

# **AMAZON LEX: CHATBOT Documentation**



Amazon Lex

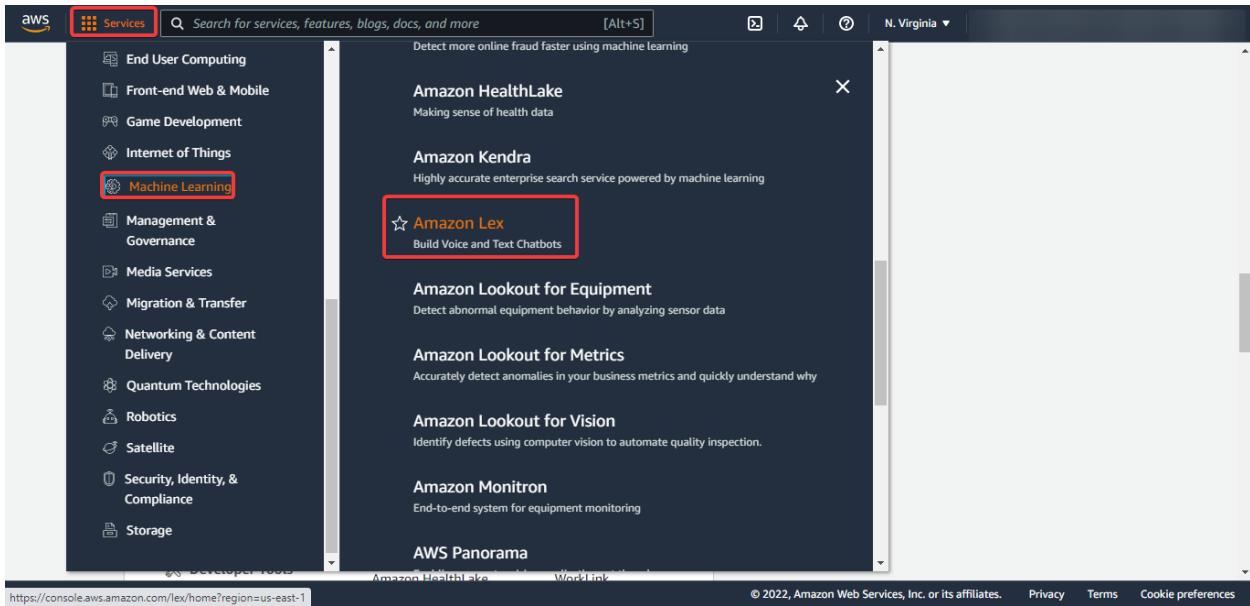
**Presented By**  
**Hem Bahadur Gurung**

# Table of Content

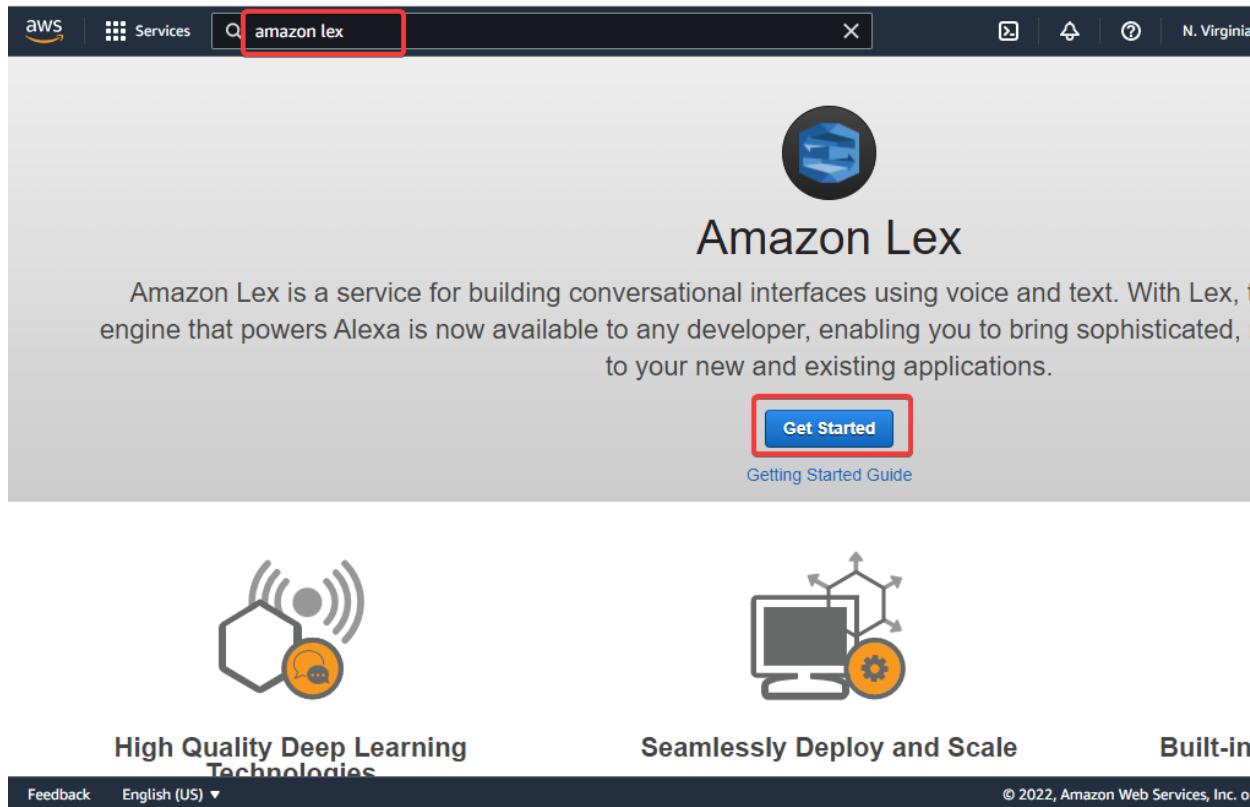
<b>Task 1: Creating a bot with Amazon Lex .....</b>	3
<b>Task 2: Testing your bot.....</b>	7
<b>Task 3: Creating an AWS Lambda function .....</b>	7
<b>Task 4: Testing your Lambda function.....</b>	11
<b>Task 5: Updating the intent of your bot.....</b>	12
<b>Task 6: Building and testing your bot.....</b>	15
<b>Task 7: Setting up an Amazon Cognito identity pool.....</b>	22
<b>Task 8: Modifying IAM roles to allow access to Amazon Lex.....</b>	25
<b>Task 9: Creating an S3 bucket.....</b>	31
<b>Task 10: Updating and testing the demonstration file .....</b>	39
<b>Amazon Lex Demo.....</b>	49
<b>Reference.....</b>	50

## Task 1: Creating a bot with Amazon Lex

- On the AWS Management Console, on the **Services** menu, choose **Amazon Lex**.

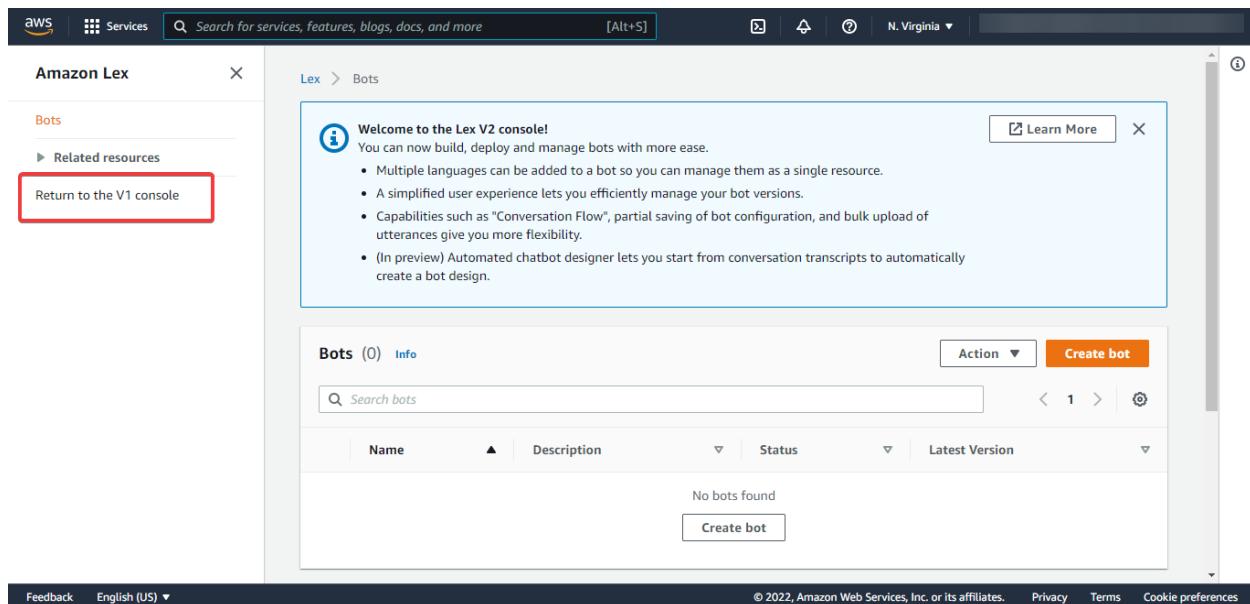


- On the **Amazon Lex** page, choose **Get Started**.



The screenshot shows the Amazon Lex homepage. At the top, there's a search bar with "amazon lex" typed in, which is highlighted with a red box. Below the search bar is the Amazon Lex logo, consisting of three blue arrows forming a triangle. The main title "Amazon Lex" is centered below the logo. A paragraph of text describes what Amazon Lex is: "Amazon Lex is a service for building conversational interfaces using voice and text. With Lex, the engine that powers Alexa is now available to any developer, enabling you to bring sophisticated, natural-sounding conversational interfaces to your new and existing applications." Below this text are two buttons: a blue "Get Started" button with white text, also highlighted with a red box, and a smaller "Getting Started Guide" link. At the bottom of the page, there are three sections with icons: "High Quality Deep Learning Technologies" (an orange brain icon), "Seamlessly Deploy and Scale" (a computer monitor icon with arrows), and "Built-in" (a gear icon). The footer contains links for "Feedback" and "English (US) ▾" on the left, and "© 2022, Amazon Web Services, Inc. or its affiliates." on the right.

- In the left navigation pane, choose **Return to the V1 console**.

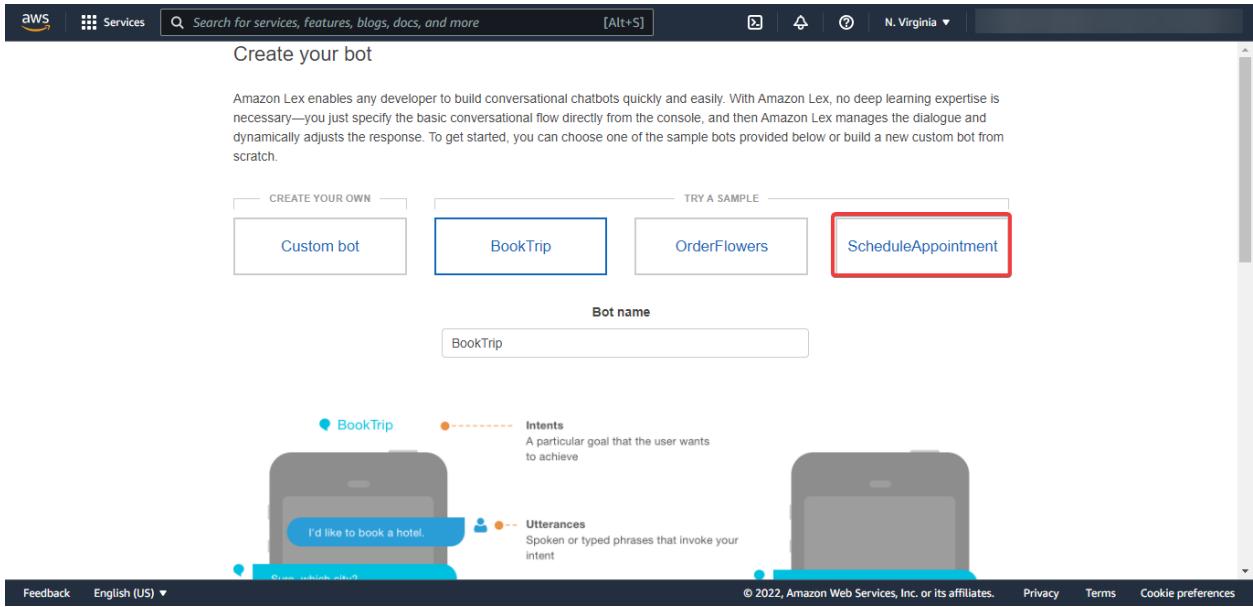


The screenshot shows the Amazon Lex V2 console. The left navigation pane has a "Bots" section with a "Return to the V1 console" link, which is highlighted with a red box. The main content area features a "Welcome to the Lex V2 console!" message box with an info icon. It says: "You can now build, deploy and manage bots with more ease." followed by a bulleted list: "Multiple languages can be added to a bot so you can manage them as a single resource.", "A simplified user experience lets you efficiently manage your bot versions.", "Capabilities such as "Conversation Flow", partial saving of bot configuration, and bulk upload of utterances give you more flexibility.", and "(In preview) Automated chatbot designer lets you start from conversation transcripts to automatically create a bot design." Below this is a "Bots (0) Info" table with columns for Name, Description, Status, and Latest Version. A search bar and a "Create bot" button are at the top of the table. The footer contains links for "Feedback" and "English (US) ▾" on the left, and "© 2022, Amazon Web Services, Inc. or its affiliates." on the right.

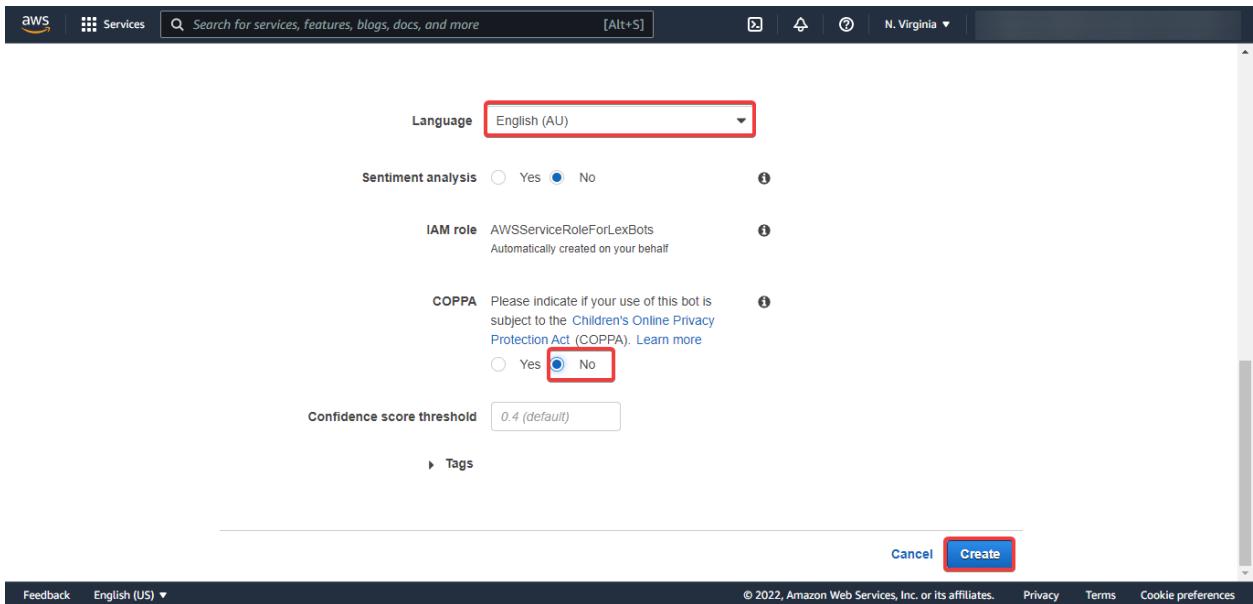
- Choose **Create**.

