

New Alexa Skills Kit Template: Build a Trivia Skill in under an Hour

This zip file contains a new trivia skill template that makes it easy for developers or non-developers to create a quiz skill for Alexa. 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 trivia idea, plug in your questions and edit a couple lines of script (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 document will walk first-time Alexa skills developers through all the required the steps involved in creating a skill using this trivia skill template, called 'Reindeer Games.' It 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 the entire process by doing it, and end up with a published Alexa skill. The document includes instructions on how to customize the skill and submit for certification. Remember, the more questions you add, the more compelling your experience will be for customers. We recommend a minimum of 20, but at least 100 is best for user engagement.

Let's Get Started

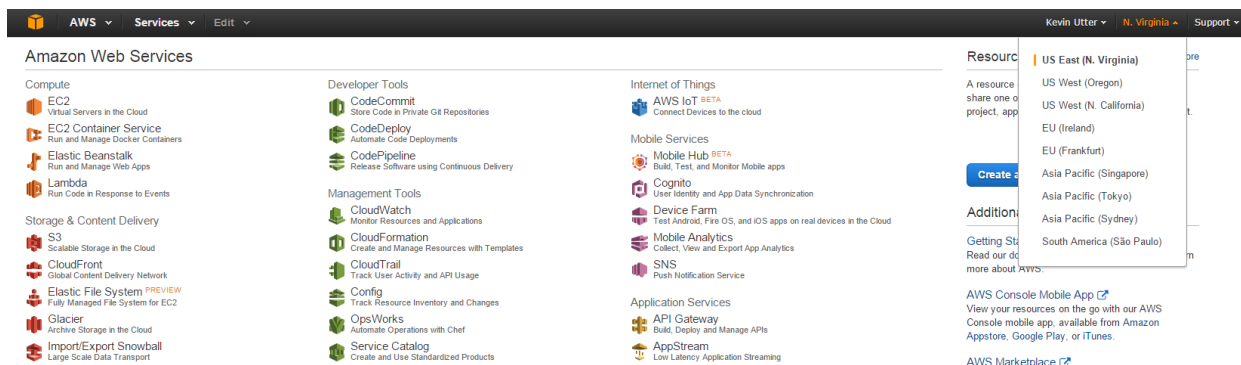
First [download the script](#) and then follow the instructions below. Be sure you have the template trivia skill set up properly before you move on to adapting it to your set of questions.

Step #1 – Create an AWS account

1. Open [aws.amazon.com](#) and then choose 'Create an Free AWS 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 this is a free tier however)
 - 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
 - a. It can sometimes take while 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

1. Select **US East (N. Virginia)** region (upper right)
2. Select Lambda from Compute services (upper left)



3. Skip 'Select Blueprint'

Select blueprint

Blueprints are sample configurations of event sources and Lambda functions. Choose a blueprint that best aligns with your desired scenario and customize as needed, or skip this step if you want to author a Lambda function and configure an event source separately. Except where otherwise noted, blueprints are licensed under [CC0](#).

Filter: All Languages << < Viewing 1-9 of 27 > >>

s3-get-object-python An Amazon S3 trigger that retrieves metadata for the object that has been updated. python2.7 · s3	dynamodb-process-stream An Amazon DynamoDB trigger that logs the updates made to a table. nodejs · dynamodb	microservice-http-endpoint A simple backend (read/write to DynamoDB) with a RESTful API endpoint using Amazon API Gateway. nodejs · api-gateway
node-exec Demonstrates running an external process using the Node.js <code>child_process</code> module. nodejs	simple-mobile-backend A simple mobile backend (read/write to DynamoDB). nodejs · mobile	kinesis-process-record-python An Amazon Kinesis stream processor that logs the data being published. python2.7 · kinesis
dynamodb-process-stream-p... An Amazon DynamoDB trigger that logs the updates made to a table. python2.7 · dynamodb	twilio-blueprint A simple backend for handling events sent from Twilio. nodejs · twilio · sms · voice · web rtc	lambda-canary Performs a periodic check of the given site, erroring out on test failure. python2.7 · cron · testing

[Cancel](#) [Skip](#)

Select Skip

4. You should be in 'Configure Function'

a. Enter the Name/Description/Runtime for your skill as in the example below:

Configure function

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

Name*

Description

Runtime*

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

Lambda function handler and role

Handler*

Role* Suggested role: Basic execution role

5. Select the 'Code Entry Type' as 'Edit code inline' and copy/paste the Lambda function code node.js script you downloaded at the beginning. Here's the [link](#) again for reference. It should look like this:

Configure function

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

Name* MyTriviaGame

Description My First Trivia Game

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

```
1- /**
2-  * Copyright 2014-2015 Amazon.com, Inc. or its affiliates. All Rights Reserved.
3-  * Licensed under the Apache License, Version 2.0 (the "license"). You may not use this file except in complia
4-  * http://aws.amazon.com/apache2.0/
5-  * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARR
6-  */
7- /**
8-  * This sample shows how to create a simple Trivia skill with a multiple choice format. The skill
9-  * supports 1 player at a time, and does not support cross-session games.
10- */
11-
12- var questions = [
13-   {
14-     "Reindeer have very thick coats, how many hairs per square inch do they have?": [
15-       "13,000",
16-       "1,200",
17-       "5,000",
18-       "700"
19-     ],
20-   },
21-   {
22-     "The 1964 classic Rudolph The Red Nosed Reindeer was filmed in": [
23-       "Japan",
24-       "United States",
25-       "France"
26-     ],
27-   }
28- ];
```

Lambda function handler and role

Handler* index.handler

Role* lambda_basic_execution Suggested role: Basic execution role

6. Set your handler and role as follows:
- Keep Handler as 'index.handler'
 - Add a new role for 'lambda_basic_execution' (note IAM role in next step. Note also if you have already used Lambda you may already have a 'lambda_basic_execution' role created you can use.)

Lambda > Functions > Reindeer_Games

Test **Actions**

Code **Configuration** **Event sources** **API endpoints** **Monitoring**

Runtime NodeJS

Handler index.handler

Role lambda_basic_execution

Description Live Template Skill for Trivia Games Vertical

Advanced settings

7. You will be asked to set up your IAM role if you have not done so.

The screenshot shows the AWS IAM console 'Role Summary' page. At the top, there's a navigation bar with 'AWS', 'Services', and 'Edit' dropdowns. Below the header, a message states 'AWS Lambda requires access to your resources' and explains that AWS Lambda uses an IAM role. A 'Hide Details' link is present. The 'Role Summary' section includes a 'Role Description' of 'Lambda execution role permissions', an 'IAM Role' dropdown set to 'Create a new IAM Role', and a 'Role Name' text input field containing 'lambda_basic_execution'. A 'View Policy Document' link is at the bottom.

8. Keep the Advanced settings as default

a. Select 'Next' and review. You should see something like below. Then 'Create your Function':

The screenshot shows the AWS Lambda console 'Review' page for a new function. The left sidebar shows the 'New function' wizard with 'Step 3: Review' selected. The main content area is titled 'Review' and contains a message about reviewing details. Below this is a 'Lambda function' section with an 'Edit' button. The function details are as follows:

Name	MyTriviaGame
Description	My First Trivia Game
Runtime	NodeJS
Handler	index.handler
Role	lambda_basic_execution
Memory (MB)	128
Timeout	3

At the bottom right, there are three buttons: 'Cancel', 'Previous', and 'Create function'.

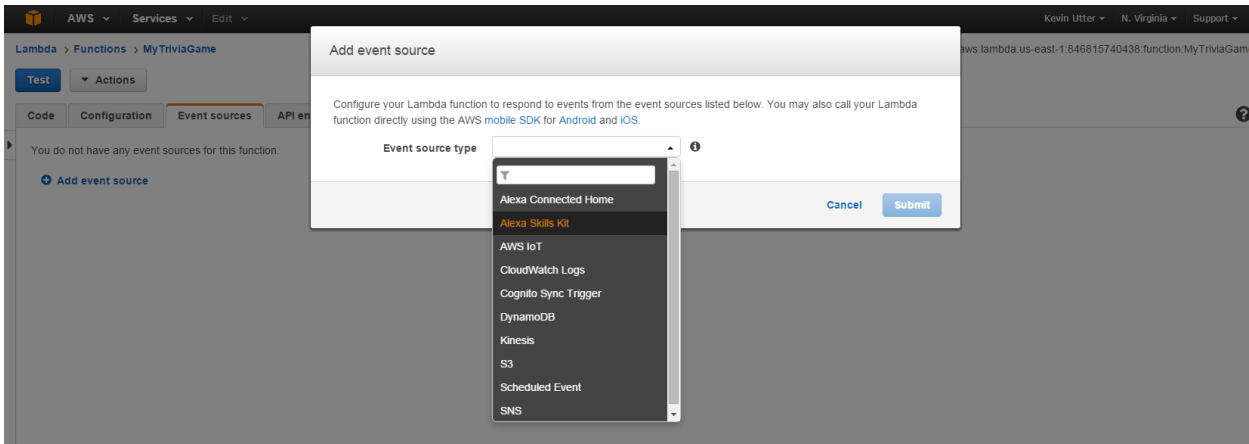
9. Next we need to create an Event source

a. In your Lambda function tabs, select 'Event Source'

