

# 10주차 Chatbot UI

- <https://github.com/aws-samples/aws-lex-web-ui>
- 코드 다운로드

https://github.com/aws-samples/aws-lex-web-ui

방문 순 Firefox 시작하기

README Code of conduct License Security

version 2.0.0 should be deployed prior to deploying lex web ui.

## Integrating into your Site and Deploying

In addition to the CloudFormation deployment mentioned above, there are other methods to integrate and deploy this project. Here is a summary of the various methods:

#	Method	Description	Use Case
1	<a href="#">CloudFormation Deployment</a> using the CloudFormation <a href="#">templates</a> provided by this project	Fully automated deployment of a hosted web application to an S3 bucket with an optional CI/CD pipeline. By default, it also creates a Cognito Identity Pool and a sample Lex bot	Use when you want to have a infrastructure as code approach that automatically builds and configures the chatbot UI resources
2	Use the pre-built <a href="#">libraries</a> from the <a href="#">dist</a> directory of this repo	We provide a pre-built version of the chatbot UI component and a loader library that you can use on your web site as a <a href="#">stand alone page</a> or as an embeddable <a href="#">iframe</a>	Use when you have an existing site and want to add the chatbot UI to it by simply copying or referencing the library files
3	Use npm to install and use the chatbot UI as a Vue component	Enables developers to consume this project as an <a href="#">npm</a> package that provides a <a href="#">Vue</a> component. See the <a href="#">Npm Install and Vue Component Use</a> section for details	Use when developing front-end based web applications built using JavaScript and bundled with tools such as <a href="#">webpack</a>

### Running Locally

This project provides a simple HTTP server to serve the sample site. You can run the server using [Node.js](#) on your local machine or a test server. Please note that running locally is only designed for testing purposes as the localhost only runs on HTTP and does not use a secure HTTPs configuration.

The chatbot UI requires proper configuration values in the files located under the [src/config](#) directory. Modify the values in the `lex-web-ui-loader-config.json` file under the `src/config` directory. If you deployed the demo site using the CloudFormation templates provided by this project, you can copy the automatically generated config files from the S3 buckets to your development host.

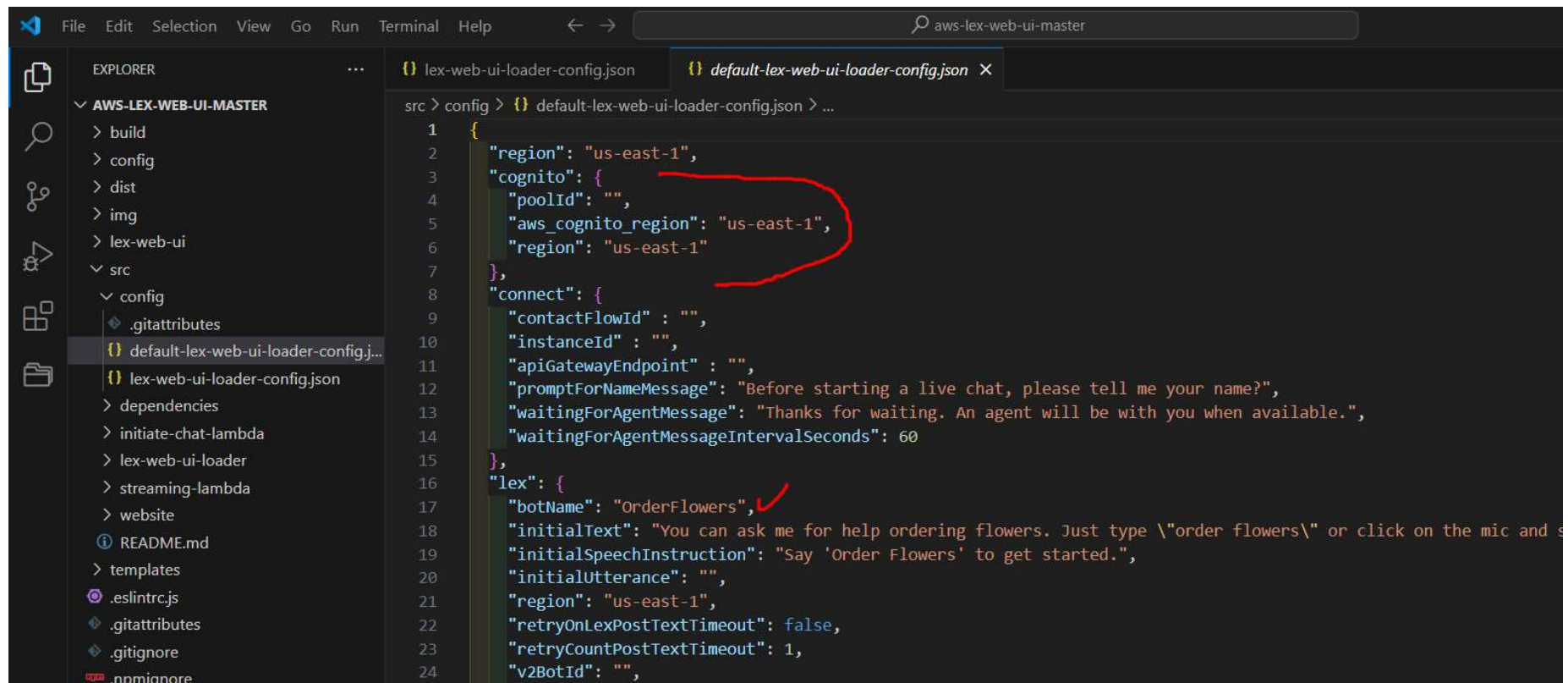
As a minimum, you would need to pass an existing Cognito Pool Id and Lex Bot name. For example, set the appropriate values in the `src/config/lex-web-ui-loader-config.json` file:

```
...
cognito: {
  "poolId": "us-east-1:deadbeef-fade-babe-cafe-0123456789ab"
},
lex: {
  "botName": "myHelpBot"
}
...
```

