UB Hacking

Workshop

Hacking with Amazon Alexa

Use this URL to view/download
the source code used in this workshop:

https://github.com/anandimous/alexa-skills-workshop

Welcome! My name is
Aniruddha Nandi.

1. **I'm will be leading this session to help you learn something new today**

2. **I'm a Senior in CS & Lead of Project AI in the Robotics Club**

3. **Things I love to do include globetrotting, aerial photography and hacking**

What will you learn today?

1. Understand Voice User Interfaces & what you can build using them.

2. Meet Alexa, an intelligent personal assistant developed by Amazon.

3. Create your first voice powered app with Amazon Alexa.

Table of Contents

What is Alexa?

Alexa is a Voice User Interface (VUI),
that lets you ***speak*** commands,
instead of clicking buttons or typing
on your keyboard.

Alexa listens to spoken input, uses it to execute tasks or skills in the cloud, and then returns output -- just like a JavaScript function.

Why do Voice UIs Matter?

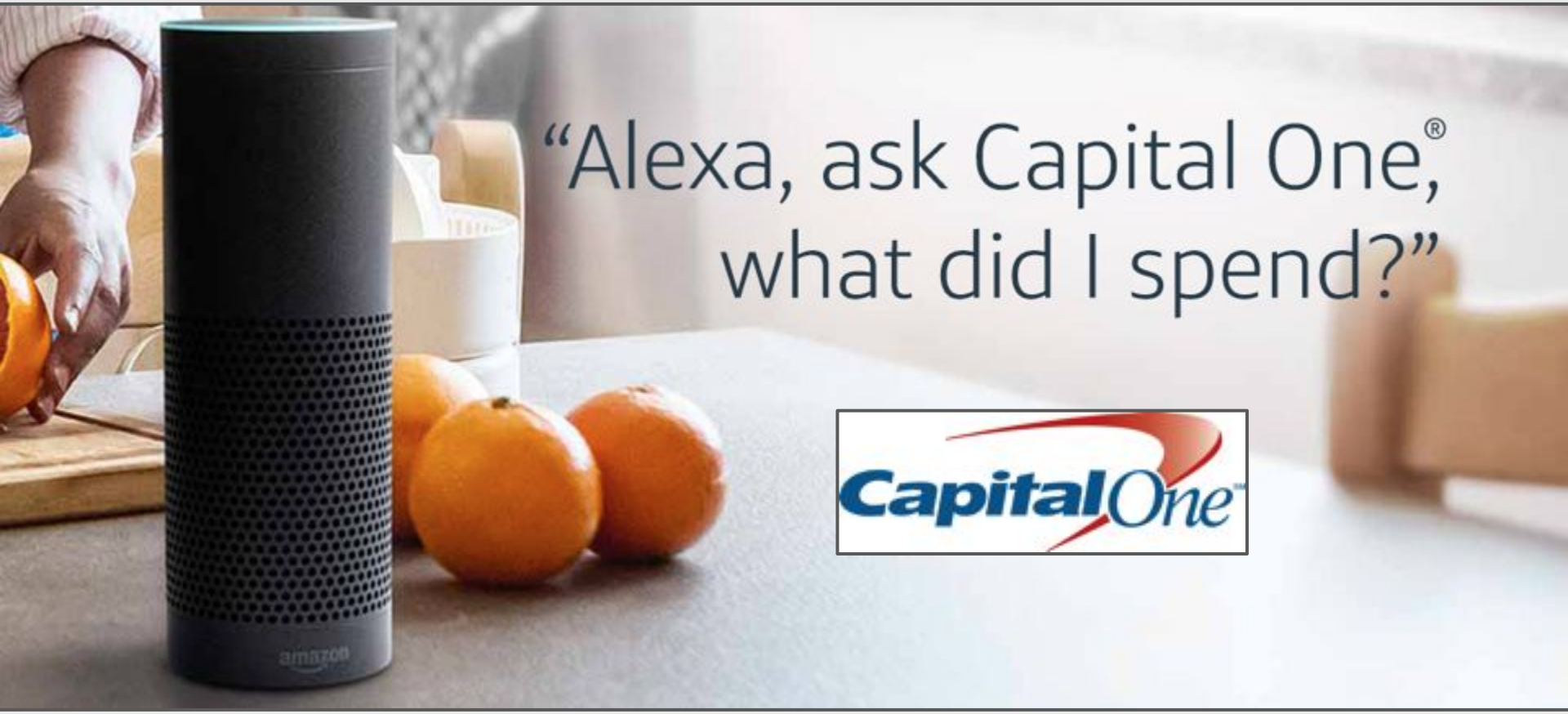
Instead of typing, clicking, or tapping - we can physically separate ourselves from our devices and speak commands naturally.

Voice UIs can run code in the cloud and communicate with IoT devices, making them ideal for homes, cars, & more.

What can you build with Alexa?

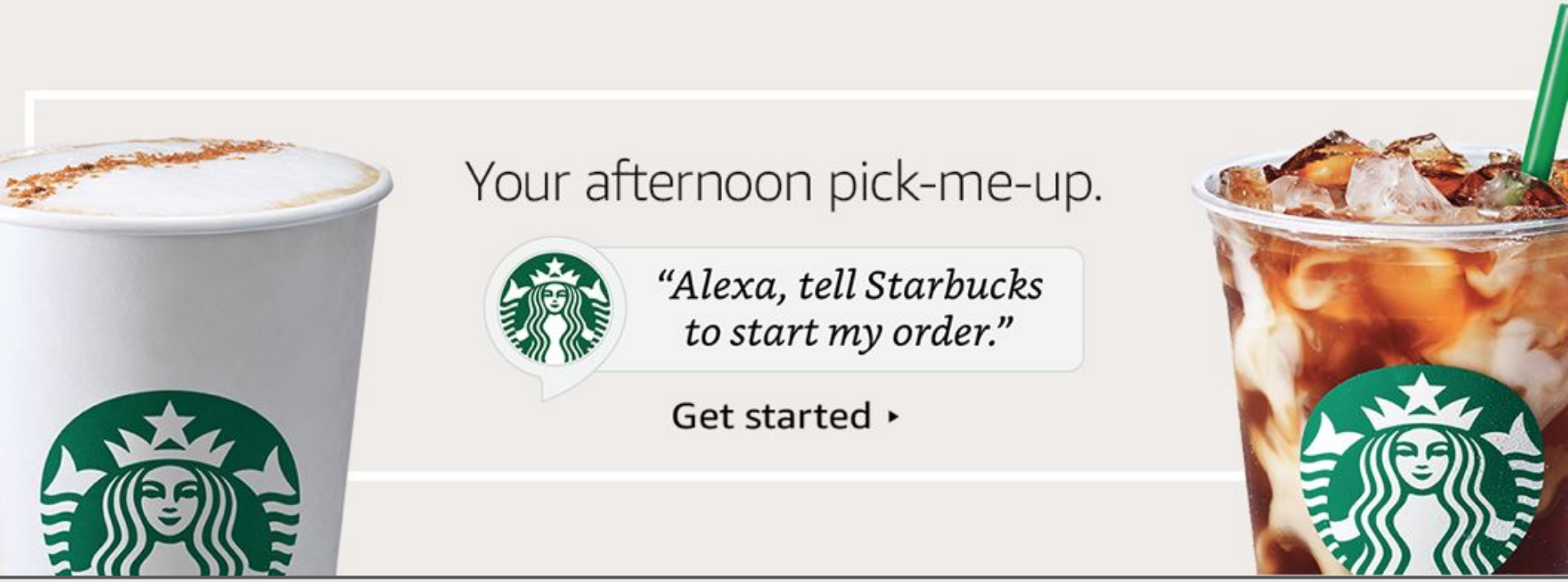
Alexa, ask Lyft for a Lyft Line to work.

What can you build with Alexa?



Alexa, ask Capital One, what did I spend?

What can you build with Alexa?



Alexa, tell Starbucks start my order.

Table of Contents

Alexa Skills are made of 2 parts:

1. **Front End** - The Alexa Voice UI handles text to speech, converting the audio into something our app can use, etc.

2. **Back End -** The logic code that actually powers our app. Usually this is written on a service called AWS Lambda.

Speech Recognition is Hard.

You said: *foːr tē tīmz.* But, what did you mean?

1. Forty Times?
2. For Tea Times?
3. For Tee Times?
4. Four Tee Times?

Parts of Speech Recognition

Automatic Speech Recognition (ASR)

Enables the recognition and translation of spoken language into text by computers.

1. Traditional phonetics-based using HMM's (Hidden Markov Models)
2. Deep feedforward neural network or RNN (recurrent neural nets) based

Parts of Speech Recognition

Natural Language Understanding (NLU)

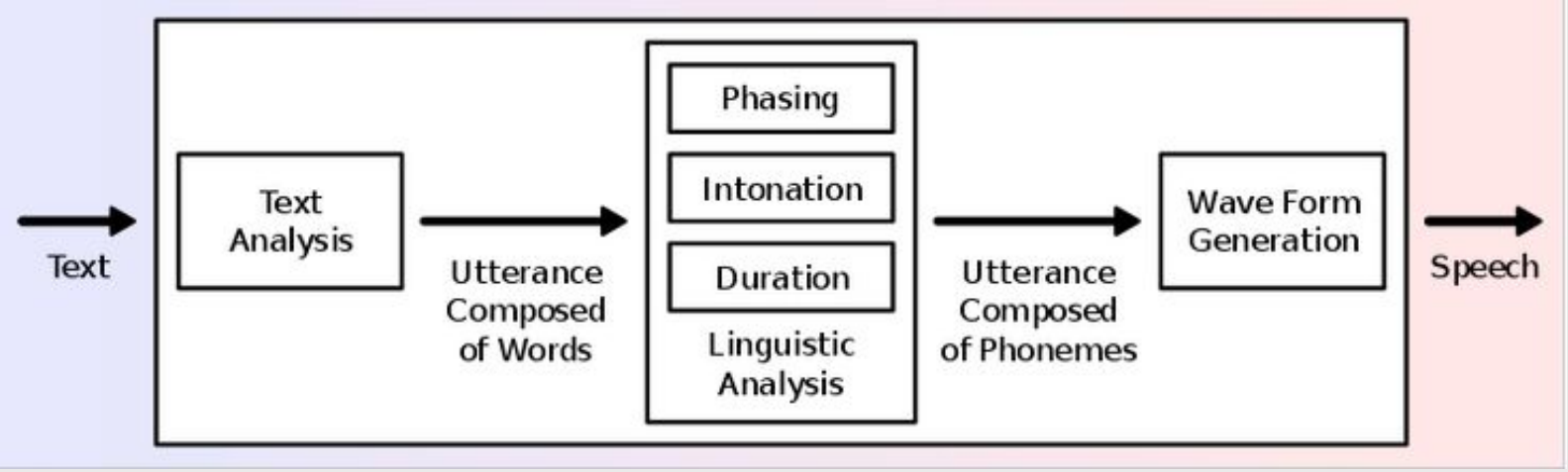
Deals with how to best handle unstructured inputs that are governed by poorly defined, flexible rules and convert them into a structured form that a machine can understand. It is an AI-hard problem.

“I need a flight and hotel in Miami from October 4 to 10”
need:flight {intent} / need:hotel {intent} / Miami {city} / Oct 4 {date} / Oct 10 {date} / sentiment: 0.5723 (neutral)

Parts of Speech Recognition

Text to Speech (TTS)

Converts text back to human-understandable speech



Alexa uses Sample Utterances for Training.

In order to **map user input** to a behavior, we provide **training data**, for each intent.

```
01  GetNewFactIntent a fact
02  GetNewFactIntent a Major League Hacking fact
03  GetNewFactIntent tell me a fact
04  GetNewFactIntent tell me a Major League Hacking fact
05  GetNewFactIntent give me a fact
06  GetNewFactIntent give me a Major League Hacking fact
07  GetNewFactIntent tell me trivia
08  GetNewFactIntent tell me a Major League Hacking trivia
09  GetNewFactIntent give me trivia
10  GetNewFactIntent give me a Major League Hacking trivia
11  GetNewFactIntent give me some information
12  GetNewFactIntent give me some Major League Hacking information
13  GetNewFactIntent tell me something
14  GetNewFactIntent give me something
```

Alexa Maps Speech Input to Intents.

Once Alexa figures out what Intent you wanted, you can easily map that back to code.

```
{
  "intents": [
    { "intent": "GetNewFactIntent" },
    { "intent": "AMAZON.HelpIntent" },
    { "intent": "AMAZON.StopIntent" },
    { "intent": "AMAZON.CancelIntent" }
  ]
}
```

“Tell me a Fact about MLH” → `GetNewFactIntent`

Table of Contents

Steps to Build Your Skill:

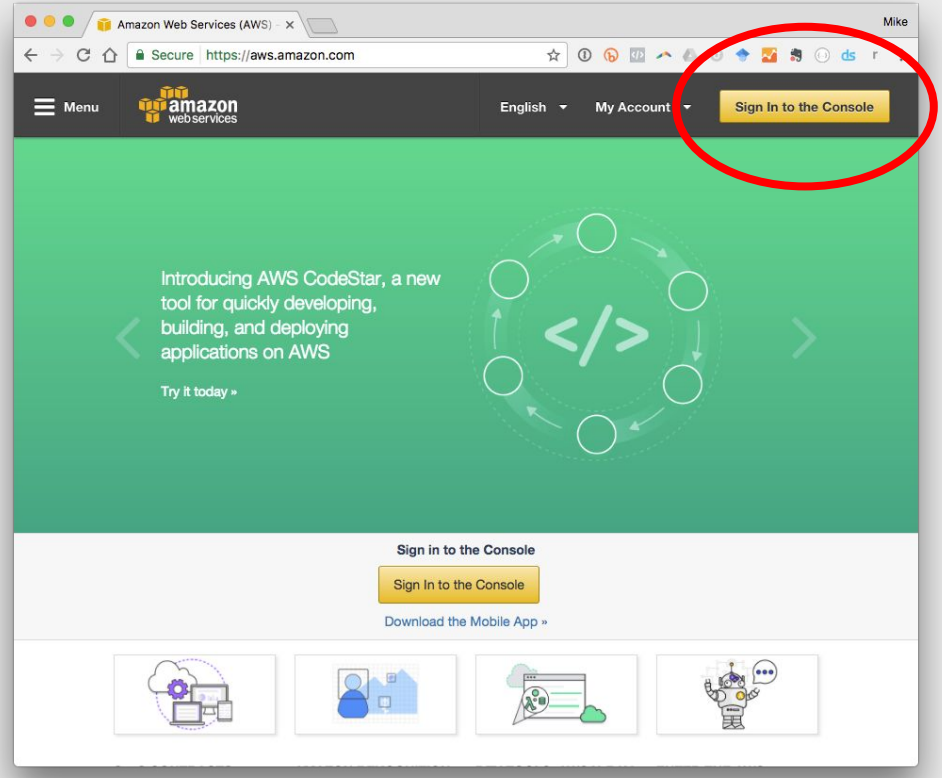
To build your first skill, we'll complete the following steps:

1. Write your skill code as an AWS Lambda function
2. Create an Alexa Skill in the Developer Portal
3. Connect Your Lambda Function to Your Skill
4. Test your Skill
5. Publish Your Skill

Sign into the AWS Console.

