

How to Develop a Fact Skill for Alexa: Step-by-Step



We want to introduce another way to help you build useful and meaningful skills for Alexa quickly. We have launched a new fact skill template that makes it easy for developers or non-developers to create a skill similar to “Fact of the Day”, “Joke of the Day”, “Flash Cards” etc. The template leverages [AWS Lambda](#) and the [Alexa Skills Kit](#), while providing the business logic, use cases, error handling and help functions for your skill. You just need to come up with a fact idea (like “Food Facts”), plug in your fact list and edit the sample provided (we walk you through how it’s done). It’s a valuable way to quickly learn the end-to-end process for building and publishing an Alexa skill.

This tutorial will walk first-time Alexa skills developers through all the required steps involved in creating a skill using this fact skill template, called ‘SpaceGeek’. This post assumes you have some familiarity with JavaScript/Node.js (or a similar programming language) and the Alexa Skills Kit. For more background information on using the Alexa Skills Kit please [watch](#) this video. For guidance on designing a voice experience with Alexa you can also [watch](#) this video.

Using the Alexa Skills Kit, you can build an application that can receive and respond to voice requests made on the Alexa platform. In this tutorial, you’ll build a web service to handle notifications from Alexa and map this service to a Skill in the Amazon Developer Portal, making it available on your device and to all Alexa users after certification.

After completing this tutorial, you’ll know how to do the following:

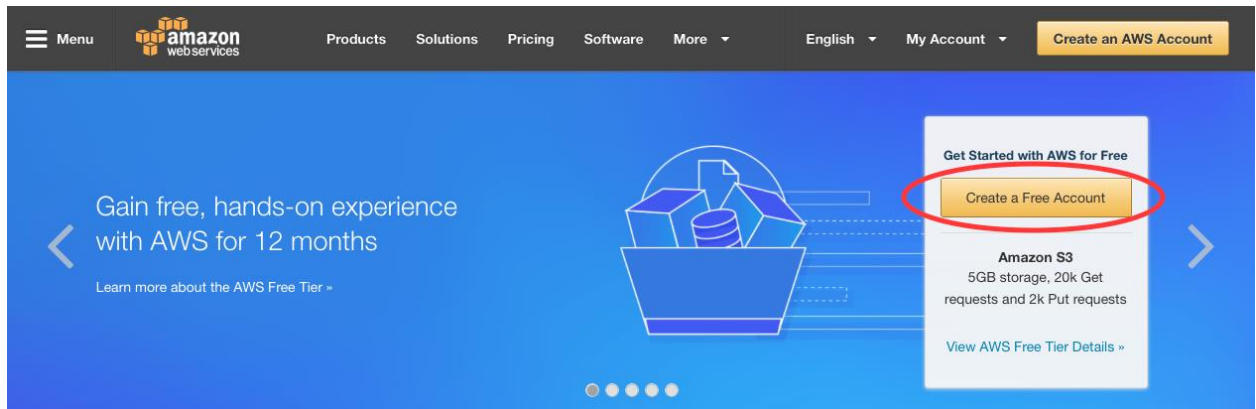
- **Create a fact-based skill** - This tutorial will walk first-time Alexa skills developers through all the required steps involved in creating a fact based skill using a template called ‘SpaceGeek’.
- **Understand the basics of VUI design** - Creating this skill will help you understand the basics of creating a working Voice User Interface (VUI) while using a cut/paste approach to development. You will learn by doing, and end up with a published Alexa skill. This tutorial includes instructions on how to customize the skill and submit for certification. For guidance on designing a voice experience with Alexa you can also [watch this video](#).
- **Use JavaScript/Node.js and the Alexa Skills Kit to create a skill** - You will use the template as a guide but the customization is up to you. For more background information on using the Alexa Skills Kit please [watch this video](#).
- **Get your skill published** - Once you have completed your skill, this tutorial will guide you through testing your skill and sending your skill through the certification process for making it available to be enabled by any Alexa user.

Let's Get Started

First, [download the code](#) and then follow the instructions below. On GitHub, click “**View Raw**” to download the zip file.

Step #1 – Create an AWS Account

1. Open aws.amazon.com and then choose ‘Create a Free Account’



- a. Follow the online instructions. Do not worry about the IAM role, we will do that later.
 - b. You will need a Valid Credit Card to set up your account (note the [AWS Free Tier](#) will suffice however. You can find out more about the [free tier here.](#))
 - c. Part of the sign-up procedure involves receiving a phone call and entering a PIN using the phone keypad.
2. Sign in to your Console



Sign In or Create an AWS Account

What is your e-mail or mobile number?

E-mail or mobile number:

☐ I am a new user.

☒ I am a returning user
and my password is:

Sign in using our secure server



[Forgot your password?](#)

- a. It can sometimes take a couple minutes for your new AWS account to go live. You will receive an e-mail when your account is active.

Step #2 – Create a Lambda Function

AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume - there is no charge when your code is not running. With Lambda, you can run code for virtually any type of application or backend service - all with zero administration. Just upload your code and Lambda takes care of everything required to run and scale your code with high availability.

Note: If you are new to Lambda and would like more information, visit the [Lambda Getting Started](#) guide.

1. **IMPORTANT:** Select **US East (N. Virginia)** region (upper right). This is the only region that currently supports Alexa Skill development.

AWS **Services** **IAM** **S3** **AWS IoT** **Edit** Noelle LaCharite **N. Virginia** **Support**

Amazon Web Services

Compute

- EC2**
Virtual Servers in the Cloud
- EC2 Container Service**
Run and Manage Docker Containers
- Elastic Beanstalk**
Run and Manage Web Apps
- Lambda**
Run Code in Response to Events

Storage & Content Delivery

- S3**
Scalable Storage in the Cloud
- CloudFront**
Global Content Delivery Network

Developer Tools

- CodeCommit**
Store Code in Private Git Repositories
- CodeDeploy**
Automate Code Deployments
- CodePipeline**
Release Software using Continuous Delivery

Management Tools

- CloudWatch**
Monitor Resources and Applications
- CloudFormation**
Create and Manage Resources with Templates

Internet of Things

- AWS IoT**
Connect Devices to the Cloud

Game Development

- GameLift**
Deploy and Scale Session-based Multiplayer Games

Mobile Services

- Mobile Hub**
Build, Test, and Monitor Mobile Apps
- Cognito**
User Identity and App Data Synchronization

Resource Groups [Learn more](#)

A resource group is a collection of resources that share one or more tags. Create a group for each project, application, or environment in your account.

[Create a Group](#) [Tag Editor](#)

Additional Resources

[Getting Started](#) [Read our documentation](#) or view our [training](#) to learn more about AWS.

2. Select Lambda from Compute services (upper left)

AWS **Services** **Edit**

Amazon Web Services

Compute

- EC2**
Virtual Servers in the Cloud
- EC2 Container Service**
Run and Manage Docker Containers
- Elastic Beanstalk**
Run and Manage Web Apps
- Lambda**
Run Code in Response to Events

Storage & Content Delivery

- S3**
Scalable Storage in the Cloud
- CloudFront**
Global Content Delivery Network
- Elastic File System** **PREVIEW**
Fully Managed File System for EC2
- Glacier**
Archive Storage in the Cloud
- Import/Export Snowball**
Large Scale Data Transport
- Storage Gateway**
Hybrid Storage Integration

Database

Developer Tools

- CodeCommit**
Store Code in Private Git Repositories
- CodeDeploy**
Automate Code Deployments
- CodePipeline**
Release Software using Continuous Delivery

Management Tools

- CloudWatch**
Monitor Resources and Applications
- CloudFormation**
Create and Manage Resources with Templates
- CloudTrail**
Track User Activity and API Usage
- Config**
Track Resource Inventory and Changes
- OpsWorks**
Automate Operations with Chef
- Service Catalog**
Create and Use Standardized Products
- Trusted Advisor**
Optimize Performance and Security

Internet of Things

- AWS IoT**
Connect Devices to the Cloud

Game Development

- GameLift**
Deploy and Scale Session-based Multiplayer Games

Mobile Services

- Mobile Hub**
Build, Test, and Monitor Mobile Apps
- Cognito**
User Identity and App Data Synchronization
- Device Farm**
Test Android, FireOS, and iOS on Real Devices in the Cloud
- Mobile Analytics**
Collect, View and Export App Data
- SNS**
Push Notification Service