Before you run the local development server, you need to install the development dependencies with the command:

```
npm install
```

# cognito, botName



```
1 {
2   "region": "us-east-1",
3   "cognito": {
4     "poolId": "",
5     "aws_cognito_region": "us-east-1",
6     "region": "us-east-1"
7   },
8   "connect": {
9     "contactFlowId": "",
10    "instanceId": "",
11    "apiGatewayEndpoint": "",
12    "promptForNameMessage": "Before starting a live chat, please tell me your name?",
13    "waitingForAgentMessage": "Thanks for waiting. An agent will be with you when available.",
14    "waitingForAgentMessageIntervalSeconds": 60
15  },
16  "lex": {
17    "botName": "OrderFlowers",
18    "initialText": "You can ask me for help ordering flowers. Just type \"order flowers\" or click on the mic and s",
19    "initialSpeechInstruction": "Say 'Order Flowers' to get started.",
20    "initialUtterance": "",
21    "region": "us-east-1",
22    "retryOnLexPostTextTimeout": false,
23    "retryCountPostTextTimeout": 1,
24    "v2BotId": "",
```

# 사용자 풀 생성

The screenshot shows the AWS Management Console for Amazon Cognito in the ap-northeast-2 region. The main heading is 'Amazon Cognito' with the subtitle '앱에 대한 보안 자격 증명 및 액세스 관리'. Below this, a paragraph explains that Cognito user pools allow adding users and logins to an app, and that Cognito user pools provide AWS credentials for accessing cloud resources. On the right sidebar, under 'Business cases to get started', there is a dropdown menu 'Add user pool to app' and a paragraph explaining that Cognito user pools can be used to add security and authorization to an app, supporting up to 100,000 users. The 'Create user pool' button is highlighted with a red checkmark. At the bottom, there is a section for 'Fees' stating that Cognito user pools are free.

← → 🏠 🌟 ↻ 🔒 https://ap-northeast-2.console.aws.amazon.com/cognito/v2/home?region=ap-northeast-2#

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

aws 서비스 🔍 검색 [알트+S] 📧 🔔 ?

☰

보안, 자격 증명 및 규정 준수

## Amazon Cognito

### 앱에 대한 보안 자격 증명 및 액세스 관리

Amazon Cognito 사용자 풀을 사용하면 앱에 등록 및 로그인을 추가할 수 있습니다. Amazon Cognito 자격 증명 풀을 사용하면 클라우드 리소스에 액세스하기 위한 AWS 자격 증명을 제공할 수 있습니다.

**비즈니스 사례에서 시작**

앱에 사용자 디렉터리 추가 ▼

Amazon Cognito 사용자 풀은 앱에 보안 인증 및 권한 부여를 추가할 수 있는 관리형 서비스로서 수백만 명의 사용자를 지원하도록 확장할 수 있습니다.

**사용자 풀 생성**

**요금**

Cognito 자격 증명 풀은 무료입니다. Cognito 사

이점 및 기능

aws

서비스

검색

[알트+S]

고급 보안 기능으로 악의적인 로그인 시도로부터 프로덕션 사용자 계정을 보호할 수 있습니다. [앱 통합](#)에서 지금 활성화하세요. [자세히 알아보기](#)

2단계  
보안 요구 사항 구성

3단계  
가입 환경 구성

4단계  
메시지 전송 구성

5단계  
앱 통합

6단계  
검토 및 생성

인증 공급자

사용자가 로그인할 때 사용할 수 있는 공급자를 구성합니다.

