optimize with generative AI
  - Creating a network of bots
- ▶ Deploying bots
- ▶ Managing conversations
- ▼ Enabling custom logic with Lambda functions
  - Interpreting the input event format
  - Preparing the response format
  - Common structures
- ▼ **Creating and attaching a Lambda function to a bot alias**

The main content area shows the following text:

alias must call the same Lambda function, you can create a router function that activates a separate function for each intent. Here is a sample router function that you can use or modify for your application:

```
import os
import json
import boto3

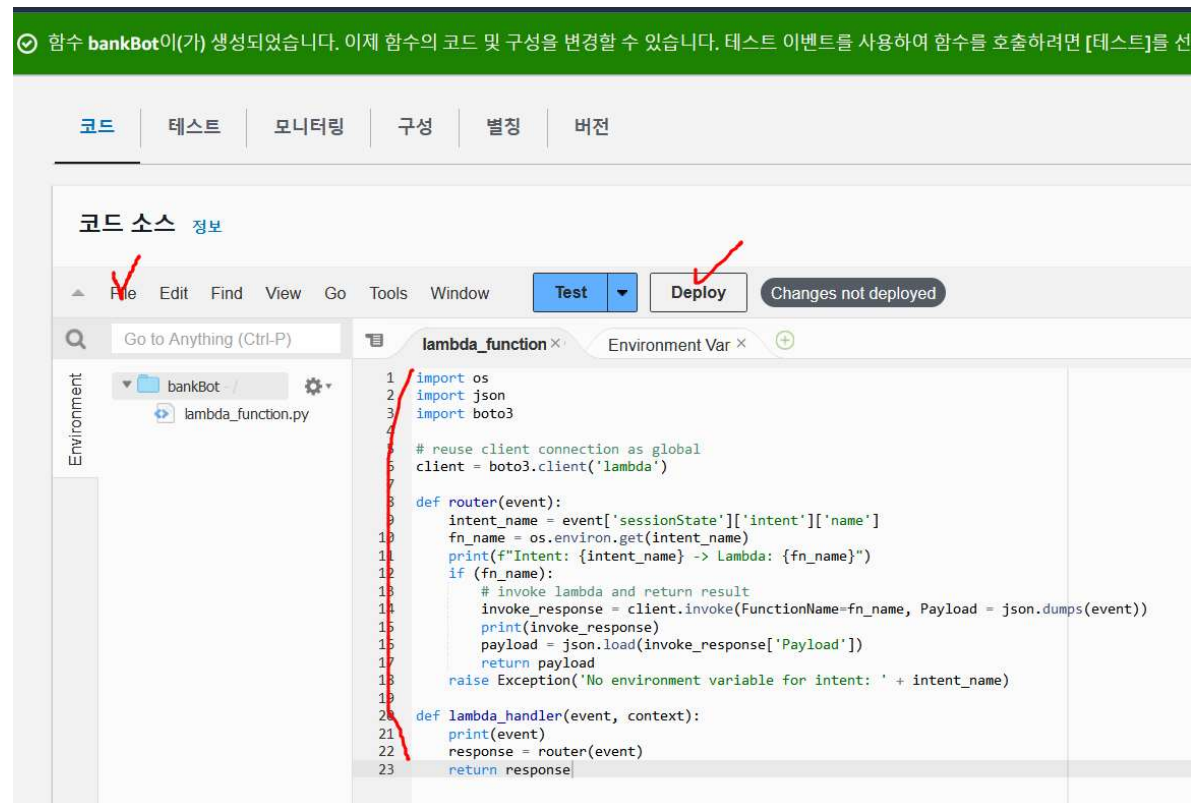
# reuse client connection as global
client = boto3.client('lambda')

def router(event):
    intent_name = event['sessionState']['intent']['name']
    fn_name = os.environ.get(intent_name)
    print(f"Intent: {intent_name} -> Lambda: {fn_name}")
    if (fn_name):
        # invoke Lambda and return result
        invoke_response = client.invoke(FunctionName=fn_name, Payload = json.dumps(event))
        print(invoke_response)
        payload = json.load(invoke_response['Payload'])
        return payload
    raise Exception('No environment variable for intent: ' + intent_name)

def lambda_handler(event, context):
    print(event)
    response = router(event)
```

# 코드복사

File-save  
Deploy



# bankBot 램다함수가 생성됨

Lex > 봇 > 봇: BankingDemo > 별칭 > 별칭: TestBotAlias > 별칭 언어 지원: 한국어(KR)

별칭 언어 지원: 한국어(KR)

▼ Lambda 함수 - 선택 사항

초기화, 검증 및 이행을 위해 Lambda 함수가 호출됩니다.

소스

▲

🔍

bankBot

IoT20240321

Lambda에 대해 자세히 알아보기

취소

저장