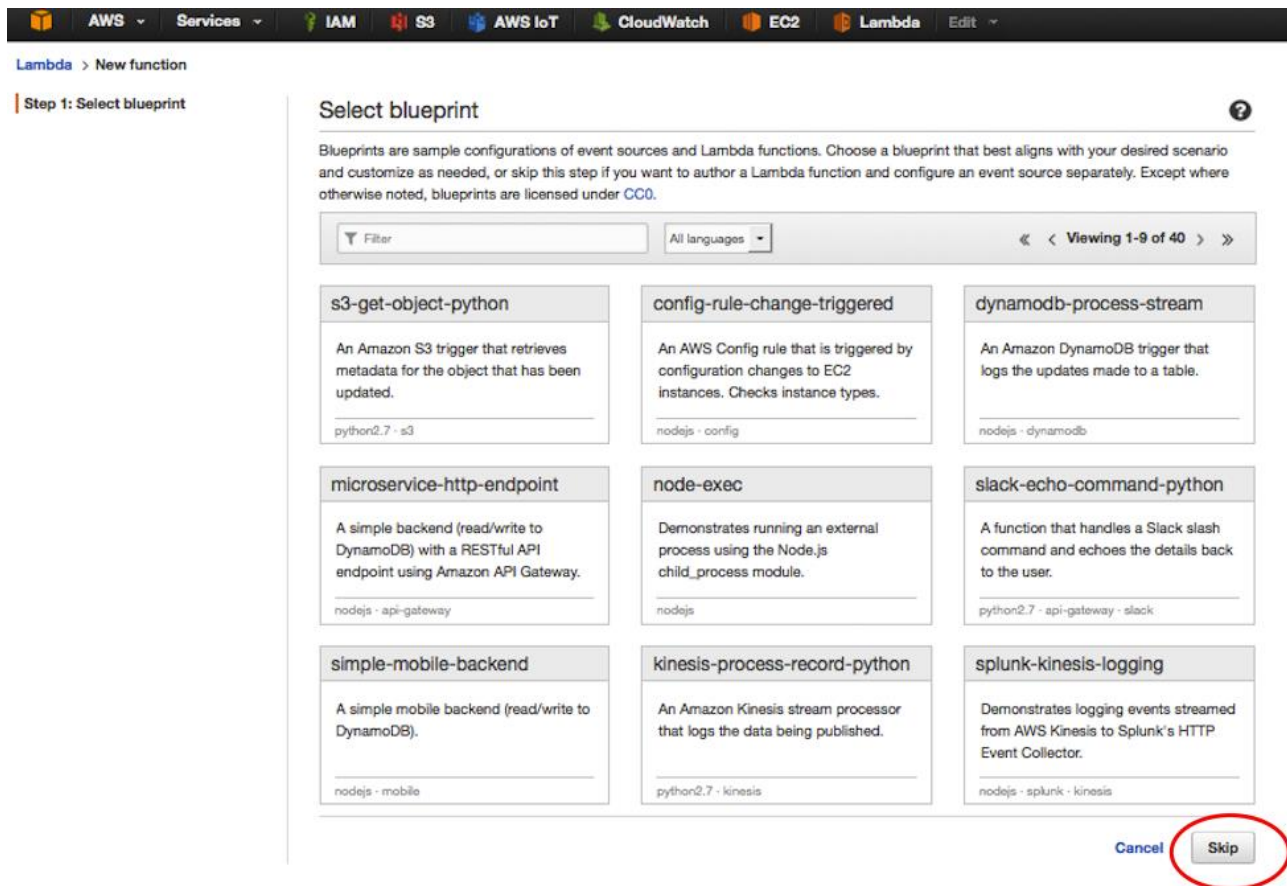
Application Services

- API Gateway**
Build, Deploy and Manage APIs

3. Select **“Create a Lambda Function”** to begin the process of defining your Lambda function.



4. At the bottom of the ‘Select Blueprint’ page, select **“Skip”**.



5. You should be in ‘Configure Function’.

Lambda > New function

Step 1: Select blueprint

Step 2: Configure function

Step 3: Review

Configure function

A Lambda function consists of the custom code you want to execute. [Learn more](#) about Lambda functions.

Name* MyFactSkill

Description My fact skill based on the SpaceGeek template.

Runtime* Node.js

a. Enter the Name, Description, and Runtime for your skill as in the example.

6. Select the 'Code Entry Type' as 'Upload Zip File' and upload the zip file containing the example. Here's the [link](#) again for reference, you will need to click on "View Raw" to download.

Lambda > New function

Step 1: Select blueprint

Step 2: Configure function

Step 3: Review

Configure function

A Lambda function consists of the custom code you want to execute. [Learn more](#) about Lambda functions.

Name* myFactSkill

Description My Fact Skill

Runtime* Node.js

Lambda function code

Provide the code for your function. Use the editor if your code does not require custom libraries (other than the aws-sdk). If you need custom libraries, you can upload your code and libraries as a .ZIP file. [Learn more](#) about deploying Lambda functions.

Code entry type ☐ Edit code inline ☒ Upload a .ZIP file ☐ Upload a .ZIP from Amazon S3

For .ZIP files larger than 10 MB, consider uploading via S3.

Upload

You have chosen to upload a .zip file but have not selected a file yet.

7. Set your handler and role as follows:

a. Keep Handler as 'index.handler'

b. Drop down the “Role” menu and select “* **Basic execution role**”. (Note: if you have already used Lambda you may already have a ‘lambda_basic_execution’ role created that you can use.) This will launch a new tab in the IAM Management Console.

Lambda function code

Provide the code for your function. Use the editor if your code does not require custom libraries (other than the aws-sdk). If you need custom libraries, you can upload your code and libraries as a .ZIP file. [Learn more](#) about deploying Lambda functions.

Code entry type ☐ Edit code inline ☒ Upload a .ZIP file ☐ Upload a .ZIP from Amazon S3

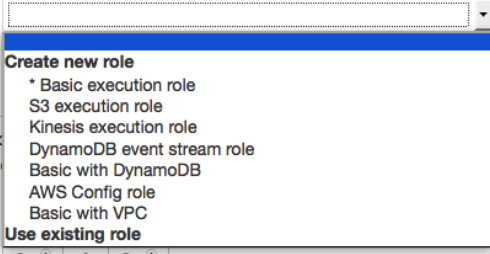
For .ZIP files larger than 10 MB, consider uploading via S3.

 Upload

You have chosen to upload a .zip file but have not selected a file yet.

Lambda function handler and role

Handler*

Role* 

Advanced settings

These settings allow you to control the function's execution (by selecting memory) or changing the function's timeout.

Memory (MB)*


Timeout* min sec

Suggested role: Basic execution role

lambda execution roles.

function. Changing your resource settings (by Lambda pricing works.

8. You will be asked to set up your Identity and Access Management or “IAM” role if you have not done so. AWS Identity and Access Management (IAM) enables you to securely control access to AWS services and resources for your users. Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources. We need to create a role that allows our skill to invoke this Lambda function.

 AWS Services Edit

AWS Lambda requires access to your resources

AWS Lambda uses an IAM role that grants your custom code permissions to access AWS resources it needs.

Hide Details

Role Summary

Role Description	Lambda execution role permissions
IAM Role	Create a new IAM Role
Role Name	lambda_basic_execution

View Policy Document

9. Accept the defaults to create a new IAM Role with a role name of `lambda_basic_execution`. Select “**Allow**” in the lower right corner and you will be returned to your Lambda function.



10. Keep the Advanced settings as default. Select ‘**Next**’ and review. You should see something like below. Then select ‘**Create Function**’:

Lambda > New function

Step 1: Select blueprint

Step 2: Configure function

Step 3: Review

Review

Please review your Lambda function details. You can go back to edit changes for each section. When you are ready, click **Create function** to complete the setup process.

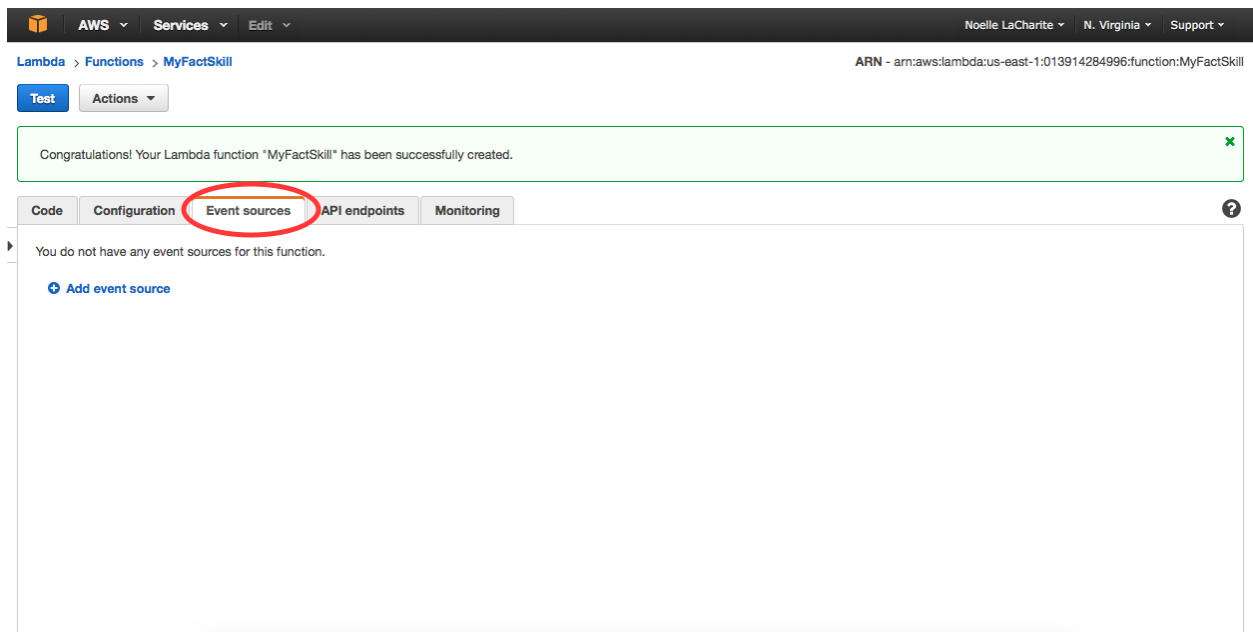
Lambda function Edit

Name	myFactSkill
Description	My Fact Skill function
Runtime	NodeJS
Handler	index.handler
Role	lambda_basic_execution
Memory (MB)	128
Timeout	3