공급자 유형

사용자가 **Cognito 사용자 풀**, **페더레이션 자격 증명 공급자** 또는 둘 다에 로그인할지 선택합니다. Amazon Cognito는 페더레이션 사용자와 사용자 풀 사용자에 따라 다른 요금 규정을 적용합니다. [요금에 대해 자세히 알아보기](#)

☒ **Cognito 사용자 풀**

사용자는 이메일 주소, 전화번호 또는 사용자 이름을 사용하여 로그인할 수 있습니다. 사용자 속성, 그룹 멤버십 및 보안 설정은 사용자 풀에 저장하고 구성됩니다.

☐ **연동 자격 증명 공급자**

사용자는 Facebook, Google, Amazon 및 Apple과 같은 소셜 자격 증명 공급자의 자격 증명을 사용하거나 SAML 또는 Open ID Connect를 통해 외부 디렉터리의 자격 증명을 사용하여 로그인할 수 있습니다. 사용자 풀의 연동 사용자에게 대한 사용자 속성 매핑 및 보안을 관리할 수 있습니다.

Cognito 사용자 풀 로그인 옵션

정보

사용자 풀에서 로그인하는 데 사용되는 속성을 선택합니다. 속성을 하나만 선택하거나 사용자 이름과 하나 이상의 다른 속성을 선택한 경우 사용자는 선택한 모든 옵션을 사용하여 로그인할 수 있습니다. 전화번호와 이메일만 선택하면 가입 시 두 가지 로그인 옵션 중 하나를 선택하라는 메시지가 표시됩니다.

☐ 사용자 이름

☒ 이메일

☐ 전화번호

⚠ 사용자 풀이 생성된 후에는 Cognito 사용자 풀 로그인 옵션을 변경할 수 없습니다.

ⓘ 고급 보안 기능으로 악의적인 로그인 시도로부터 프로덕션 사용자 계정을 보호할 수 있습니다. **앱 통합**에서 지금 활성화하세요. [자세히 알아보기](#)

[Amazon Cognito](#) > [사용자 풀](#) > 사용자 풀 생성

1단계  
[로그인 환경 구성](#)

2단계  
[보안 요구 사항 구성](#)

3단계  
[가입 환경 구성](#)

4단계  
[메시지 전송 구성](#)

5단계  
**앱 통합**

6단계

## 앱 통합 정보

Cognito의 기본 제공 인증 및 권한 부여 흐름을 사용하여 사용자 풀에 대한 앱 통합을 설정합니다.

### 사용자 풀 이름

사용자 풀에 대해 기억하기 쉬운 이름을 생성합니다.

사용자 풀 이름

lex-web-ui-test

사용자 풀 이름은 128자 이하로 제한됩니다. 이름에는 영숫자, 공백 및 특수 문자 + = , . @ -만 포함될 수 있습니다.

⚠ 이 사용자 풀을 생성한 후에는 사용자 풀 이름을 변경할 수 없습니다.

### 앱 클라이언트 이름 | 정보

앱 클라이언트의 기억하기 쉬운 이름을 입력합니다.


lex-web-ui-test-client

앱 클라이언트 이름은 128자 이하로 제한됩니다. 이름에는 영숫자, 공백 및 특수 문자 + = , . @ -만 포함될 수 있습니다.

### 클라이언트 보안키 | 정보

앱 클라이언트에서 클라이언트 보안키가 생성될지 여부를 선택합니다. 클라이언트 보안키는 앱의 서버 측 구성 요소에서 API 요청을 승인하는 데 사용됩니다. 클라이언트 보안키를 사용하면 서드 파티가 클라이언트를 가장하는 것을 방지할 수 있습니다.

- ☐ 클라이언트 보안키 생성
- ☒ 클라이언트 보안키를 생성하지 않음

 Amazon Cognito로 하여금 앱 클라이언트에 대한 클라이언트 보안키를 생성하도록 허용한 후에는 클라이언트 보안키를 변경하거나 제거할 수 없습니다.

### ▶ 고급 앱 클라이언트 설정

앞서 선택한 내용에 따라 제한된 인증 흐름, OAuth 2.0 권한 부여 유형 및 OIDC 범위가 채워집니다.

### ▶ 속성 읽기 및 쓰기 권한 | 정보

이 앱이 읽고 쓸 수 있는 표준 및 사용자 지정 속성을 선택합니다. 필수 속성은 쓰기 가능으로 잠깁니다. 앱 클라이언트가 가입 중에 초기 값을 설정할 수 있도록 하려면 변경 불가능한 사용자 지정 속성을 쓰기 가능으로 설정하는 것이 좋습니다.



태그 (0)

키

값

태그를 찾을 수 없음

취소

이전

사용자 풀 생성

## Amazon Cognito



사용자 풀 New

자격 증명 풀

🟢 사용자 풀 "lex-web-ui-test"이(가) 성공적으로 생성되었습니다.