The screenshot shows the Amazon Lex Bots page. On the left, there's a sidebar with options like 'Bots', 'Intents', 'Slot types', 'Migration tool', and 'Switch to the new Lex V2 console'. The main area has a heading 'Bots' and a table with columns 'Name', 'Status', 'Locale', 'Last updated', and 'Date Created'. A prominent blue 'Create' button is at the top left of the table. Above the table, a message box says 'Try the new Lex V2 Console' with a link to 'Start building!'. The message also mentions support for multiple languages, simplified versioning, and interactive conversation flow. At the bottom of the page, there are links for 'Feedback', 'English (US) ▾', '© 2022, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

- On the Create your bot page, choose the **ScheduleAppointment** blueprint.



- For the **Language** section, select **English (US)**.
- For the **COPPA** section, select **No**.
- Choose **Create**.

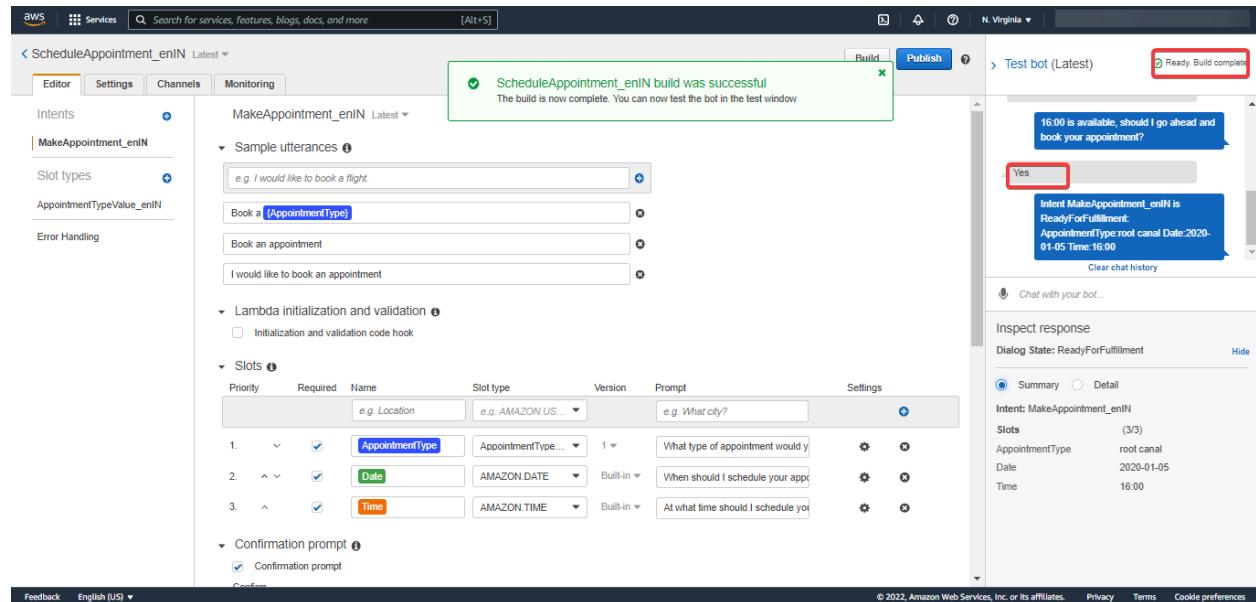


## Task 2: Testing your bot

Now, you will test your bot by using the test window in the console.

- When you see the status *Ready Build complete*, test your bot by entering the following values:
- I would like to make an appointment
- A root canal
- 5/1/2020
- 4:00 PM
- Yes

You should see the following confirmation.



## Task 3: Creating an AWS Lambda function

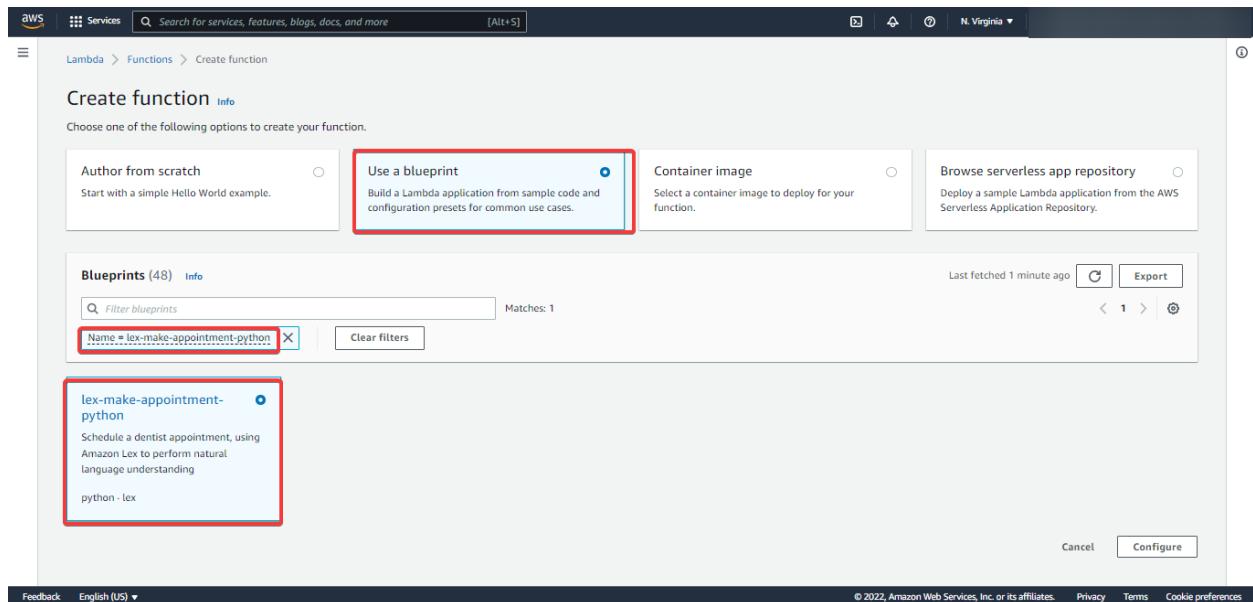
- On the AWS Management Console, on the **Services** menu, choose **Lambda**.

The screenshot shows the AWS Management Console homepage. At the top, there is a banner with the message: "The new AWS Console Home will replace your existing experience soon. Starting April 2022, the new AWS Console Home will replace your current experience. Switch now to customize your Console Home and view valuable insights. Learn more or let us know what you think." A "Switch now" button is located in the top right corner of the banner. Below the banner, the title "AWS Management Console" is displayed. On the left, the "AWS services" sidebar is visible, showing sections for "Recently visited services" (Amazon Lex, Amazon SageMaker) and "All services". Under "Compute", the "Lambda" service is highlighted with a red box. Other services listed under Compute include EC2, Lightsail, Batch, Elastic Beanstalk, Serverless Application Repository, AWS Outposts, EC2 Image Builder, and AWS App Runner. To the right of the sidebar, there is a "New AWS Console Home" section with a preview image and a "Switch now" button. Another section below it discusses the "AWS Console Mobile App" with a preview image and a "Learn more" link.

- Choose **Create function.**

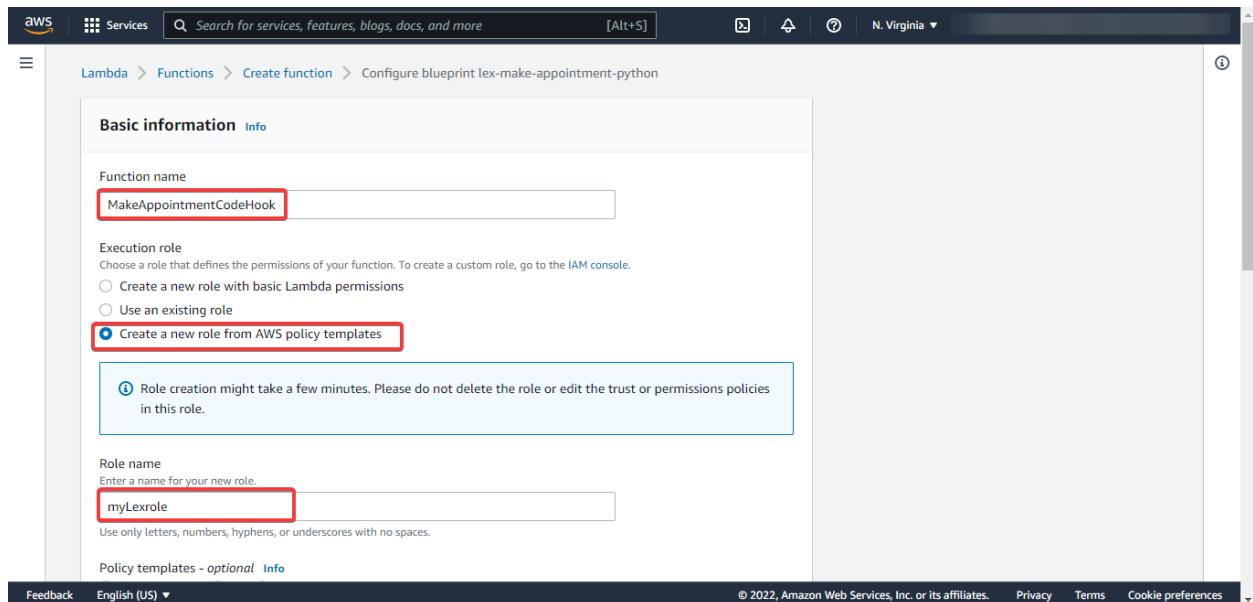
The screenshot shows the AWS Lambda Functions page. The left sidebar includes links for Dashboard, Applications, Functions (which is selected), Additional resources, and Related AWS resources. The main content area shows a message: "Tags failed to load. The filter doesn't include tags." Below this, a table titled "Functions (0)" is displayed, indicating "Last fetched 10 seconds ago". The table has columns for Function name, Description, Package type, Runtime, Code size, and Last modified. A "Create function" button is located at the top right of the table area. The bottom of the page includes standard AWS footer links: Feedback, English (US), © 2022, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

- Choose the **Use a blueprint** tab.
- In the search box, filter for Amazon Lex blueprints by entering **Lex**.
- Open the **lex-make-appointment-python** blueprint by selecting it.

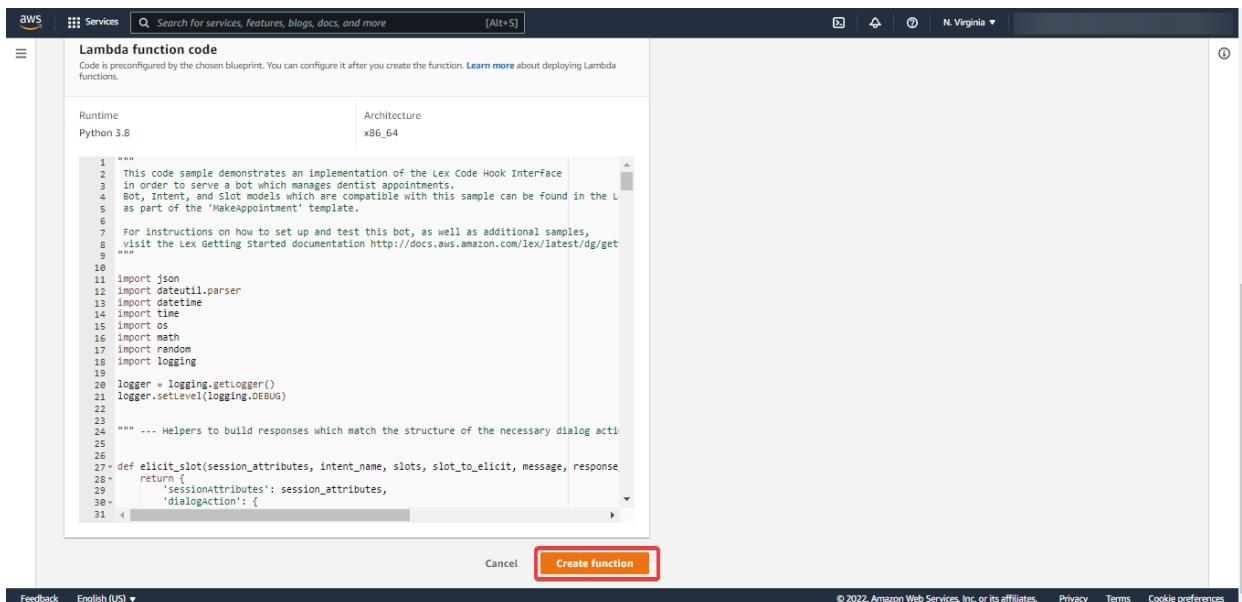


- For the **Function name**, enter MakeAppointmentCodeHook.
- For the **Execution role**, select **Create a new role from AWS policy templates**.
- For the **Role name**, enter myLexrole.

Take a few minutes to review the Python code in the function.



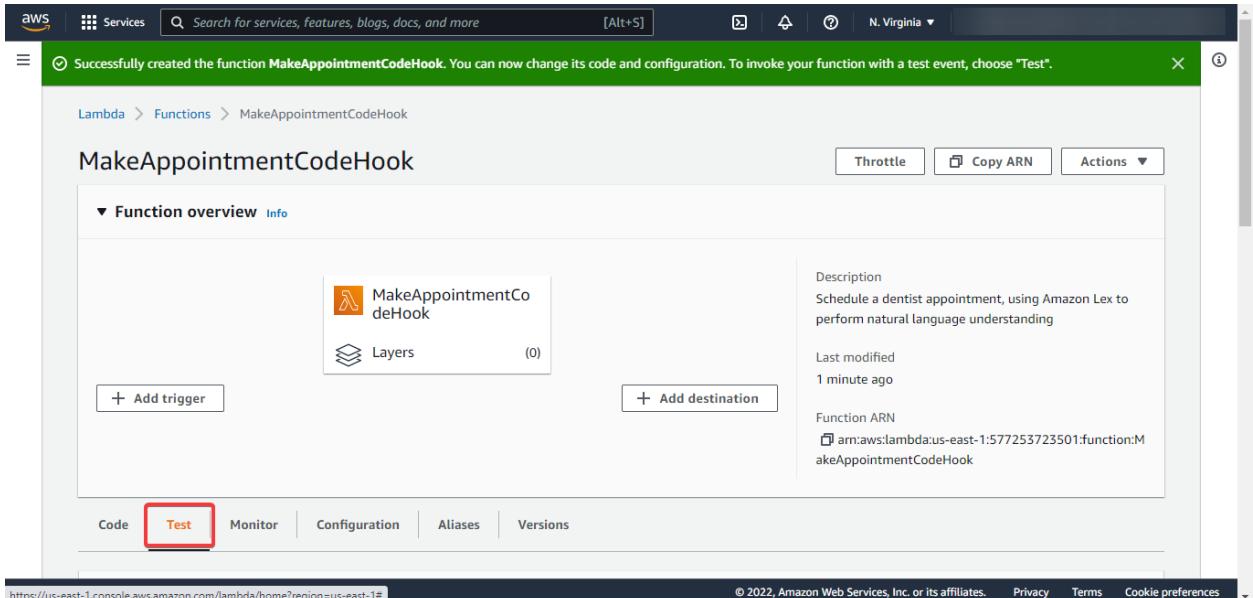
- Choose **Create function**.



When your function is ready, it should be in the **Lambda Designer** window.

## Task 4: Testing your Lambda function

- On the **Designer** page, choose **Test**.



- On the **Configure test event** page, for the name of the event, enter **MyMakeAppointmentEvent**.
- Choose **Create**.
- Choose **Test**.

The screenshot shows the AWS Lambda console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and account information. Below the navigation bar, the main menu includes 'Code', 'Test' (which is selected), 'Monitor', 'Configuration', 'Aliases', and 'Versions'. The 'Test' section is titled 'Test event' and contains fields for 'Test event action' (set to 'Create new event'), 'Event name' ('MyMakeAppointmentEvent'), and 'Event sharing settings' (set to 'Private'). A note says 'This event is only available in the Lambda console and to the event creator. You can configure a total of 10.' There's also a link to 'Learn more'. At the bottom of the test section, there are 'Save' and 'Test' buttons. The footer of the page includes links for 'Feedback', 'English (US) ▾', and various AWS terms like 'Privacy', 'Terms', and 'Cookie preferences'.

You see the message: *Execution result: succeeded*. You can examine the details of the test by expanding the **Details** section.