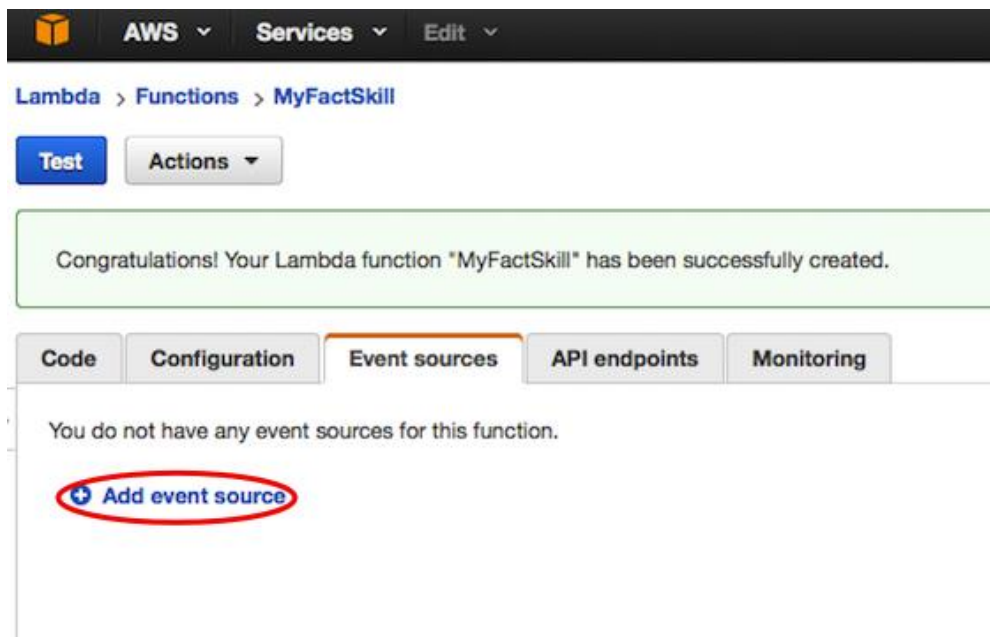
Cancel Previous Create function

11. Next we need to create an Event source. *Event sources* publish events that cause the Lambda function to be invoked. Upon invocation, AWS Lambda executes your code by passing the event to the handler in your Lambda function code. You associate an event source with your Lambda function using an event source mapping. We will use the Alexa Skills Kit event source and map it to this function.

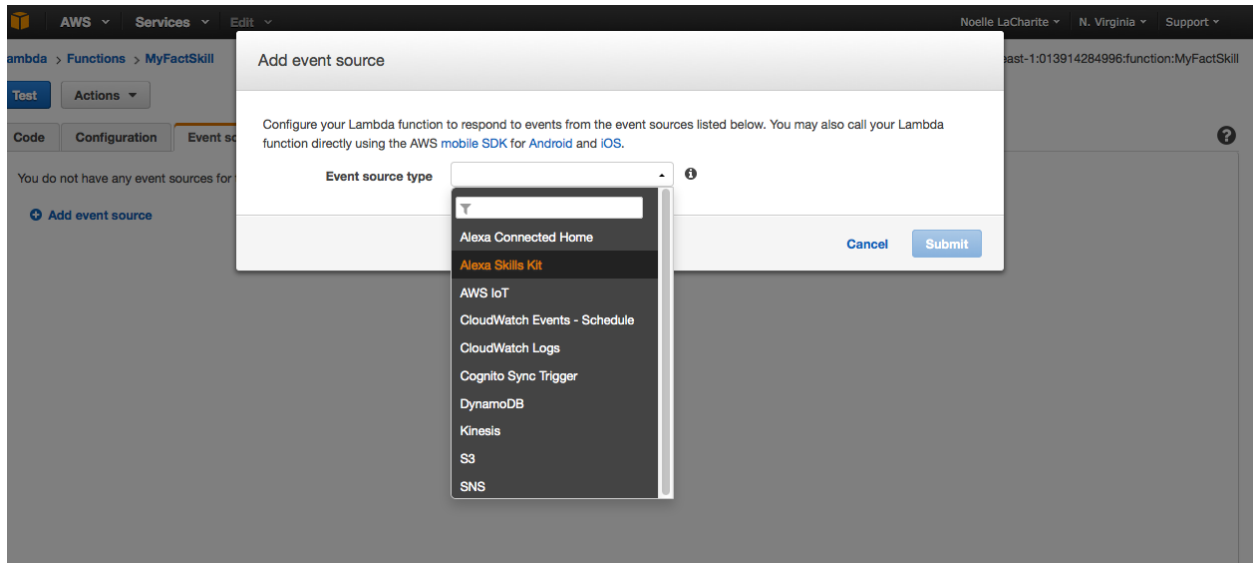
- a. In your Lambda function tabs, select ‘**Event Sources**’



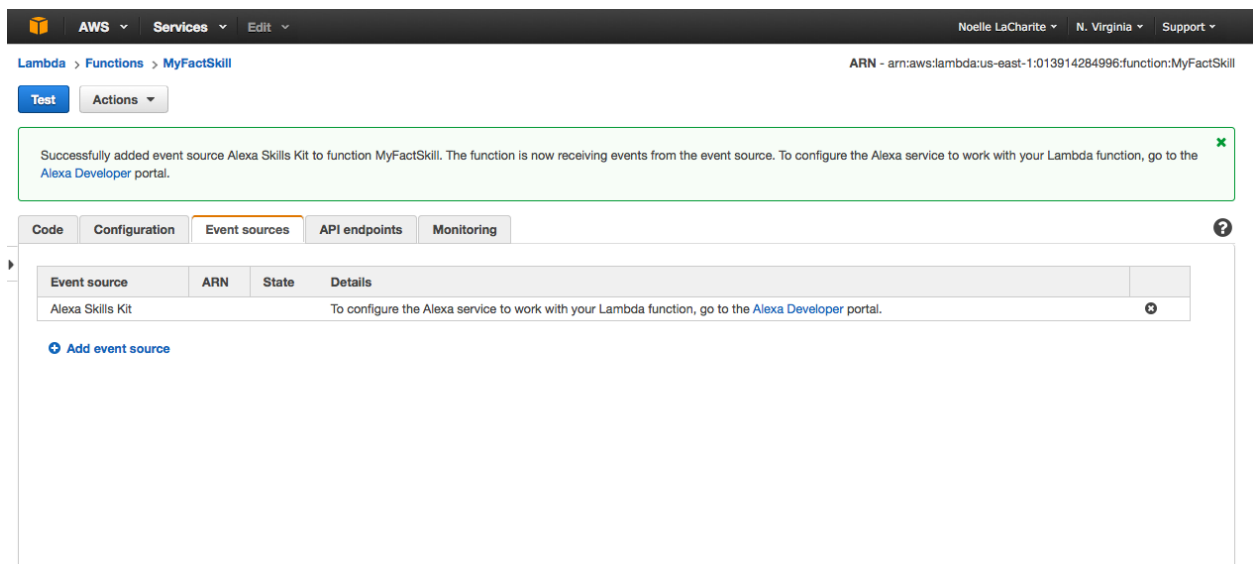
12. Select **'Add event source'**.




a. Select **'Alexa Skill Kit'** from the dropdown:



13. You should see the confirmation of creation of the free tier event source “Alexa Skills Kit”:



14. Keep this tab open as you will need to copy the ARN for your Lambda function for use in the developer portal. You can find your ARN at the top right corner of the function page:



AWS

Services

Edit

Noelle LaCharite

N. Virginia

Support

Lambda > Functions > MyFactSkill

ARN: arn:aws:lambda:us-east-1:013914284996:function:MyFactSkill

Test

Actions

Successfully added event source Alexa Skills Kit to function MyFactSkill. The function is now receiving events from the event source. To configure the Alexa service to work with your Lambda function, go to the [Alexa Developer](#) portal.

Code

Configuration

Event sources

API endpoints

Monitoring

Event source	ARN	State	Details
Alexa Skills Kit			To configure the Alexa service to work with your Lambda function, go to the Alexa Developer portal.