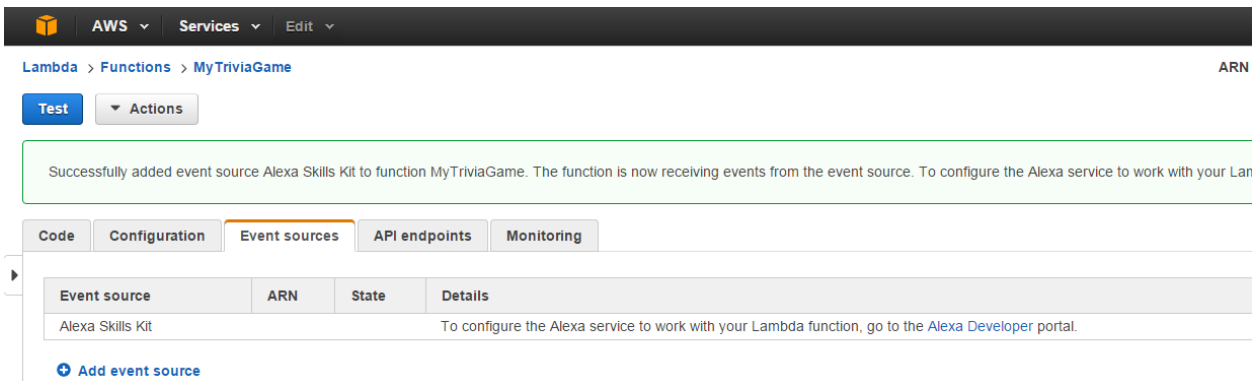
The screenshot shows the AWS Lambda console 'Event sources' tab for a function named 'MyTriviaGame'. The top navigation bar includes 'AWS', 'Services', and 'Edit' dropdowns. The breadcrumb trail is 'Lambda > Functions > MyTriviaGame'. There's a 'Test' button and an 'Actions' dropdown. Below these are tabs for 'Code', 'Configuration', 'Event sources' (which is selected), 'API endpoints', and 'Monitoring'. A message states 'You do not have any event sources for this function.' and there is a '+ Add event source' link.

10. Select 'Add event source'

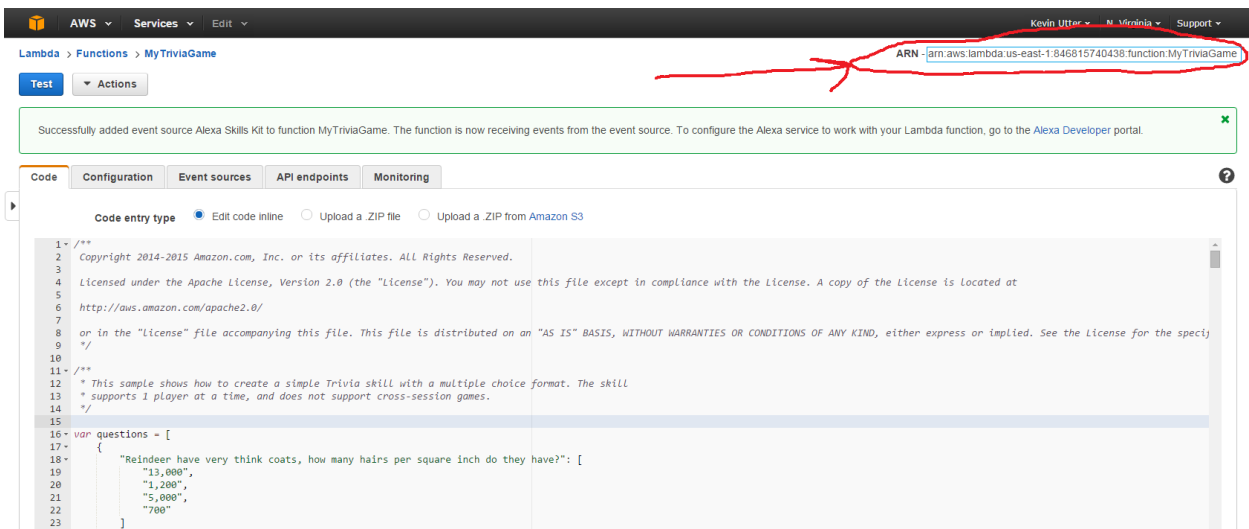
a. Select type as 'Alexa Skill Kit'