세부 정보 보기

④ 고급 보안 기능으로 악의적인 로그인 시도로부터 프로덕션 사용자 계정을 보호할 수 있습니다. [앱 통합](#)에서 지금 활성화하세요. [자세히 알아보기](#)

Amazon Cognito > 사용자 풀



### Amazon Verified Permissions의 새로운 소식! API 게이트웨이용 Cognito 사용자 그룹 인증

이제 애플리케이션에 대한 세분화된 권한 부여 서비스인 Amazon Verified Permissions를 사용하여 API의 그룹 인식 권한 부여 정책을 생성할 수 있습니다. [자세히 알아보기](#)

[Amazon Verified Permissions로 이동](#)



### 사용자 풀 (1) 정보

사용자 풀을 보고 구성합니다. 사용자 풀은 연동 및 로컬 사용자 프로필의 디렉터리이며, 사용자에게 인증 옵션을 제공합니다.



삭제

사용자 풀 생성

🔍 이름 또는 ID로 사용자 풀 검색

< 1 > ⚙️

사용자 풀 이름 ▲

사용자 풀 ID ▼

생성 시간 ▼

마지막 업데이트 시간 ▼



[lex-web-ui-test](#)

ap-northeast-2\_APAV1CmVr

1초 전

1초 전

aws

서비스

검색

[알트+S]

서울

roads

Amazon Cognito

사용자 풀 New  
자격 증명 풀

고급 보안 기능으로 악의적인 로그인 시도로부터 프로덕션 사용자 계정을 보호할 수 있습니다. 앱 통합에서 지금 활성화하세요. 자세히 알아보기

사용자

그룹

로그인 경험

가입 경험

메시징

앱 통합

사용자 풀 속성

사용자 (0) 정보

사용자 삭제

사용자 생성

속성: 사용자 이름

속성으로 사용자 검색

< 1 >

사용자 이름

이메일 주소

이메일 확인됨

확인 상태

상태

사용자를 찾을 수 없음

사용자 생성

사용자 가져오기 (0) 정보

시작

중지

가져오기 작업 생성

작업 이름으로 가져오기 작업 검색

< 1 >

aws

서비스

검색

[알트+S]

ⓘ 고급 보안 기능으로 악의적인 로그인 시도로부터 프로덕션 사용자 계정을 보호할 수 있습니다. 앱 통합에서 지금 활성화하세요. 자세히 보기

초대 메시지 | 정보

메시징 탭에서 초대 메시지 템플릿 구성

☒ 초대 보내지 않음

☐ 이메일 초대 보내기

이메일 주소

이 사용자의 이메일 주소를 입력합니다. 사용자의 이메일 주소는 로그인, 계정 복구 및 계정 확인에 사용할 수 있습니다.

kilsoon.park80@gmail.com

☐ 이메일 주소를 확인됨으로 표시

임시 암호

Amazon Cognito는 사용자가 생성한 암호를 이메일 메시지로 사용자에게 전송합니다.

☒ 암호 설정

☐ 암호 생성

암호

이 사용자의 임시 암호를 입력합니다. 임시 암호는 초대 메시지와 함께 사용자에게 전송됩니다.

●●●●●●●●

☐ 암호 표시

aws 서비스 검색 [알트+S]

### Amazon Cognito

Amazon Cognito > 사용자 풀 > lex-web-ui-test > 사용자: 54f80d1c-80e1-7006-6a3c-7c420b92a1c7

#### 사용자 풀 New

자격 증명 풀

#### 사용자 정보

사용자 ID(하위)  
54f80d1c-80e1-7006-6a3c-7c420b92a1c7

로그인에 사용되는 별칭 속성  
이메일

MFA 설정  
MFA 비활성

계정 상태  
✓ 활성화됨

확인 상태  
암호 강제 변경

As a minimum, you would need to pass an existing Cognito Pool Id and Lex Bot name. For example, set the appropriate values in the `src/config/lex-web-ui-loader-config.json` file:

```
...  
cognito: {  
  "poolId": "us-east-1:deadbeef-fade-babe-cafe-0123456789ab"  
},  
lex: {  
  "botName": "myHelpBot"  
}  
...
```

Before you run the local development server, you need to install the development dependencies with the command:

```
npm install
```

