**Amazon Lex Master Chatbot Workshop**

Amazon Lex is a service for building conversational interfaces into any application using voice and text. Amazon Lex provides the advanced deep learning functionalities of automatic speech recognition (ASR) for converting speech to text, and natural language understanding (NLU) to recognize the intent of the text, to enable you to build applications with highly engaging user experiences and lifelike conversational interactions. With Amazon Lex, the same deep learning technologies that power Amazon Alexa are now available to any developer, enabling you to quickly and easily build sophisticated, natural language, conversational bots ("chatbots").

This workshop walks through the steps of creating an interactive Master chatbot using the AWS Lex service.

**MasterBot**

MasterBot is a transactional chat bot that can help one get a quick answer or transfer to a worker bot (covered in a separate workshop), e.g. AddressBot.

Consider this conversation:

MasterBot: Welcome to CRA. How can I help you?

User: I want to do my personal taxes

MasterBot: You'd like more information on personal income tax return. Is that right?

User: yes

MasterBot: Action (Lambda) – Direct to a How-to article.

Or this conversation (covered in a separate workshop):

MasterBot: Welcome to CRA. How can I help you?

User: I want to change my address.

MasterBot: Action (Lambda) – redirect to AddressUpdateBot.

First, we will create the Amazon Lex bot. Then, we'll add some Lambda Functions to bring it to life.

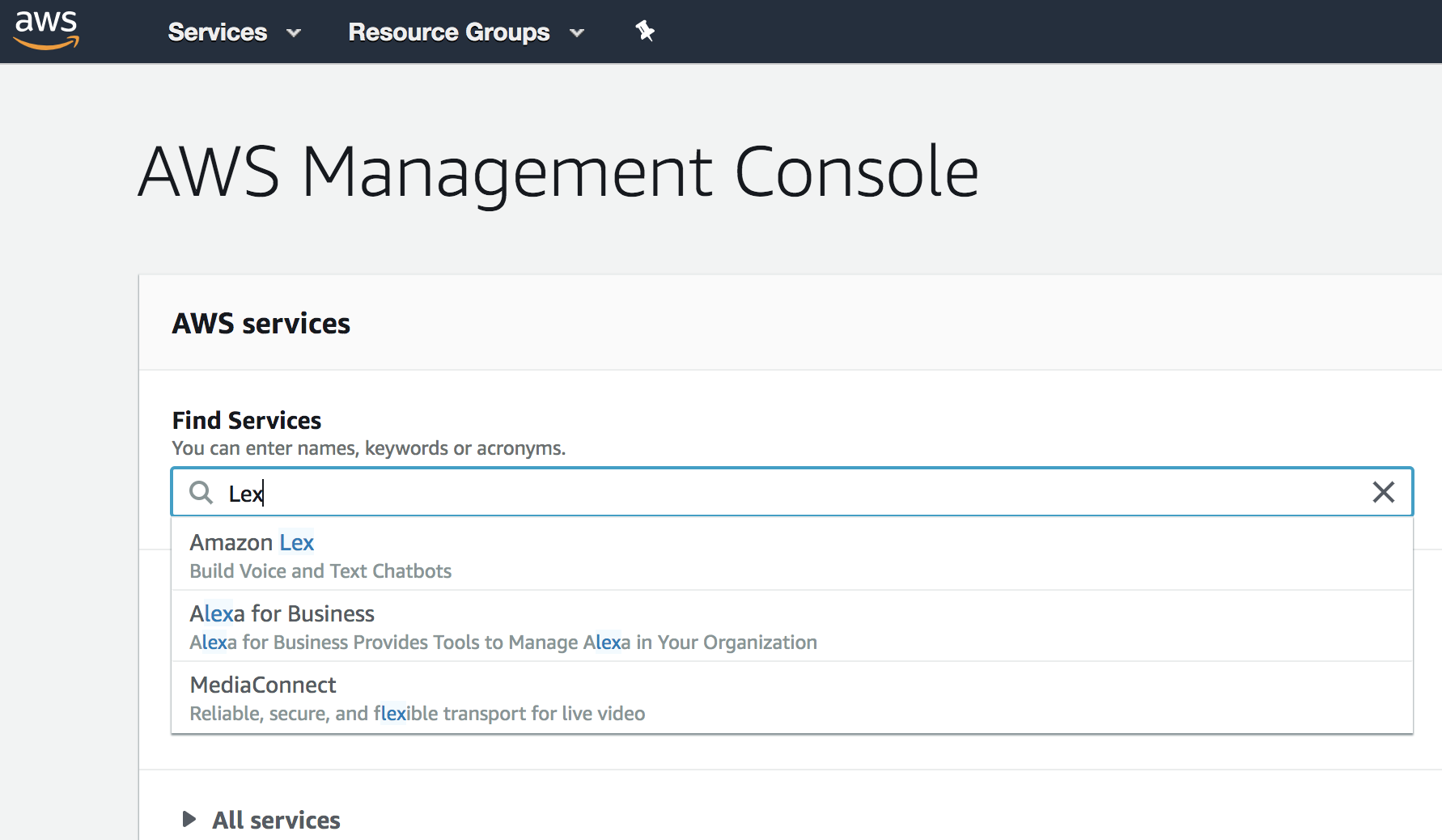
**Amazon Lex bot**

We will be creating a custom bot in the console and applying the necessary intents and custom slots in order to understand the visitor needs.