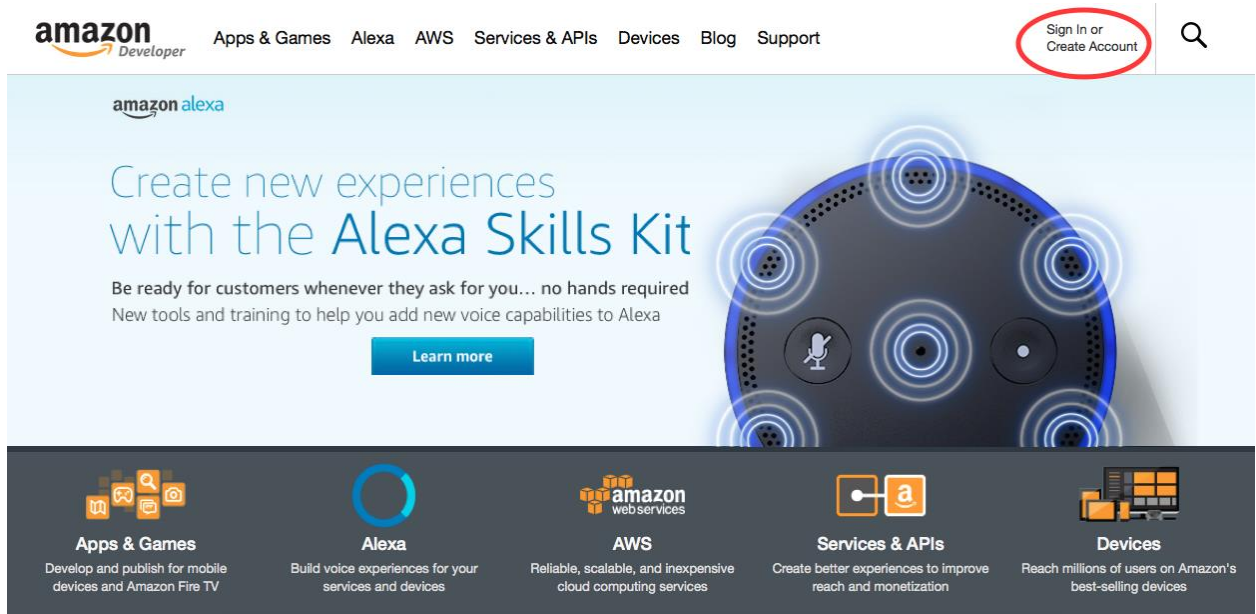
+

 Add event source

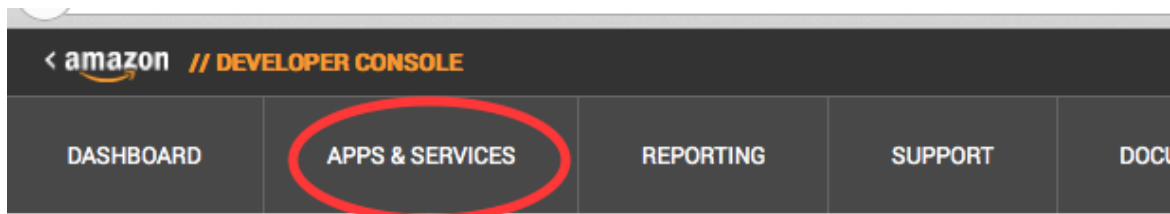
Step #3 – Set-up Your Alexa Skill in the Developer Portal

Skills are managed through the Amazon Developer Portal. You'll link the Lambda function you created above to a Skill defined in the Developer Portal.

1. Open a new tab and go to the [Developer Portal](#). Sign in or create a free account (upper right). You might see a different image if you have registered already or our page may have changed. If you see a similar menu and the ability to create an account or sign in, you are in the right place.



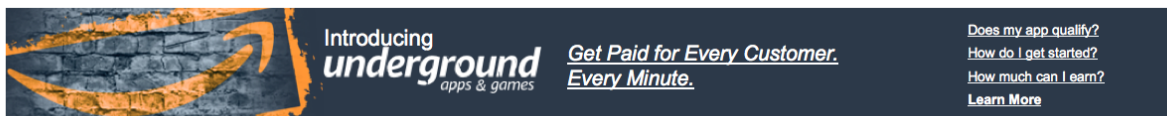
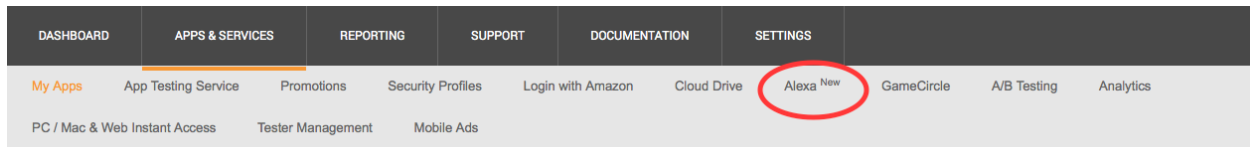
2. Once you've signed in, navigate to **Apps & Services**.



Developer Communications

[Announcements](#) | [Notifications](#)

3. Then select **Alexa**. You can also bookmark this page for future reference:
<https://developer.amazon.com/edw/home.html#/skills/>

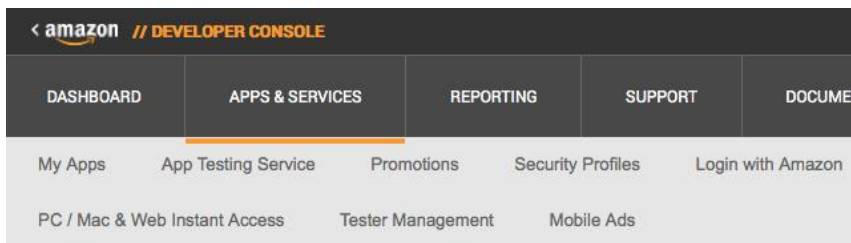


0 apps under this account

Nothing Found

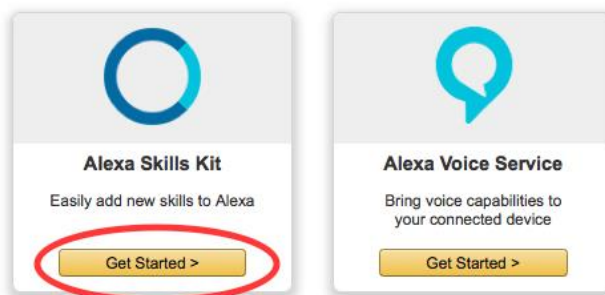
Add a New App

4. Choose **Alexa Skills Kit**:



Get started with Alexa

Add new voice-enabled capabilities using the Alexa Skills Kit, or add voice-powered experiences to yo



FAQs Contact Us App Distribution Agreement Trademark Guidelines Terms of Use

5. Here is where you will define and manage your skill.

Select “Add New Skill”:

The screenshot shows the top navigation bar of the Alexa Developer Console. The 'APPS & SERVICES' tab is selected. Below the navigation bar, the 'Add a New Skill' button is highlighted with a red circle. The page title is 'Building Alexa Skills with the Alexa Skills Kit'.

Building Alexa Skills with the Alexa Skills Kit

To learn more about building Alexa skills, see [Getting Started with the Alexa Skills Kit](#). To start building an Alexa skill for free using AWS Lambda, see [Developing an Alexa Skill as a Lambda Function](#).

We encourage you to visit the [Alexa Developer Forum](#) to collaborate with Alexa team members and fellow Alexa developers.

6. Add the name of the skill. You can use “My Fact Skill” for this example. Remember, when you create a skill that you will publish, you will use a name that you define for your skill. That name will be the one that shows up in the Alexa App.
7. Add the invocation name. Since we are using the sample, type “space geek”.

The screenshot shows the 'Add New Skill' form. It has three main sections: 'Name', 'Invocation Name', and 'Endpoint'. The 'Name' field is filled with 'My Fact Skill'. The 'Invocation Name' field is filled with 'space geek'. The 'Endpoint' section has two radio buttons: 'HTTPS' (selected) and 'Lambda ARN (Amazon Resource Name)'. Below the radio buttons is a text input field. At the bottom, there are three buttons: 'Save', 'Submit for Certification', and 'Next'.

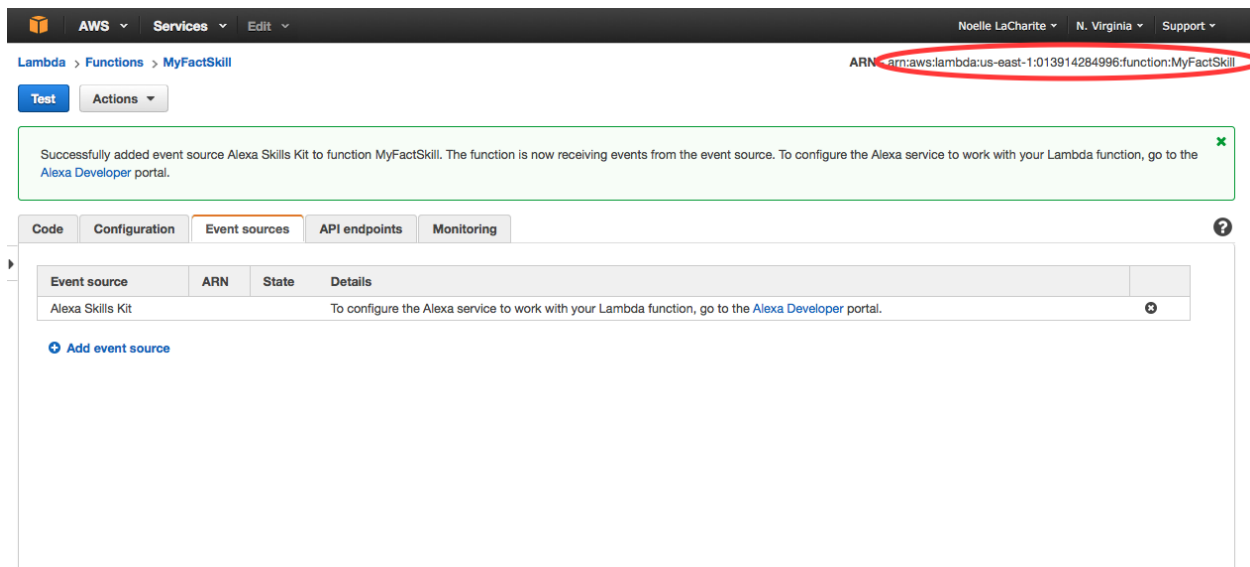
Name *
The name of this skill. This is the name displayed in the Alexa App.

Invocation Name *
The name users will say to interact with this skill. This does not have to be the same as the skill name. The invocation name must comply with the [Invocation Name Guidelines](#).

Endpoint *
The URL for the service endpoint, e.g. [https://myskill.ishere.com/somepath](#), or the Lambda ARN.
[More info about AWS Lambda](#)
[How to integrate AWS Lambda with Alexa](#)

☐ HTTPS ☒ **Lambda ARN (Amazon Resource Name)** ?