This screenshot shows the AWS Lambda function configuration page. At the top, it displays a success message: 'The test event MyMakeAppointmentEvent was successfully saved.' Below this, there are sections for 'Layers' (0), '+ Add trigger', '+ Add destination', and 'Last modified 10 minutes ago'. To the right, the 'Function ARN' is listed as 'arn:aws:lambda:us-east-1:577253723501:function:MyMakeAppointmentCodeHook'. The main content area has tabs for 'Code', 'Test' (selected), 'Monitor', 'Configuration', 'Aliases', and 'Versions'. Under the 'Test' tab, a green box highlights the 'Execution result: succeeded (logs)' message, which includes a 'Details' link. At the bottom, there are 'Delete', 'Save', and 'Test' buttons. The footer includes 'Feedback', 'English (US) ▾', and standard AWS links.

## Task 5: Updating the intent of your bot

Each bot you create in Amazon Lex has an *intent*. The intent is an action that the bot will fulfill. In this task, you will update the intent to use the AWS Lambda function that you created.

Each bot you create in Amazon Lex has an *intent*. The intent is an action that the bot will fulfill. In this task, you will update the intent to use the AWS Lambda function that you created.

- On the AWS Management Console, on the **Services** menu, choose **Amazon Lex**.
- From the **Bots** list, select the **ScheduleAppointment** bot that you created in Task 1.

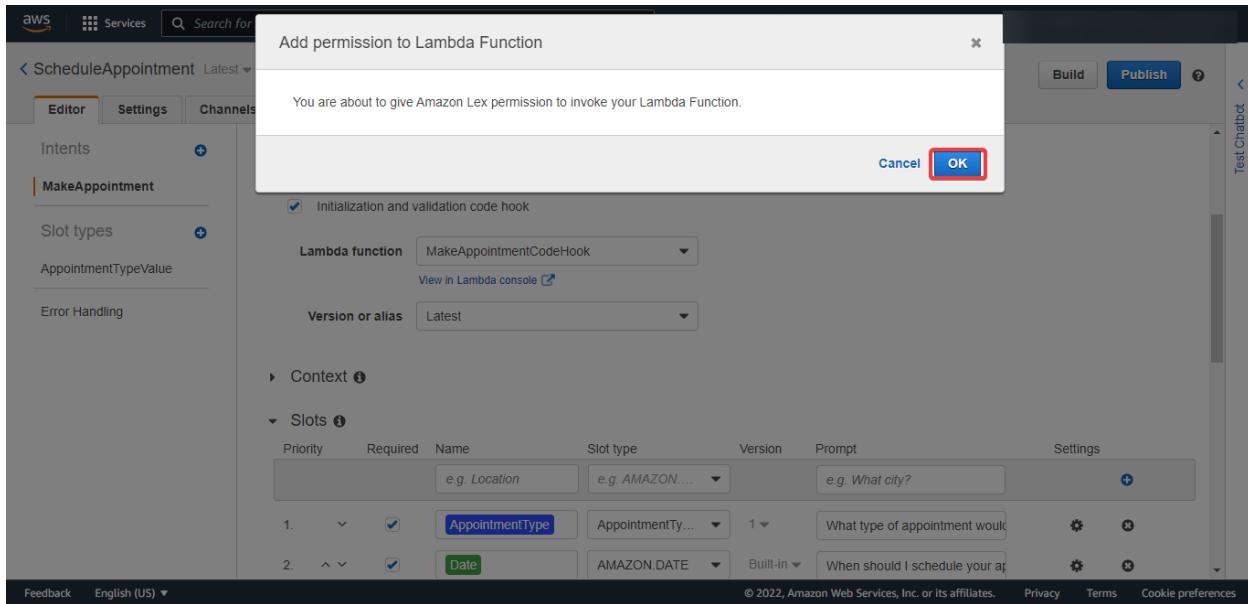
The screenshot shows the AWS Management Console with the 'Amazon Lex' service selected. The 'Bots' section is open, displaying a list of bots. The 'ScheduleAppointment' bot is selected and highlighted with a red box. The list includes columns for Name, Status, Locale, Last updated, and Date Created. The status for 'ScheduleAppointment' is 'READY'.

- Expand the **Lambda initialization and validation** section.
- Choose **Edit**.

The screenshot shows the configuration page for the 'ScheduleAppointment' bot. The 'Editor' tab is selected. Under the 'Lambda initialization and validation' section, the 'Initialization and validation code hook' checkbox is checked, and the 'Lambda function' dropdown is set to 'MakeAppointmentCodeHook', which is also highlighted with a red box.

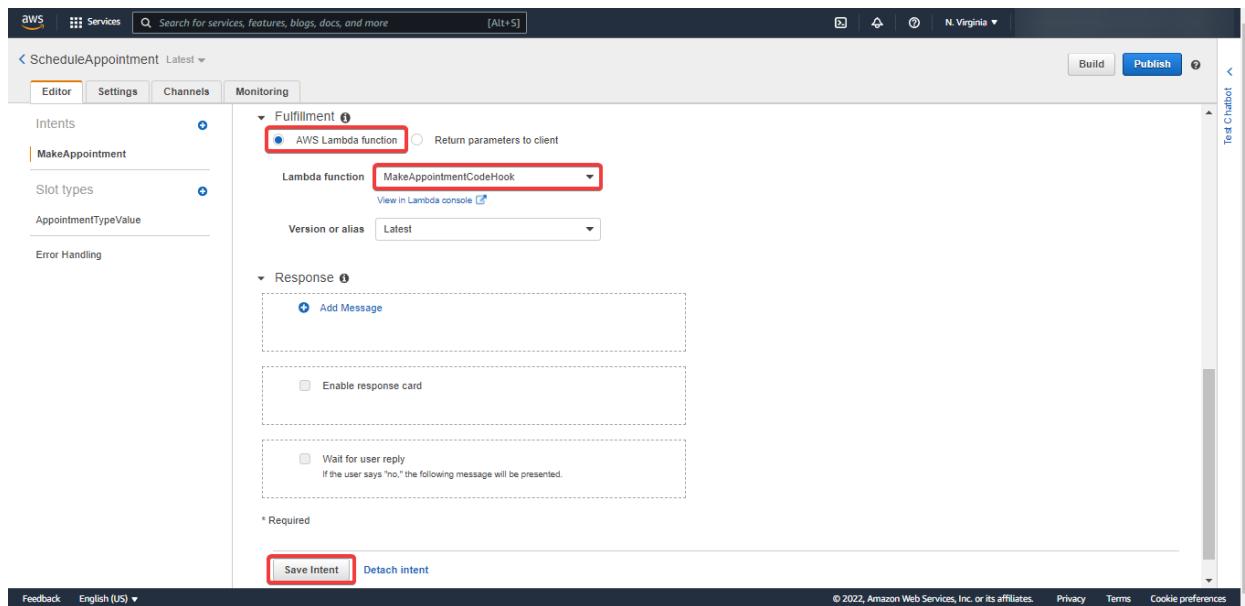
You should get a message that you are going to give Amazon Lex permission to invoke your Lambda function.

- Choose **OK**.



In addition to adding the code for initializing the bot, you must also add code to fulfill the request.

- In the **Fulfillment section**, choose the **AWS Lambda function** radio button.
- From the dropdown list of Lambda functions, select **MakeAppointmentCodeHook**.
- Choose **Save Intent**.

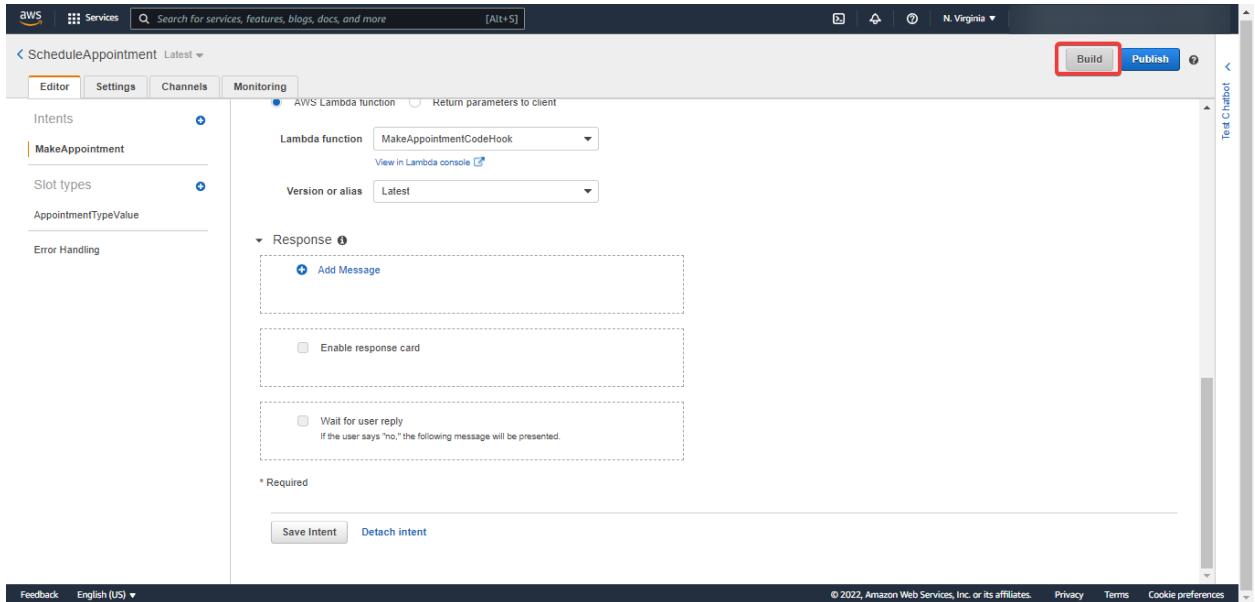


## Task 6: Building and testing your bot

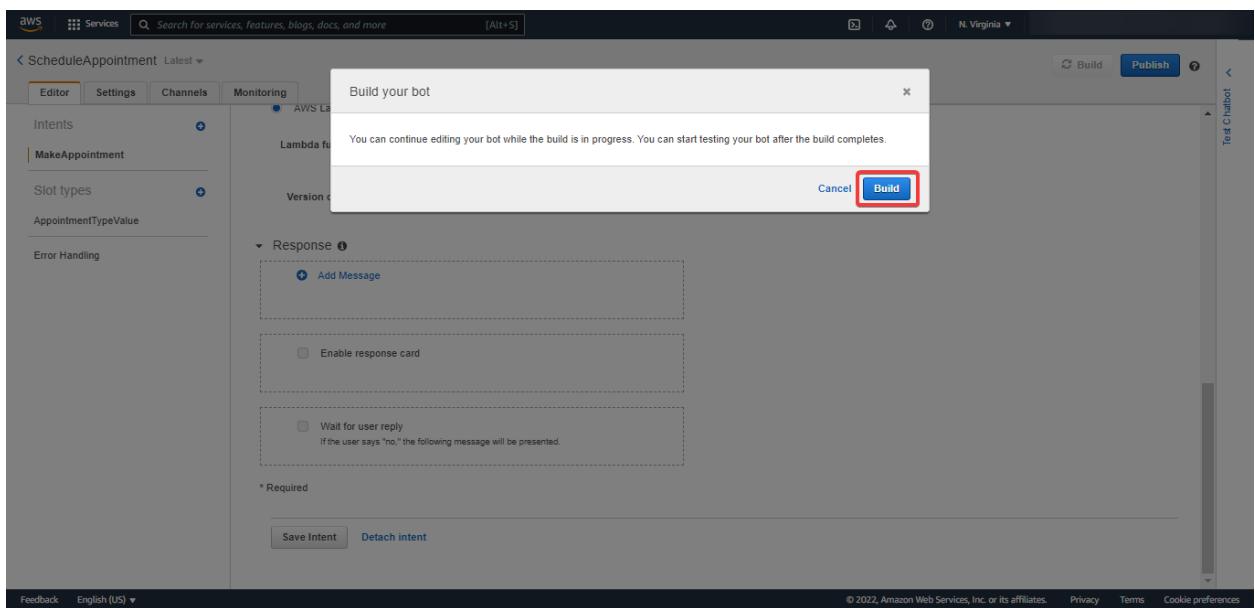
Now, you will test your bot to make sure that it uses the Lambda function.

- Select **Build**.

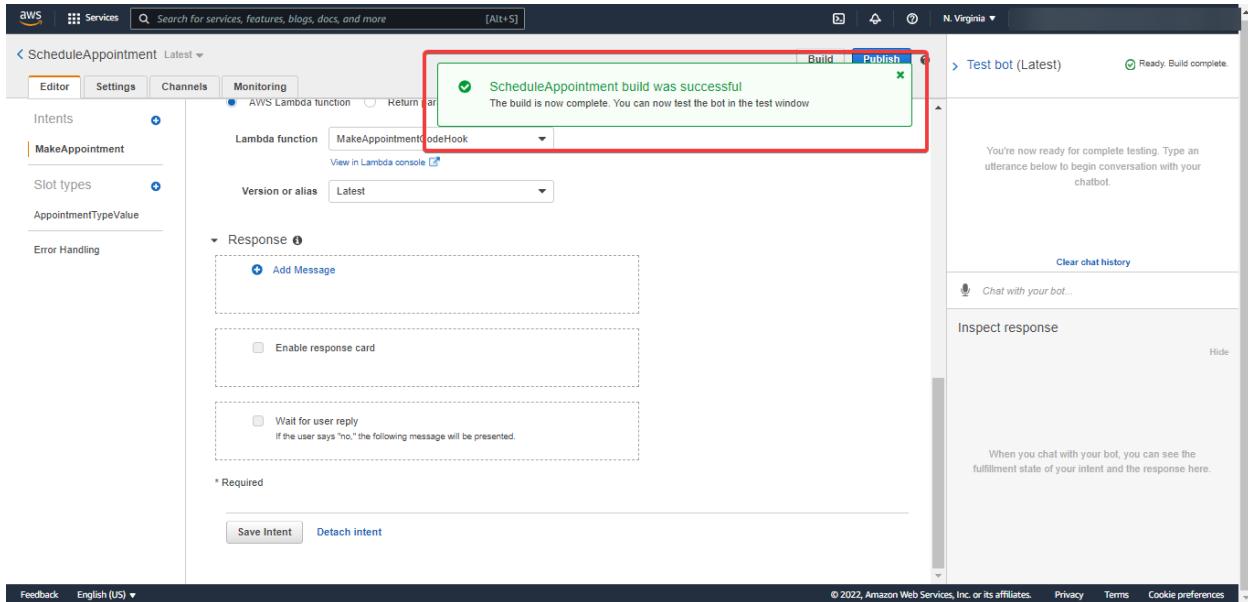
You are prompted with a message that says that you can continue to edit your bot while it builds.



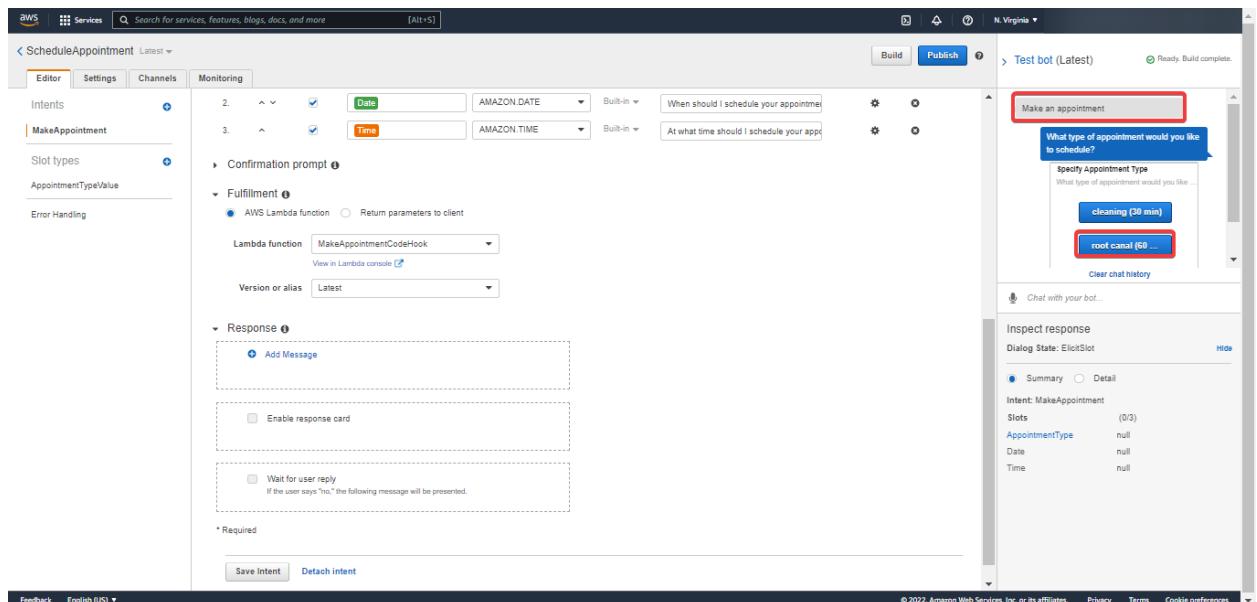
- Select **Build**.