Navigate to:

<AWS URL here>

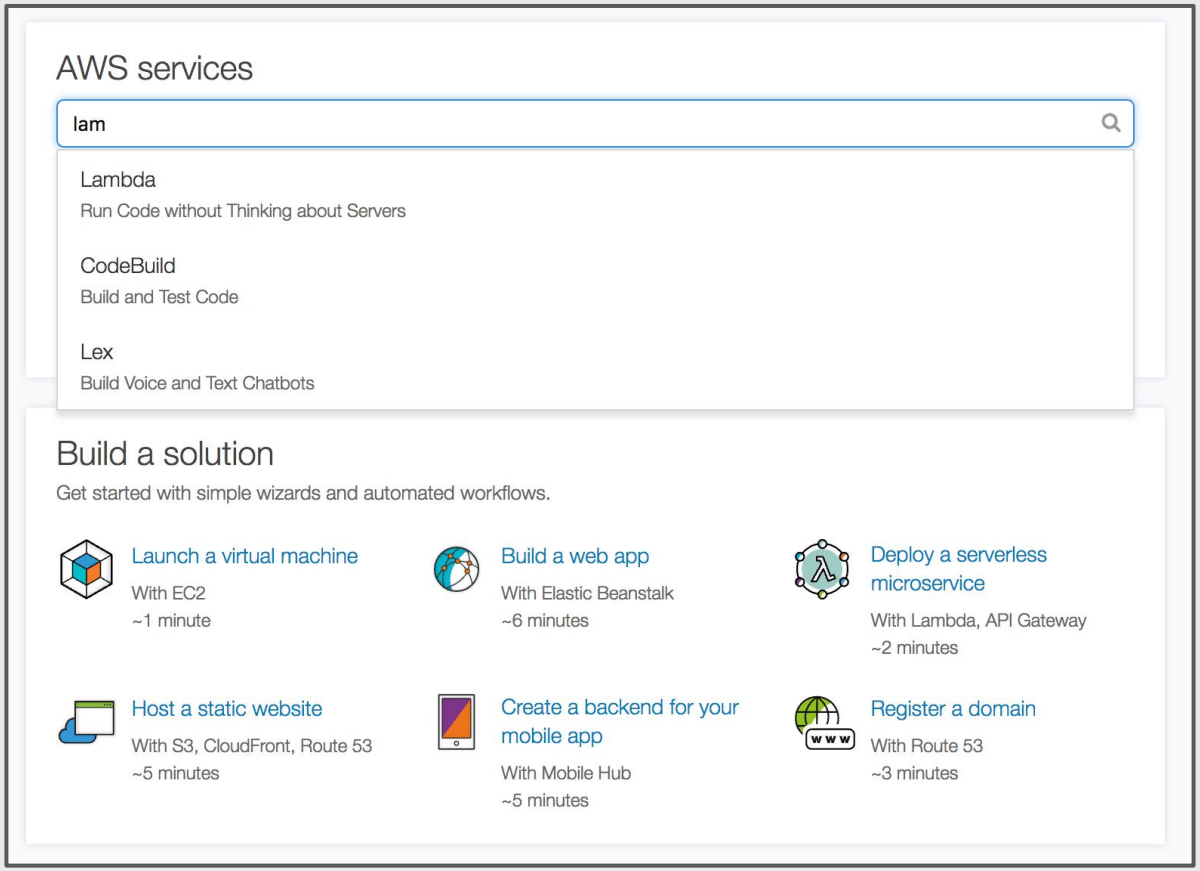


Instructions

Click on “Sign in to Console” to get started.

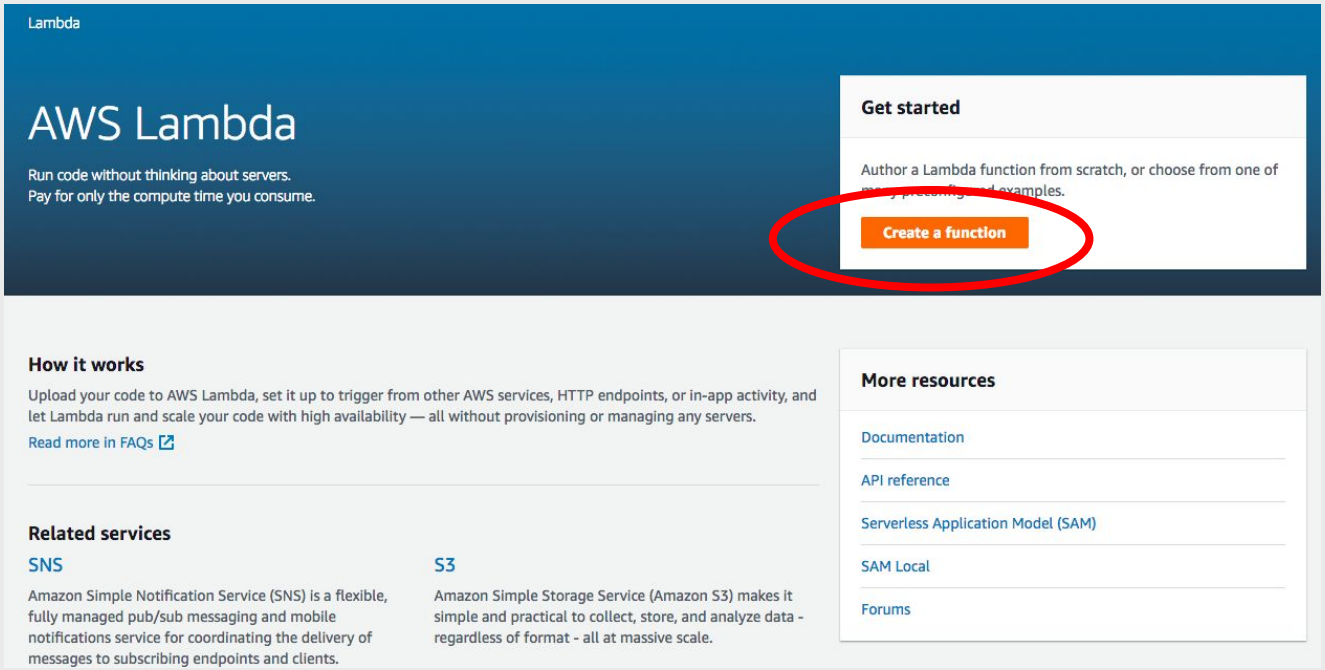
Navigate to the Lambda Manager.

You can search for Lambda in the search box or find it on the list of available services.



Create your First Lambda Function.

Each AWS Lambda Function is responsible for one thing (like returning facts about you!).



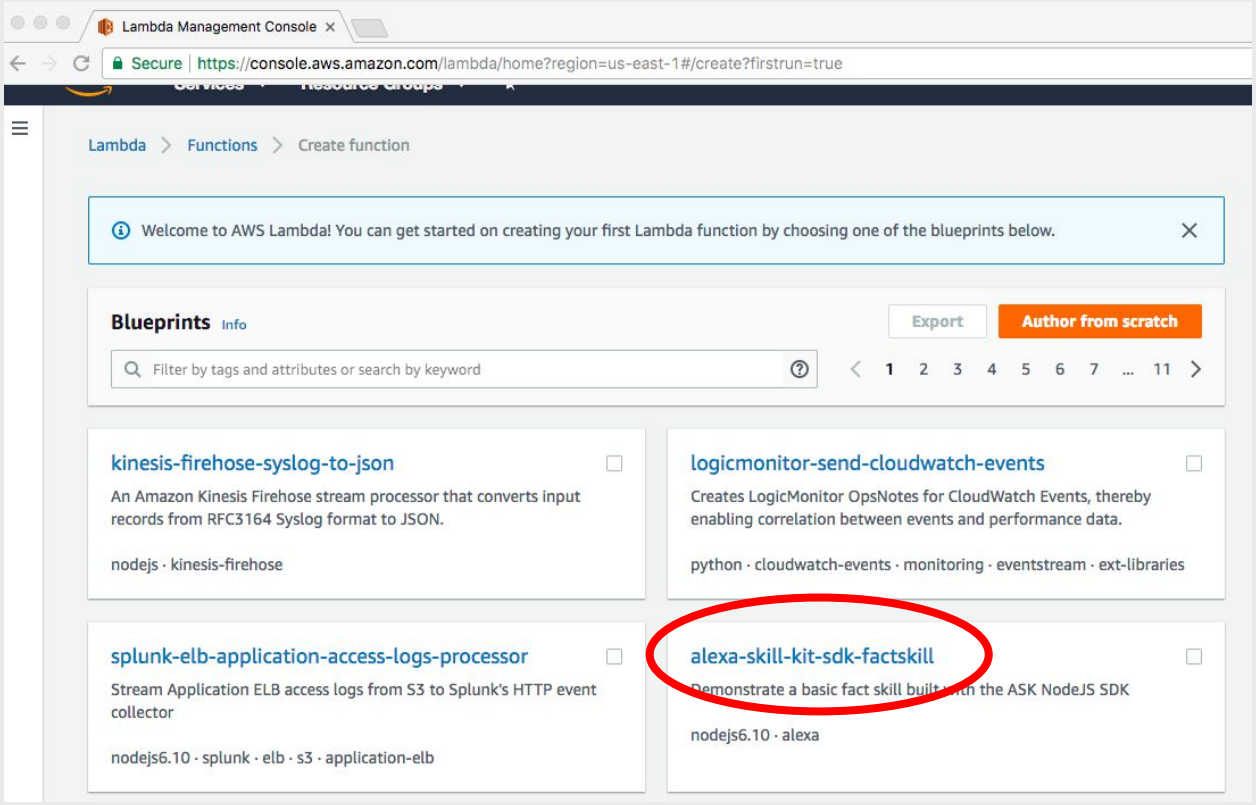
Instructions

Click “Create a Function” to create your first function!

Choose a Blueprint.

Instructions

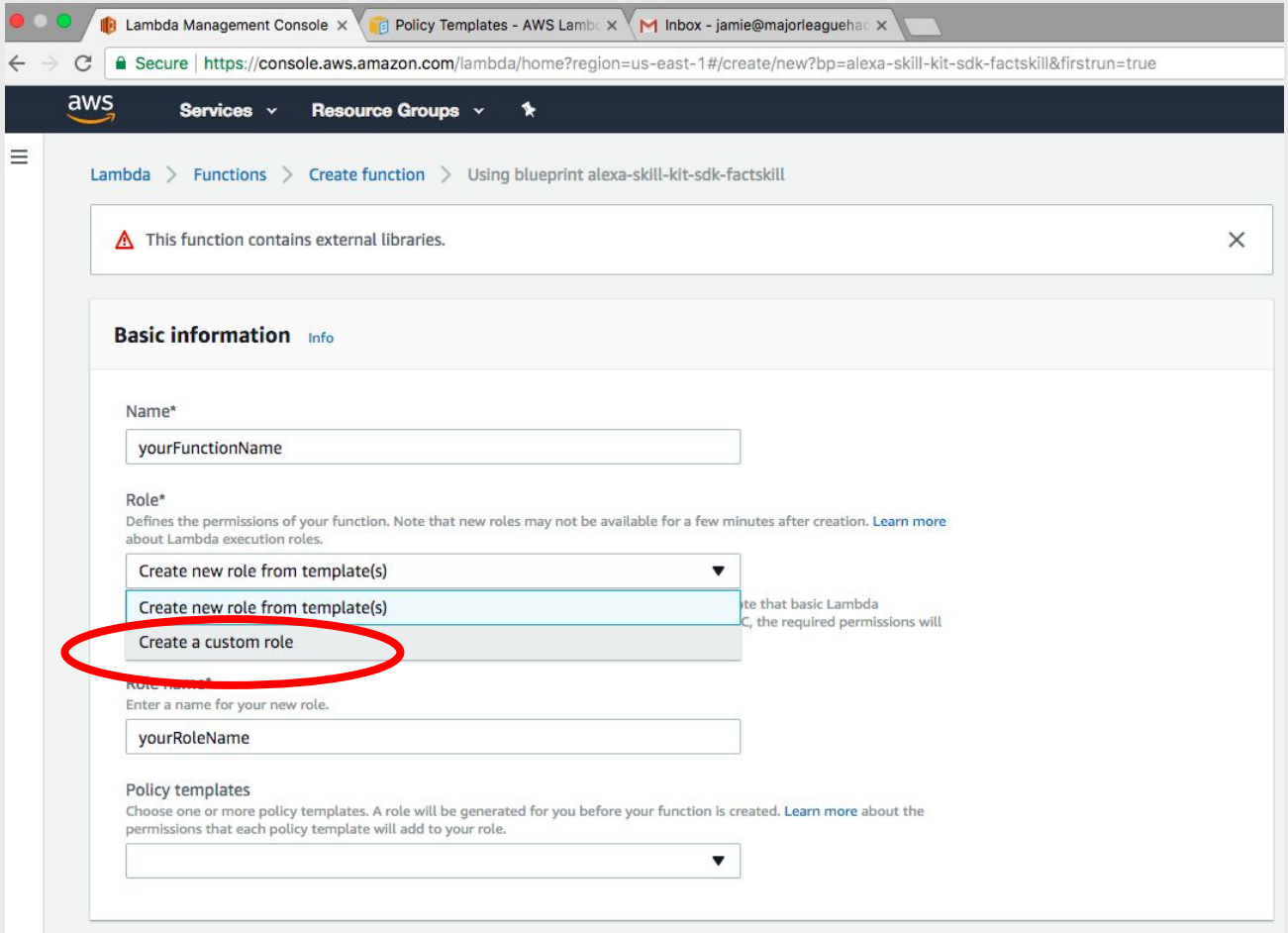
Select “alexa-skill-kit-sdk-factskill” from the options.



Set Basic Information.