EXPLORER

- AWS-LEX-WEB-UI-MASTER
  - build
  - config
  - dist
  - img
  - lex-web-ui
    - src
      - config
        - .gitattributes
        - default-lex-web-ui-loader-config.j...
        - lex-web-ui-loader-config.json
      - dependencies
      - initiate-chat-lambda
      - lex-web-ui-loader

src > config > default-lex-web-ui-loader-config.json > ...

```
1 {  
2   "region": "us-east-1",  
3   "cognito": {  
4     "poolId": "ap-northeast-2:54f80d1c-80e1-7006-6a3c-7c420b92a1c7",  
5     "aws_cognito_region": "us-east-1",  
6     "region": "us-east-1"  
7   },  
8   "connect": {  
9     "contactFlowId": "",  
10    "instanceId": "",  
11    "apiGatewayEndpoint": "",  
12    "promptForNameMessage": "Before starting a live chat, please tell me your name?",  
13    "waitingForAgentMessage": "Thanks for waiting. An agent will be with you when available",  
14    "waitingForAgentMessageIntervalSeconds": 60  
15  },  
}
```

# poolId 이거로 해야함.

The image shows a development environment with VS Code and the AWS Cognito console. In VS Code, the file `default-lex-web-ui-loader-config.json` is open, showing a configuration for a Cognito user pool. The `poolId` is set to `ap-northeast-2_APAV1CmVn`. The AWS Cognito console shows the '자격 증명 풀' (User Pools) page. A table lists the user pools, with the `lex-web-ui-test` pool highlighted. The `poolId` for this pool is `ap-northeast-2:359eccaa-46c4-4217-8bc8-e5ee15e5606c`, which is circled in red. The console also shows a message indicating that the user pool has been created successfully.

```
{
  "region": "ap-northeast-2",
  "cognito": {
    "poolId": "ap-northeast-2_APAV1CmVn",
    "aws_cognito_region": "ap-northeast-2",
    "region": "ap-northeast-2"
  }
},
```

Amazon Cognito > 자격 증명 풀

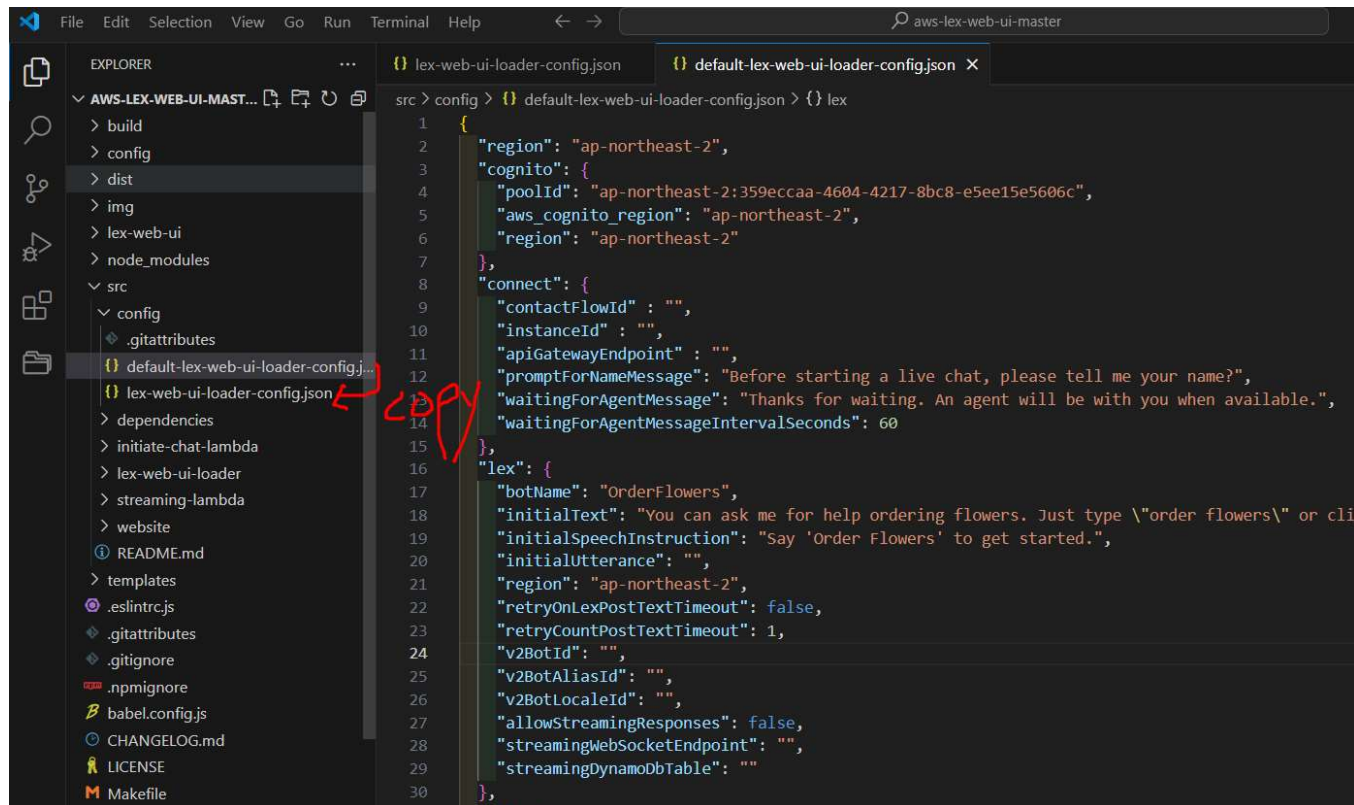
자격 증명 풀 (1) 정보

자격 증명 풀을 보고 구성합니다. 자격 증명 풀을 사용하면 고유한 자격 증명을 생성하고 사용자에게 권한을 할당할 수 있습니다.