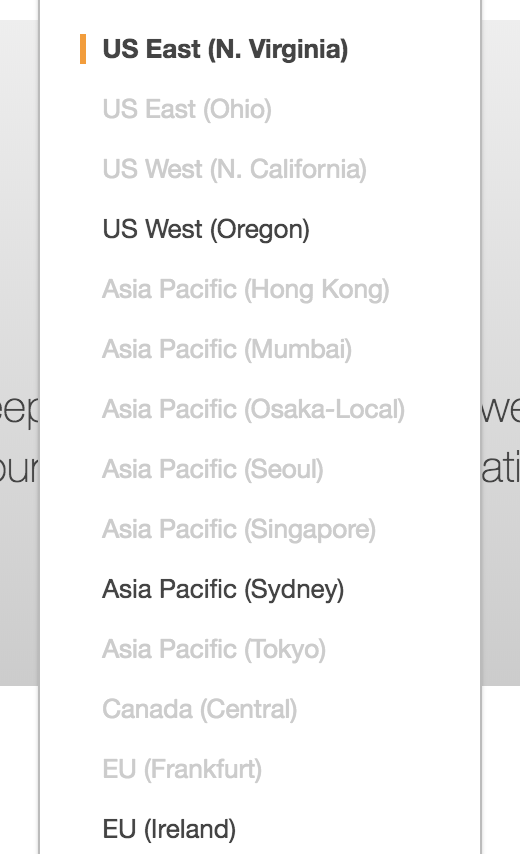
**Navigate to Lex in the AWS Console**

The login credentials are on the screen.

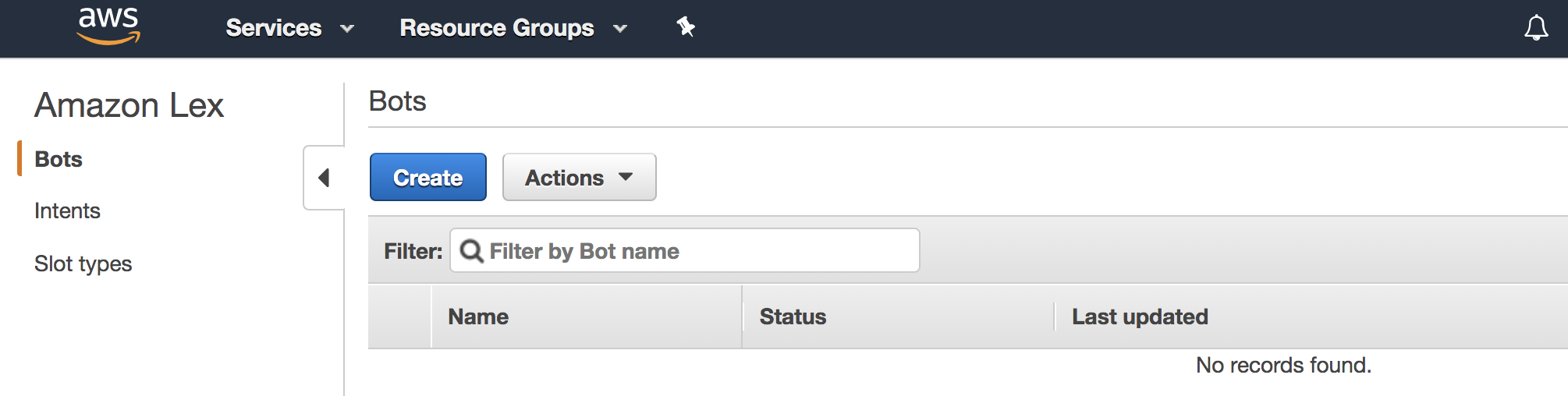
Once you are logged into your AWS account, you will be presented with a list of AWS services and a search box. Type "Lex" into the search box and choose the "Amazon Lex" service.



You may be directed to select at region. At the time of writing, Lex is only available in the N. Virginia, Oregon, Sydney and Ireland regions. Choose the US East (N. Virginia) region.