Instructions

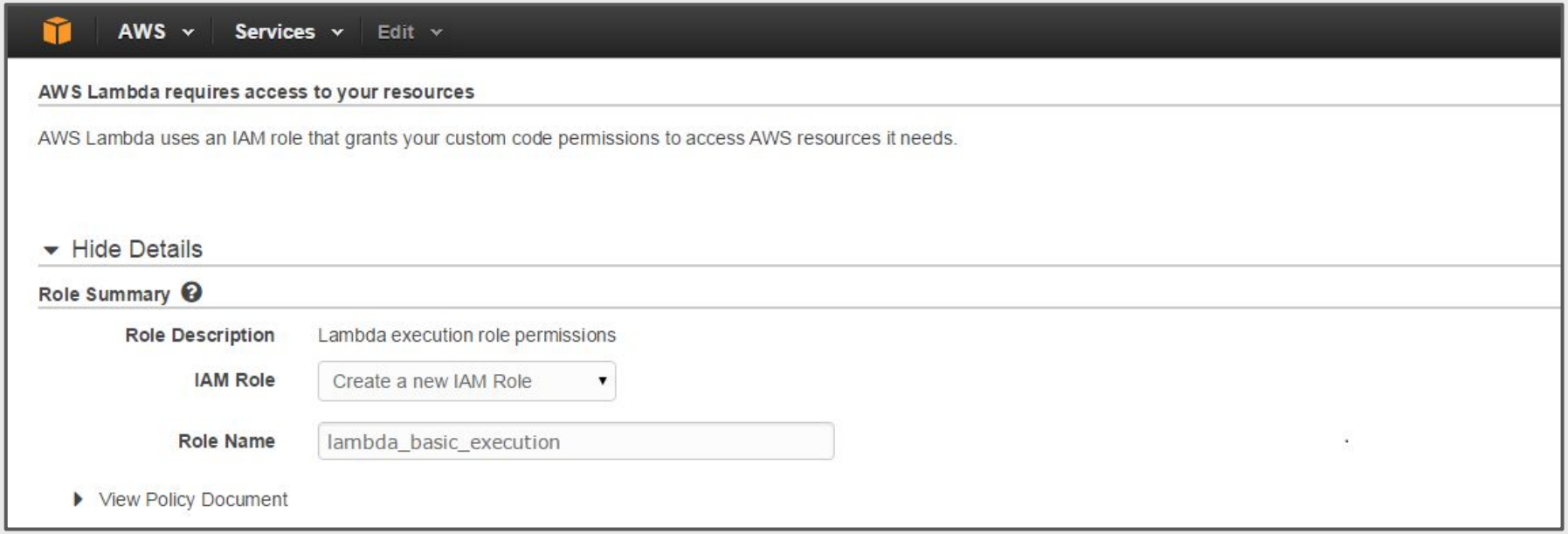
Enter a name, select “Create a custom role,” and name your role.



Create your IAM Role.

Instructions

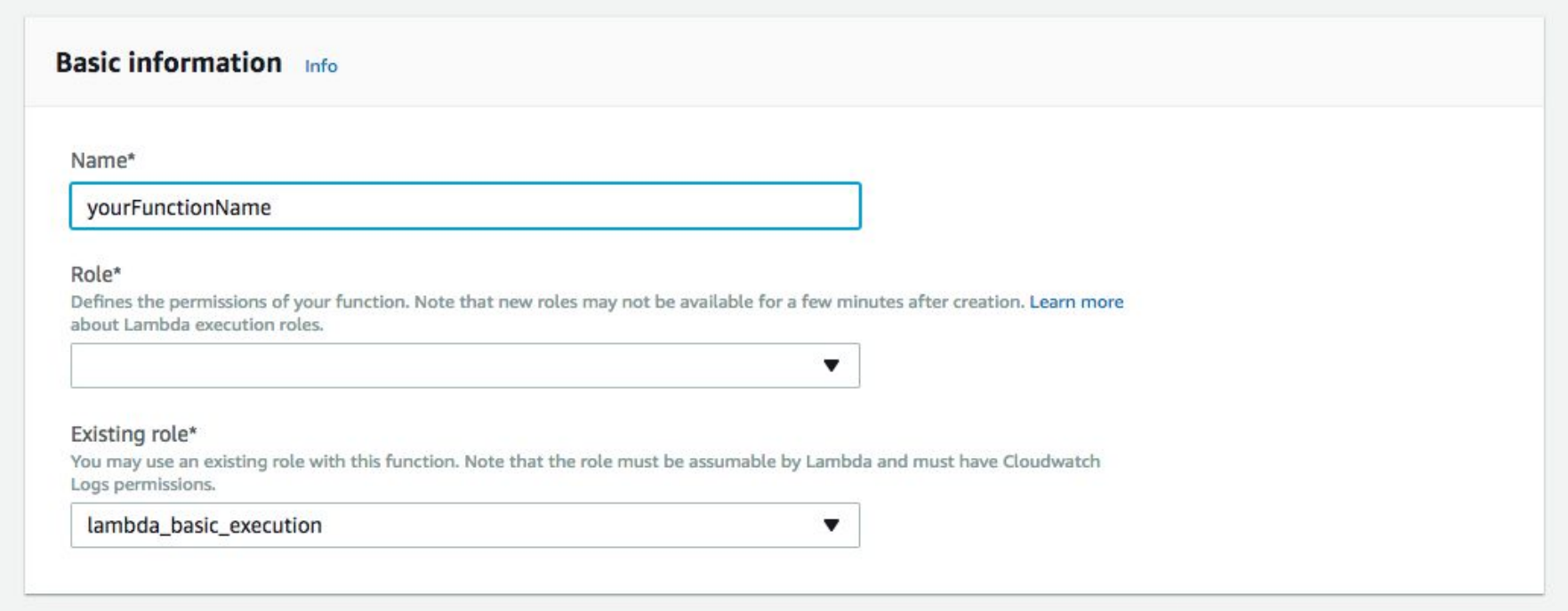
Select "Create a new IAM Role" from dropdown menu.
Role Name & policy will automatically populate.



Select “Allow” in the lower right corner and you will be returned to your Lambda function.

Create your IAM Role.

Notice that “lambda_basic_execution” is now in the “Existing Role*” field.



Step 1: Download the Sample Code

To get the sample code, head to this URL:

<same code url here>

Step 2: Open Index.js

Unzip the directory and open `src/index.js` in your favorite code editor.


Customize your Facts

Instructions

Find the `FACTS` array inside `index.js`.

Replace the facts about MLH with facts of your choice!

```
12  // TODO: replace with facts about yourself
13  const FACTS = [
14    "Major League Hacking is commonly called MLH.",
15    "Major League Hacking's mission is to empower hackers.",
16    "Over 65,000 student hackers participated in Major League Hacking...",
17    "Major League Hacking was founded in 2013 by Swift and Jon.",
18    "Hackers created over 12,000 projects at MLH hackathons in 2016.",
19    "Over 200 schools around the world hosted MLH hackathons in 2016.",
20    "Major League Hacking is headquartered in New York City."
21  ];
```

Code Review: The Handlers Object

The `handlers` object tells Alexa how to handle various actions. `GetFact` is the main logic of our application.

```
23  var handlers = {
24    'LaunchRequest': function() { this.emit('GetFact'); },
25    'GetNewFactIntent': function() { this.emit('GetFact'); },
26    'GetFact': function() {
27      // Randomly select a fact from the array
28      const factIndex = Math.floor(Math.random() * FACTS.length);
29      const randomFact = FACTS[factIndex];
30
31      // Create speech output
32      const speechOutput = "Here's your fact: " + randomFact;
33      this.emit(':tellWithCard', speechOutput, "MLH Facts", randomFact);
34    }
35  };
```

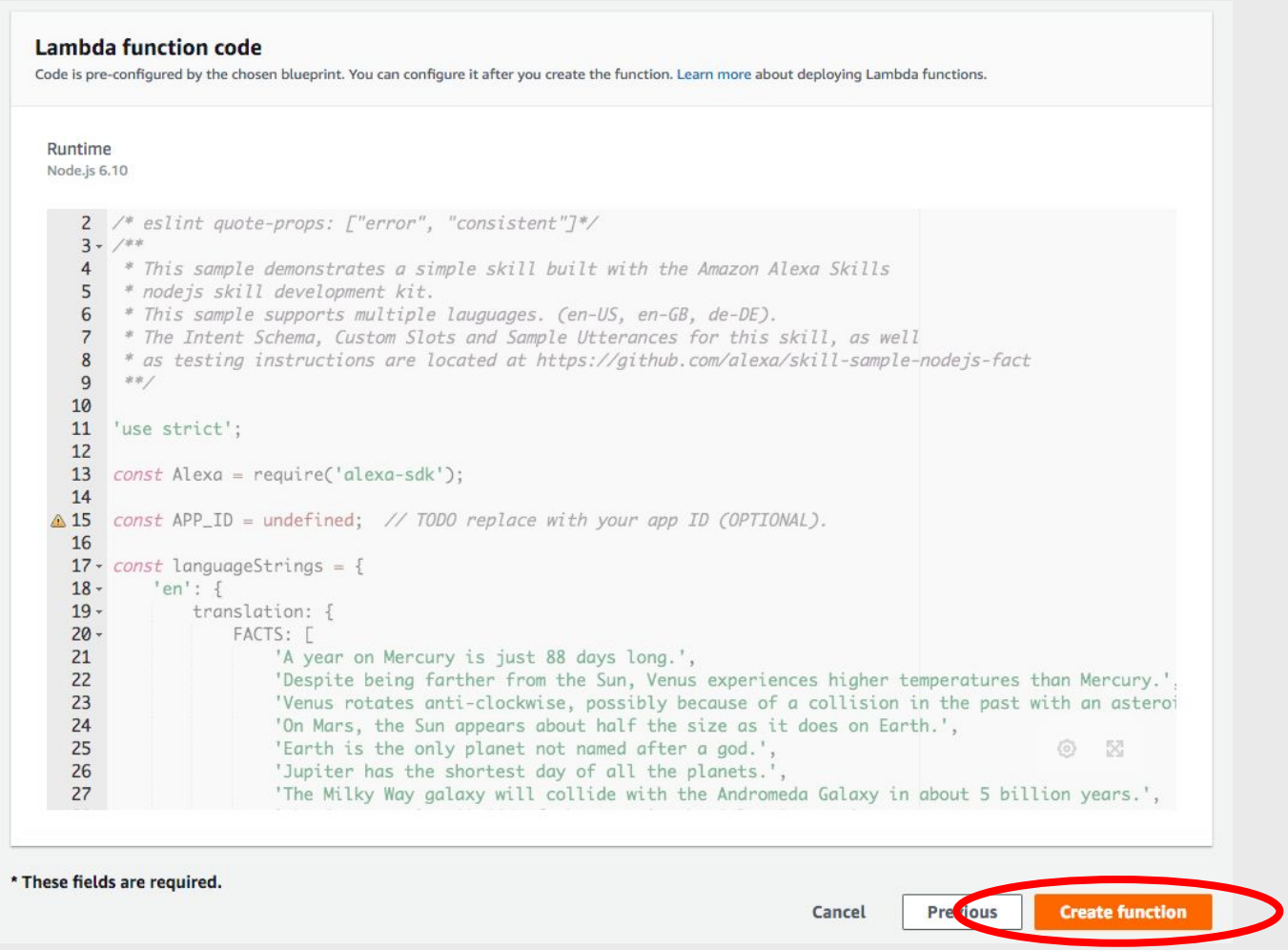
Code Review: The Handler Function

The `handler` function tells Alexa how to route voice commands by passing a copy of the `handlers` object.

```
37  // AWS Lambda calls this function every time Alexa uses our skill.
38  exports.handler = function(event, context, callback) {
39    // Include the AWS Alexa Library.
40    const Alexa = require("alexa-sdk");
41
42    // Create an instance of the Alexa library & pass it the requested command.
43    var alexa = Alexa.handler(event, context);
44
45    // Give our Alexa instance handling instructions & execute the request.
46    alexa.registerHandlers(handlers);
47    alexa.execute();
48  };
```

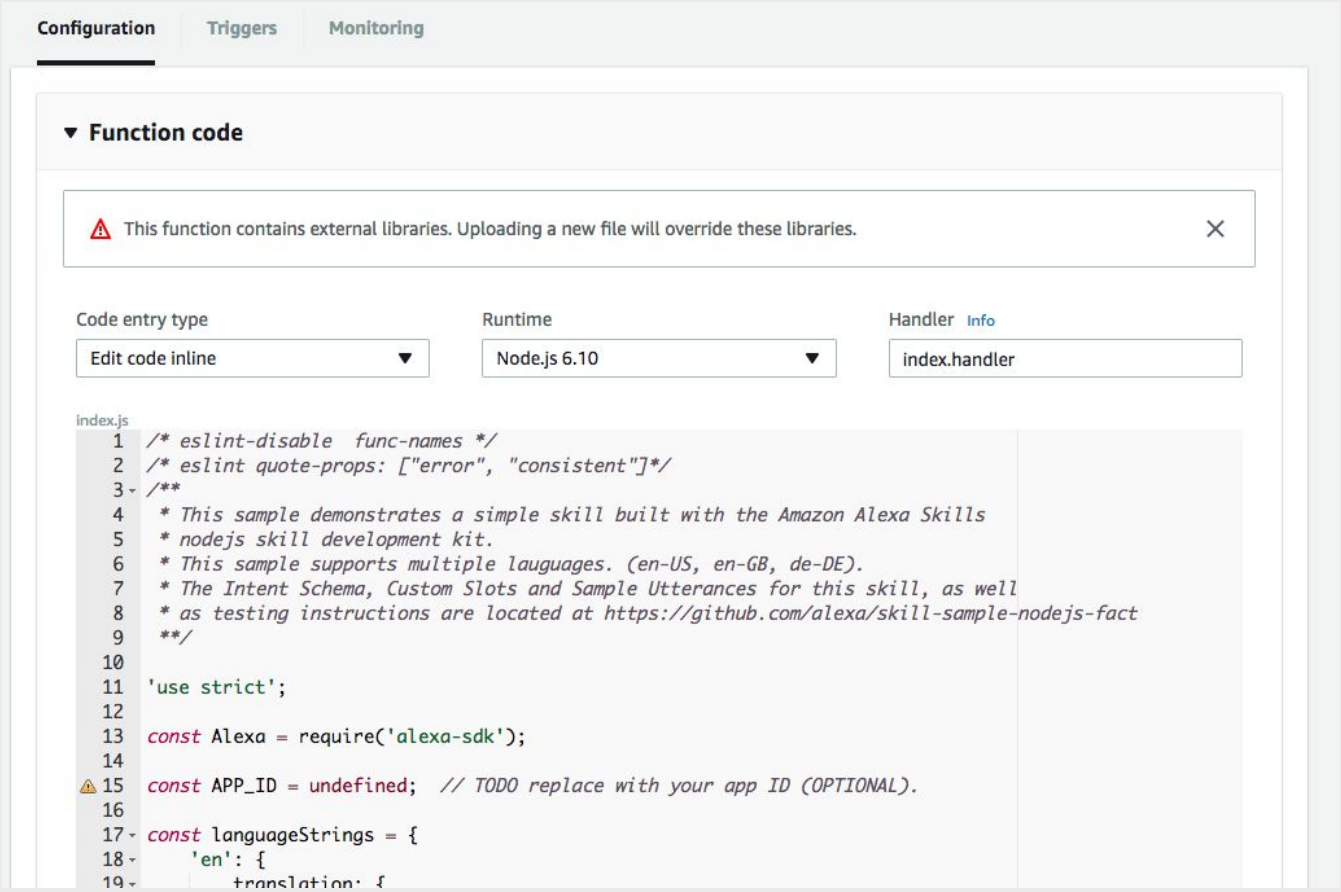
Create your Function.