8. Add the ARN from the Lambda function you created in the AWS Console earlier. Select the **Lambda ARN (Amazon Resource Name)** radio button.
 - a) You should still have that browser tab open. In the AWS Management Console, copy the ARN in the upper right corner.



b) Paste the ARN into the **Endpoint** field. Here's an example:

*Fields required for certification

Application Id
The ID for this skill
amzn1.echo-sdk-ams.app.dffbcd1a-309d-4270-a890-1c2092e77edb

Name *
The name of this skill. This is the name displayed in the Alexa App.
My Fact Skill

Invocation Name *
The name users will say to interact with this skill. This does not have to be the same as the skill name. The invocation name must comply with the [Invocation Name Guidelines](#)
space geek

Endpoint *
The URL for the service endpoint, e.g. <https://myskill.ishere.com/somepath>, or the Lambda ARN, [More info about AWS Lambda](#) [How to integrate AWS Lambda with Alexa](#)
☐ HTTPS ☒ Lambda ARN (Amazon Resource Name) ?
 arn:aws:lambda:us-east-1:013914284996:function:MyFactSkill

Save Submit for Certification Next

9. Select **Next**.

10. Now we need to define our skill's interaction model. Let's begin with the intent schema. In the context of Alexa, an *intent* represents an action that fulfills a user's spoken request. Intents can optionally have arguments called *slots*. We will not be using custom slots in this template, but they are very useful if you want to parameterize your intents.

- a) **Review** the Intent Schema below. This is written in JSON and provides the information needed to map the intents we want to handle programmatically. **Copy** this from the intent schema in the [GitHub repository here](#).

[< Back to the list of skills](#)

**My Fact Skill**
DEVELOPMENT
3/24/16

3/24/16

Fields required for certification

Getting started

Skill Information 

Interaction Model 

Configuration 

Test 

Publishing Information 

Privacy & Compliance 

Intent Schema

The schema of user intents in JSON format. For more information, see [Intent Schema](#). Also see [built-in slots](#) and [built-in intents](#).

```
1 {
2   "intents": [
3     {
4       "intent": "GetNewFactIntent"
5     },
6     {
7       "intent": "AMAZON.HelpIntent"
8     },
9     {
10      "intent": "AMAZON.StopIntent"
11    },
12    {
13      "intent": "AMAZON.CancelIntent"
14    }
15  ]
16 }
```

Below you will see the intents for getting a new fact, and then a collection of built-in intents to simplify handling common user tasks. Help intent will handle any time the user asks for help, stop and cancel are built-in intents to make it easier for you to handle when a user wants to exit the application. For more on the use of built-in intents, [go here](#).

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


11. The next step is to build the utterance list.

Given the flexibility and variation of spoken language in the real world, there will often be many different ways to express the same request. Providing these different phrases in your sample utterances will help improve voice recognition for the abilities you add to Alexa. It is important to include as wide a range of representative samples as you can — all the phrases that you can think of that are possible in use (though do not include samples that users will never speak). Alexa also attempts to generalize based on the samples you provide to interpret spoken phrases that differ in minor ways from the samples specified.

- a. Now it is time to add the Utterances. Copy/paste the sample utterances from [GitHub](#). An example of utterances is listed below. Once they are copied, the screen should look similar to the following image:

Sample Utterances*

Phrases end users say to interact with the skill. For better results, provide as many samples as you can. Note that you must select three of these to use as your Example Phrases on the Description tab.


For more information, see [Defining the Voice Interface for an Alexa skill](#).

```
1 GetNewFactIntent a fact
2 GetNewFactIntent a space fact
3 GetNewFactIntent tell me a fact
4 GetNewFactIntent tell me a space fact
5 GetNewFactIntent give me a fact
6 GetNewFactIntent give me a space fact
7 GetNewFactIntent tell me trivia
8 GetNewFactIntent tell me a space trivia
9 GetNewFactIntent give me trivia
10 GetNewFactIntent give me a space trivia
11 GetNewFactIntent give me some information
12 GetNewFactIntent give me some space information
13 GetNewFactIntent tell me something
14 GetNewFactIntent give me something
```

12. Select **Save**. You should see the interaction model being built (this make a take a minute or 2). If you select next, your changes will be saved and you will go directly to the test screen. After selecting **Save**, it should now look like this:

[< Back to the list of skills](#)

[Getting started](#)

**My Fact Skill**
DEVELOPMENT
3/24/16

Skill Information

Interaction Model

Configuration

Test

Publishing Information

Privacy & Compliance

*Fields required for certification

Intent Schema*

The schema of user intents in JSON format. For more information, see [Intent Schema](#). Also see [built-in slots](#) and [built-in intents](#).

```
1 {  
2   "intents": [  
3     {  
4       "intent": "GetNewFactIntent"  
5     },  
6     {  
7       "intent": "AMAZON.HelpIntent"  
8     },  
9     {  
10      "intent": "AMAZON.StopIntent"  
11    },  
12    {  
13      "intent": "AMAZON.CancelIntent"  
14    }  
15  ]  
16 }
```

Custom Slot Types

Custom slot types to be referenced by the Intent Schema and Sample Utterances

For general information about custom slots, see [Custom Slot Types](#).

Example: TOPPINGS - cheese | onions | ham (note: newlines displayed as | for brevity)

Add Slot Type

Sample Utterances*

Phrases end users say to interact with the skill. For better results, provide as many samples as you can. Note that you must select three of these to use as your Example Phrases on the Description tab.

For more information, see [Sample Utterances](#).

```
1 GetNewFactIntent a fact  
2 GetNewFactIntent a space fact  
3 GetNewFactIntent tell me a fact  
4 GetNewFactIntent tell me a space fact  
5 GetNewFactIntent give me a fact  
6 GetNewFactIntent give me a space fact  
7 GetNewFactIntent tell me trivia  
8 GetNewFactIntent tell me a space trivia  
9 GetNewFactIntent give me trivia  
10 GetNewFactIntent give me a space trivia  
11 GetNewFactIntent give me some information  
12 GetNewFactIntent give me some space information  
13 GetNewFactIntent tell me something  
14 GetNewFactIntent give me something  
15
```

Save


Submit for Certification

Next

13. Select **Next**, and you will be taken to the Configuration screen. Select “No” for account linking and **Next** to move to the Test screen.

[< Back to the list of skills](#)

[Getting started](#)

**My Fact Skill**
DEVELOPMENT
3/24/16

Skill Information

Interaction Model

Configuration

Test

Publishing Information

Privacy & Compliance

*Fields required for certification

Account Linking

Do you allow users to create an account or link to an existing account with you? [Learn more](#) *

☐ Yes ☒ No

Save

Submit for Certification

Next

14. You have now completed the initial development of your skill. You should see a screen that looks like this.

[< Back to the list of skills](#)



[Getting started](#)

Skill Information	✓
Interaction Model	✓
Configuration	✓
Test	✓
Publishing Information	✓
Privacy & Compliance	✓

You should now see 4 green checkmarks.

Start testing this skill

Enable ☐ This skill is enabled for testing on your account.?

Once you have completed testing on your device, please complete the Description and Publishing Information tab, then submit the skill for certification.

If it passes Amazon's testing and certification process, it will become available to Alexa end users.

Try this on your Echo: **Alexa ask space geek**

Voice Simulator

Hear how Alexa will speak a response entered in plain text or SSML. [Learn more about supported SSML tags.](#)

For example: Here is a word spelled out: `<say-as interpret-as="spell-out">hello</say-as>`.

Here is a word spelled out: `<say-as interpret-as="spell-out">hello</say-as>`.

Listen

Service Simulator

Use Service Simulator to test your lambda function.

Text

Json

Enter Utterance *

Ask My Fact Skill

Reset

We are now ready to test.

15. In the Test area, we are going to enter a sample utterance in the service simulator section and see how Alexa will respond. In this example, we have called the skill 'Space Geek'. This is the 'Invocation Name' we set up on the Skill Information line in the "Skill Information" section.

- a) In the Service Simulator, type 'open Space Geek' and select "Ask My Fact Skill".

Service Simulator

Use Service Simulator to test your lambda function.

Text

Json

Enter Utterance *

open Space Geek

Ask My Fact Skill

Reset

- b) You should see the formatted JSON request from the Alexa Service and the response coming back. Verify that you get a correct Lambda response, and notice the card output. You will want to customize this output later.

Service Simulator

Use Service Simulator to test your lambda function.