After the build is complete, you should receive a confirmation that your bot build was successful.



- Test your bot by entering the following details.
- Enter **Make an appointment**
- Select **root canal**.



- Select one of the times that are displayed. If you select any of the available times and do not get a prompt to confirm the appointment, continue to the next step.

The screenshot shows the AWS Lambda Test bot interface. On the right, a conversation window displays a user message: "We do not have any availability on that date, is there another day which works for you?". Below it, a list of days is shown: "3-31 (Thu)", "4-1 (Fr)" (which is highlighted with a red box), and "Clear chat history". On the left, the Lambda function configuration is visible, including the intent "MakeAppointment" and its slots: "AppointmentType" (AppointmentTypeValue), "Date" (AMAZON.DATE), and "Time" (AMAZON.TIME). The Lambda function is set to "MakeAppointmentCodeHook".

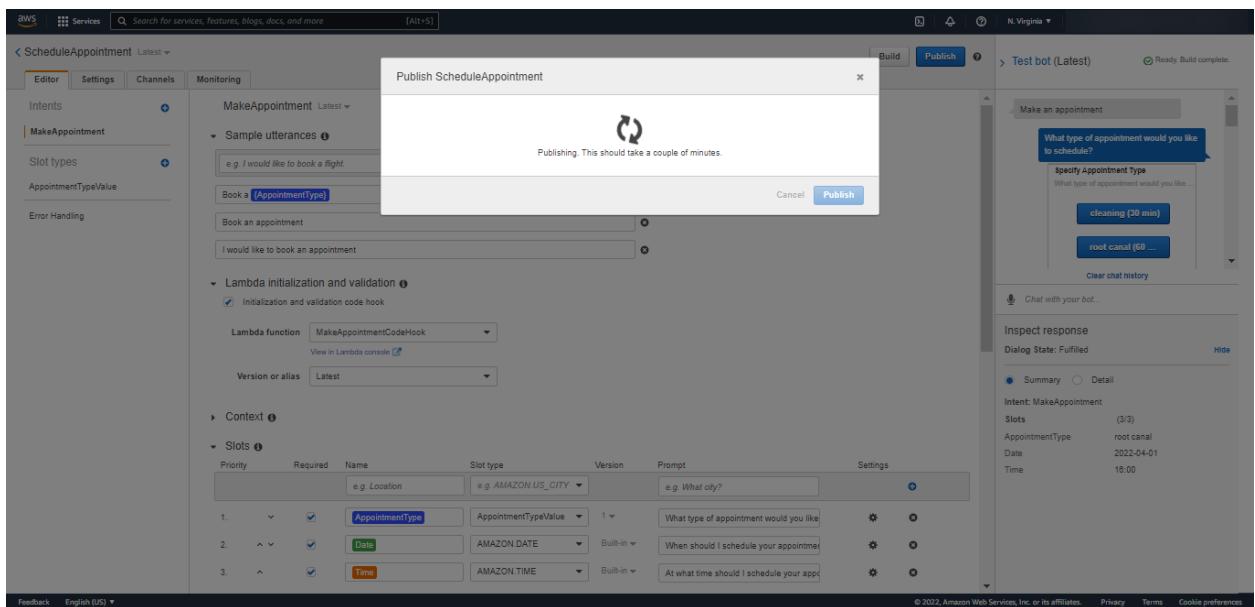
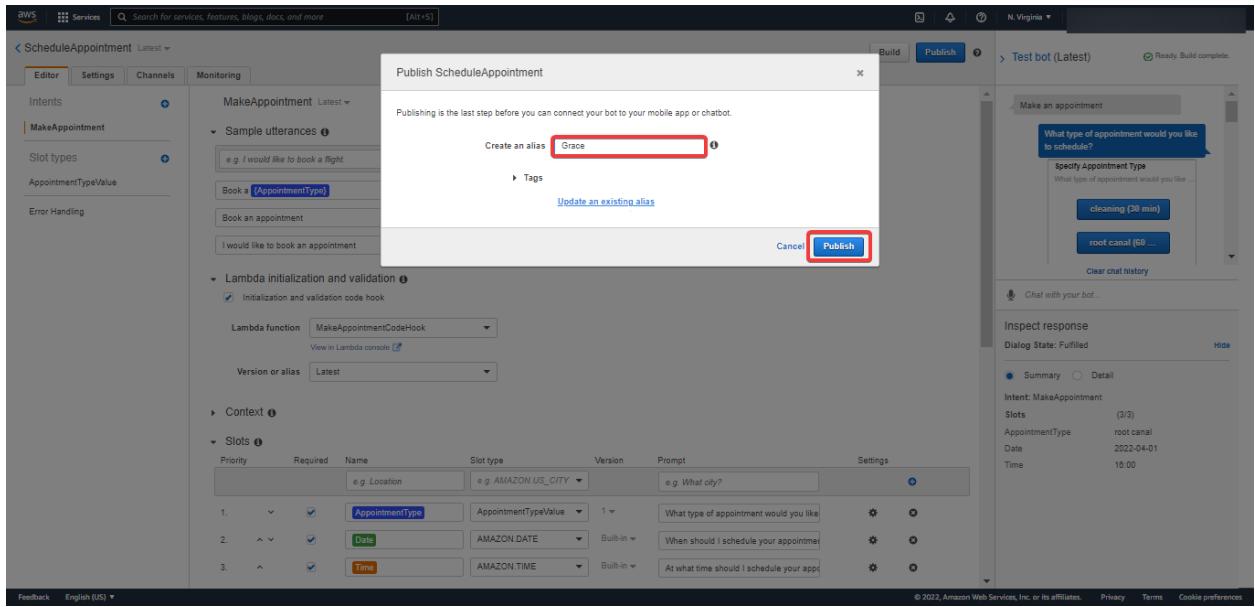
- Confirm the appointment by selecting yes.

The screenshot shows the AWS Lambda Test bot interface. On the right, a conversation window displays a user message: "What time on 2022-04-01 works for you? 4:00 p.m. is our only availability, does that work for you?". Below it, a list of responses is shown: "Confirm Appointment" (with a sub-message "Is 4:00 p.m. on 2022-04-01 okay?") and "yes" (which is highlighted with a red box). On the left, the Lambda function configuration is visible, including the intent "MakeAppointment" and its slots: "AppointmentType" (AppointmentTypeValue), "Date" (AMAZON.DATE), and "Time" (AMAZON.TIME). The Lambda function is set to "MakeAppointmentCodeHook".

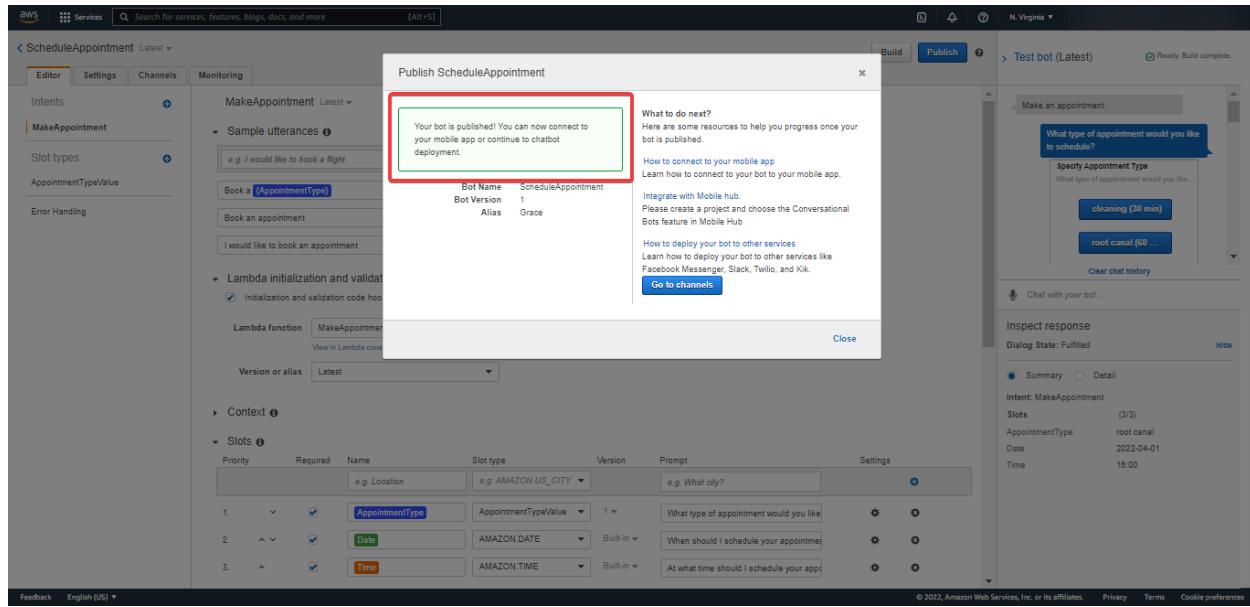
Now that you have a working version of your bot, you must publish it so that you can test it by calling it from a webpage.

- Choose **Publish**.

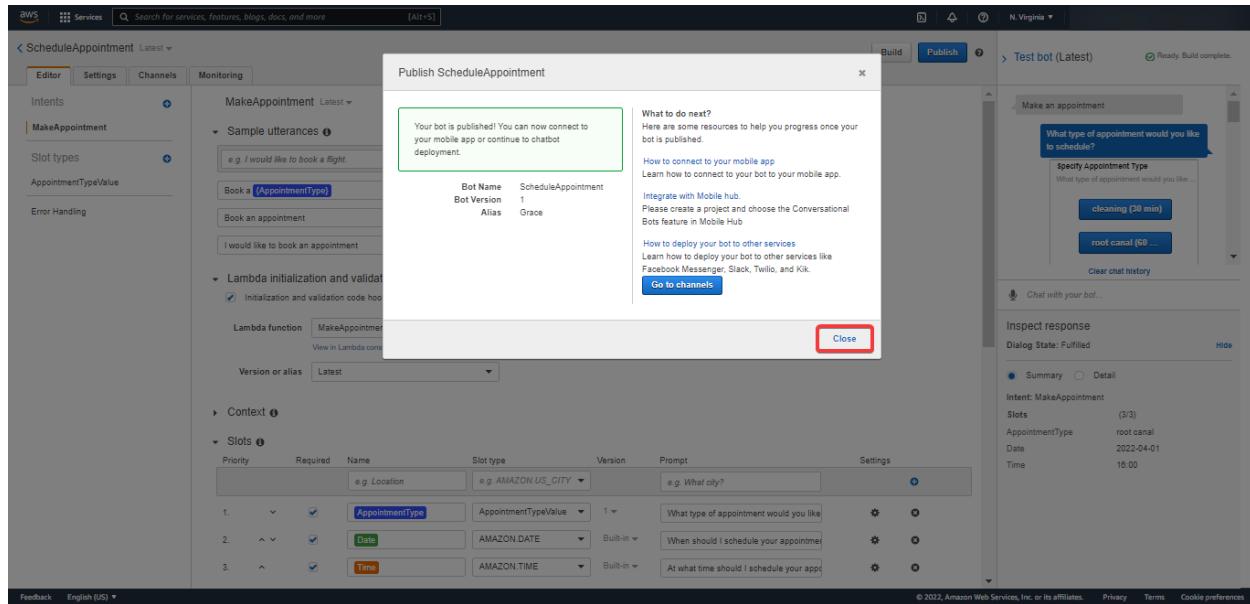
- For your bot's alias, enter a unique name.
- Choose **Publish**.



After the bot finishes publishing, you should get a confirmation notification. Make a note of the alias because you will need it in the next task.



- Close the Publish ScheduleAppointment notification by choosing Close.



## Incorporating your bot into a webpage

Now that your bot is working, it's time to test it by running it from a webpage. The easiest way to do the test is to create a static webpage and host it on

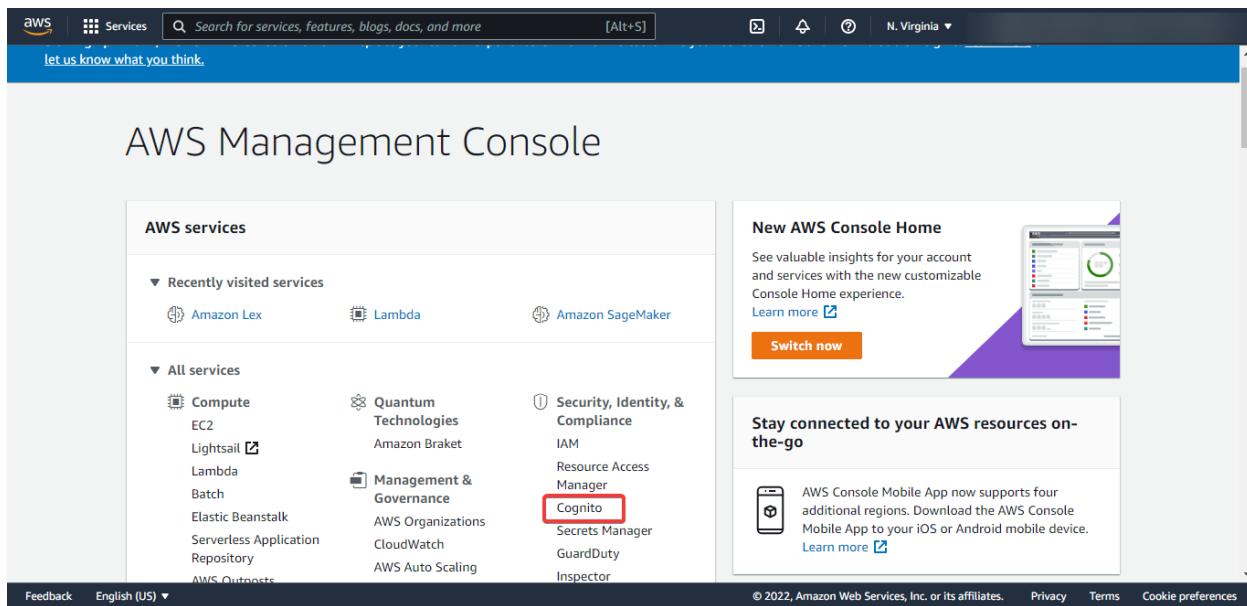
Amazon S3. The webpage will invoke the Amazon Lex API to load your bot. Visitors to the webpage can then interact with it.

## Task 7: Setting up an Amazon Cognito identity pool

You will now set up a webpage for testing your appointment bot. This webpage will be hosted in Amazon S3 as a static webpage. To add security to this page, you must first set up an Amazon Cognito identity pool.