Click “Create Function” at the bottom of the page



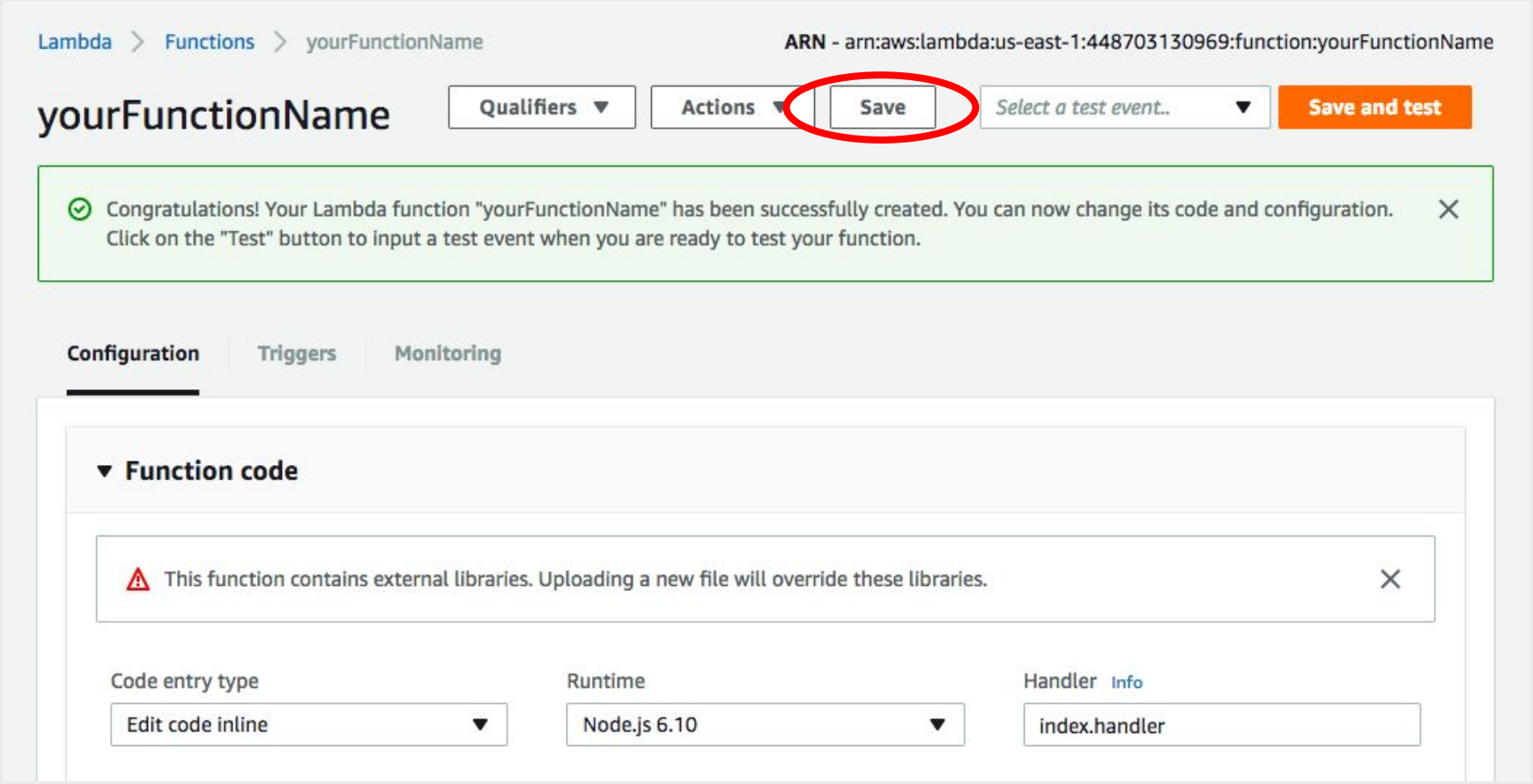
Copy & Paste your Code into the Lambda Editor

Copy the entire contents of index.js and paste it over the code in the inline editor on the AWS Console.



Save your Code.

Click the white “Save” button (not the orange one).

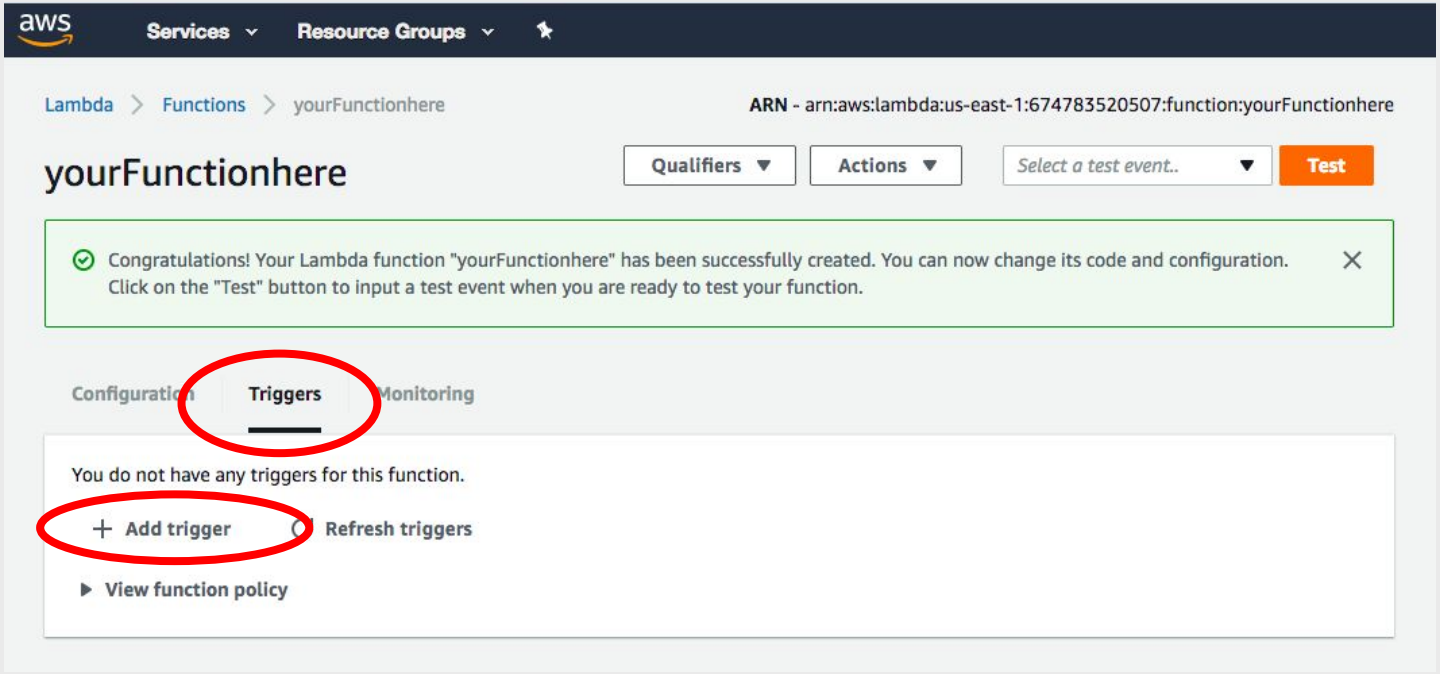


Configure your Trigger.

We need to setup a trigger to call our new Lambda Function (*in our case Alexa Skills Kit*).

Click “Triggers”

Click “+ Add Trigger”

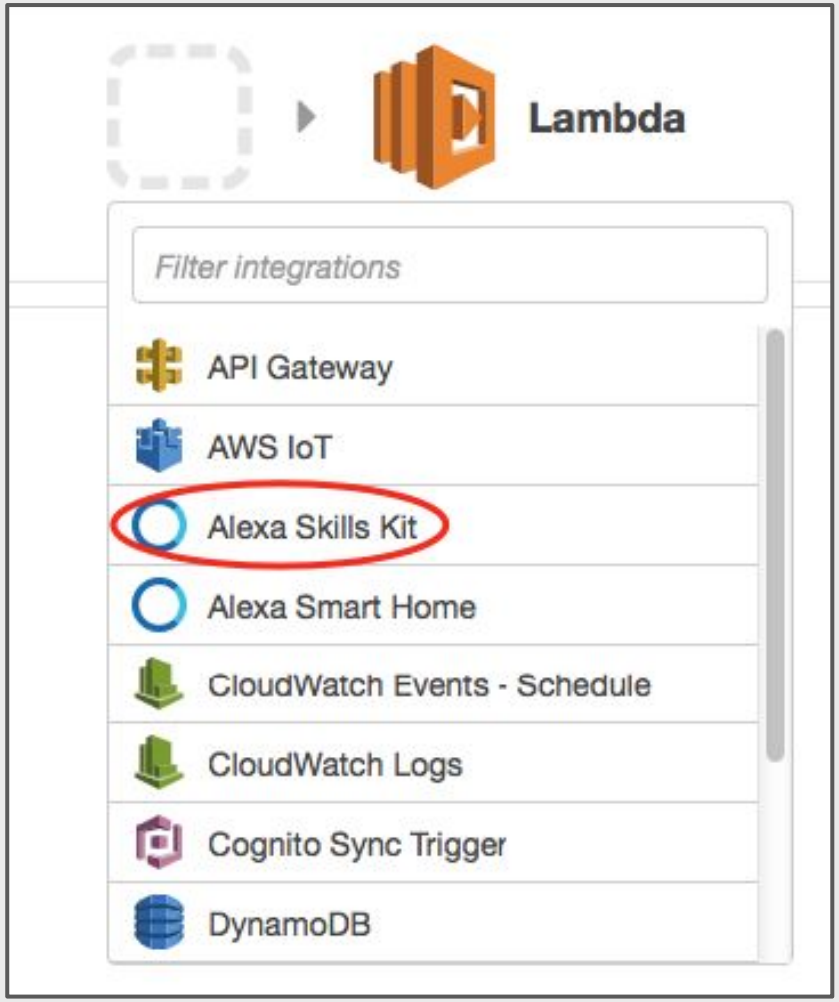


Configure your Trigger.