Text

Json

Enter Utterance *

open Space Geek

Ask My Fact Skill

Reset

Lambda Request

```
1 {
2   "session": {
3     "sessionId": "SessionId.2ebc7cf2-9dfe-47a3-81
4     "application": {
5       "applicationId": "amzn1.echo-sdk-ams.app.df
6     },
7     "user": {
8       "userId": "amzn1.ask.account.AFP3ZWPOS2BGJR
9     },
10    "new": true
11  },
12  "request": {
13    "type": "IntentRequest",
14    "requestId": "EdwRequestId.b78134bf-fe25-455e
15    "timestamp": "2016-03-25T04:48:39Z",
16    "intent": {
17      "name": "GetNewFactIntent",
18      "slots": {}
19    },
20  },
21  "version": "1.0"
22 }
```

Lambda Response

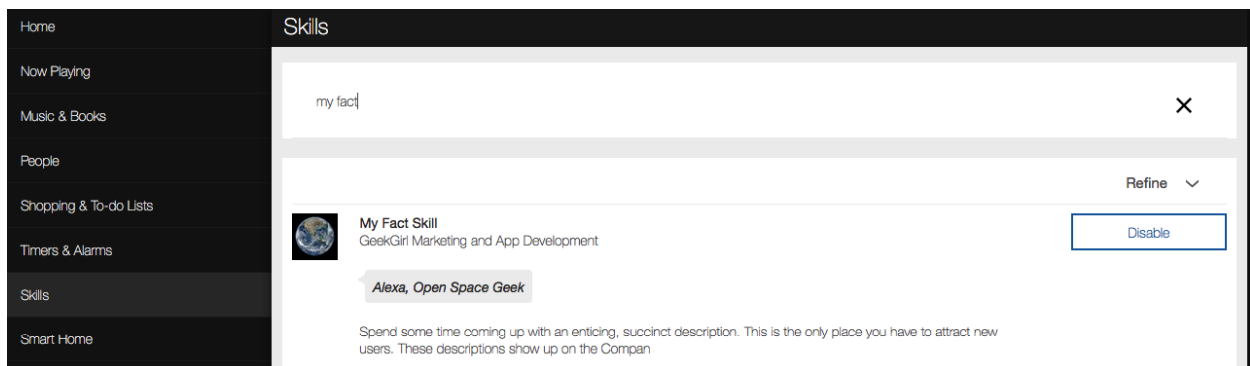
```
1 {
2   "version": "1.0",
3   "response": {
4     "outputSpeech": {
5       "type": "PlainText",
6       "text": "Here's your space fact: On Mars, t
7     },
8     "card": {
9       "content": "Here's your space fact: On Mars
10      "title": "SpaceGeek",
11      "type": "Simple"
12    },
13    "shouldEndSession": true
14  },
15  "sessionAttributes": {}
16 }
```

Listen

Submit for Certification

Next

- c) (Optional) Testing with your device. This is optional as you can do all the testing in the portal. Assuming your Echo device is on-line (and logged in with the same account as your developer account), you should now see your skill enabled in the Alexa app and ask Alexa to launch your skill. For more information on testing an Alexa skill and registering an Alexa-enabled device, [check here](#).



* Not working (getting an invalid response)?

- Do you have the right ARN copied from your Lambda function into your Developer Portal / Skill?
- Are you calling the right invocation name?
 - Are you saying launch, start or open?
 - Are you sure you have no other skills in your accounts with the same invocation name?
- For this template specifically, you should have a minimum of 20 facts for a good customer experience.

Step #4 – Make it Yours

1. In the Skill Information section in the Developer Console, edit the Skill Information Tab to reflect your new Fact Skill:
 - b. Provide a skill name that represents the new skill you are creating.
 - c. Come up with a cool ***Invocation Name that users will use to invoke your skill***
 - d. Create a fun ***icon***. Be sure you have the rights to whatever icons you are uploading – you will need to provide both 108x108px and 512x512px images. Need help finding an image, see [PixelBay](#) as a possible source for royalty-free images. Use an image editor (such as Paint on Windows or Preview on Mac) to change the size of the image.
 - e. Everything else can stay as-is for now in the Developer Portal
2. Open the source file for your Lambda function, index.js, in an editor of your choice. The zip file was downloaded in an earlier step or you can find it [here](#). You will see on line 25 the definition of the facts used in the SpaceGeek example. These are the strings you will want to edit to customize this fact for your use.

```

25
26 ▾ /**
27  * Array containing space facts.
28  */
29 ▾ var SPACE_FACTS = [
30     "A year on Mercury is just 88 days long.",
31     "Despite being farther from the Sun, Venus experiences
32     higher temperatures than Mercury.",
33     "Venus rotates counter-clockwise, possibly because of a
34     collision in the past with an asteroid.",
35     "On Mars, the Sun appears about half the size as it does on
36     Earth.",
37     "Earth is the only planet not named after a god.",
38     "Jupiter has the shortest day of all the planets.",
39     "The Milky Way galaxy will collide with the Andromeda
40     Galaxy in about 5 billion years.",
41     "The Sun contains 99.86% of the mass in the Solar System.",
42     "The Sun is an almost perfect sphere.",
43     "A total solar eclipse can happen once every 1 to 2 years.
44     This makes them a rare event.",
45     "Saturn radiates two and a half times more energy into
46     space than it receives from the sun.",
47     "The temperature inside the Sun can reach 15 million
48     degrees Celsius.",
49     "The Moon is moving approximately 3.8 cm away from our
50     planet every year."
51 ];

```

3. Edit the name of the array. Change the name SPACE_FACTS to something that represents your fact type. Then use “Find” to locate the other instances of SPACE_FACTS in the code and rename them as well.
4. Edit the strings to contain whatever facts or information you would like to make randomly available when a user invokes your skill. A few suggestions:
 - a) Only change the text between the double quotes. These are your facts.
 - b) Ensure you don’t accidentally delete and quotes or commas. You can always go back to GitHub and copy it again if you make a mistake.
 - c) The skill uses a mathematical randomization on your list of facts. It is a good idea to have at least 20, but more like 100, facts in the skill to ensure that the facts do not repeat too quickly. Also remember that because it is random, it is possible that the same fact can be repeated twice.
 - d) For extra credit and completely optional- If you would like to ensure that the facts don’t repeat (for a “Daily Fact Skill” for example), you can use a datastore like DynamoDB to store an id that you can check when the user accesses the skill and iterate through the facts. For more information on using DynamoDB with Lambda, [go here](#).

5. You will also want to make sure to change the “Space Geek” references to the name of your skill. You don’t have to edit them all, but the following reference changes are required for certification.

- a) Find this code in the HelpIntent, and change Space Geek to your custom fact name:

```
"AMAZON.HelpIntent": function (intent, session, response) {  
    response.ask("You can ask Space Geek tell me a  
space fact, or, you can say exit... What can I help you  
with?", "What can I help you with?");
```

- b) Find the HandleNewFactRequest function, and change the highlighted references to your custom fact name. By now, the SPACE_FACTS array should have already been renamed:

```
function handleNewFactRequest(response) {  
  
    // Get a random space fact from the space facts list  
  
    var factIndex = Math.floor(Math.random() *  
SPACE_FACTS.length);  
  
    var fact = SPACE_FACTS[factIndex];  
  
  
    // Create speech output  
  
    var speechOutput = "Here's your space fact: " + fact;  
  
  
    response.tellWithCard(speechOutput, "SpaceGeek",  
speechOutput);  
  
}
```


- c) **Optionally**, you can change the console log output to have the custom name as well. Look for the following code, and change SpaceGeek to your custom fact name.

```
console.log("SpaceGeek
```

6. In order to control who accesses your web service, we should validate the application id in requests made to your web service. Let’s go back to your Alexa skill in your






Developer Portal for a moment. **Copy** in your Application ID from the ‘Skill Information’ section in your developer portal / skill and copy it into your Lambda function.

[< Back to the list of skills](#)

**MyFactSkill-SpaceGeek template**
DEVELOPMENT
Version 1.0 | 3/5/16

[Getting started](#)

*Fields required for certification

Skill Information 	Application Id The ID for this skill amzn1.echo-sdk-ams.app.8b7e6904-f4f1-4715-b2df-a8c0a9583acd
Interaction Model 	Name * The name of this skill. This is the name displayed in the Alexa App. MyFactSkill-SpaceGeek template
Test 	Invocation Name * The name users will say to interact with this skill. This does not have to be the same as the skill name. The invocation name must comply with the Invocation Name Guidelines spacegeek
Description 	Version The serial number of the skill e.g. 1.0, 1.1 1.0
Publishing Information 	

15. In your AWS lambda function, in the index.js file, find the applicationId section and copy **YOUR** Application ID into the section indicated, notice the quotes. (*do not copy the ID below, which is just an example*). It looks something like this when you are done: (*be sure to SAVE*)

```
// Route the incoming request based on type (LaunchRequest, IntentRequest,
// etc.) The JSON body of the request is provided in the event parameter.

exports.handler = function (event, context) {

    try {

        console.log("event.session.application.applicationId=" +
event.session.application.applicationId);

        /**

         * Uncomment this if statement and populate with your skill's
         application ID to

         * prevent someone else from configuring a skill that sends
         requests to this function.

         */

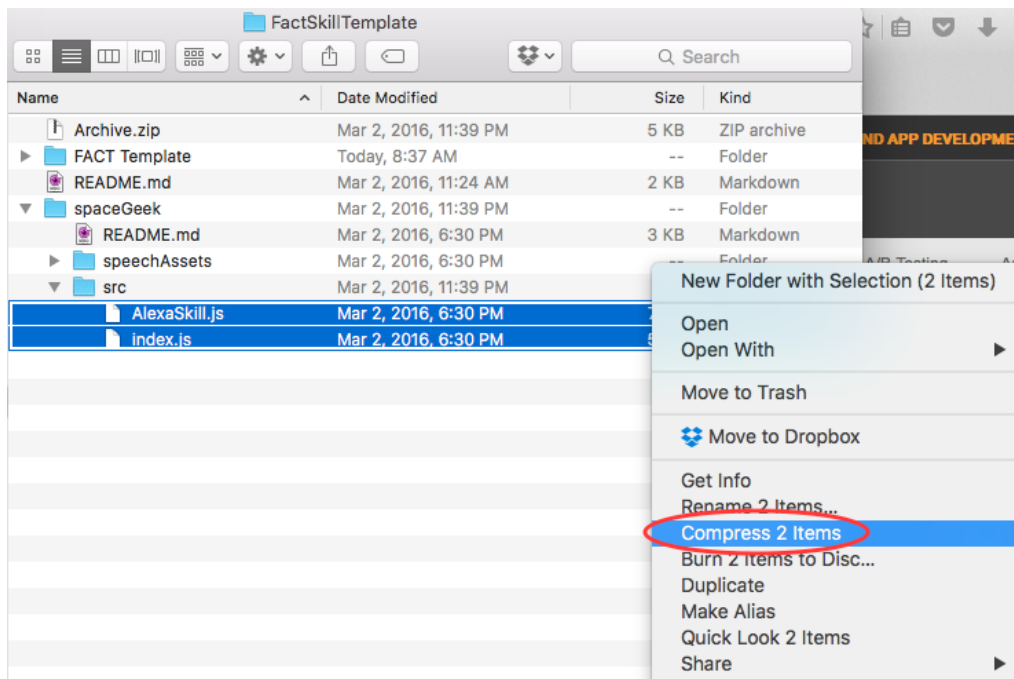
        if (event.session.application.applicationId !== "amzn1.echo-sdk-
ams.app.8b7e6904-f4f1-4715-b2df-a8c0a9583acd") {
```