If this is your first time in the Amazon Lex console, you may be prompted with a welcome page. Choose "Get Started" and then "Cancel" so that you are at the Lex page with "Bots", "Intents", and "Slot Types" on the left hand navigation. If this isn't your first time, you will be brought directly to the Lex console.

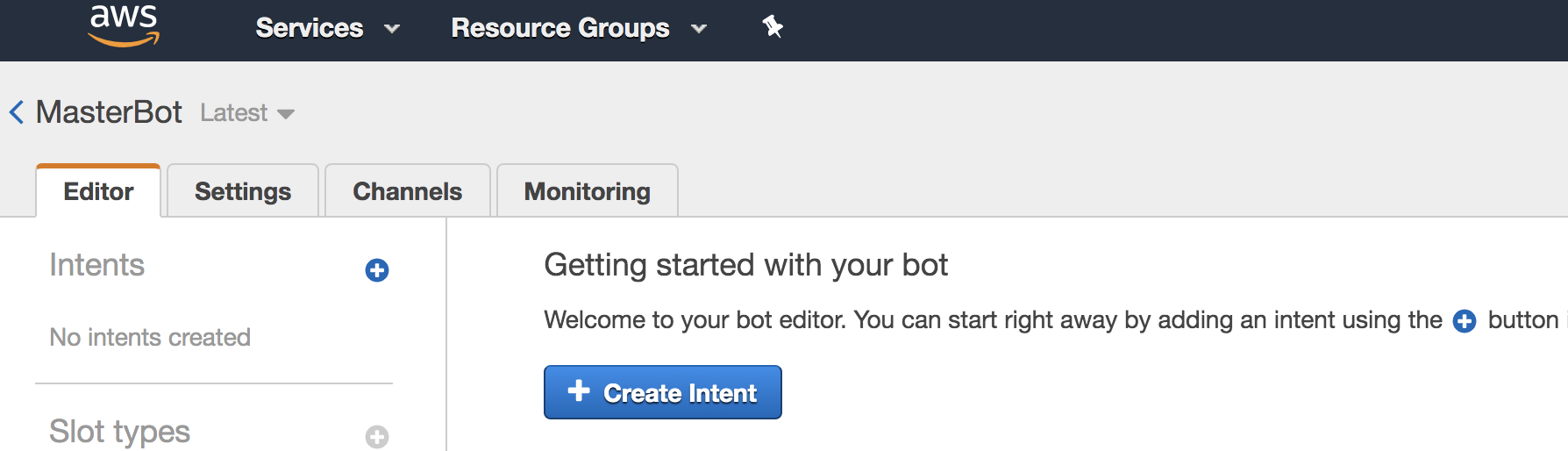


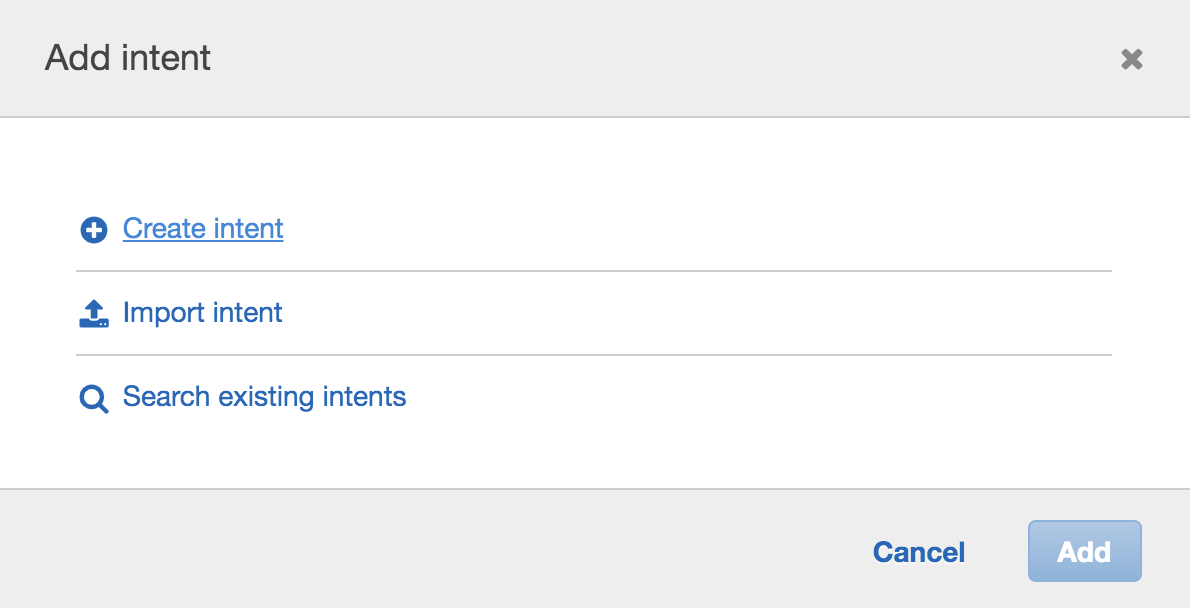
**Create bot**

First, we'll create the bot.