Instructions

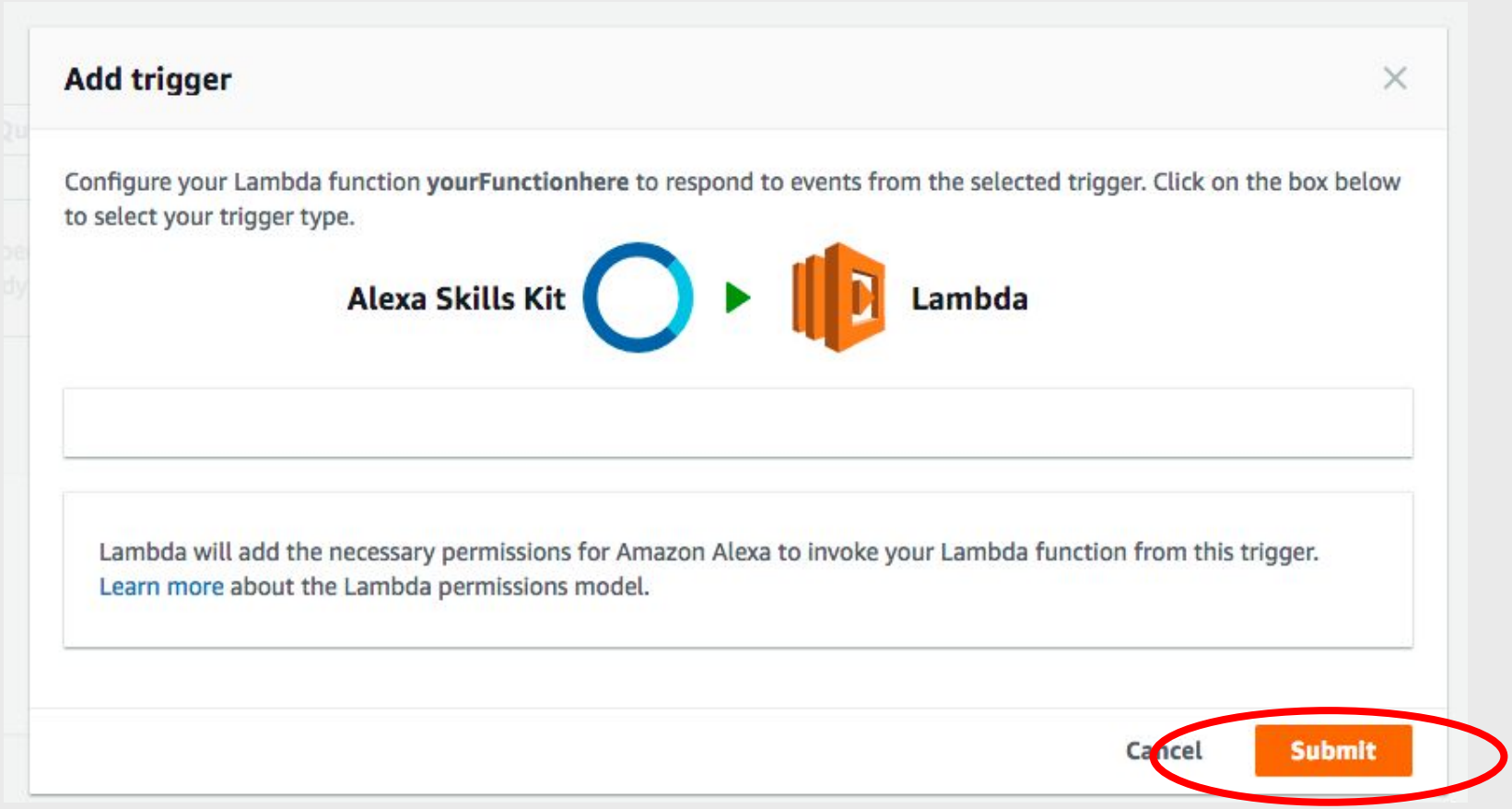
Click on the gray dash-lined box

Select “Alexa Skills Kit” from the dropdown menu



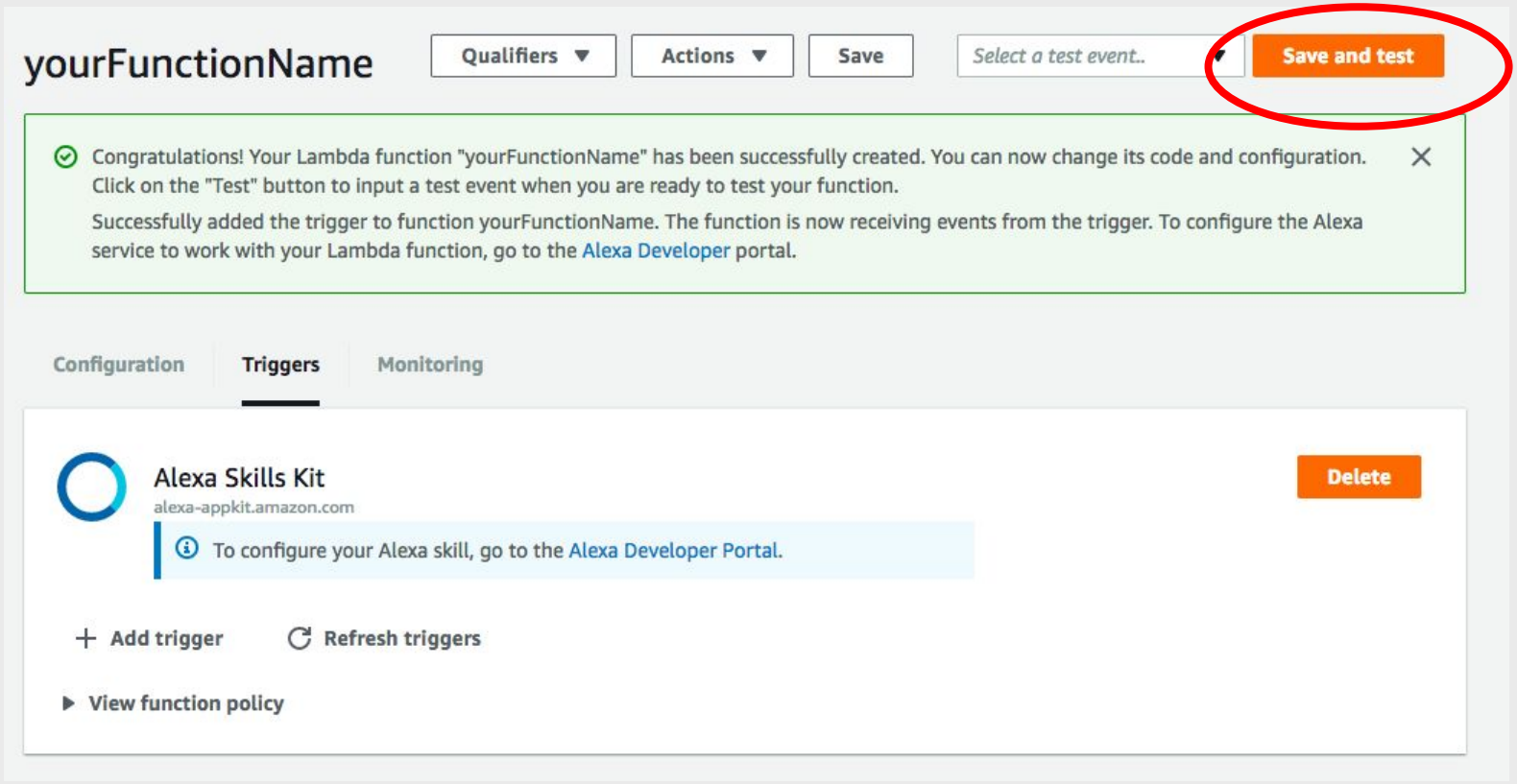
Configure your Trigger.

Click “Submit.”



Test your Trigger.

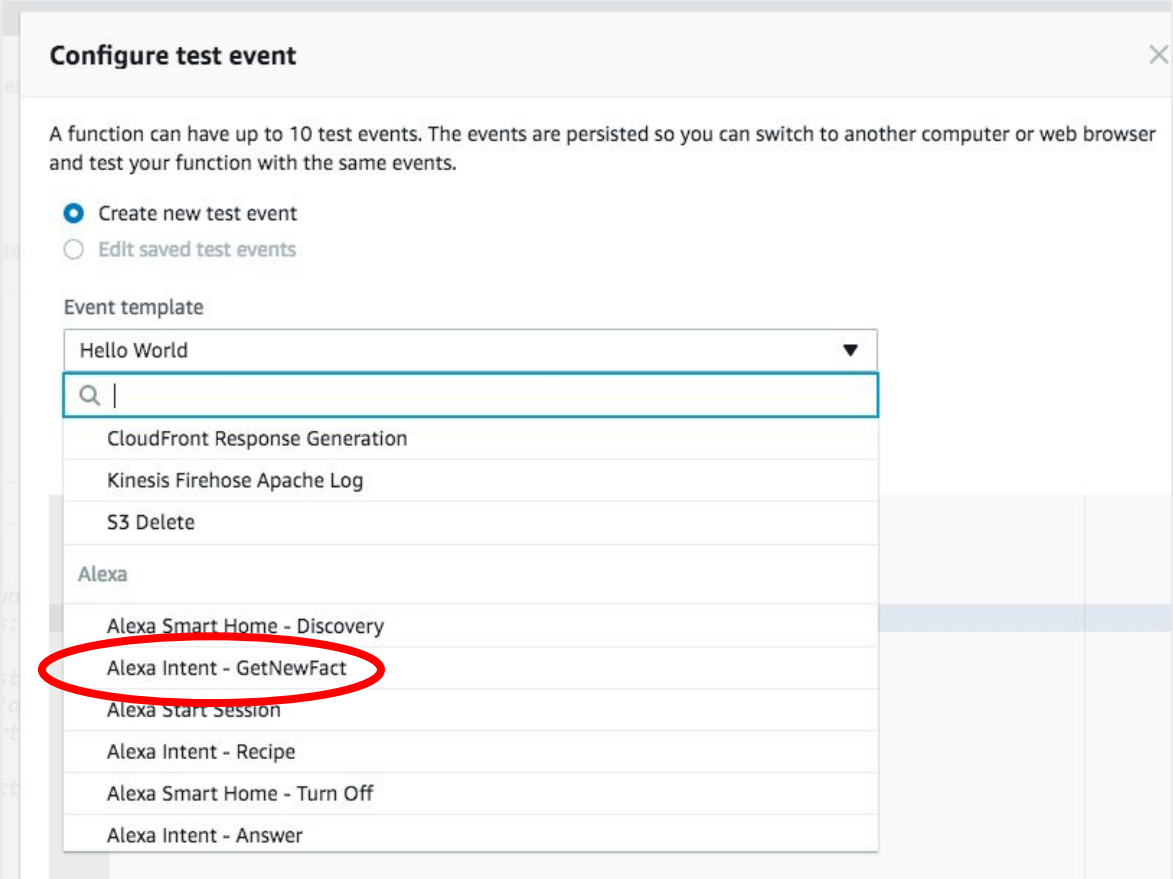
Click “Save and Test.”



Test your Trigger.

Click “Hello World.”

Scroll down and select “Alexa Intent - GetNewFact.”

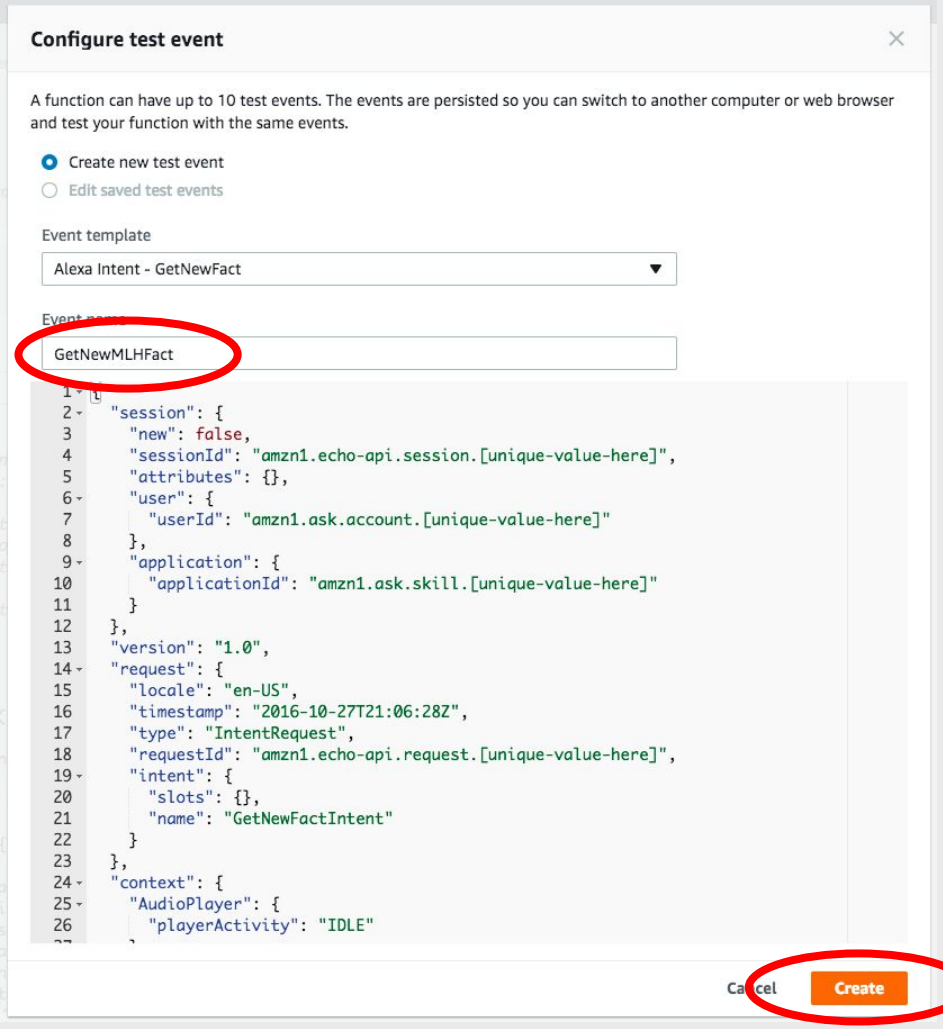


Test your Trigger.

We need to setup a trigger to call our new Lambda

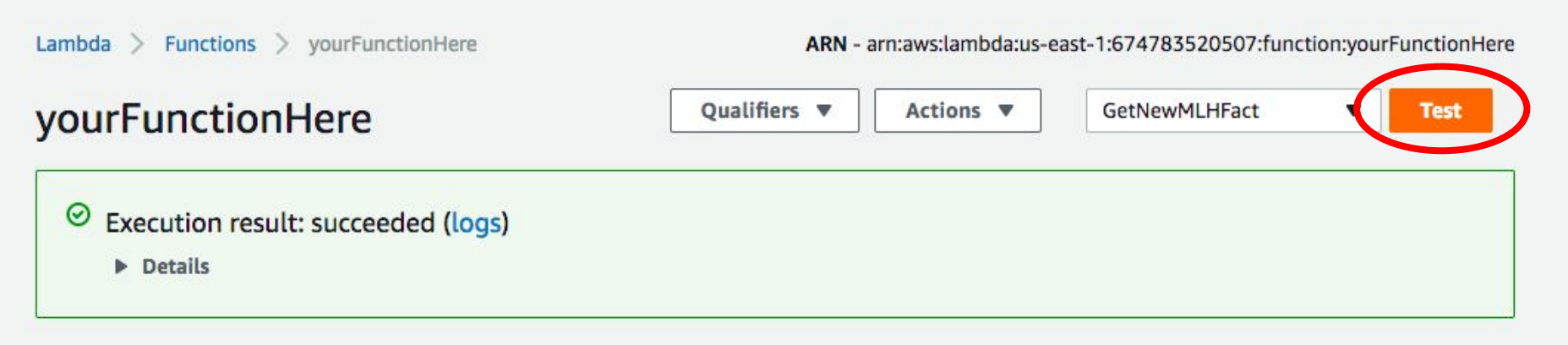
Name your test event.

Click “Create.”



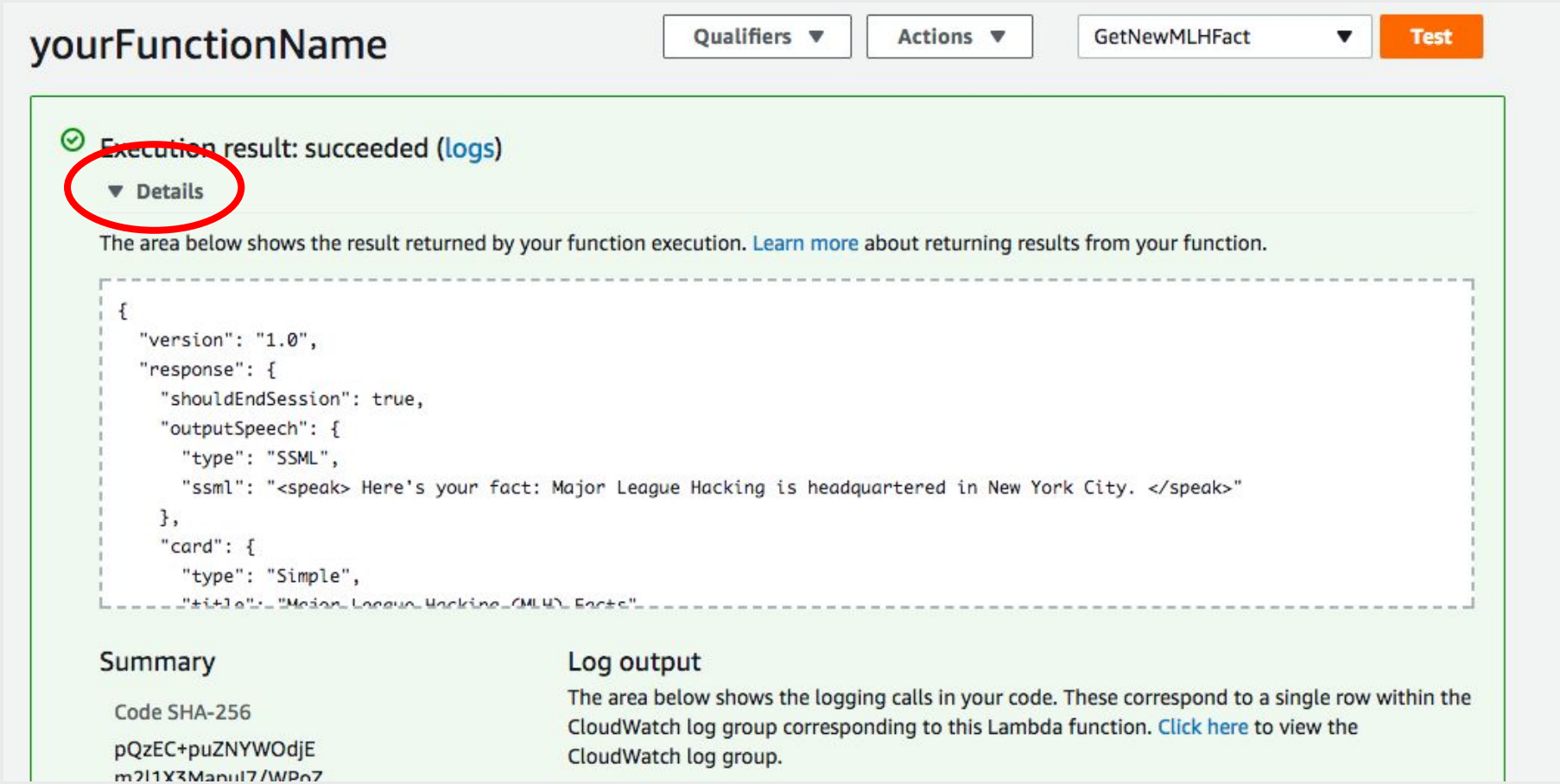
Test your Trigger.