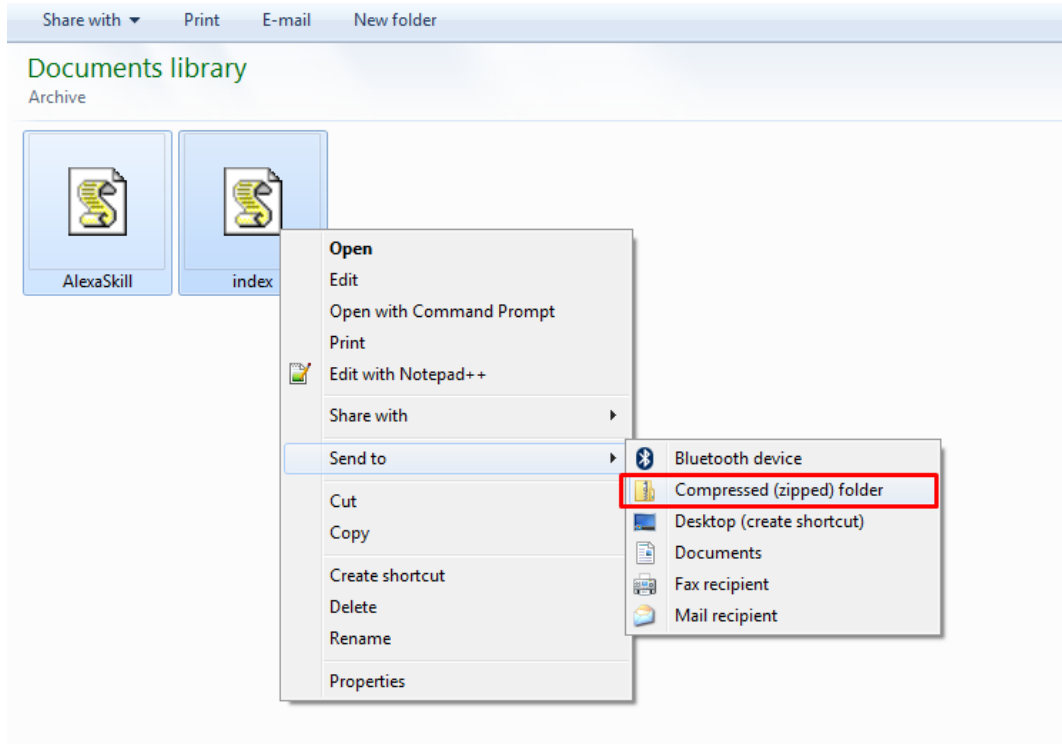
```
context.fail("Invalid Application ID");  
}
```

16. A minimum of 20 facts is needed to get started, about 100 is a good number to keep users engaged. The more the better.
17. ***Be sure to select SAVE when you are all done. Note: we test in the Developer Portal, not in our Lambda function (AWS).***
18. Log back into your AWS console and upload the changes you have just made. First you will need to zip up the files into a new archive. You can do this by selecting the 2 files that you need (AlexaSkill.js and your updated index.js) into a new archive. Be sure that those are the only 2 files in your archive file. Here is an example of what this looks like on a Mac, but the context menu on Windows also provides a similar option for creating a compressed file.

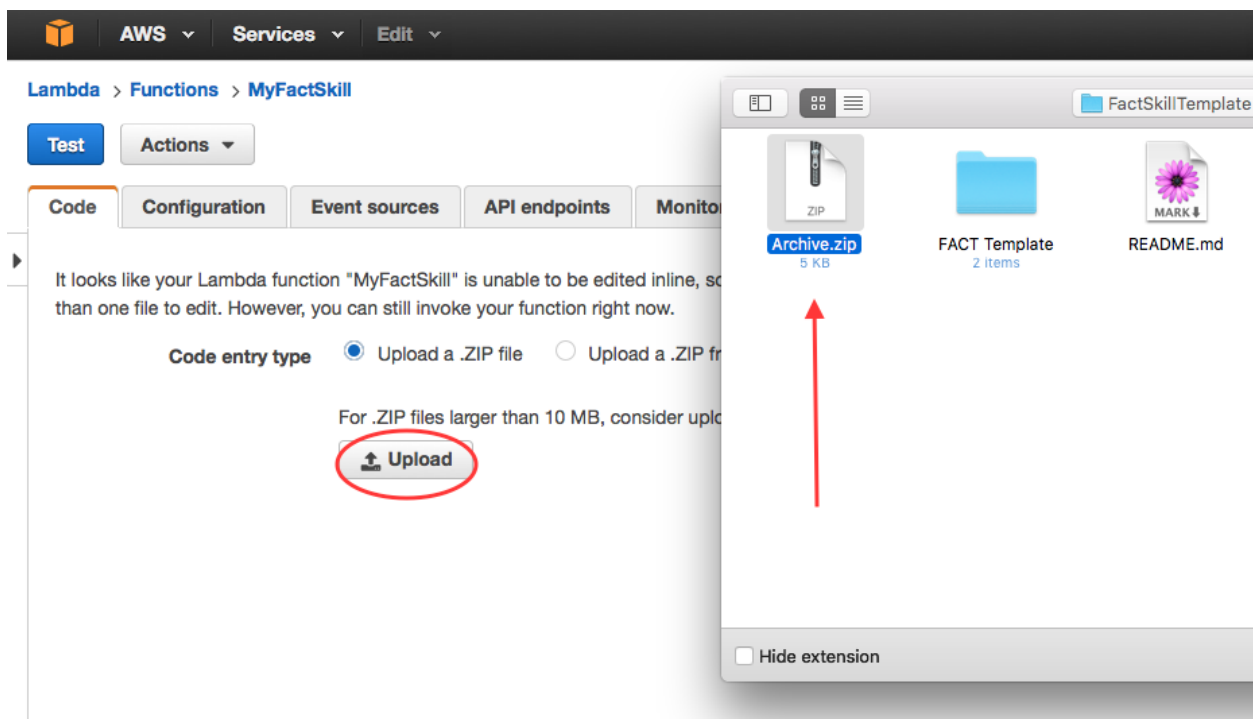
On a Mac:



On Windows:



19. Select your Lambda function and on the Code tab, select “Upload” to add the archive you just created.



20. Once you have successfully added the file you will see it on the screen, then select “Save”:

Lambda > Functions > MyFactSkill

Save **Save and test** Actions ▾

Code Configuration Event sources API endpoints Monitoring

It looks like your Lambda function "MyFactSkill" is unable to be edited inline, so you need to re-upload a file to edit. However, you can still invoke your function right now.

Code entry type ☒ Upload a .ZIP file ☐ Upload a .ZIP from Amazon S3

For .ZIP files larger than 10 MB, consider uploading via S3.

Upload Archive.zip **Successfully selected.**

21. Repeat the tests you performed earlier to ensure your changes are functioning properly. See step 12 for a review of how to perform functional tests. We will not be submitting to certification until we have customized this skill.

Step #5 – Publish Your Skill

Now we need to go back to our Developer Portal to test and edit our skill and we will be ready for certification.

1. In your skills Test section, enter your Utterances into the Simulator to make sure everything is working with your new facts.
2. Optionally, you can test with your Alexa-enabled device to make sure everything is working correctly. You may find a few words that need to be changed for a better user experience.

Some things to think about:

* Does every fact sound correct? Do you need to change any words to make them sound correct? Because we are randomizing our facts, this could take a while. Instead, you can use the Voice Simulator in the Test section to simulate Alexa's responses. In the Voice Simulator, type in each fact that you are using to test how Alexa will say it. Use

additional punctuation or possibly SSML if you need to better control how Alexa responds. You can find out more about [SSML here](#).

* Have you added in YOUR ApplicationID as per the previous instruction?

3. Select the Description area of your skill next:

[< Back to the list of skills](#)

?

My Fact Skill
DEVELOPMENT

3/24/16

Skill Information

Interaction Model

Configuration

Test

Publishing Information

Privacy & Compliance

*Fields required for certification

Note: You can preview your skill's detail page in

Short Skill Description *
A quick introductory description, which will be shown in the Alexa App in the main list of skills, along with the first example phrase you enter below. Maximum characters: 160

Full Skill Description *
Explanation of the skill's benefits, what it does, how it works, how the user gets started, and any prerequisites, such as an account with your company or particular hardware. Use a conversational tone and correct grammar and punctuation. This description is shown to users in the Alexa App, on the skill's detail card.

Example Phrases *
Important: Many developers fail certification due to this step so please read carefully. Provide three phrases from your Sample Utterances, with any slots filled in with a valid value. These are displayed on the detail card in the Alexa App and should teach users how to interact with the skill. Include the wake word and your invocation name in the first phrase.