1. From the Amazon Lex console, select "Bots" on the left and then choose "Create" and choose "Custom Bot". Create a Custom bot with these settings (you can see these in the "Settings" tab later)
   * Bot name: {Your Initials}\_MasterBot
     + E.g. RP\_MasterBot
   * Output voice: Salli (or another)
   * Session timeout: 5 min
   * IAM role: (accept the default) AWSServiceRoleForLexBots
   * COPPA: (our bot is not directed at children) No

**Create CRA Inquiry Intent**





Click on “Create Intent”, add a new Intent called {Your Initials}\_CRAInquiryIntent with the following settings and click "Save Intent" to save the Intent.  
e.g. RP\_CRAInquiryIntent

Add the following options to your intent:

1. Sample Utterances: add these to the list of sample utterances so the bot recognizes similar phrases (each entry on a separate line)

I would like to do my {TaxCategory} taxes

I want to do my {TaxCategory} taxes

How can I file a {TaxCategory} income tax return?

I want to file my {TaxCategory} income tax return

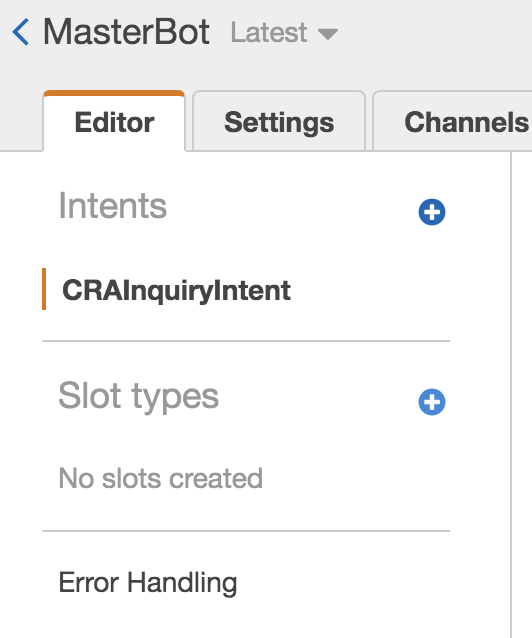
I need help with my {TaxCategory} income tax return

1. Lambda initialization and validation (leave unchecked)
2. Fulfillment: choose "Return parameters to client" for now
3. Confirmation prompt: You'd like more information on {TaxCategory} income tax return. Is that right? to confirm and Please rephrase your question. to cancel.

When everything is entered, choose "Save Intent" at the bottom.

**Create Slot types**

In the upper left hand corner, select the "+" next to "Slot Type" to add new slots to your account.



Add the following Slot types (each value should be a separate entry). To work independently in a shared environment, use your initials in the names {Your Initials} taxCategoryType. e.g., {RP}taxCategoryType.

Note: Although they are saved with the AWS Account, ***Slot Types will only show up in the list on the left when they are associated in the next step.*** Keep creating slots until you've completed all in the following table:

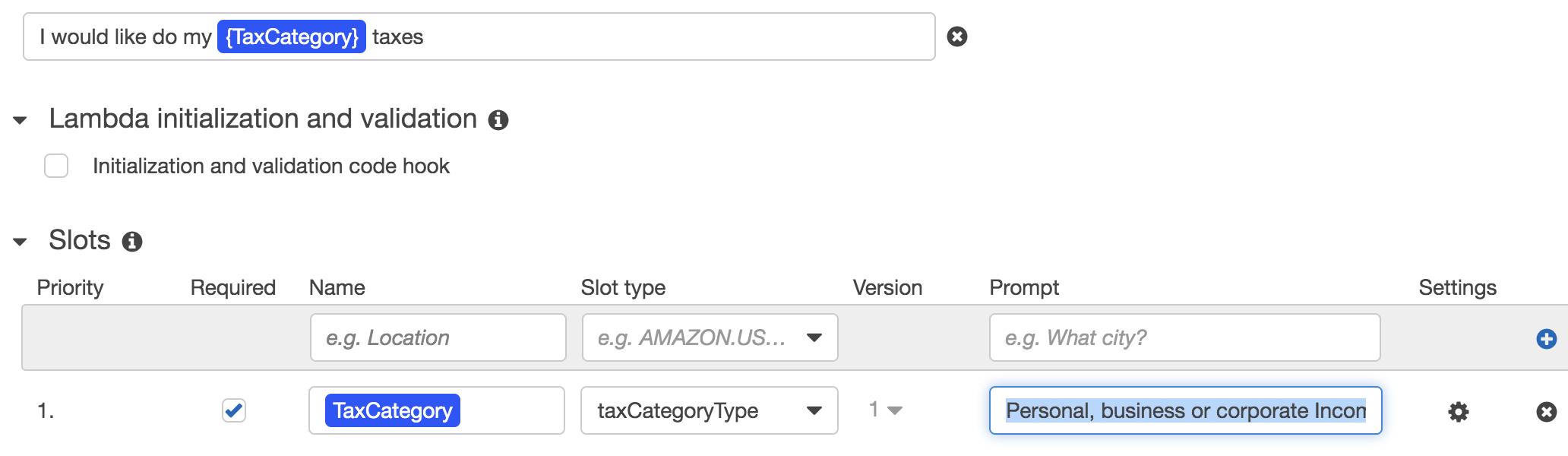
Choose "Save slot type" when each one is completed.