Click “Test.”



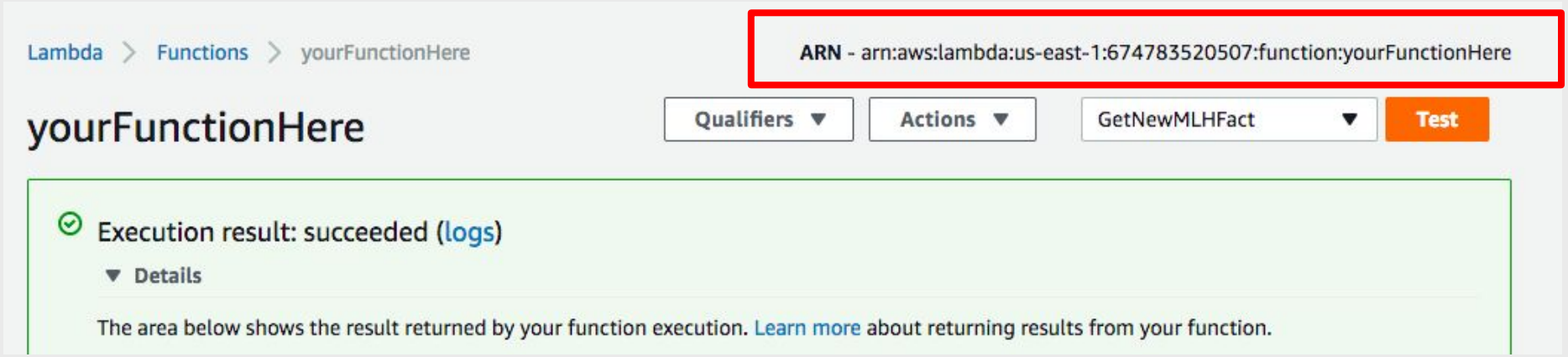
Test your AWS Lambda Function

If it's working, you can click "Details" to see output similar to the following:



Note Down your Function's ARN.

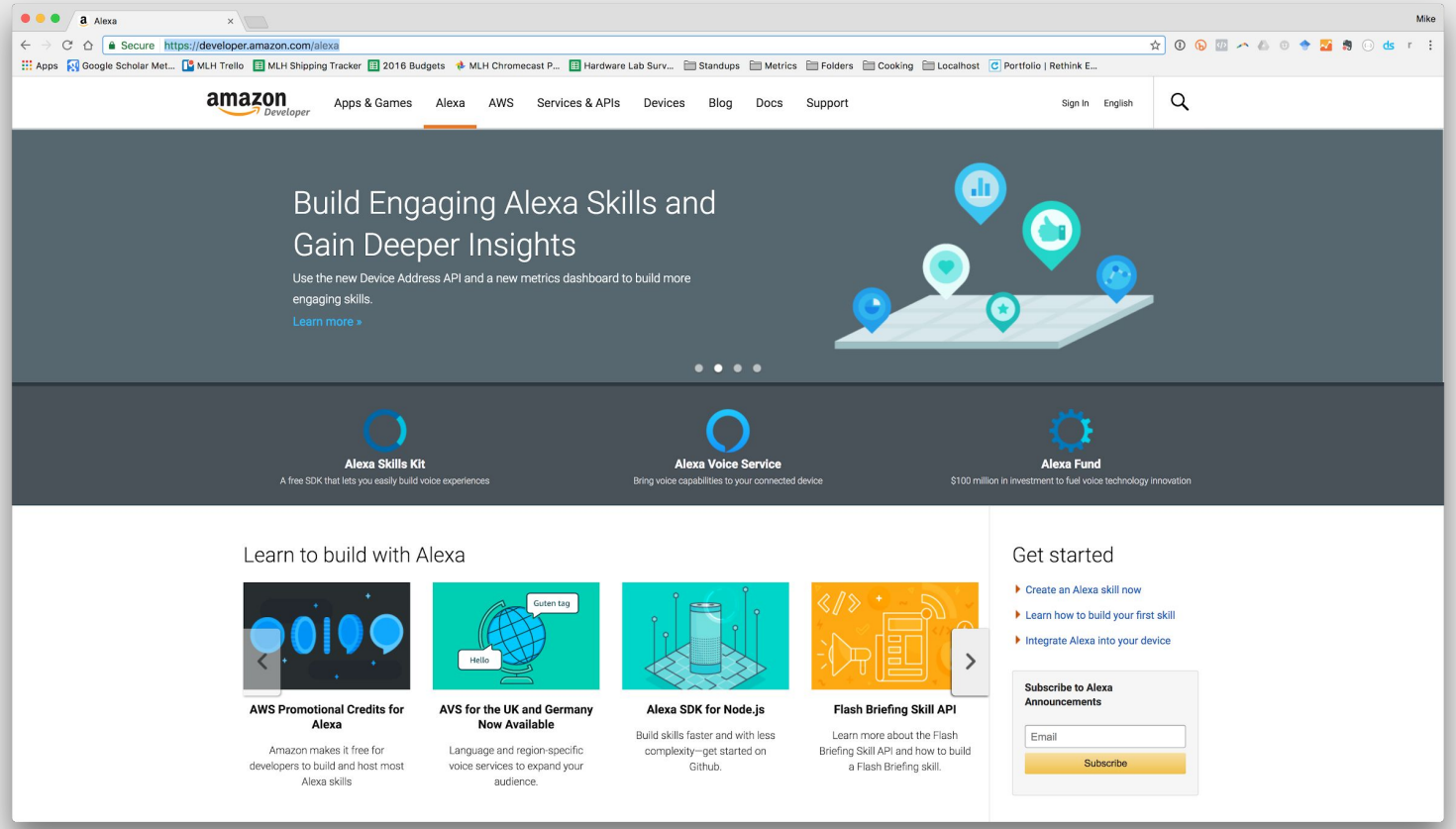
You'll need it in a few steps, so either keep this tab open or save it in a file somewhere now.



Sign into the Amazon Developer Portal.

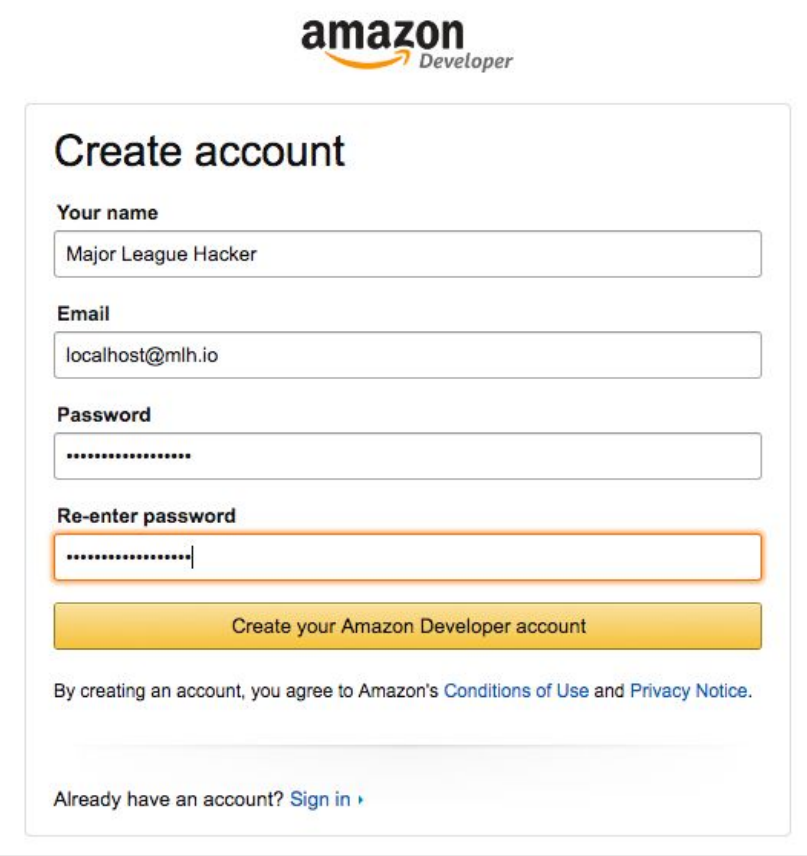
Head over to:

<Alexa Developer Portal url>



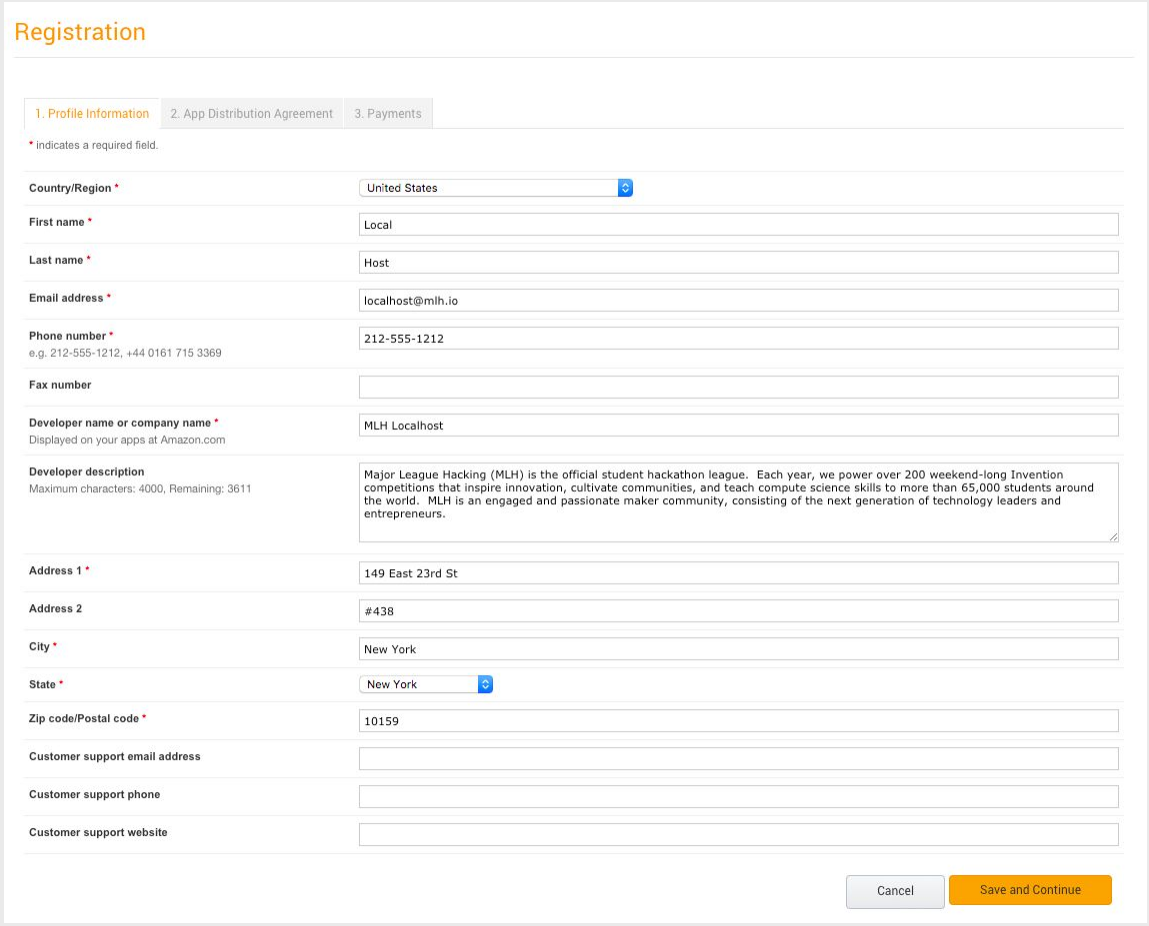
Create your account.

Fill in this form.



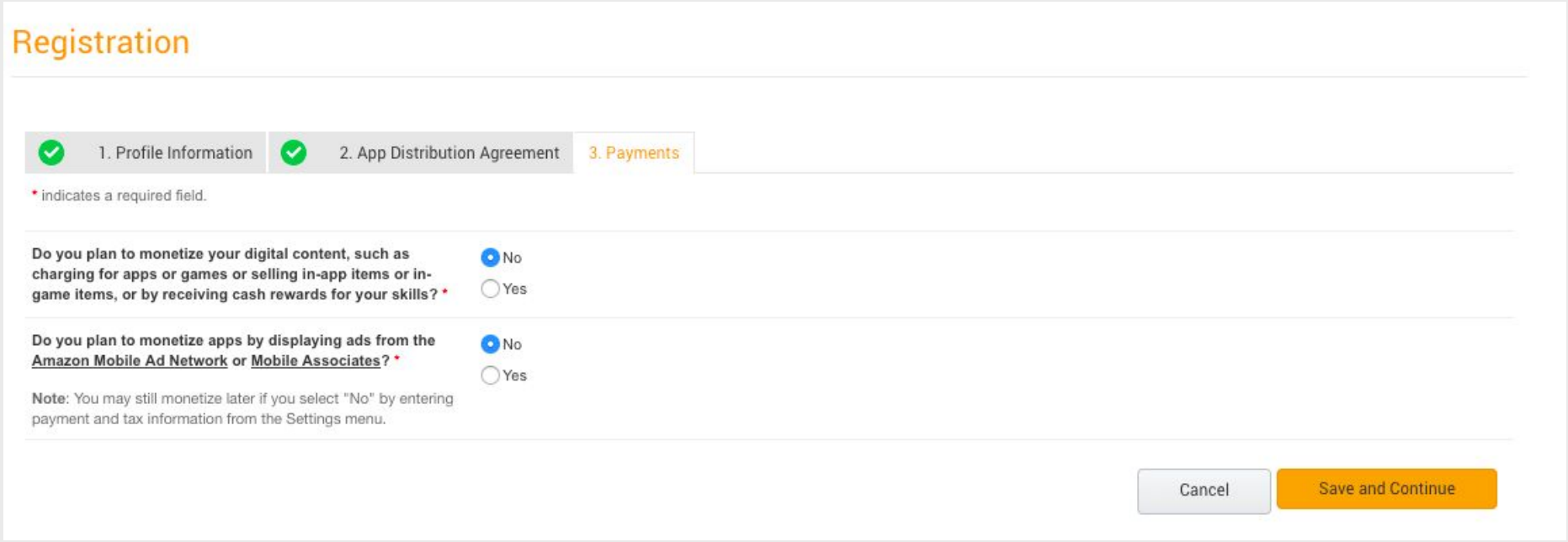
Create your Developer Profile.