11. You should see the following free tier event source:

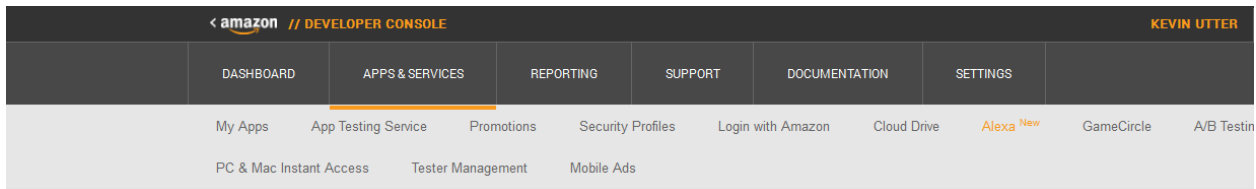


12. Copy the ARN for your Lambda function. You will need it for setting up your skill in the Developer Portal. You can find your ARN here:



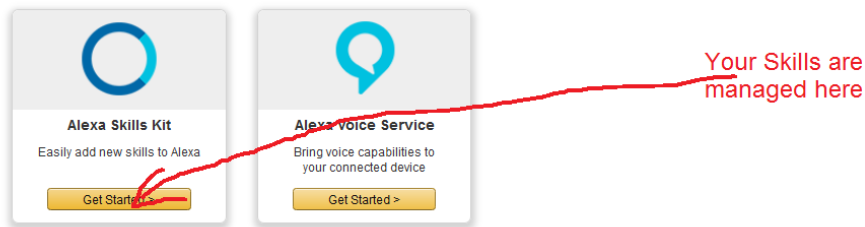
Step #3 – Set up skill in Developer Portal

1. Sign in or create a free account on the [Developer Portal](#) (upper right)
2. Once you've signed in, navigate to Apps & Services/Alexa/Alexa Skills Kit in the post-login experience



Get started with Alexa

Add new voice-enabled capabilities using the Alexa Skills Kit, or add voice-powered experiences to your connected devices with the Alexa Voice Service.



3. This is where your skill will be defined and managed
 - a. Select add new skill and add your name/invocation name/version and your ARN endpoint. Example here:

[Back to the list of skills](#)

[Getting started](#)

Reindeer Games
CERTIFICATION
Version 1.1 | 12/4/15

*Fields required for certification

Skill Information ✓	Application Id The ID for this skill	amzn1.echo-sdk-ams.app.05aebcb3-1461-48fb-a008-8dded1e2b
Interaction Model ✓	Name * The name of this skill. This is the name displayed in the Alexa App.	Reindeer Games
Description ✓	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	reindeer games
Publishing Information ✓	Version The serial number of the skill e.g. 1.0, 1.1	1.1
	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	<input type="radio"/> HTTPS @ Lambda ARN (Amazon Resource Name) ? <u>arn:aws:lambda:us-east-1:73.51086626:function:Reindeer_Games</u>

[Withdraw from Certification](#) ? [Back to the List of Skills](#)

This is an example ARN, be sure to put YOURS here.

- b. Select Save and Next

c. We need to define our skill's interaction model.

i. Copy/paste the following Intent Schema

```
{
  "intents": [
    {
      "intent": "AnswerIntent",
      "slots": [
        {
          "name": "Answer",
          "type": "LIST_OF_ANSWERS"
        }
      ]
    },
    {
      "intent": "AnswerOnlyIntent",
      "slots": [
        {
          "name": "Answer",
          "type": "LIST_OF_ANSWERS"
        }
      ]
    },
    {
      "intent": "AMAZON.StartOverIntent"
    },
    {
      "intent": "AMAZON.RepeatIntent"
    },
    {
      "intent": "AMAZON.HelpIntent"
    },
    {
      "intent": "AMAZON.StopIntent"
    },
    {
      "intent": "AMAZON.CancelIntent"
    }
  ]
}
```

ii. Add Slot Type as below screenshot:

Editing slot type

Enter Type *

LIST_OF_ANSWERS

Enter Values *

Values must be line-separated

1 1
2 2
3 3
4 4

iii. Add the Utterances

1. Copy/paste the following:

AnswerIntent the answer is {Answer}
AnswerIntent my answer is {Answer}
AnswerIntent is it {Answer}
AnswerIntent {Answer} is my answer
AnswerOnlyIntent {Answer}