(*Slot types are shared at the account level so Description text would help other developers determine if they can reuse this Slot type.)*

| **Slot type name** | **Description** | **Values (each entry on a separate line)** |
| --- | --- | --- |
| XXXtaxCategoryType | *Income tax return type* | personal; business; corporate |
|  |  |  |
|  |  |  |

**Add Slots to the Intent**

Navigate back to the intent page of your CRAInquiryIntentXXX, locate "Slots" midway down the page.

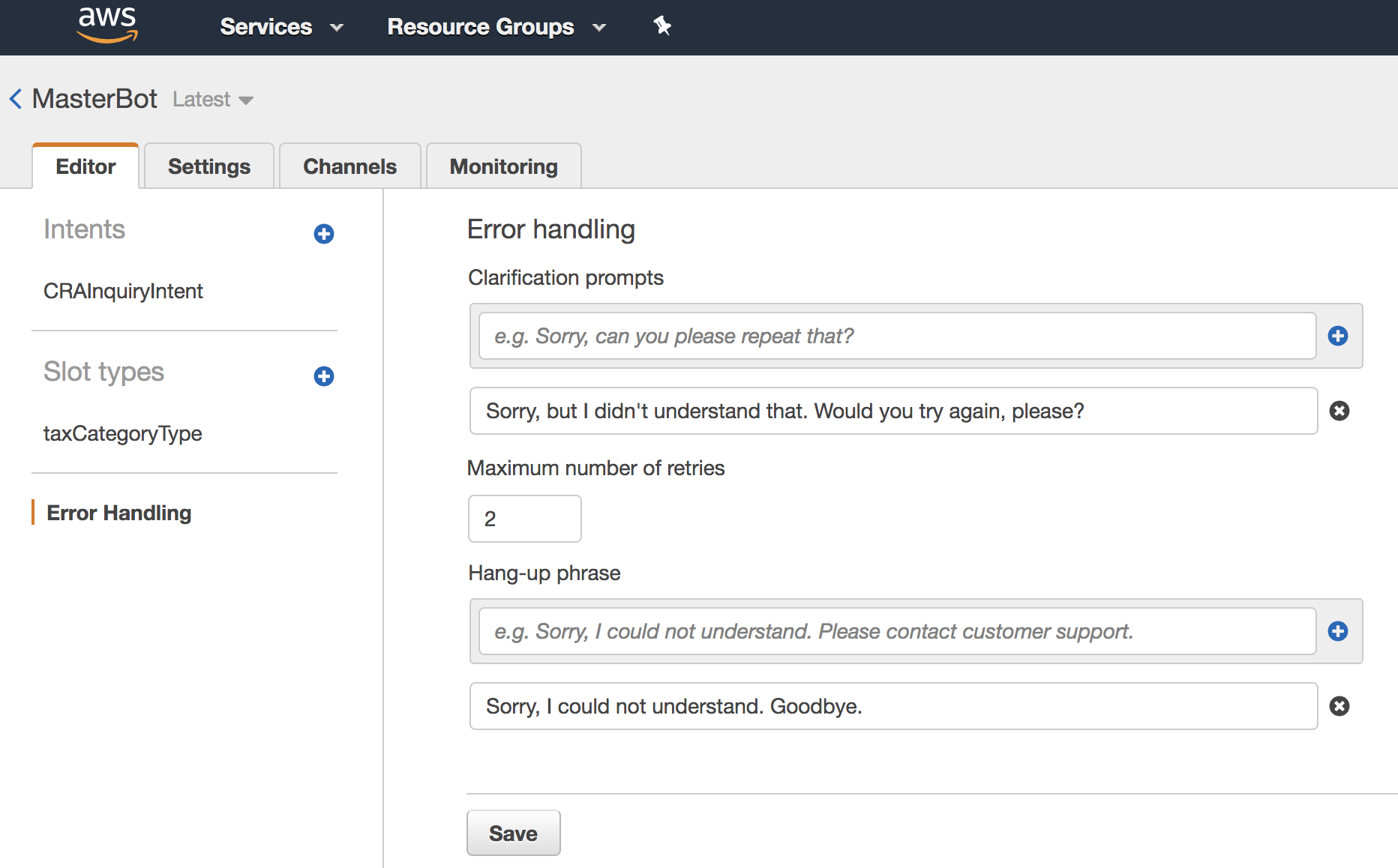


Add the following entries to the list of Slots, choosing the Slot Types created above. Click "Save Intent".