이름 또는 ID로 자격 증명 풀 검색

자격 증명 풀 이름	자격 증명 풀 ID	생성 시간
lex-web-ui-test	ap-northeast-2:359eccaa-46c4-4217-8bc8-e5ee15e5606c	2초 전

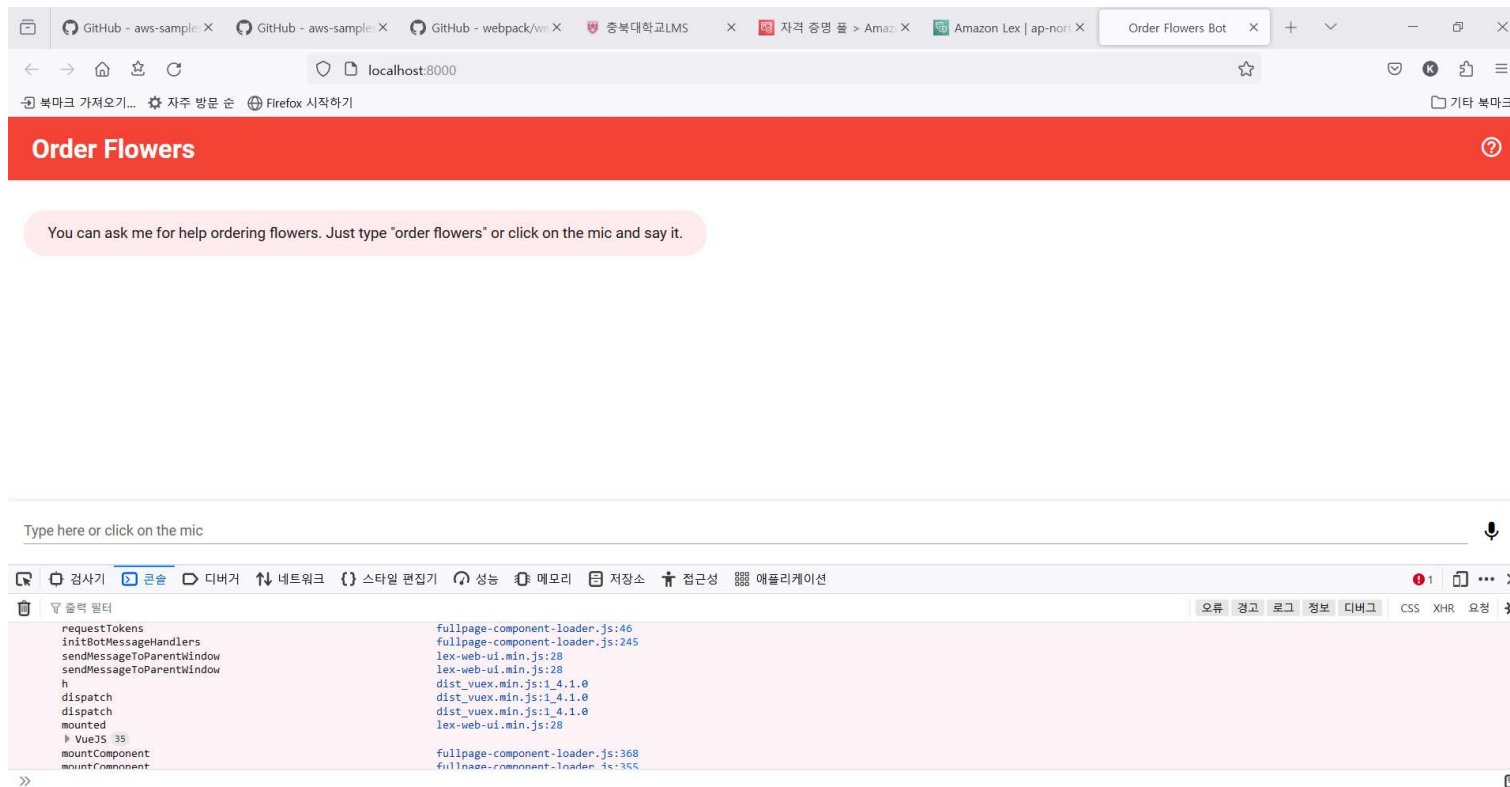
# 파일 복사



- npm install
- npm start <- 실행

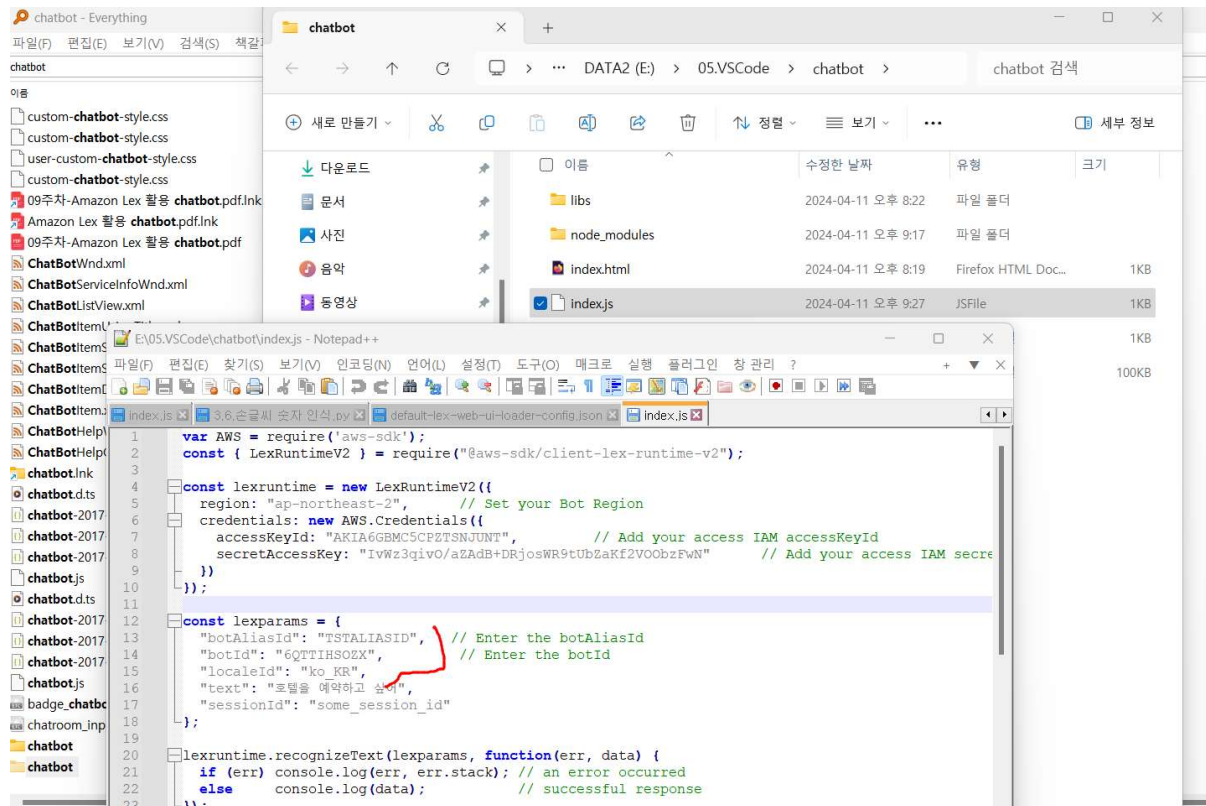


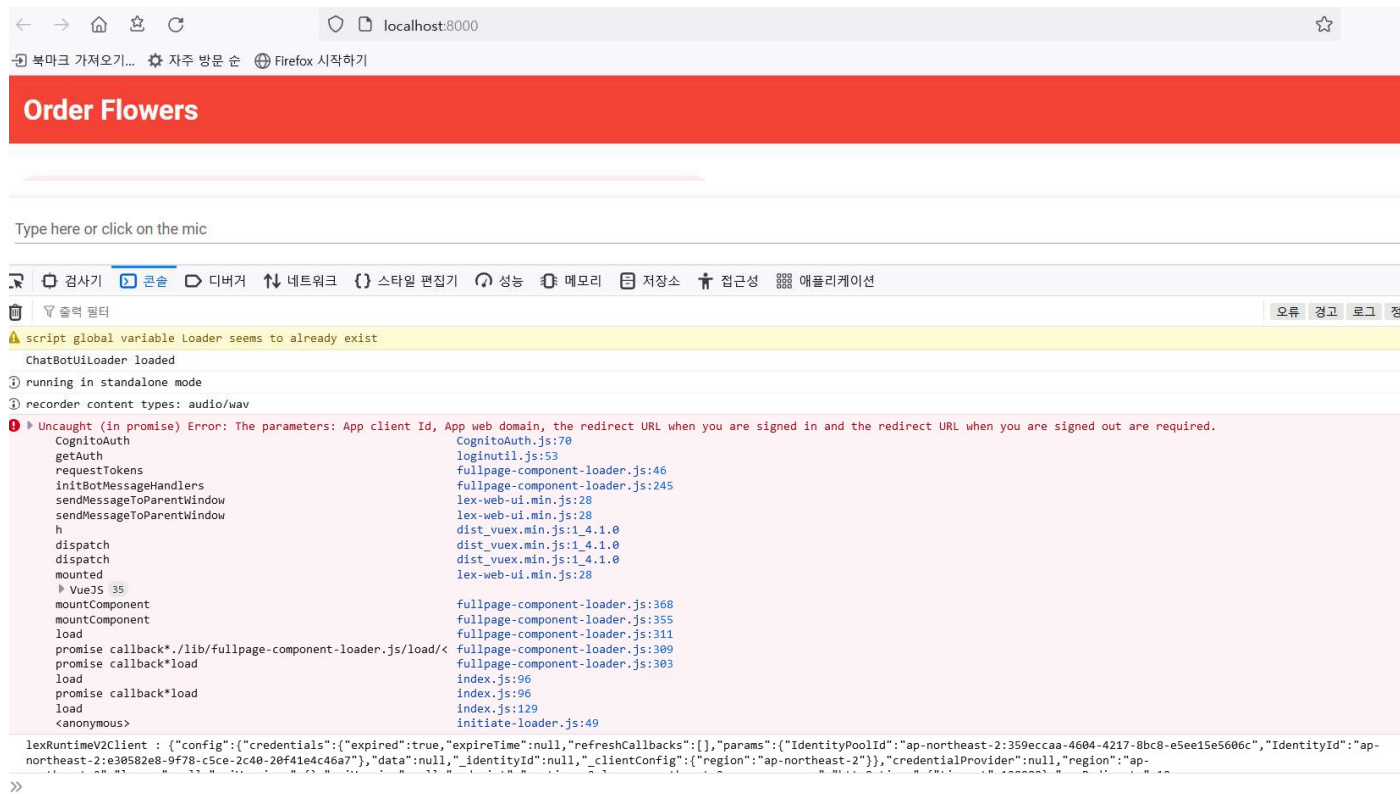
# 콘솔창



F12 누르면 콘솔창 볼수 있음.  
기타도구-웹개발자도구 활성화

# index.js





# IAM/권한추가

The screenshot shows the AWS IAM console interface. On the left, the 'Identity and Access Management(IAM)' sidebar is visible, with the 'Roles' link highlighted by a red circle. The main content area displays the 'Roles (8) 정보' page, which includes a search bar and a table of IAM roles. The