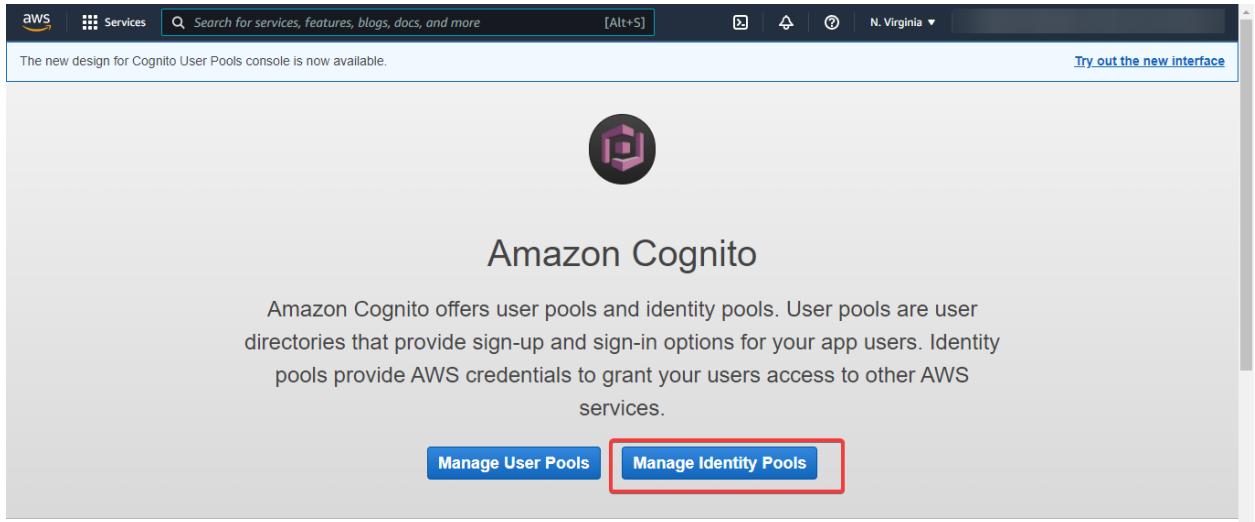
To set up the identity pool:

- On the AWS Management Console, on the **Services** menu, choose **Cognito**.



The screenshot shows the AWS Management Console homepage. At the top, there's a navigation bar with the AWS logo, a "Services" button, a search bar, and account information. Below the search bar, a blue banner says "let us know what you think." The main area is titled "AWS Management Console". On the left, there's a sidebar titled "AWS services" with sections for "Recently visited services" (Amazon Lex, Lambda, Amazon SageMaker) and "All services". Under "All services", there are several categories: Compute (EC2, Lightsail, Lambda, Batch, Elastic Beanstalk, Serverless Application Repository, AWS Outposts), Quantum Technologies (Amazon Braket), Management & Governance (AWS Organizations, CloudWatch, AWS Auto Scaling), Security, Identity, & Compliance (IAM, Resource Access Manager, Secrets Manager, GuardDuty, Inspector), and a "Cognito" service icon which is highlighted with a red box. To the right of the sidebar, there are two promotional boxes: one for the "New AWS Console Home" and another for "Stay connected to your AWS resources on-the-go" (AWS Console Mobile App). At the bottom, there are links for "Feedback", "English (US)", and various legal links like "Privacy", "Terms", and "Cookie preferences".

- Choose **Manage Identity Pools**.

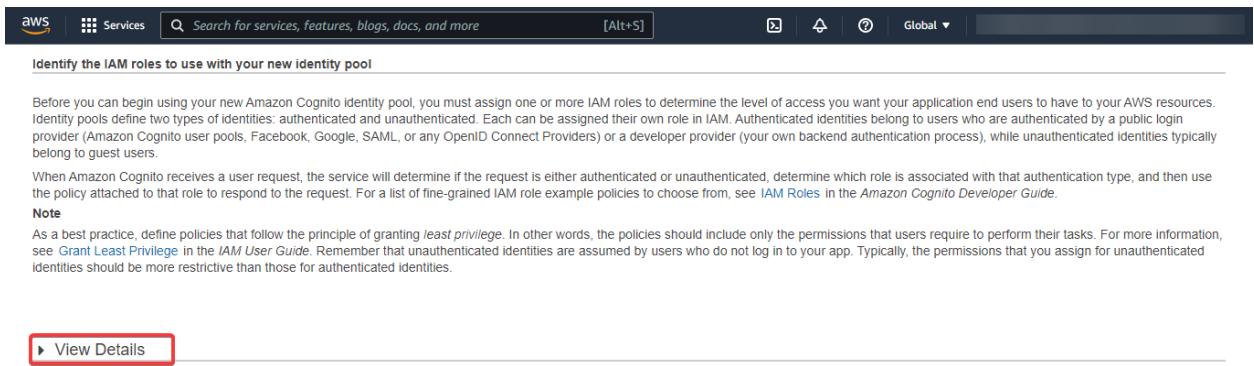


- In the **Getting started wizard** page, for the **Identity pool name**, enter: myidentitypool
- Select **Enable access to unauthenticated identities** and choose **Create Pool**.

The screenshot shows the "Create new identity pool" wizard, Step 1: Create identity pool. It has two tabs: "Step 1: Create identity pool" (selected) and "Step 2: Set permissions". The main area is titled "Create new identity pool" with a sub-instruction: "Identity pools are used to store end user identities. To declare a new identity pool, enter a unique name." A text input field contains "myidentitypool", which is highlighted with a red box. Below the input field is an example: "Example: My App Name". There are two expandable sections: "Unauthenticated identities" (with a checked checkbox for "Enable access to unauthenticated identities") and "Authentication flow settings". At the bottom, there's a note about authentication flows, a "Required" indicator, and a "Create Pool" button, which is also highlighted with a red box.

Expand the View Details section.

- Make a note of the two role names (one role will end with *Auth\_Role* and the other role will end with *Unauth\_Role*). You will need both of these role names later in this lab.
- Choose **Allow**.



Identify the IAM roles to use with your new identity pool

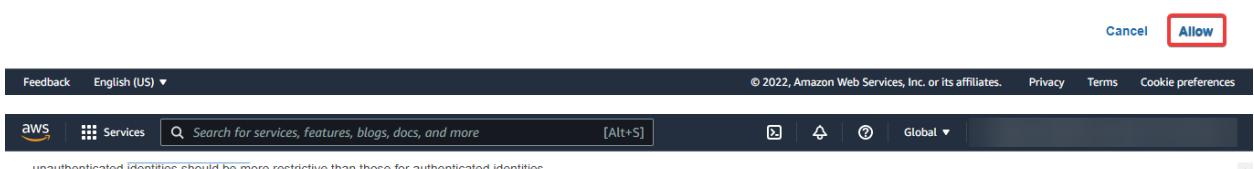
Before you can begin using your new Amazon Cognito identity pool, you must assign one or more IAM roles to determine the level of access you want your application end users to have to your AWS resources. Identity pools define two types of identities: authenticated and unauthenticated. Each can be assigned their own role in IAM. Authenticated identities belong to users who are authenticated by a public login provider (Amazon Cognito user pools, Facebook, Google, SAML, or any OpenID Connect Providers) or a developer provider (your own backend authentication process), while unauthenticated identities typically belong to guest users.

When Amazon Cognito receives a user request, the service will determine if the request is either authenticated or unauthenticated, determine which role is associated with that authentication type, and then use the policy attached to that role to respond to the request. For a list of fine-grained IAM role example policies to choose from, see [IAM Roles in the Amazon Cognito Developer Guide](#).

**Note**

As a best practice, define policies that follow the principle of granting *least privilege*. In other words, the policies should include only the permissions that users require to perform their tasks. For more information, see [Grant Least Privilege](#) in the *IAM User Guide*. Remember that unauthenticated identities are assumed by users who do not log in to your app. Typically, the permissions that you assign for unauthenticated identities should be more restrictive than those for authenticated identities.

▶ View Details

Feedback English (US) ▾ © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

unauthenticated identities should be more restrictive than those for authenticated identities.

▼ Hide Details

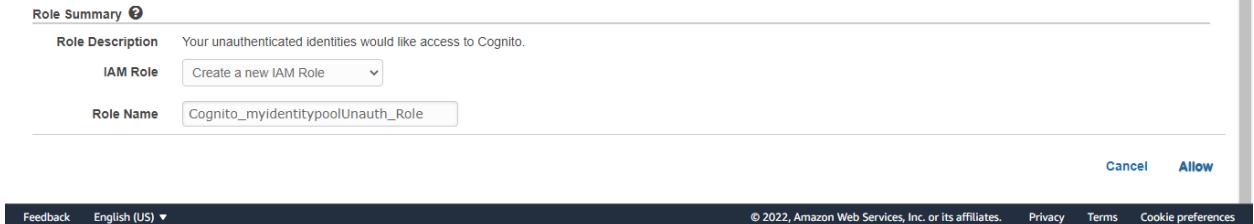
**Role Summary** ⓘ

**Role Description** Your authenticated identities would like access to Cognito.

**IAM Role** Create a new IAM Role

**Role Name** Cognito\_myidentitypoolAuth\_Role

▶ View Policy Document

Feedback English (US) ▾ © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

**Role Summary** ⓘ

**Role Description** Your unauthenticated identities would like access to Cognito.

**IAM Role** Create a new IAM Role

**Role Name** Cognito\_myidentitypoolUnauth\_Role

Cancel Allow

- Make a note of the **IdentityPoolID** because you will need this ID later in the lab.

Cognito\_myidentitypoolAuth\_Role

Cognito\_myidentitypoolUnauth\_Role

The screenshot shows the AWS Cognito 'Getting started' page for the Android platform. On the left, there's a sidebar with links like 'Identity pool', 'Dashboard', 'Sample code' (which is selected), and 'Identity browser'. The main content area has a heading 'Getting started with Amazon Cognito' and a 'Platform' dropdown set to 'Android'. Below this, there are two sections: 'Download the AWS SDK for Android' (with a download button) and 'Get AWS Credentials'. A code snippet for initializing a credentials provider is shown:

```
// Initialize the Amazon Cognito credentials provider
CognitoCachingCredentialsProvider credentialsProvider = new CognitoCachingCredentialsProvider(
    getApplicationContext(),
    "us-east-1:1577cedd-ba2c-4565-8b89-ef588640f619", // Identity pool ID
    Regions.US_EAST_1 // Region
);
```

Below the code, there's another section 'Then initialize the credentials provider:' with a single item 'Getting Started with Cognito Identity'. At the bottom, there's a 'Go To Dashboard' button.

## Task 8: Modifying IAM roles to allow access to Amazon Lex

The webpage that hosts your bot must be allowed to access Amazon Lex. To do this, the two roles that were created in the Amazon Cognito identity pool need permissions to access Amazon Lex. You must configure AWS Identity and Access Management (IAM) to grant these permissions to the identity pool roles.

- On the AWS Management Console, on the **Services** menu, choose **IAM**.

AWS Management Console

**AWS services**

▼ Recently visited services

- Cognito
- Amazon Lex
- Amazon SageMaker
- IAM
- Lambda

▼ All services

Compute	Quantum Technologies	Security, Identity, & Compliance
EC2	Amazon Braket	IAM
Lightsail	Management & Governance	Resource Access Manager
Lambda	AWS Organizations	Cognito
Batch	CloudWatch	Secrets Manager
Elastic Beanstalk	AWS Auto Scaling	GuardDuty
Serverless Application Repository	CloudFormation	Inspector
AWS Outposts		

New AWS Console Home

See valuable insights for your account and services with the new customizable Console Home experience.

Learn more [Switch now](#)

Stay connected to your AWS resources on-the-go

AWS Console Mobile App now supports four additional regions. Download the AWS Console Mobile App to your iOS or Android mobile device.

Learn more [AWS Console Mobile App](#)

Explore AWS

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Choose Roles.

Identity and Access Management (IAM)

Introducing the new IAM dashboard experience

We've redesigned the IAM dashboard experience to make it easier to use. [Let us know what you think.](#)

**IAM dashboard**

**Security recommendations**

Add MFA for root user

Sign in as the root user (or contact your administrator) and register a multi-factor authentication (MFA) device for the root user to improve security for this account.

**IAM resources**

User groups	Users	Roles	Policies	Identity providers
0	0	19	2	0

**What's new**

Updates for features in IAM

View all [View all](#)

**AWS Account**

Account ID: 577253723501

Account Alias: 577253723501 [Create](#)

Sign-in URL for IAM users in this account: <https://577253723501.signin.amazonaws.com/console>

**Tools**

Policy simulator

The simulator evaluates the policies that you choose and determines the effective

https://us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles

https://us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Enter the first few characters of the *Auth\_Role* name you noted in Task 7.
- From the list of search results, select the role that you created.

The screenshot shows the AWS IAM Roles experience. The left sidebar includes 'Access management' (User groups, Users, Roles), 'Access reports' (Access analyzer, Archive rules, Analyzers, Settings), and 'Feedback'. The main area displays a table of roles with columns for Role name, Trusted entities, and Last activity. Two roles are listed:

Role name	Trusted entities
Cognito_myidentitypoolAuth_Role	Identity Provider: cognito-identity.amazonaws.com
Cognito_myidentitypoolUnauth_Role	Identity Provider: cognito-identity.amazonaws.com

A red box highlights the 'Create role' button at the top right and the role names in the table.

The screenshot shows the AWS IAM Roles experience after performing a search for 'auth'. The search results show the same two roles: Cognito\_myidentitypoolAuth\_Role and Cognito\_myidentitypoolUnauth\_Role. A red box highlights the search bar containing 'auth'.

- Choose **Attach policies**.

Identity and Access Management (IAM)

Creation date: March 30, 2022, 16:09 (UTC+05:45)

Last activity: None

ARN: arn:aws:iam::577253723501:role/Cognito\_myidentitypoolAuth\_Role

Maximum session duration: 1 hour

Permissions | Trust relationships | Tags | Access Advisor | Revoke sessions

**Permissions policies (1)**

You can attach up to 10 managed policies.

Filter policies by property or policy name and press enter

Policy name Type Description

oneClick\_Cognito\_myidentitypoolAuth\_Role\_1648635547052 Customer inline

Permissions boundary - (not set)

Set a permissions boundary to control the maximum permissions this role can have.

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- In the search box, enter AmazonLex and Press enter on PC.
- Select **AmazonLexReadOnly & AmazonLexRunBotsOnly** policies.
- Choose **Attach policy**.

Attach policy to Cognito\_myidentitypoolAuth\_Role

Current permissions policies (0)

Other permissions policies (Selected 2/740)

Filter policies by property or policy name and press enter

"AmazonLex" Clear filters

Policy name	Type	Description
<input checked="" type="checkbox"/> <span>AmazonLexReadOnly</span>	AWS managed	Provides read-only access to Amazon Lex.
<input type="checkbox"/> <span>AmazonLexFullAccess</span>	AWS managed	Provides full access to Amazon Lex via the AWS M...
<input checked="" type="checkbox"/> <span>AmazonLexRunBotsOnly</span>	AWS managed	Provides access to Amazon Lex conversational APIs.

Cancel Attach policies

Feedback English (US) © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Finally these two policies have been attached.

The screenshot shows the AWS IAM Permissions Policies page. On the left, there's a sidebar with 'Identity and Access Management (IAM)' selected. The main area shows a table of policies:

Policy name	Type	Description
AmazonLexReadOnly	AWS managed	Provides read-only access to Amazon Lex
AmazonLexRunBotsOnly	AWS managed	Provides access to Amazon Lex
oneClick_Cognito_myidentitypoolAuth_Role_1648635547052	Customer inline	

Two policies are highlighted with a red box: 'AmazonLexReadOnly' and 'AmazonLexRunBotsOnly'. Below the table, there's a section for 'Permissions boundary - (not set)' with a 'Set permissions boundary' button.

- Use the same process to attach the same two policies to the *Unauth\_Role*.

The screenshot shows the AWS IAM Roles page. On the left, there's a sidebar with 'Identity and Access Management (IAM)' selected. The main area shows a table of roles:

Role name	Trusted entities
Cognito_myidentitypoolAuth_Role	Identity Provider: cognito-identity.amazonaws.com
<b>Cognito_myidentitypoolUnauth_Role</b>	Identity Provider: cognito-identity.amazonaws.com
EMR_AutoScaling_DefaultRole	AWS Service: elasticmapreduce, and 1 more.
EMR_DefaultRole	AWS Service: elasticmapreduce
EMR_EC2_DefaultRole	AWS Service: ec2

The role 'Cognito\_myidentitypoolUnauth\_Role' is highlighted with a red box and has a checked checkbox next to it. A blue banner at the top says 'Introducing the new IAM roles experience'.

Screenshot of the AWS IAM Roles page showing the 'Cognito\_myidentitypoolUnauth\_Role' configuration.

**Summary**

Creation date	March 30, 2022, 16:09 (UTC+05:45)
Last activity	None
ARN	arn:aws:iam::577253723501:role/Cognito_myidentitypoolUnauth_Role
Maximum session duration	1 hour

**Permissions** | Trust relationships | Tags | Access Advisor | Revoke sessions

**Permissions policies (1)**  
You can attach up to 10 managed policies.

Filter policies by property or policy name and press enter

Add permissions ▾

Attach policies (highlighted with a red box)

Create inline policy

Policy name ▾

Type Description

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the 'Attach policy to Cognito\_myidentitypoolUnauth\_Role' dialog.