Fill out the registration form & agree to the terms of use.



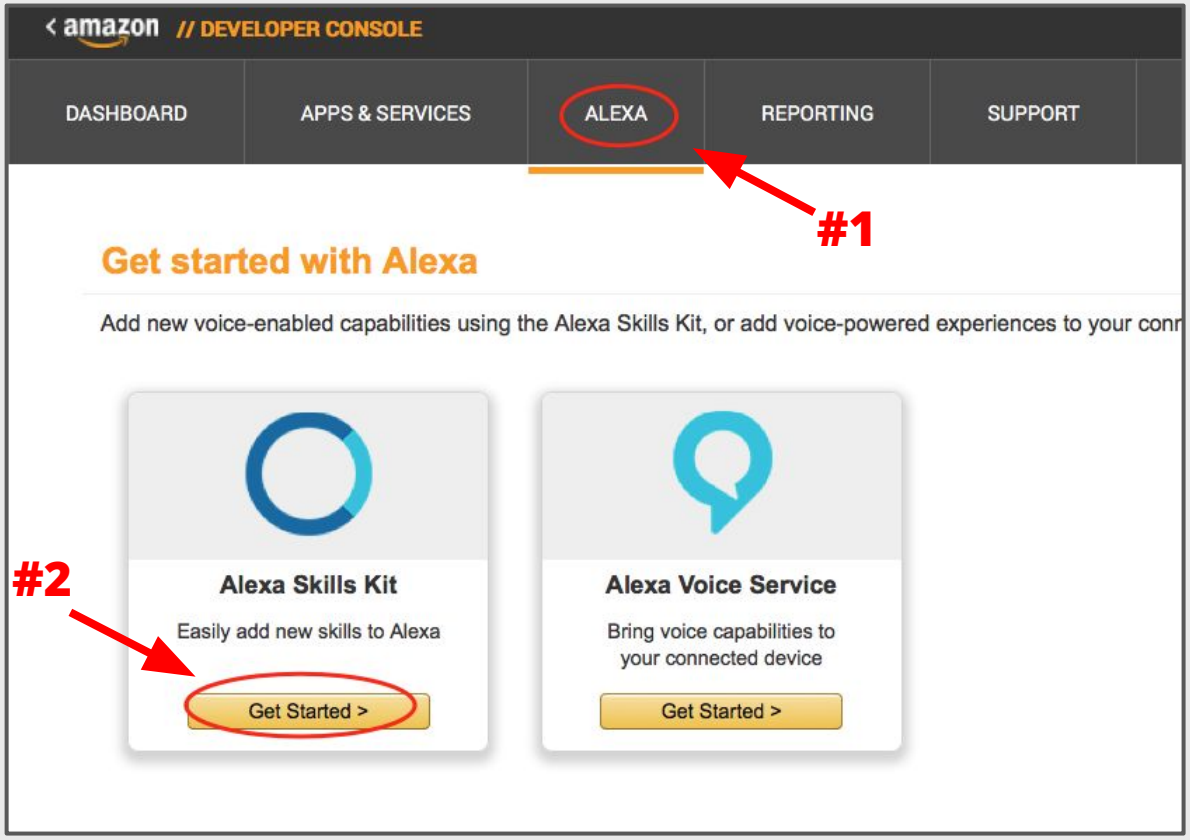
Create your Developer Profile.

Fill out the registration form & agree to the terms of use.



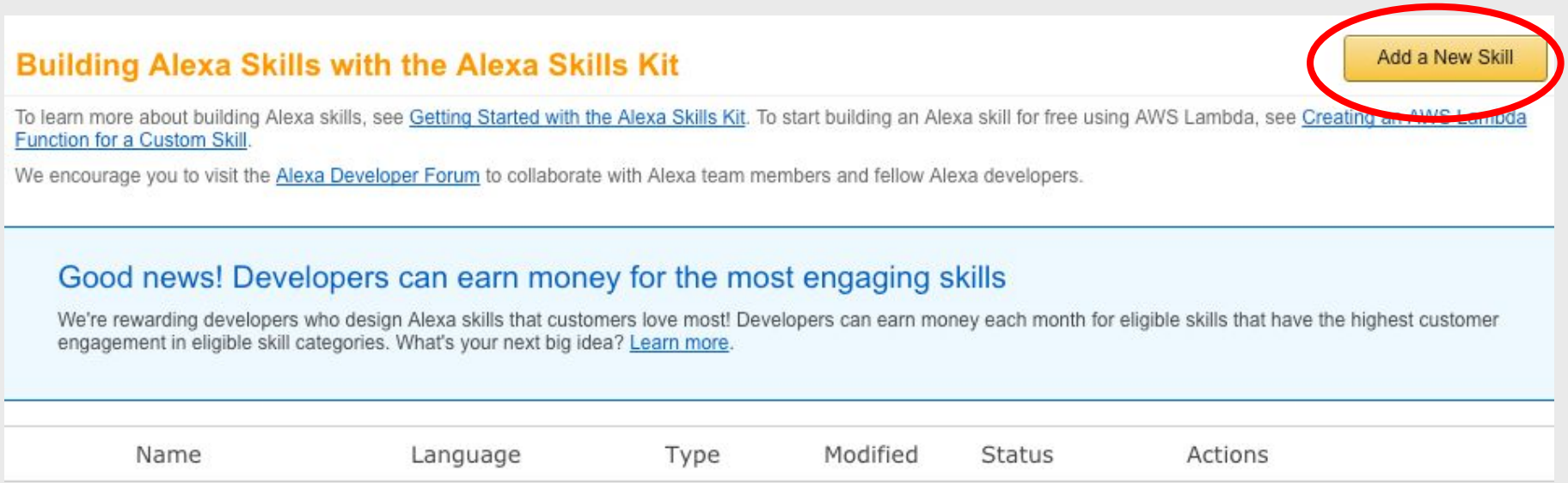
Create a New Alexa Skill.

Navigate to the Alexa tab.



Create a New Alexa Skill

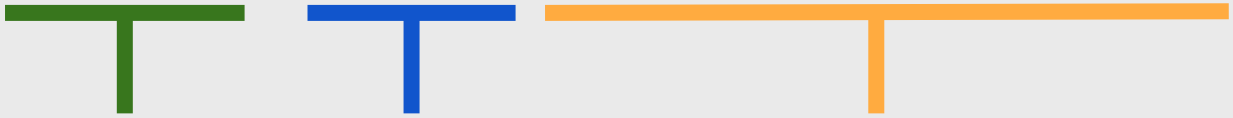
Click on "Getting Started" under “Alexa Skills Kit.”
Click “Add a New Skill” in the top right.



What is the Invocation Name?

This is a **1-3 word phrase** that users will say out loud to launch your skill.

Alexa, open UBHacking facts.



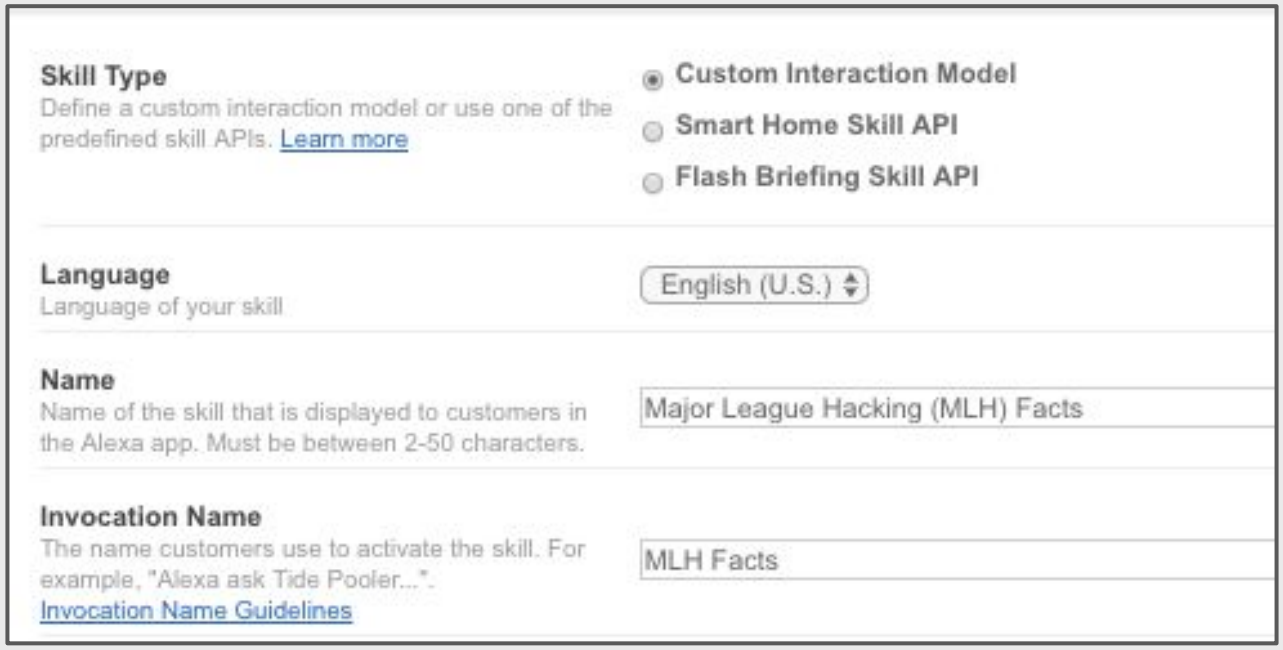
Wake Word

Launch

Skill Invocation Name

Name your Skill

1. Select *“Custom Interaction Model”* for Skill Type
2. Enter a Name for your Skill.
3. Enter an Invocation Name for your Skill.
4. Select the "No" option under Audio Player and click “Next”.



Grab the Intent Schema JSON

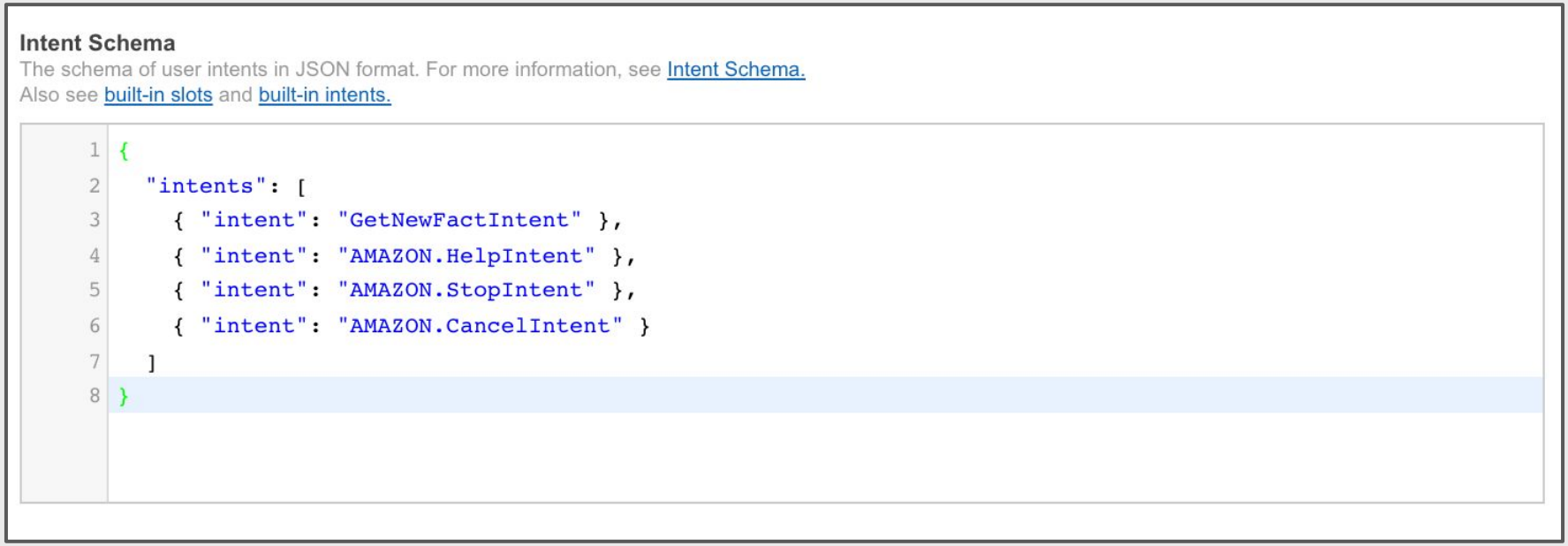
There's a file called `SpeechAssets/IntentSchema.json` inside the code you downloaded earlier with the following JSON:

```
{
  "intents": [
    { "intent": "GetNewFactIntent" },
    { "intent": "AMAZON.HelpIntent" },
    { "intent": "AMAZON.StopIntent" },
    { "intent": "AMAZON.CancelIntent" }
  ]
}
```

This code tells Alexa which code to trigger in our Lambda Function and some basic defaults like "STOP" and "HELP".

Paste the JSON into the Intent Schema Box

Copy the code you found in `SpeechAssets/IntentSchema.json` into the box on the Alexa Developer dashboard.



Customize your Utterances

Open up `SpeechAssets/SampleUtterances.txt` and replace “Major League Hacking” with your name:

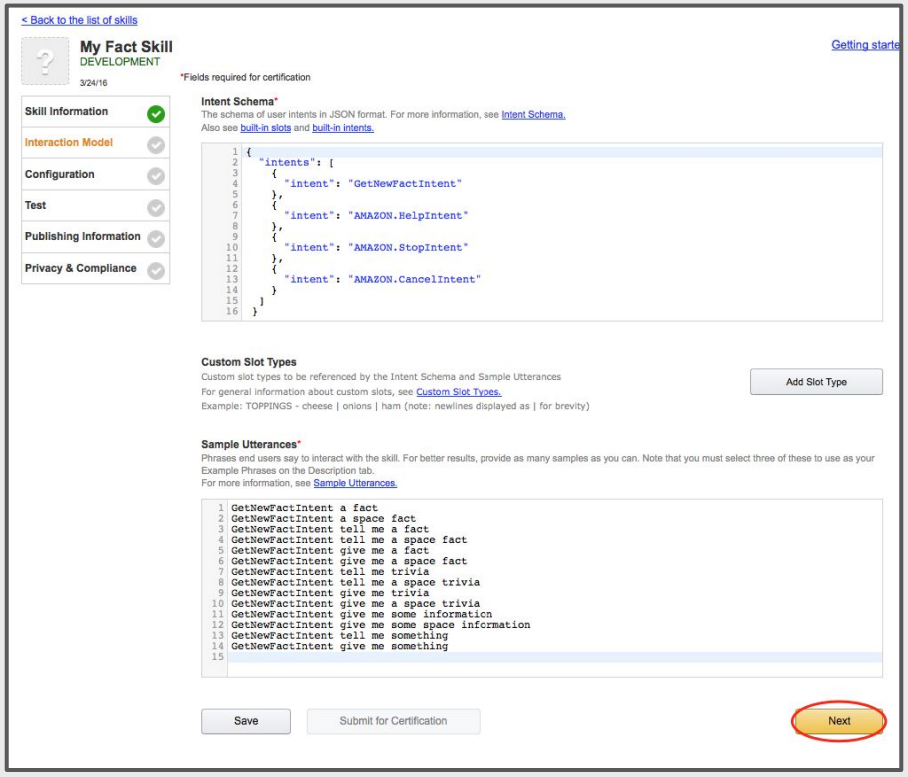
```
GetNewFactIntent a fact
GetNewFactIntent a Major League Hacking fact
GetNewFactIntent tell me a fact
GetNewFactIntent tell me a Major League Hacking fact
GetNewFactIntent give me a fact
GetNewFactIntent give me a Major League Hacking fact
GetNewFactIntent tell me trivia
GetNewFactIntent tell me a Major League Hacking trivia
GetNewFactIntent give me trivia
GetNewFactIntent give me a Major League Hacking trivia
GetNewFactIntent give me some information
GetNewFactIntent give me some Major League Hacking information
GetNewFactIntent tell me something
GetNewFactIntent give me something
```

Paste your Custom Utterances into the Sample Utterances Box

Copy your customized `SpeechAssets/SampleUtterances.txt` into the box on the Alexa Developer dashboard.