| **Required** | **Name** | **Slot type** | **Prompt** |
| --- | --- | --- | --- |
| Yes | TaxCategory | XXXtaxCategoryType | Personal, business or corporate Income Tax return? |
|  |  |  |  |
|  |  |  |  |

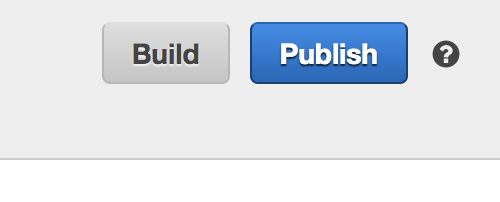
**Error Handling**

1. Navigate to the Error handling settings and set the following



* Prompts: (one prompt) Sorry, but I didn't understand that. Would you try again, please?
* Maximum number of retries: 2
* Hang-up phrase: (one phrase) Sorry, I could not understand. Goodbye.

**Test**



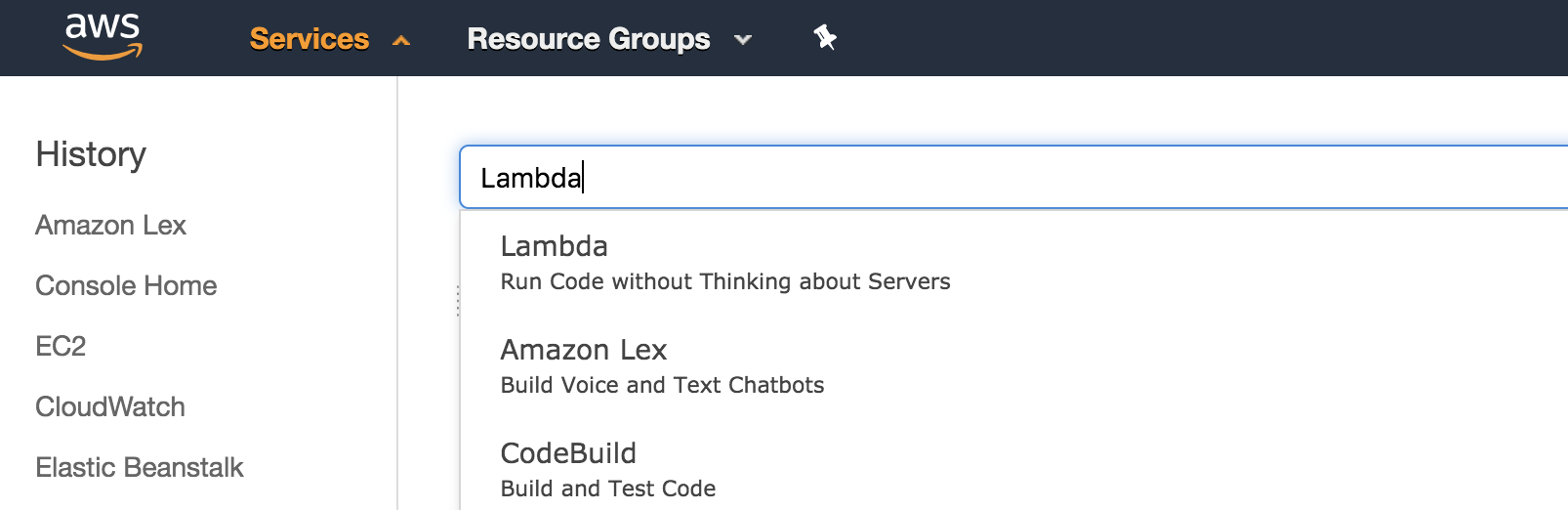
Click the Build icon in the upper right hand corner to build the app. This can take a few minutes.

Once built, a new panel will appear on the right of the Amazon Lex Console where you can test some of the Utterances in the Test Bot dialog. For example, if you say I want to do my personal taxes does Lex correctly map personal to the TaxCategory slot?