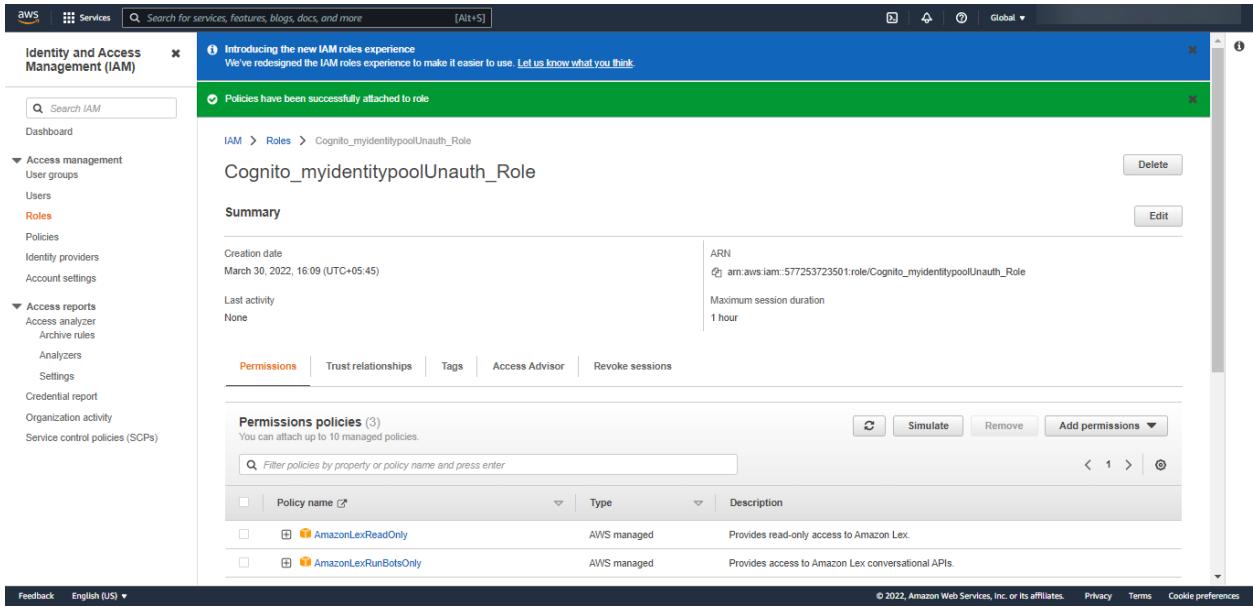
**Current permissions policies (0)**

**Other permissions policies (Selected 2/740)**

Policy name	Type	Description
<input checked="" type="checkbox"/>  AmazonLexReadOnly	AWS managed	Provides read-only access to Amazon Lex.
<input type="checkbox"/>  AmazonLexFullAccess	AWS managed	Provides full access to Amazon Lex via the AWS M...
<input checked="" type="checkbox"/>  AmazonLexRunBotsOnly	AWS managed	Provides access to Amazon Lex conversational APIs.

Cancel Attach policies (highlighted with a red box)

Feedback English (US) ▾ © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



## Task 9: Creating an S3 bucket

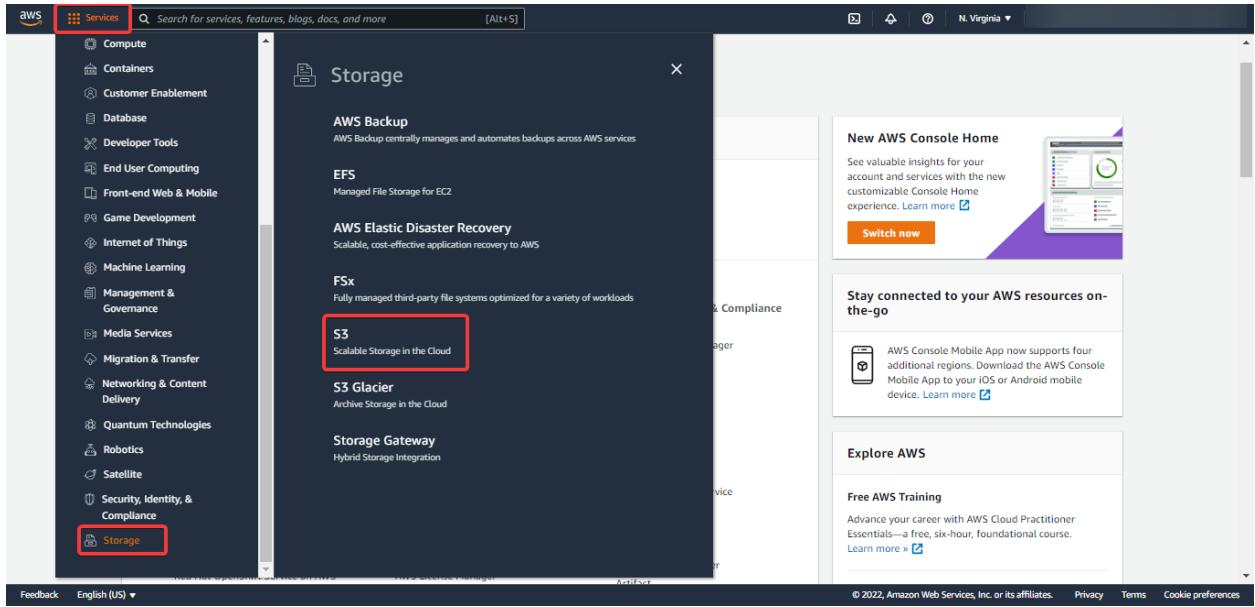
Now that you set up the security permissions, you must create an S3 bucket to host your webpage.

- First, download the following two webpage files and extract them to a local file directory.
- index.html
- error.html

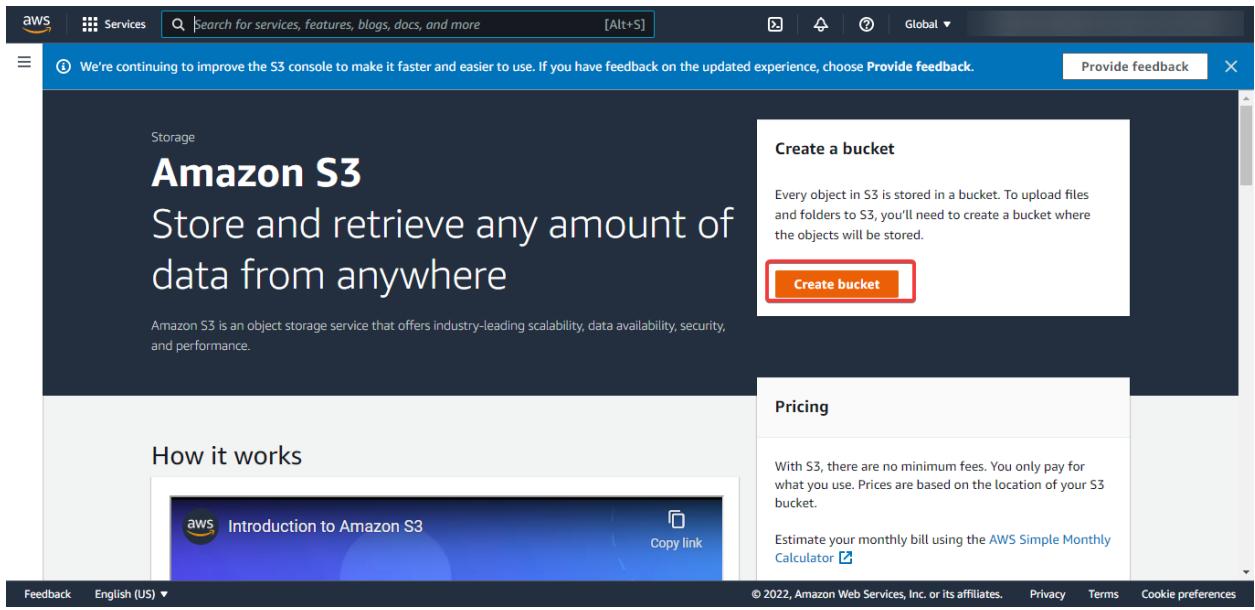
The index.html file includes the script that will load your bot.

[https://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/CUR-TF-200-ACMLFO-1/lab-6/202011/en\\_us/lab6.zip](https://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/CUR-TF-200-ACMLFO-1/lab-6/202011/en_us/lab6.zip)

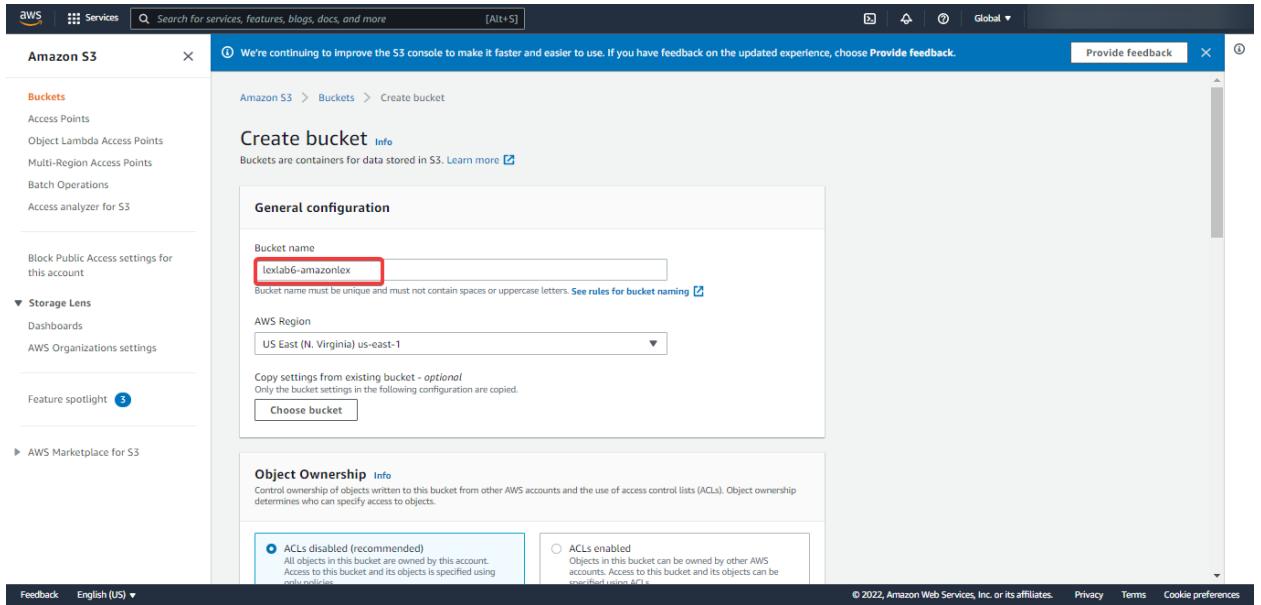
- On the AWS Management Console, on the **Services** menu, choose **S3**.



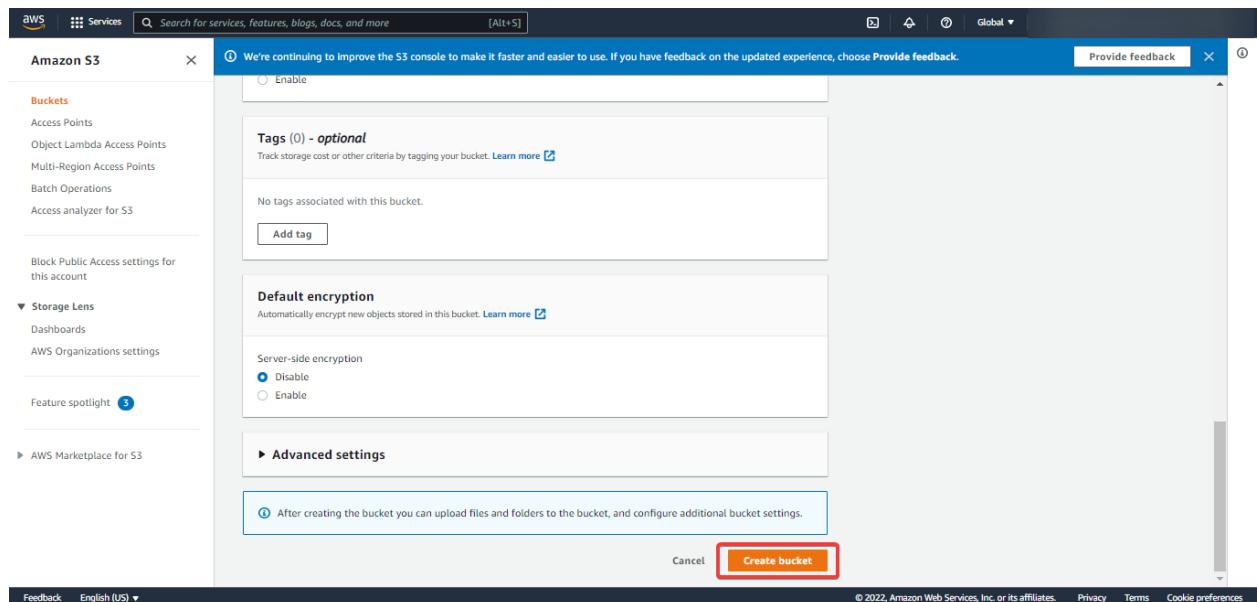
- On the **Amazon S3** page, choose **Create bucket**.



- Enter a name for your bucket. Because all S3 bucket names must be unique, try entering lexlab6 and six random letters.



- Select **Create bucket**.



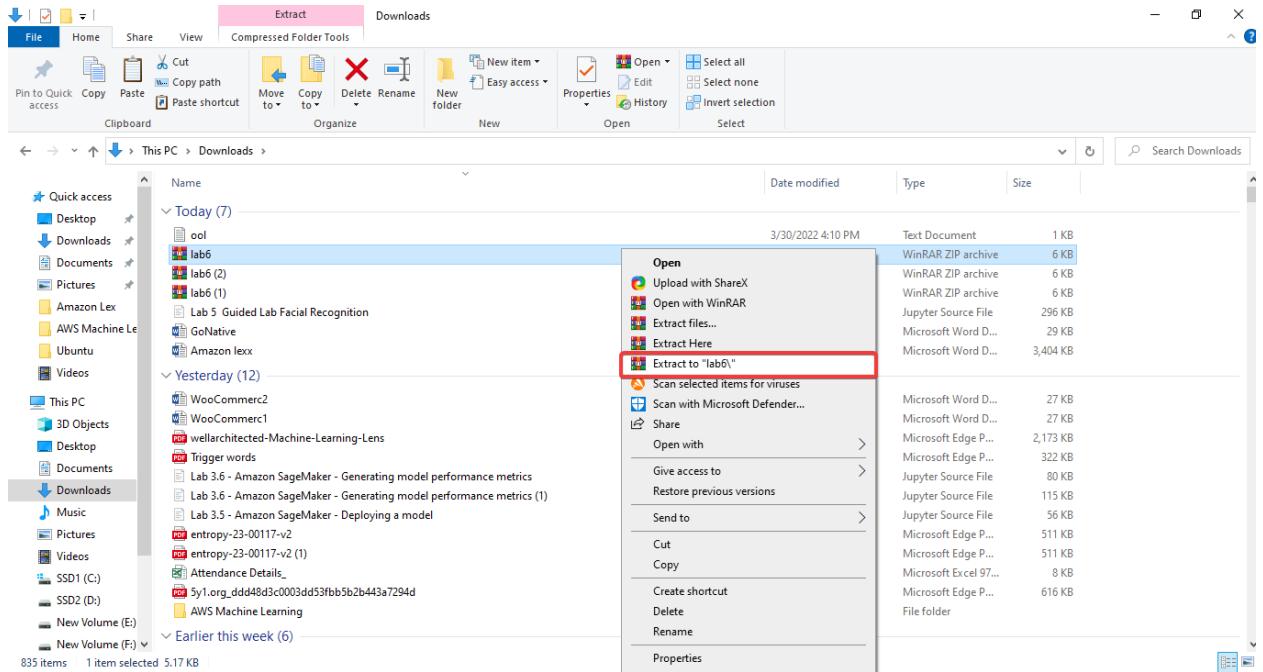
- On the **Buckets** page, select the bucket that you just created.