table lists the role name, a checkbox, and the associated AWS service. A red checkmark is placed next to the role 'lex-web-ui-test-iam'.

<input type="checkbox"/>	역할 이름	신뢰할 수 있는 개체
<input type="checkbox"/>	<a href="#">AWSServiceRoleForLexV2Bots_PL4L698162</a>	AWS 서비스: lexv2 (서비스)
<input type="checkbox"/>	<a href="#">AWSServiceRoleForLexV2Bots_UW6JX1W7QTJ</a>	AWS 서비스: lexv2 (서비스)
<input type="checkbox"/>	<a href="#">AWSServiceRoleForSupport</a>	AWS 서비스: support (서비스)
<input type="checkbox"/>	<a href="#">AWSServiceRoleForTrustedAdvisor</a>	AWS 서비스: trustedadvisor (서비스)
<input type="checkbox"/>	<a href="#">bankBot-role-uws5pxsj</a>	AWS 서비스: lambda
<input type="checkbox"/>	<a href="#">IoT20240321-role-3n5rrbmw</a>	AWS 서비스: lambda
<input type="checkbox"/>	<a href="#">lex-web-ui-test-iam</a> ✓	자격 증명 공급자: cognito
<input type="checkbox"/>	<a href="#">thisdot-detectText-role-csi4oxhe</a>	AWS 서비스: lambda

# AmazonLexFullAccess 권한추가

[권한](#) | [신뢰 관계](#) | [태그](#) | [액세스 관리자](#) | [세션 취소](#)

**권한 정책 (2) 정보**

최대 10개의 관리형 정책을 연결할 수 있습니다.

필터링 기준 유형  
모든 유형

< 1 > ⚙

<input type="checkbox"/>	정책 이름 <a href="#">↗</a>	▲	유형	▼	연결된 엔터티	▼
<input type="checkbox"/>	<a href="#">+ AmazonLexFullAccess</a>		AWS 관리형		2	
<input type="checkbox"/>	<a href="#">+ Cognito-unauthenticated-1715253770905</a>		고객 관리형		1	

▶ 권한 경계 (설정되지 않음)