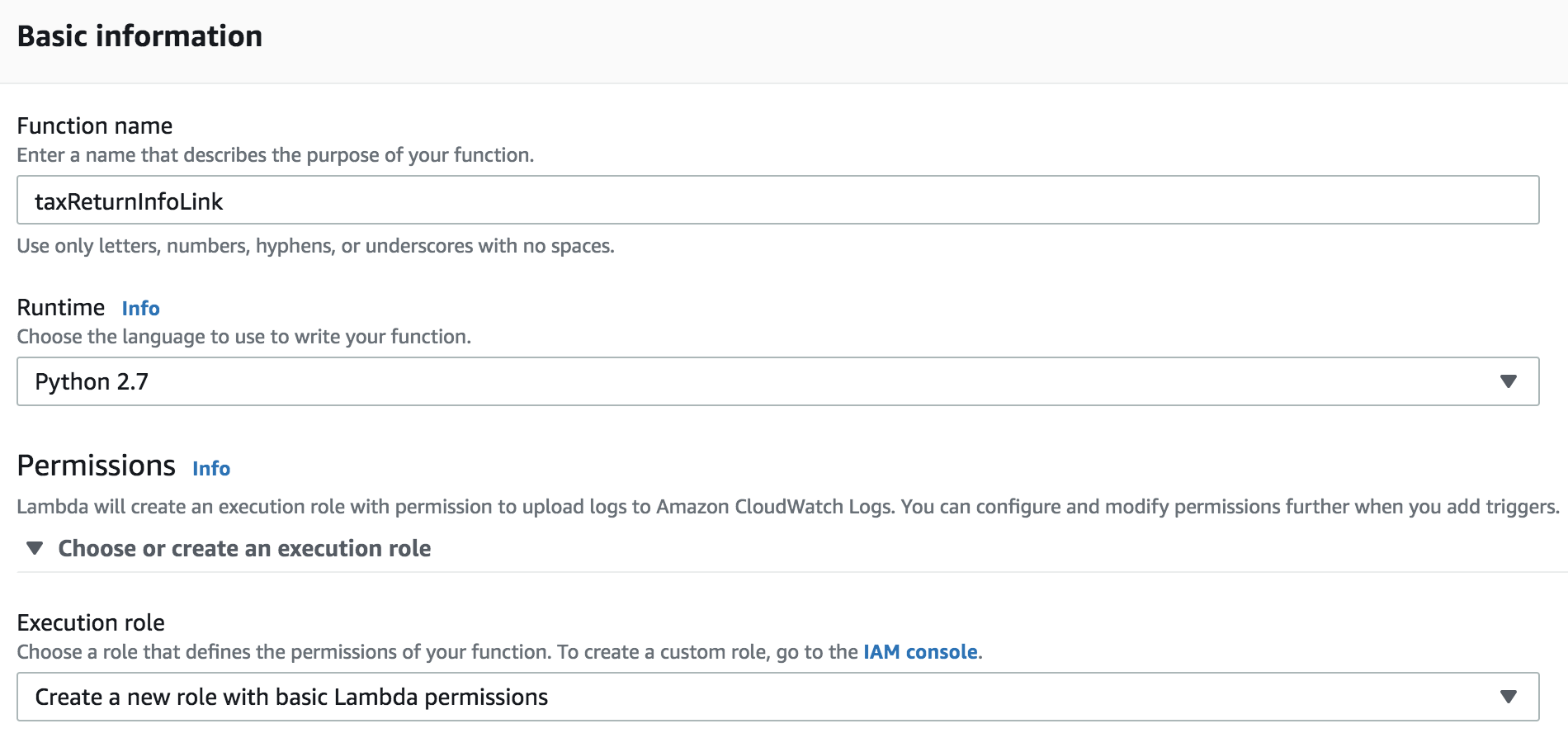
**Lambda Function**

Now that we've tested that our application works, let's add more logic to validate our choices and handle the processing of the order. We'll be using another service called AWS Lambda to do this. AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume.