The screenshot shows the AWS S3 Bucket Overview page. At the top, there are several informational banners: one about improving the console, one confirming the creation of the bucket 'lexlab6-amazonlex', and one encouraging users to read the resources page. On the left sidebar, there are links for Buckets, Storage Lens, Feature spotlight, and AWS Marketplace for S3. The main content area displays an 'Account snapshot' and a table of buckets. The table has columns for Name, AWS Region, Access, and Creation date. A single row is shown for the bucket 'lexlab6-amazonlex', which is highlighted with a red box. The table header includes 'Name', 'AWS Region', 'Access', and 'Creation date'. The bucket details show it is located in 'US East (N. Virginia) us-east-1' and has 'Bucket and objects not public' access. It was created on 'March 30, 2022, 16:56:47 (UTC+05:45)'. At the bottom of the page, there is a URL bar with the address 'https://s3.console.aws.amazon.com/s3/buckets/lexlab6-amazonlex?region=us-eas...', and standard footer links for Privacy, Terms, and Cookie preferences.

- On the **Bucket overview** page, select **Upload**.

The screenshot shows the AWS S3 Bucket Objects page for the 'lexlab6-amazonlex' bucket. The top navigation bar is identical to the previous screenshot. The main content area shows an 'Objects (0)' section with a message stating 'No objects'. Below this, there is a toolbar with various actions: Copy, Copy S3 URI, Copy URL, Download, Open, Delete, Actions (with a dropdown arrow), Create folder, and Upload. The 'Upload' button is highlighted with a red box. There is also a search bar labeled 'Find objects by prefix'. At the bottom of the page, there are links for Feedback, English (US) ▾, and standard footer links for Privacy, Terms, and Cookie preferences.

Note: Extract downloaded zip files



- On the Upload page, select Add files.

The screenshot shows the AWS S3 Upload page. The URL is [Amazon S3 > Buckets > lexlab6-amazonlex > Upload](#). The main area is titled 'Upload' with an 'Info' link. It says 'Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more' with a link icon. Below is a dashed blue box containing the text 'Drag and drop files and folders you want to upload here, or choose Add files, or Add folders.' A 'Files and folders (0)' table is shown with columns: Name, Folder, Type, and Size. The table is empty. At the bottom, it says 'No files or folders' and 'You have not chosen any files or folders to upload.' Buttons for 'Feedback', 'English (US) ▾', '© 2022, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences' are at the bottom.

Browse to the location for the *index.html* and *error.html* files that you downloaded previously.

The screenshot shows the AWS S3 'Upload' interface. At the top, there's a search bar and a message: 'We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose Provide feedback.' Below the search bar, the breadcrumb navigation shows: Amazon S3 > Buckets > lexlab6-amazonlex > Upload. The main area is titled 'Upload' with an 'Info' link. A note says: 'Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more.' A large dashed blue box allows dragging and dropping files. Below it, a table lists 'Files and folders (2 Total, 17.0 KB)'. The table has columns: Name, Folder, Type, and Size. Two files are listed: 'error.html' (3.0 KB) and 'index.html' (13.9 KB). Both files are highlighted with a red box. At the bottom, there's a 'Destination' section with a dropdown set to 's3://lexlab6-amazonlex'. The footer includes links for Feedback, English (US), © 2022, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

- Select both files, and then select **Upload**.

The screenshot shows the AWS S3 'Upload' interface after selecting both files. The 'Files and folders' table now has two rows, both with the 'Name' column checked. The 'error.html' and 'index.html' files are highlighted with a red box. The 'Destination' section remains the same. Below it, there are sections for 'Permissions' (Grant public access and access to other AWS accounts) and 'Properties' (Specify storage class, encryption settings, tags, and more). At the bottom right, there are 'Cancel' and 'Upload' buttons, with the 'Upload' button highlighted with a red box. The footer is identical to the first screenshot.

Finally uploaded successfully.

The screenshot shows the AWS S3 console's 'Upload: status' page. At the top, a blue banner displays a message: 'We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose Provide feedback.' Below this, a green bar indicates a successful upload: 'Upload succeeded' with a link to 'View details below.' The main title 'Upload: status' is centered above a summary table. A note in a box says: 'The information below will no longer be available after you navigate away from this page.' The summary table has three columns: 'Destination' (s3://lexlab6-amazonlex), 'Succeeded' (2 files, 17.0 KB (100.00%)), and 'Failed' (0 files, 0 B (0%)). Below the table, tabs for 'Files and folders' (selected) and 'Configuration' are visible. A large section titled 'Files and folders (2 Total, 17.0 KB)' follows, which is currently empty. The bottom navigation bar includes links for 'Feedback', 'English (US)', and copyright information: '© 2022, Amazon Web Services, Inc. or its affiliates.' and 'Privacy Terms Cookie preferences'.

- After the upload is complete, return to the **Bucket overview** page by selecting **Close**.

This screenshot is identical to the one above, showing the 'Upload: status' page with a successful upload. However, the 'Close' button in the top right corner of the main content area is now highlighted with a red box, indicating the user action required to proceed.

- Go to **Properties** tab

The screenshot shows the AWS S3 console interface. At the top, there's a navigation bar with the AWS logo, 'Services', a search bar, and global settings. Below the navigation bar, a blue header bar contains two informational messages: 'We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose Provide feedback.' and 'Read the S3 resources page for documentation and technical content.' The main content area shows the 'lexlab6-amazonlex' bucket. A breadcrumb navigation path 'Amazon S3 > Buckets > lexlab6-amazonlex' is visible. Below the path, the bucket name 'lexlab6-amazonlex' is shown with an 'Info' link. A navigation bar below the bucket name includes tabs for 'Objects' (selected), 'Properties' (highlighted with a red box), 'Permissions', 'Metrics', 'Management', and 'Access Points'. The 'Properties' tab displays the 'Objects (2)' section, which lists 'error.html' as an HTML file last modified on March 30, 2022. Below this, the 'Static website hosting' section is partially visible.

- On the **Properties** tab, scroll down to the **Static website hosting** section and select **Edit**.

The screenshot shows the 'Properties' tab of the AWS S3 console for the 'lexlab6-amazonlex' bucket. The 'Static website hosting' section is open, revealing its configuration options. The 'Edit' button for this section is highlighted with a red box. The section includes a note: 'Use this bucket to host a website or redirect requests.' Below this, there are two dropdown menus: 'Static website hosting' (set to 'Disabled') and 'Index document' (set to 'index.html'). Other sections like 'Requester pays' and 'API response' are also visible but not currently being edited.

- Select **Enable**.
- For the **Index document**, enter: index.html
- For the error page, enter: error.html

The screenshot shows the 'Edit static website hosting' configuration page for a bucket named 'lexab6-amazonlex'. The 'Static website hosting' section is active, with the 'Enable' radio button selected. Under 'Hosting type', the 'Host a static website' option is chosen. A note below explains that content must be publicly readable. The 'Index document' field contains 'index.html', and the 'Error document - optional' field contains 'error.html'. The bottom right corner features a red box around the 'Save changes' button.

- Choose **Save changes**.

This screenshot is identical to the previous one, showing the 'Edit static website hosting' configuration page. The 'Save changes' button at the bottom right is now highlighted with a red box.

## Task 10: Updating and testing the demonstration file

You must now update the demonstration HTML file so that it uses the Amazon Cognito identity pool that you created in Task 6.

Use a text editor to make the following changes to the HTML page.

- On line 144 of the script, add the *IdentitypoolID* for the identity pool that you created in Task 6.

```
index.html
131 <form id="chatform" style="margin-bottom: 10px" onsubmit="return pushChat();">
132   <p style="width: 380px; font-size: 0.8em; line-height: 1.2em">
133     <input type="text" id="wisdom" size="80" value="" placeholder="What do you want to do?">
134   </p>
135 </form>
136
137 <script type="text/javascript">
138   // set the focus to the input box
139   document.getElementById("wisdom").focus();
140
141   // Initialize the Amazon Cognito credentials provider
142   AWS.config.region = 'us-east-1'; // Region
143   AWS.config.credentials = new AWS.CognitoIdentityCredentials({
144     IdentityPoolId: 'us-east-1:1577cedd-ba2c-4565-8b89-ef588640f619',
145   });
146
147   var lexruntime = new AWS.LexRuntime();
148   var lexUserId = 'chatbot-demo' + Date.now();
149   var sessionAttributes = {};
150   var responseCardOptions = null;
151
152   function pushChat(textToPush, labelToShow) {
153
154     // if there is text to be sent...
155     var wisdomText = document.getElementById("wisdom").value;
```

- On lines 185 and 186 of the script, add the *botAlias* and *botName* for your bot.

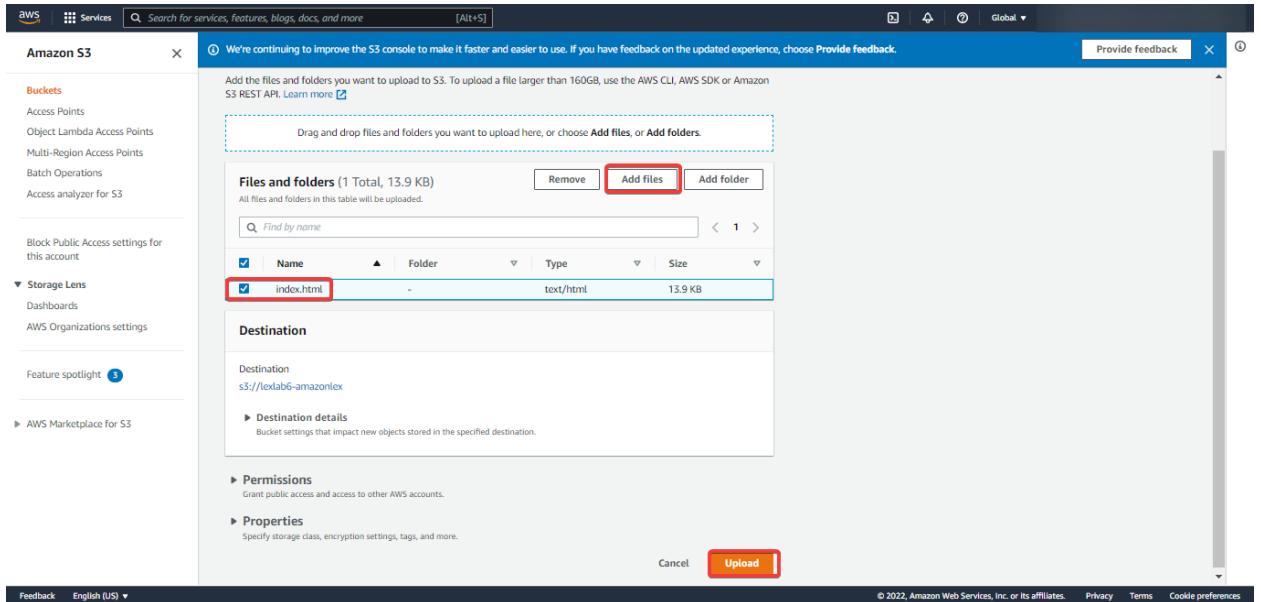
```
index.html
176           labelToShow = item.text;
177           // break out of the loop
178           i = responseCardOptions.length + 1;
179         }
180       }
181     }
182
183     // send it to the Lex runtime
184     var params = {
185       botAlias: 'Grace',
186       botName: 'ScheduleAppointment',
187       // if the caller has specifically provided text to be pushed, use it
188       inputText: (textToPush ? textToPush : wisdom),
189       userId: lexUserId,
190       sessionAttributes: sessionAttributes
191     };
192
193     // if the caller has specifically provided a label to be shown, show it
194     showRequest((labelToShow ? labelToShow : wisdom));
195
196     // do a KB search if requested (not used when the input is from a response card)
```

- Save your updated HTML page locally.

The screenshot shows the AWS S3 console. On the left, there's a sidebar with links like 'Buckets', 'Object Lambda Access Points', 'Multi-Region Access Points', 'Batch Operations', 'Access analyzer for S3', 'Block Public Access settings for this account', 'Storage Lens', 'Dashboards', 'AWS Organizations settings', 'Feature spotlight', and 'AWS Marketplace for S3'. The main area is titled 'Amazon S3' and shows an 'Account snapshot' with a 'View Storage Lens dashboard' button. Below it is a table titled 'Buckets (1) Info' with one item: 'lexlab6-amazonlex' (US East (N. Virginia) us-east-1). The 'lexlab6-amazonlex' row is highlighted with a red box.

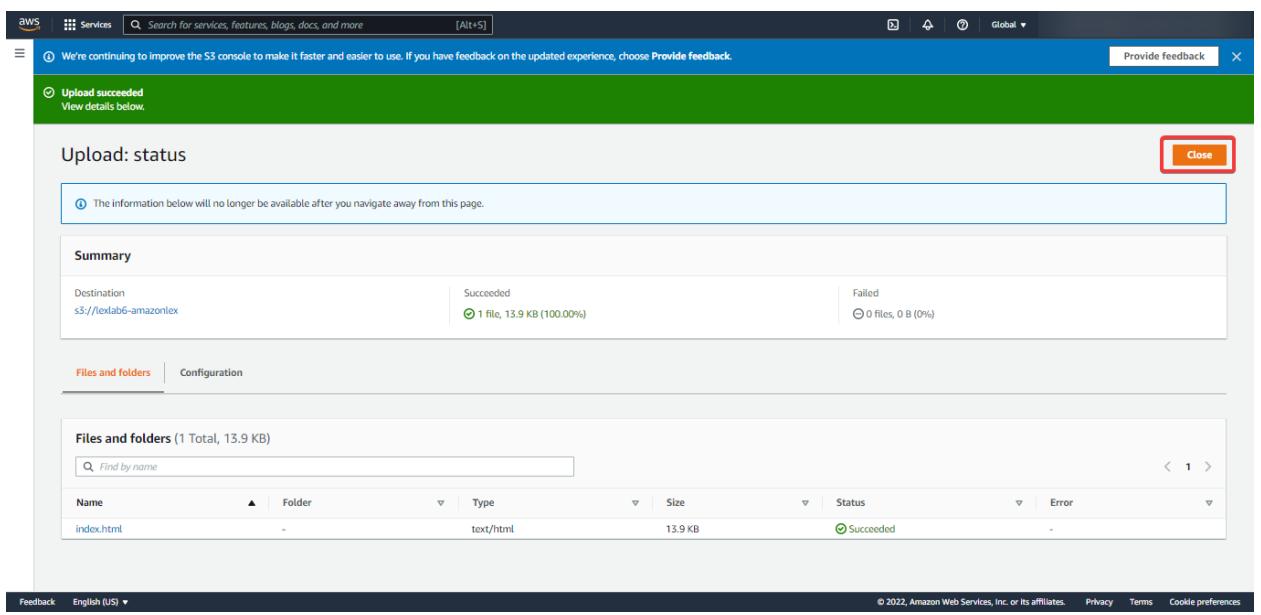
The screenshot shows the AWS S3 'Upload: status' page. At the top, there's a green banner with the message 'Upload succeeded' and 'View details below.' Below the banner, the title is 'Upload: status' with a 'Close' button. A note says 'The information below will no longer be available after you navigate away from this page.' Under 'Summary', it shows 'Destination s3://lexlab6-amazonlex' and 'Succeeded 1 file, 13.9 KB (100.00%)'. There are tabs for 'Files and folders' (selected) and 'Configuration'. The 'Files and folders' section shows a table with one item: 'index.html' (text/html, 13.9 KB, Succeeded). The bottom of the page includes standard AWS footer links: Feedback, English (US), © 2022, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

- Choose the **Objects** tab for the bucket that you created in Task
- Choose **Upload** and upload the HTML file that you edited.



The screenshot shows the AWS S3 console upload interface. On the left, a sidebar lists various AWS services like Buckets, Access Points, and Storage Lens. The main area has a message about improving the console. Below it, a table shows one file ('index.html') selected. The 'Add files' button is highlighted with a red box. The 'Destination' section shows the bucket ('s3://lexlab6-amazonlex'). The 'Permissions' and 'Properties' sections are collapsed. At the bottom right are 'Cancel' and 'Upload' buttons, with 'Upload' highlighted with a red box.