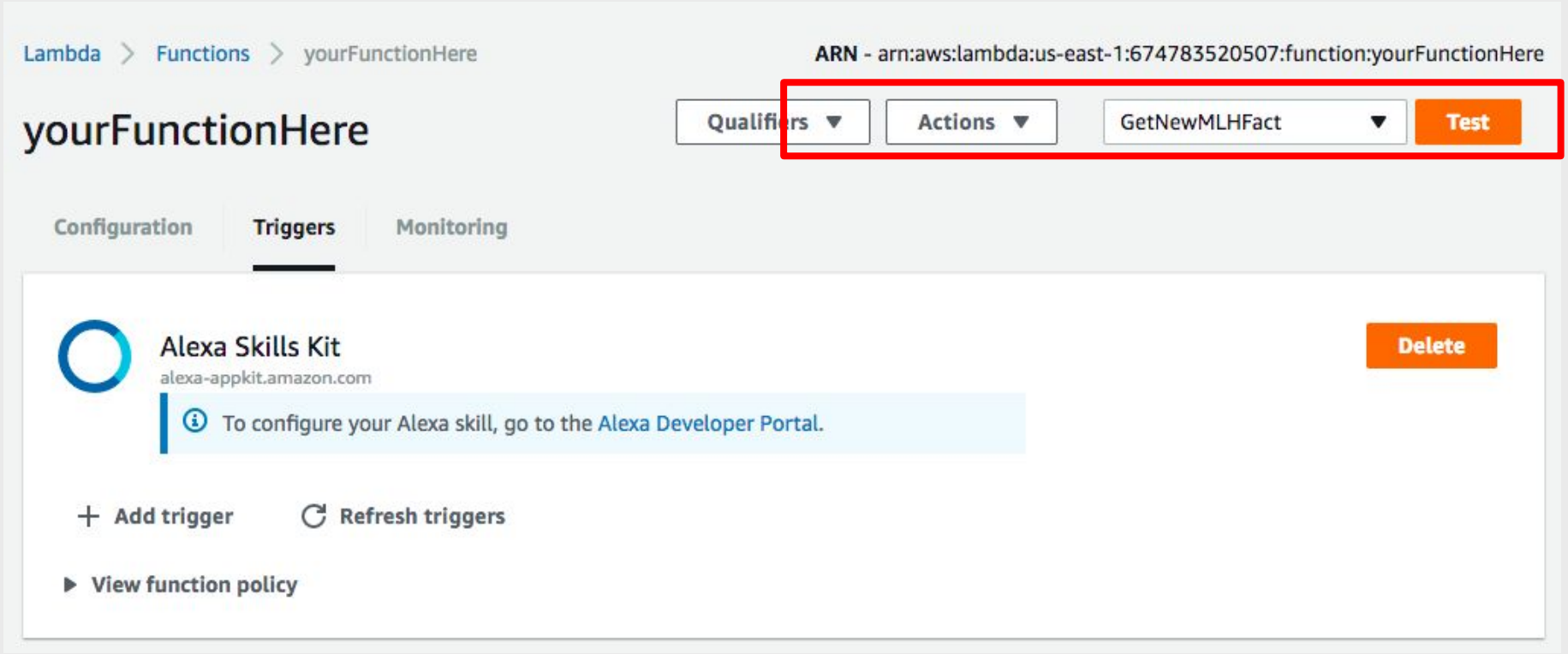
Save your Interaction Model and Continue.

Once you've defined your intent schema & sample utterances, hit "next" in the bottom corner to move on.



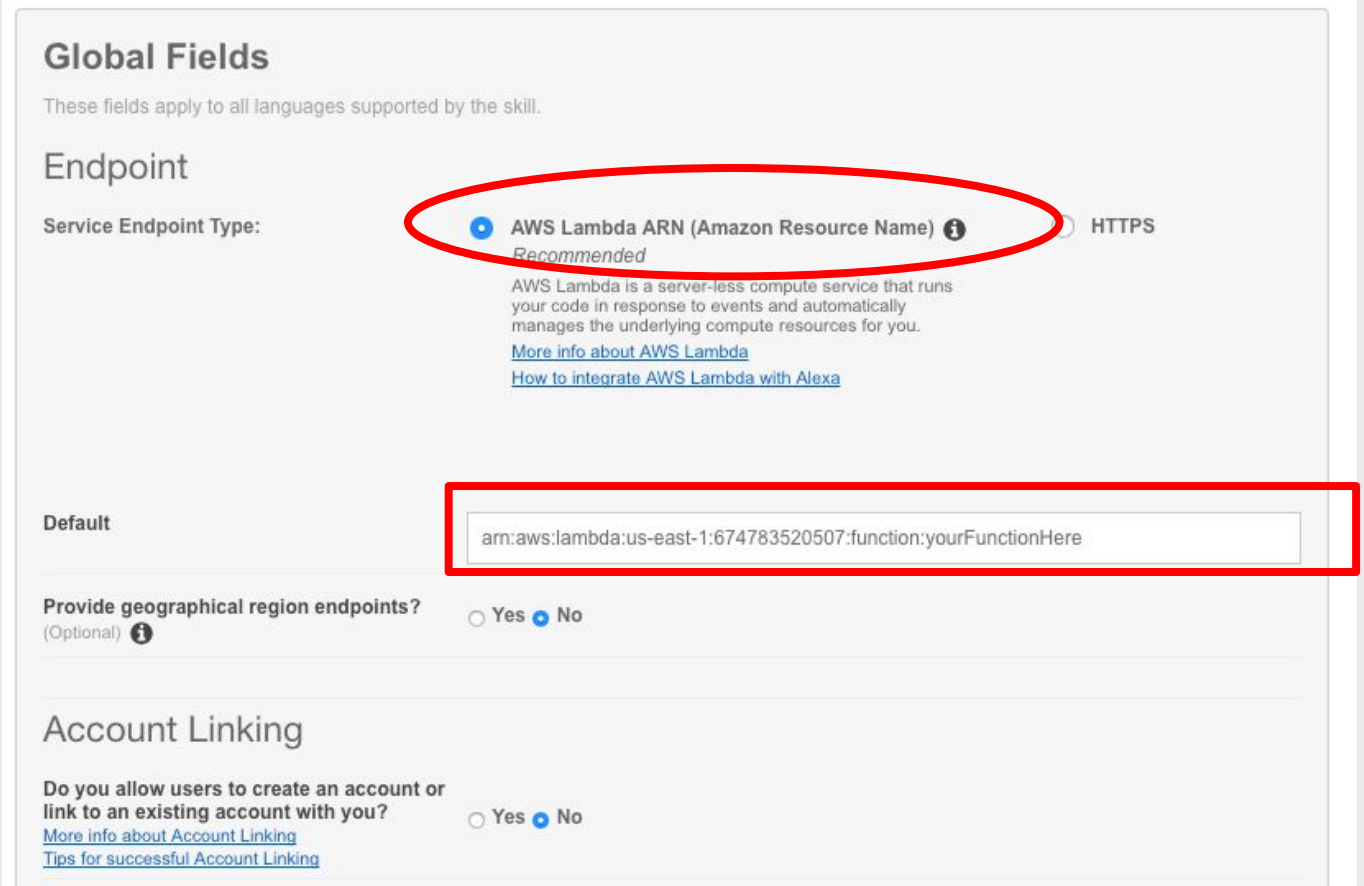
Go get your Lambda Function's ARN

Remember the ARN we noted down earlier?
Time to put it to use



Connect Your Lambda Function to Your Skill.

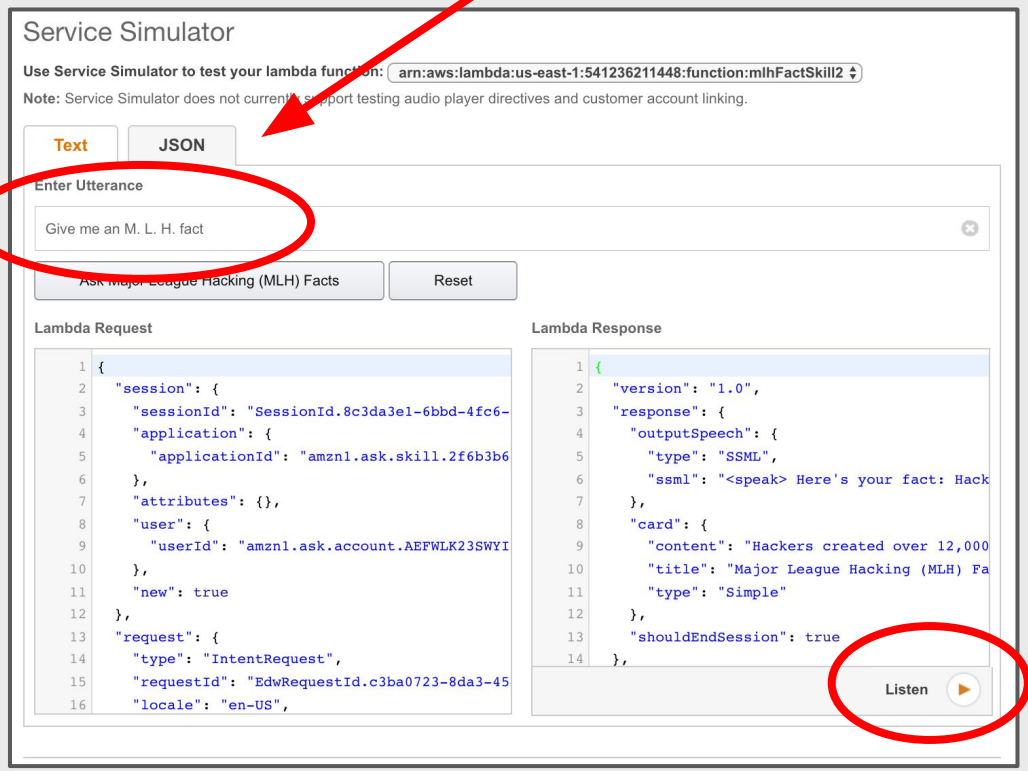
Select AWS Lambda ARN and the Region closest to you.
Paste the ARN from your function in the text box.



Test your Skill

Instructions

Type one of your sample utterances into the service simulator to see how Alexa would respond.



Hear what Alexa would say by clicking "Listen".

Troubleshooting Your Skill

If you aren't getting a valid response, check the following:

1. Do you have the right ARN copied from your Developer Portal/Skill into your your Lambda function?
2. Are you calling the right invocation name?
3. Are you saying launch, start or open?
4. Are you sure you have no other skills in your accounts with the same invocation name?

Table of Contents

Let's recap quickly...

1. Voice User Interfaces allow us to physically separate ourselves from devices.

2. Amazon Alexa makes it easy for you to create apps (skills) that utilize Voice User Interfaces.

3. Alexa takes care of speech recognition and context so you can focus on making a great app.

Feedback Form

If you liked this workshop or have any feedback, feel free to enter this form

<Form url>

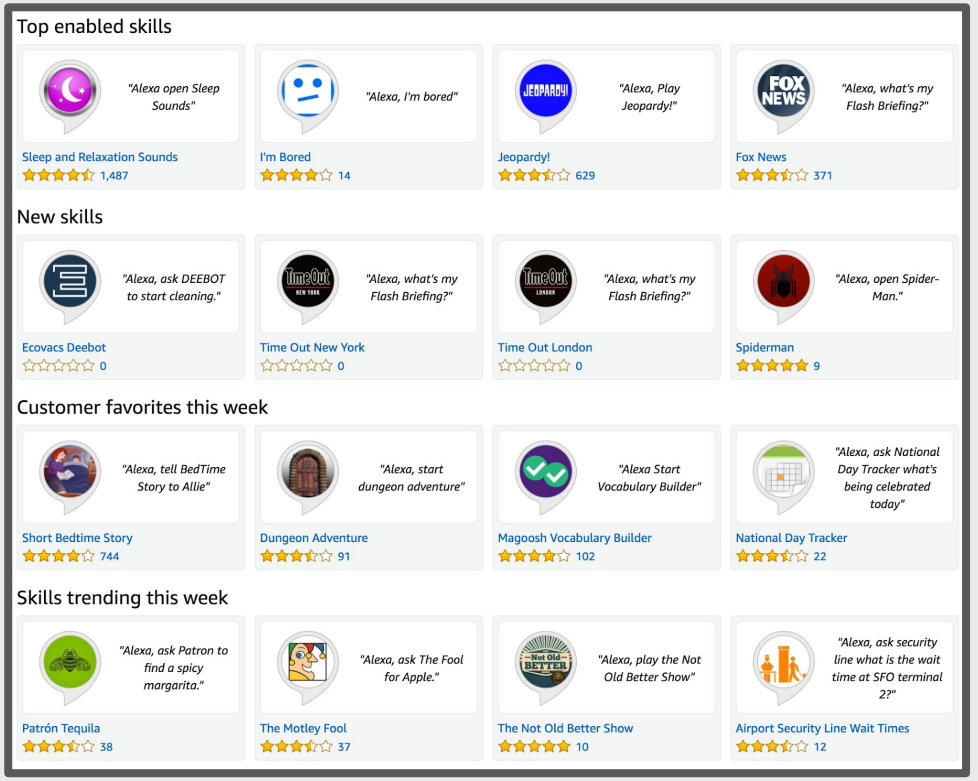
Table of Contents

Next Steps: Publish your Skill!

You have a working skill, the next step is to publish it on the Alexa Skill Marketplace.

Instructions

1. *Complete the "Publishing Information" and "Privacy and Compliance" sections on the developer portal.*
2. *Ensure that your skill meets the basic requirements.*



UB Hacking

Workshop

Hacking with Amazon Alexa