Category *
The general area of functionality of this skill.

Keywords
Search terms used to increase the discoverability of your skill. Use a comma or white space to separate your terms.

- Spend some time coming up with an enticing, succinct description. This is the only place you have to attract new users. These descriptions show up on the list of [skills available](#) in the Alexa app.
- In your example phrases, be sure that the examples you use match the utterances that you created in the Interaction Model section. Remember, there are built-in intents such as help and cancel. You can learn more about [built-in intents here](#). You can also find a list of [supported phrases to begin a conversation here](#).
- Be sure you have the rights to whatever icons you are uploading – you will need to provide both 108x108px and 512x512px images. If there is any question the Amazon certification team will fail your Alexa skill submission.

[< Back to the list of skills](#)



My Fact Skill DEVELOPMENT

3/24/16

*Fields required for certification

[Getting started](#)

Skill Information	✓
Interaction Model	✓
Configuration	✓
Test	✓
Publishing Information	⌵
Privacy & Compliance	⌵

Note: You can preview your skill's detail page in the Alexa App during development. In the Alexa App, click Skills, then find your skill in the list.

Short Skill Description *

A quick introductory description, which will be shown in the Alexa App in the main list of skills, along with the first example phrase you enter below. Maximum characters: 160

Spend some time coming up with an enticing, succinct description. This is the only place you have to attract new users. These descriptions show up on the Compan

Full Skill Description *

Explanation of the skill's benefits, what it does, how it works, how the user gets started, and any prerequisites, such as an account with your company or particular hardware. Use a conversational tone and correct grammar and punctuation. This description is shown to users in the Alexa App, on the skill's detail card.

This is where you can help train your users on how to use your skill. You can also provide information on how to contact the developers and connect with your company.

Example Phrases *

Important: Many developers fail certification due to this step so please read carefully. Provide three phrases from your Sample Utterances, with any slots filled in with a valid value. These are displayed on the detail card in the Alexa App and should teach users how to interact with the skill. Include the wake word and your invocation name in the first phrase.

Alexa, Open Space Geek

Alexa, Launch Space Geek and tell me a fact

Alexa, Start Space Geek and give me a fact

Category *

The general area of functionality of this skill.

Education

Keywords

Search terms used to increase the discoverability of your skill. Use a comma or white space to separate your terms.

facts, education, learning

- Once you have uploaded your icons, you should see a success message at the bottom of the screen.
- IMPORTANT:** Add the text “This is based on the Fact Skill Template” to the Testing Instructions section. This alerts the Certification team of your submission using this standardized template, smoothing the road to a faster publish. Finally, select **Next**.

[< Back to the list of skills](#)



My Fact Skill DEVELOPMENT

3/24/16

*Fields required for certification

[Getting started](#)

Skill Information	✓
Interaction Model	✓
Configuration	✓
Test	✓
Publishing Information	⊖
Privacy & Compliance	⊖

Note: You can preview your skill's detail page in the Alexa App during development. In the Alexa App, click Skills, then find your skill in the list.

Short Skill Description *

A quick introductory description, which will be shown in the Alexa App in the main list of skills, along with the first example phrase you enter below. Maximum characters: 160

Spend some time coming up with an enticing, succinct description. This is the only place you have to attract new users. These descriptions show up on the Companion

Full Skill Description *

Explanation of the skill's benefits, what it does, how it works, how the user gets started, and any prerequisites, such as an account with your company or particular hardware. Use a conversational tone and correct grammar and punctuation. This description is shown to users in the Alexa App, on the skill's detail card.

This is where you can help train your users on how to use your skill. You can also provide information on how to contact the developers and connect with your company.

Example Phrases *

Important: Many developers fail certification due to this step so please read carefully. Provide three phrases from your Sample Utterances, with any slots filled in with a valid value. These are displayed on the detail card in the Alexa App and should teach users how to interact with the skill. Include the wake word and your invocation name in the first phrase.

Alexa, Open Space Geek

Alexa, Launch Space Geek and tell me a fact

Alexa, Start Space Geek and give me a fact

Category *

The general area of functionality of this skill.

Education

Keywords

Search terms used to increase the discoverability of your skill. Use a comma or white space to separate your terms.

facts, education, learning

Images

Small Icon *

108 x 108px PNG(with transparency) or JPG. This is displayed in the Alexa App.



Large Icon *

512 x 512px PNG (with transparency) or JPG. This is used when displaying the skill in the Alexa App on larger screens.



Testing Instructions *

Please detail any special instructions to test your skill. Include any account or hardware requirements. This information is used by the certification team and is not displayed to customers.

This is based on the Fact Skill Template.

Save

Submit for Certification

Next

Image uploaded successfully.

6. Privacy and Compliance.

- On the Privacy and Compliance section, select 'No' for spending real money and collecting personal information. Privacy and Terms URL's are optional. Choose to certify that your skill can be imported to and exported from the US.

[< Back to the list of skills](#)

[Getting started](#)



My Fact Skill
DEVELOPMENT

3/24/16

*Fields required for certification

Skill Information	✓
Interaction Model	✓
Configuration	✓
Test	✓
Publishing Information	✓
Privacy & Compliance	Ⓜ

Privacy

Does this skill allow users to make purchases or spend real money? *

☐ Yes ☒ No

Does this Alexa skill collect users' personal information? *

This includes anything that can identify the user: name, email, password, phone number, date of birth, etc.

☐ Yes ☒ No

Privacy Policy URL

Link to the Privacy Policy for this skill.

Terms of Use URL

Link to the Terms of Use for this skill.

Compliance

Export Compliance *

☐ I certify that this Alexa skill may be imported to and exported from the United States and all other countries and regions in which we operate our program or in which you've authorized sales to end users (without the need for us to obtain any license or clearance or take any other action) and is in full compliance with all applicable laws and regulations governing imports and exports, including those applicable to software that makes use of encryption technology.

Save

Submit for Certification

b. Select "Save".

[< Back to the list of skills](#)

[Getting started](#)



My Fact Skill
DEVELOPMENT

3/24/16

*Fields required for certification

Skill Information	✓
Interaction Model	✓
Configuration	✓
Test	✓
Publishing Information	✓
Privacy & Compliance	Ⓜ

Privacy

Does this skill allow users to make purchases or spend real money? *

☐ Yes ☒ No

Does this Alexa skill collect users' personal information? *

This includes anything that can identify the user: name, email, password, phone number, date of birth, etc.

☐ Yes ☒ No

Privacy Policy URL

Link to the Privacy Policy for this skill.

Terms of Use URL

Link to the Terms of Use for this skill.

Compliance

Export Compliance *


☒ I certify that this Alexa skill may be imported to and exported from the United States and all other countries and regions in which we operate our program or in which you've authorized sales to end users (without the need for us to obtain any license or clearance or take any other action) and is in full compliance with all applicable laws and regulations governing imports and exports, including those applicable to software that makes use of encryption technology.

Save

Submit for Certification

c. Select "Submit for Certification"

[< Back to the list of skills](#) [Getting started](#)



My Fact Skill
 DEVELOPMENT
 3/24/16

*Fields required for certification

Skill Information ✓
Interaction Model ✓
Configuration ✓
Test ✓
Publishing Information ✓
Privacy & Compliance ✓

Does this skill allow users to make purchases or spend real money? ☐ Yes ☒ No

Does this Alexa skill collect users' personal information? ☐ Yes ☒ No
This includes anything that can identify the user: name, email, password, phone number, date of birth, etc.

Privacy Policy URL

Terms of Use URL

that this Alexa skill may be imported to and exported from the United States, other countries and regions in which we operate our program or in which you've made sales to end users (without the need for us to obtain any license or approval or take any other action) and is in full compliance with all applicable laws and regulations governing imports and exports, including those applicable to software that makes use of encryption technology.

Save Submit for Certification


Successfully updated the publishing information.

All 6 checkmarks are green

You have successfully updated the publishing information

d. Finally, confirm your submission. Select “Yes” to submit your skill.

[< Back to the list of skills](#) [Getting started](#)



My Fact Skill
 DEVELOPMENT
 3/24/16

*Fields required for certification

Skill Information ✓
Interaction Model ✓
Configuration ✓
Test ✓
Publishing Information ✓
Privacy & Compliance ✓

Privacy

Does this skill allow users to make purchases or spend real money? ☐ Yes ☒ No

Does this Alexa skill collect users' personal information? ☐ Yes ☒ No
This includes anything that can identify the user: name, email, password, phone number, date of birth, etc.

Privacy Policy URL

Terms of Use URL

Compliance

Export Compliance *

that makes use of encryption technology.

Save Submit for Certification

Successfully updated the publishing information.

Submit for certification

Are you sure you want to submit this skill for certification? You cannot edit the skill while it is under review.

No Yes

to and exported from the United States we operate our program or in which you've made sales to end users (without the need for us to obtain any license or approval or take any other action) and is in full compliance with all applicable laws and regulations governing imports and exports, including those applicable to software that makes use of encryption technology.

Congratulations! You have published a new skill. You will receive progress e-mails and possibly other suggestions from the team on how you can make your skill even better. You can update your skills at any time.

Get Started Today

Check out these Alexa developer resources:

[Alexa Skills Kit \(ASK\)](#)

[Alexa Developer Forums](#)

[Knowledge Base](#)

[Intro to Alexa Skills Kit - On Demand Webinar](#)

[Voice Design 101 - On Demand Webinar](#)

[Developer Office Hours](#)