- Choose **Exit** to close the **Upload** page.



The screenshot shows the AWS S3 console after a successful upload. A green banner at the top says 'Upload succeeded'. Below it, a summary table shows the destination bucket ('s3://lexlab6-amazonlex') and upload results ('Succeeded: 1 file, 13.9 KB (100.00%)'). The 'Failed' section is empty. The 'Files and folders' tab is selected, showing the uploaded file 'index.html' in a table with columns for Name, Type, Size, Status, and Error. The 'Status' column for 'index.html' shows a green checkmark and 'Succeeded'. At the bottom right are 'Feedback', 'English (US)', and 'Close' buttons, with 'Close' highlighted with a red box.

- Choose the **Permissions** tab.

We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose [Provide feedback](#).

[How to optimize your costs on S3.](#)

Amazon S3 > Buckets > lexlab6-amazonlex

lexlab6-amazonlex [Info](#)

Objects Properties **Permissions** Metrics Management Access Points

**Permissions overview**

Access

Bucket and objects not public

**Block public access (bucket settings)**

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Feedback English (US) © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- In the **Block public access** section, choose **Edit**.

We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose [Provide feedback](#).

Access

Bucket and objects not public

**Block public access (bucket settings)**

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

**Edit**

**Block all public access**

On

► Individual Block Public Access settings for this bucket

**Bucket policy**

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

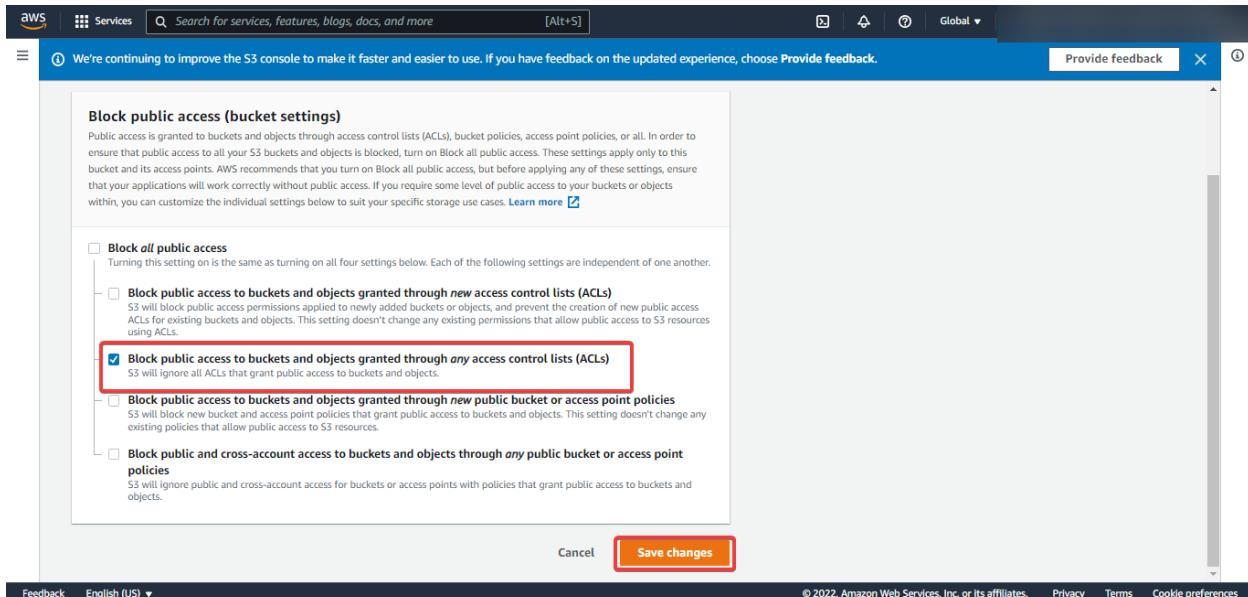
**Public access is blocked because Block Public Access settings are turned on for this bucket**

To determine which settings are turned on, check your Block Public Access settings for this bucket. Learn more about using Amazon S3 Block Public Access [Learn more](#)

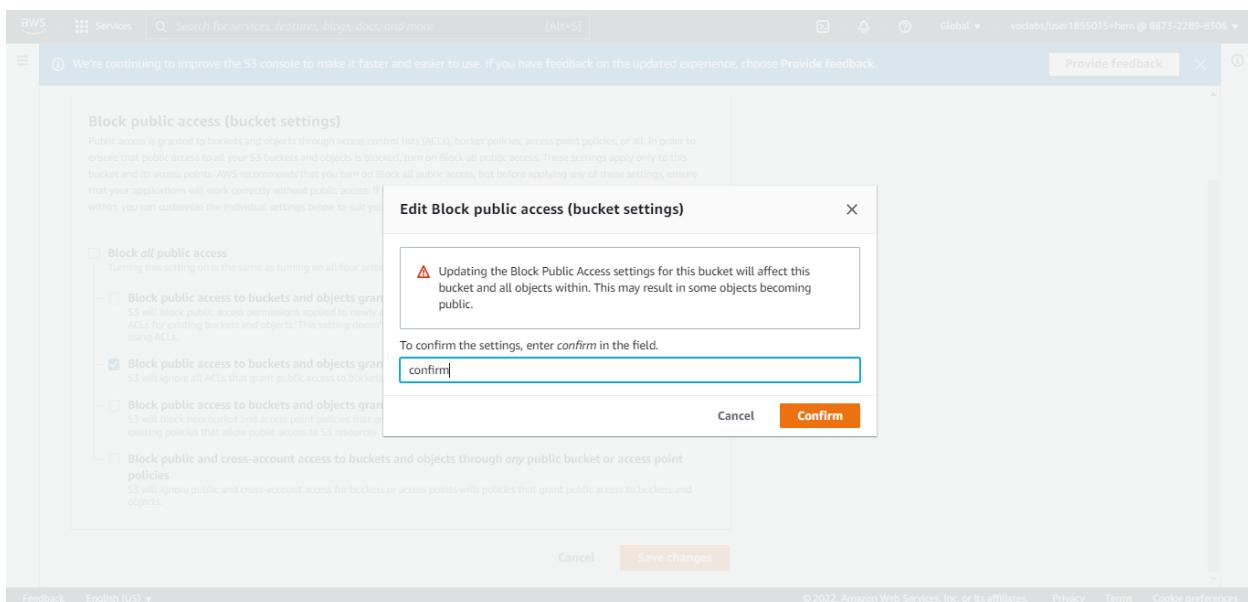
https://s3.console.aws.amazon.com/s3/# © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Clear the following options:
- Block all public access**
- Block public access to buckets and objects granted through *new* public bucket or access point policies**
- Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**

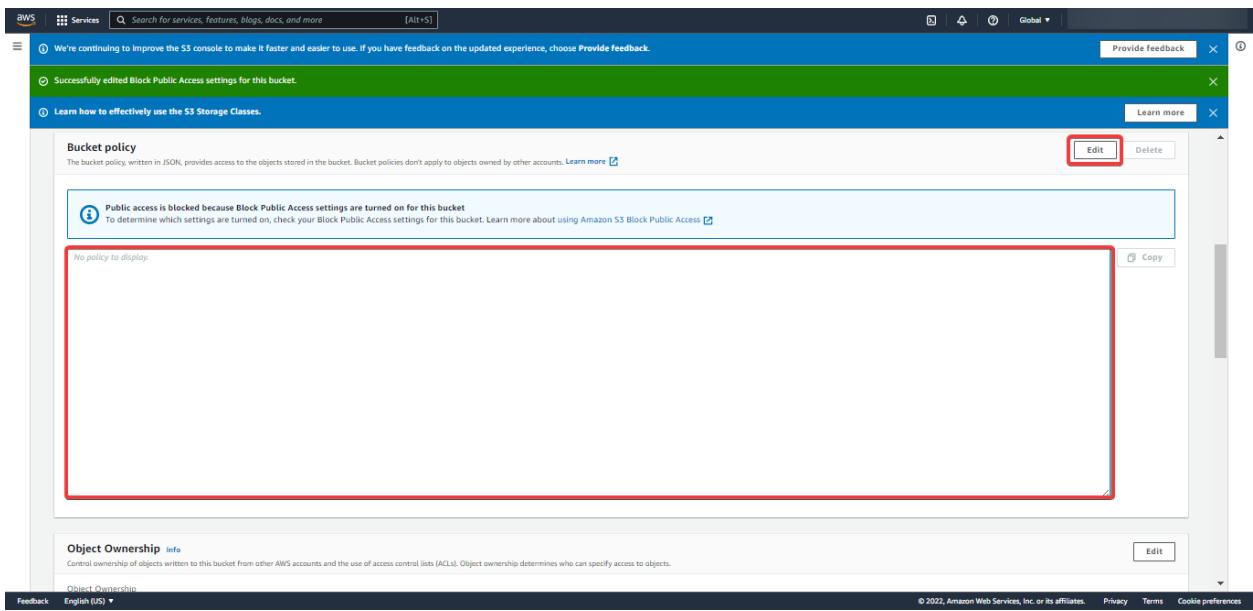
- Choose **Save changes**.



- In the confirmation dialog box, enter Confirm and then choose **Confirm**.



- In the **Bucket policy** section, choose **Edit**.

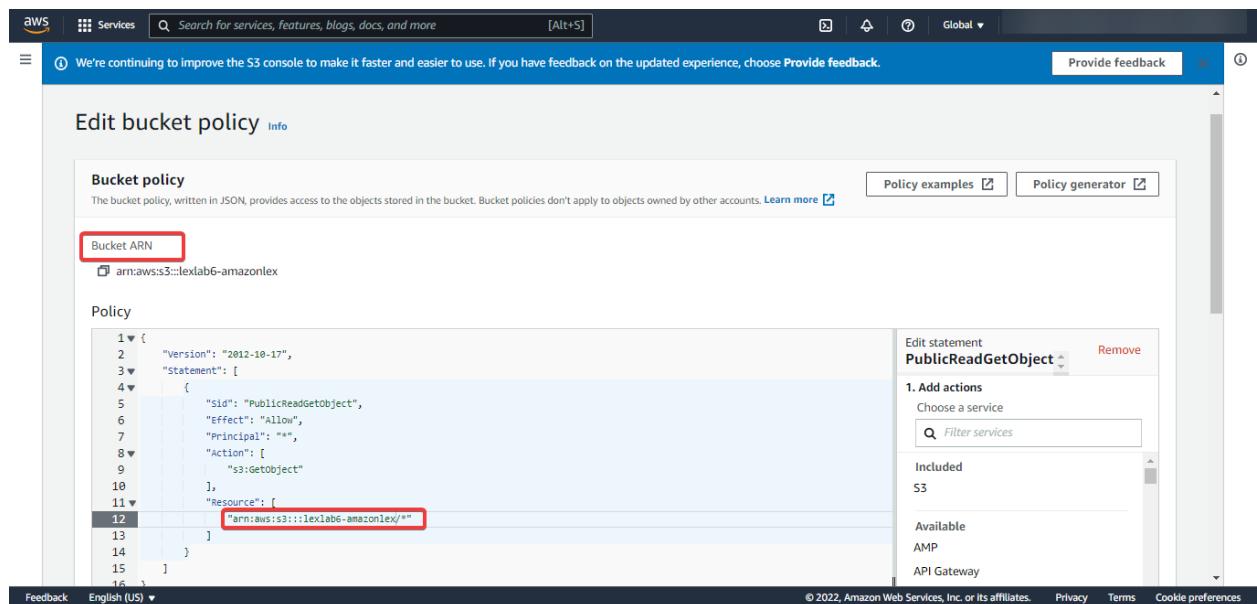


- To grant public read access to the webpage, copy the following bucket policy. In the **Bucket policy editor**, paste the policy.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "PublicReadGetObject",  
      "Effect": "Allow",  
      "Principal": "*",  
      "Action": [  
        "s3:GetObject"  
      ],  
      "Resource": [  
        "arn:aws:s3:::example.com/*"  
      ]  
    }  
  ]  
}
```

```
]  
}
```

- In the **Resource** part of the policy, edit the value of the Amazon Resource Name (ARN) by replacing *example.com* with the name of your bucket.



- Choose **Save changes**. You will get a warning that the bucket has public access.

The screenshot shows the AWS IAM Policy Editor interface. At the top, there's a banner stating, "We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose Provide feedback." Below this, there's a JSON editor area with the text "JSON Ln 12, Col 47". It includes status indicators: Security: 0, Errors: 0, Warnings: 0, and Suggestions: 0. A red box highlights an "Unexpected error" message: "Network Failure - verify permissions to access-analyzer:ValidatePolicy in IAM to use policy validation." To the right, there are sections for "2. Add a resource" and "3. Add a condition (optional)". At the bottom right are "Cancel" and "Save changes" buttons.

The screenshot shows the AWS S3 Bucket Permissions page for the bucket "lexlab6-amazonlex". The top navigation bar includes links for Feedback, English (US), and various AWS services. A green banner at the top says, "Successfully edited bucket policy." Below this, another banner says, "Learn how to effectively use the S3 Storage Classes." The main content area shows the bucket name "lexlab6-amazonlex" with an "Info" link and a "Publicly accessible" button. A red box highlights the "Permissions" tab in the navigation bar. Under the "Permissions overview" section, there's an "Access" section with a "Public" link. A callout box titled "Block public access (bucket settings)" provides information about public access settings. At the bottom, there's a URL bar with the address "https://s3.console.aws.amazon.com/s3/buckets/lexlab6-amazonlex?region=us-east-1#permissions&tab=permissions" and standard footer links for Privacy, Terms, and Cookie preferences.

- Choose the **Properties** tab.

The screenshot shows the AWS S3 console. At the top, there are two informational banners: one about improving the S3 console and another encouraging users to learn about S3 Storage Classes. Below these, the navigation path is 'Amazon S3 > Buckets > lexlab6-amazonlex'. The main title is 'lexlab6-amazonlex' with a 'Info' link. A red box highlights the 'Properties' tab in the navigation bar. Under the 'Bucket overview' section, there is a 'Requester pays' configuration panel and a 'Static website hosting' configuration panel. The URL for the static website is highlighted with a red box.

- Scroll down to the **Static website hosting** section.
- At the bottom of the **Static website hosting** section, choose the URL

This screenshot shows the 'Static website hosting' configuration for the 'lexlab6-amazonlex' bucket. It includes fields for 'Static website hosting' (set to 'Enabled'), 'Hosting type' (set to 'Bucket hosting'), and the 'Bucket website endpoint' (set to 'http://lexlab6-amazonlex.s3-website-us-east-1.amazonaws.com'). The URL 'http://lexlab6-amazonlex.s3-website-us-east-1.amazonaws.com' is highlighted with a red box.

- Visit <http://lexlab6-amazonlex.s3-website-us-east-1.amazonaws.com/>

## Amazon Lex Demo

← → ⌂ Not secure | lexlab6-amazonlex.s3-website-us-east-1.amazonaws.com

# Amazon Lex - Appointment BOT

The screenshot shows a web-based chat interface for the Amazon Lex Appointment Bot. The conversation history is as follows:

- User: Do you have appartment ?
- Bot: **Specify Appointment Type**  
What type of appointment would you like to schedule?
- User: root canal (60 min)
- Bot: **Specify Date**  
When would you like to schedule your root canal?
- User: 4-1 (Fri)
- Bot: **Confirm Appointment**  
Is 4:00 p.m. on 2022-04-01 okay?
- User: yes
- Bot: Okay, I have booked your appointment. We will see you at 4:00 p.m. on 2022-04-01

At the bottom, there is a text input field with the placeholder "What do you want to do?"

## **Reference**

*Amazon Lex* (no date). Available at: <https://us-east-1.console.aws.amazon.com/lex/home?region=us-east-1> (Accessed: 30 March 2022).

Ananta Wijaya (2021) *Lab 6 - Amazon Lex - Create a chatbot*. Available at: [https://www.youtube.com/watch?v=N\\_IwwYoaYHM](https://www.youtube.com/watch?v=N_IwwYoaYHM) (Accessed: 30 March 2022).

Lab 6 - Amazon Lex - Create a chatbot (no date). Available at: <https://awsacademy.instructure.com/courses/15943/modules/items/1322353> (Accessed: 30 March 2022).