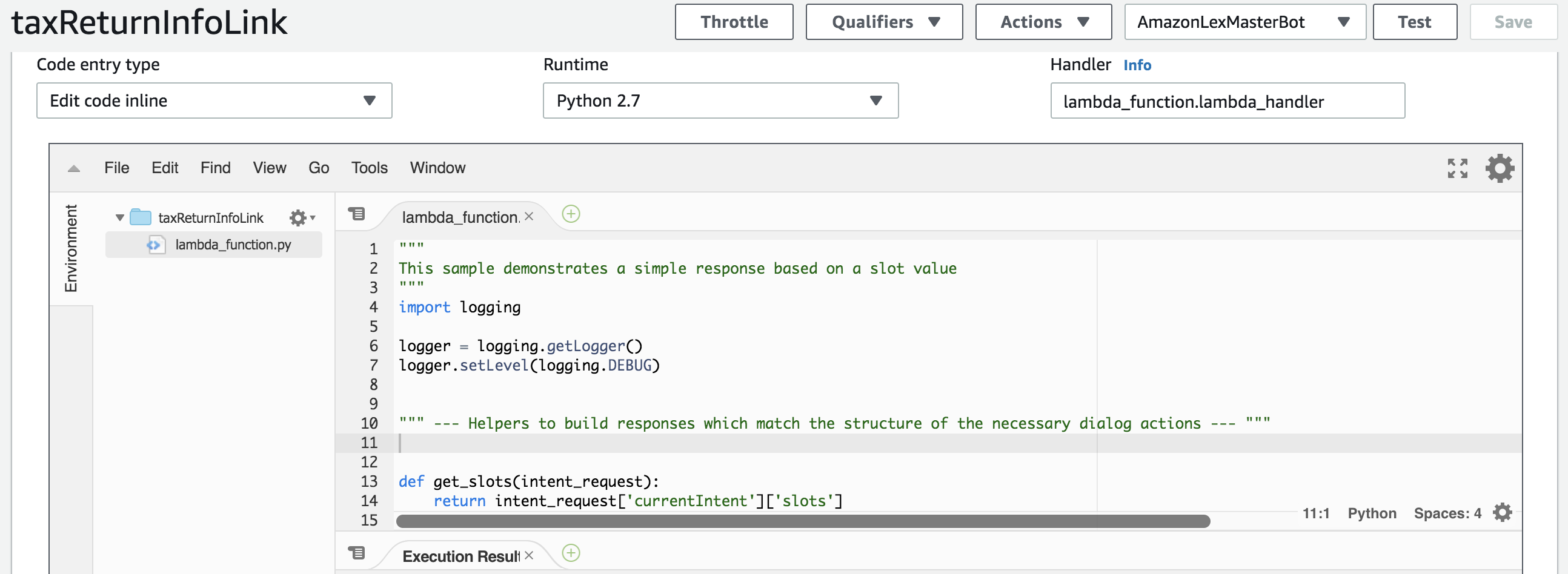
Choose "Services" from the navigation bar at the top of the page, and search for "Lambda".



1. Select "Create function" in the upper right-hand corner
2. To work independently in a shared environment, use your initials in the function name (e.g., taxReturnInfoLinkXXX)
3. Pick Python 2.7 as the runtime.
4. Choose an IAM role that includes the AWSLambdaBasicExecutionRole Managed Policy. There may already be one named masterbot-lambda-role
   * If no such role exists, you can create a new IAM Role using one of these approaches:
     + Choose "Create new role from template(s)", provide a role name, and choose Basic Lambda permissions from the "Policy templates" dropdown
     + Choose "Create a Custom role", which should open up a new tab where an IAM role is shown; review the policy document and click "Allow"



1. Once created, you can scroll to the bottom of the page to edit the code for you function.
   * You can get the function source from lambda\_function.py
   * There is no need to set up a trigger; you can accept default values for most of the configuration



1. In the code block you'll notice that the function checks the intent name it receives (if intent\_name == 'CRAInquiryIntent':); You will need to change this value in the function and in the test event to match the name you used for your bot. Click "Save" at the top to save your Lambda function.

Now we'll configure a test event to ensure our Lambda function works.

1. At the top of the page, choose "Select a test event".
2. Configure the Test event and test to confirm the function works as expected
   * You can use the following test event code:

{

"messageVersion": "1.0",

"invocationSource": "DialogCodeHook",

"userId": "",

"sessionAttributes": {},

"bot": {

"name": "MasterBotXXX",

"alias": "$LATEST",

"version": "$LATEST"

},

"outputDialogMode": "Text",

"currentIntent": {

"name": "CRAInquiryIntentXXX",

"slots": {

"TaxCategory": "business"

},

"confirmationStatus": "None"

}

}

* + After saving your test event, you can choose "test" in the console to ensure it works as expected.

**Test the bot**

Navigate back to the Amazon Lex console to reconfigure your bot to use the Lambda function.

1. From the Lex Console, select the MasterBotXXX bot and choose Latest from the version drop down to make changes
2. Modify the CRAInquiryIntentXXX Intent
   * Associate it with the new taxReturnInfoLinkXX Lambda function (select "Lambda function" in the "Lambda initialization and validation" area)
     + When prompted, allow Amazon Lex to call your new function
   * Associate it with the new taxReturnInfoLinkXXX Lambda function for (select "Lambda function" in the "Fulfillment" area); remember to click "Save Intent"
3. Build the bot
4. Test using the Amazon Lex Console; do you see any responses when you ask I want to do my personal taxes.

Try asking for different types of Income Tax return. You should see a message directing you to the appropriate link containing more information.