AMAZON.StartOverIntent start game
AMAZON.StartOverIntent new game
AMAZON.StartOverIntent start
AMAZON.StartOverIntent start new game

3. Select save. You should see the model being built (this make a take a minute or 2)

a. It should now look like this:

Reindeer Games
CERTIFICATION
Version 1.1 | 12/11/15 *Fields required for certification

[Getting started](#)

Skill Information ✓
Interaction Model ✓
Description ✓
Publishing Information ✓

Intent Schema*
The schema of user intents in JSON format.
For more information, see [Defining the Voice Interface for an Alexa skill](#).

```
1 {  
2   "intents": [  
3     {  
4       "intent": "AnswerIntent",  
5       "slots": [  
6         {  
7           "name": "Answer",  
8           "type": "LIST_OF_ANSWERS"  
9         }  
10      ]  
11    },  
12    {  
13      "intent": "AnswerOnlyIntent",  
14      "slots": [  
15        {  
16          "name": "Answer",  
17          "type": "LIST_OF_ANSWERS"  
18        }  
19      ]  
20    }  
21  ]  
22 }
```

Custom Slot Types
Custom slot types to be referenced by the Intent Schema and Sample Utterances
For more information, see [Defining the Voice Interface for an Alexa skill](#).
Example: TOPPINGS - cheese | onions | ham (note: newlines displayed as | for brevity)

Type	Values
LIST_OF_ANSWERS	1 2 3 4

Sample Utterances*
Phrases end users say to interact with this skill. For better results, provide as many samples as you can.
For more information, see [Defining the Voice Interface for an Alexa skill](#).

```
1 AnswerIntent the answer is {Answer}  
2 AnswerIntent my answer is {Answer}  
3 AnswerIntent is it {Answer}  
4 AnswerIntent {Answer} is my answer  
5 AnswerIntent {Answer}  
6  
7 AnswerOnlyIntent the answer is {Answer}  
8 AnswerOnlyIntent my answer is {Answer}  
9 AnswerOnlyIntent is it {Answer}  
10 AnswerOnlyIntent {Answer} is my answer  
11 AnswerOnlyIntent {Answer}  
12  
13 AMAZON.StartOverIntent start game  
14 AMAZON.StartOverIntent new game  
15 AMAZON.StartOverIntent start  
16 AMAZON.StartOverIntent start new game
```

b. Select Next

4. We are ready to test!

- d. In the Test tab, we are going to enter a sample utterance in the service simulator tab.
 - i. In this example, we have called the skill 'reindeer games.' This is the 'Invocation Name' we set up on the Skill Information line in step #2.
 - ii. Enter 'open reindeer games' and select Ask.
 - iii. You should see the formatted JSON request from the Alexa Service and the response coming back.
 - iv. 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 Companion app and ask Alexa to launch your skill!

Interaction Model ☒

Test ☒

Description ☐

Publishing Information ☐

Enable ☐ This skill is enabled for testing on this 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 kevin trivia

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>

Service Simulator

Use Service Simulator to test your lambda function.

Text ☒ Json ☐

Enter Utterance *

open reindeer games

Ask ReindeerTrivaGame Reset

Lambda Request

```
1 {
2   "session": {
3     "sessionId": "a64e72da-0f6a-42b5-a
4     "applicationId": "amzn1.echo-sdk-ams.app.d
5     "applicationId": "amzn1.echo-sdk-ams.app.d
6   },
7   "user": {
8     "userId": "amzn1.account.AF4EJZW7M5O5IDBM8
9     "new": true
10  },
11  "request": {
12    "type": "LaunchRequest",
13    "requestId": "EdwRequestId.a7f9c1b0-ea92-4c7
14    "timestamp": 1448998687920
15  }
16 }
17 }
```

Lambda Response

```
1 {
2   "version": "1.0",
3   "response": {
4     "outputSpeech": {
5       "type": "PlainText",
6       "text": "Reindeer Trivia. Lets start a r
7     },
8     "card": {
9       "type": "Simple",
10      "content": "Reindeer Trivia. Lets start
11      "title": "Reindeer Trivia"
12    },
13    "prompt": {
14      "type": "PlainText",
15      "text": "Question 1. Who was the voice
16    },
17    "shouldEndSession": false
18  },
19  "shouldEndSession": false
20 }
21 }
```

Submit for Certification Next

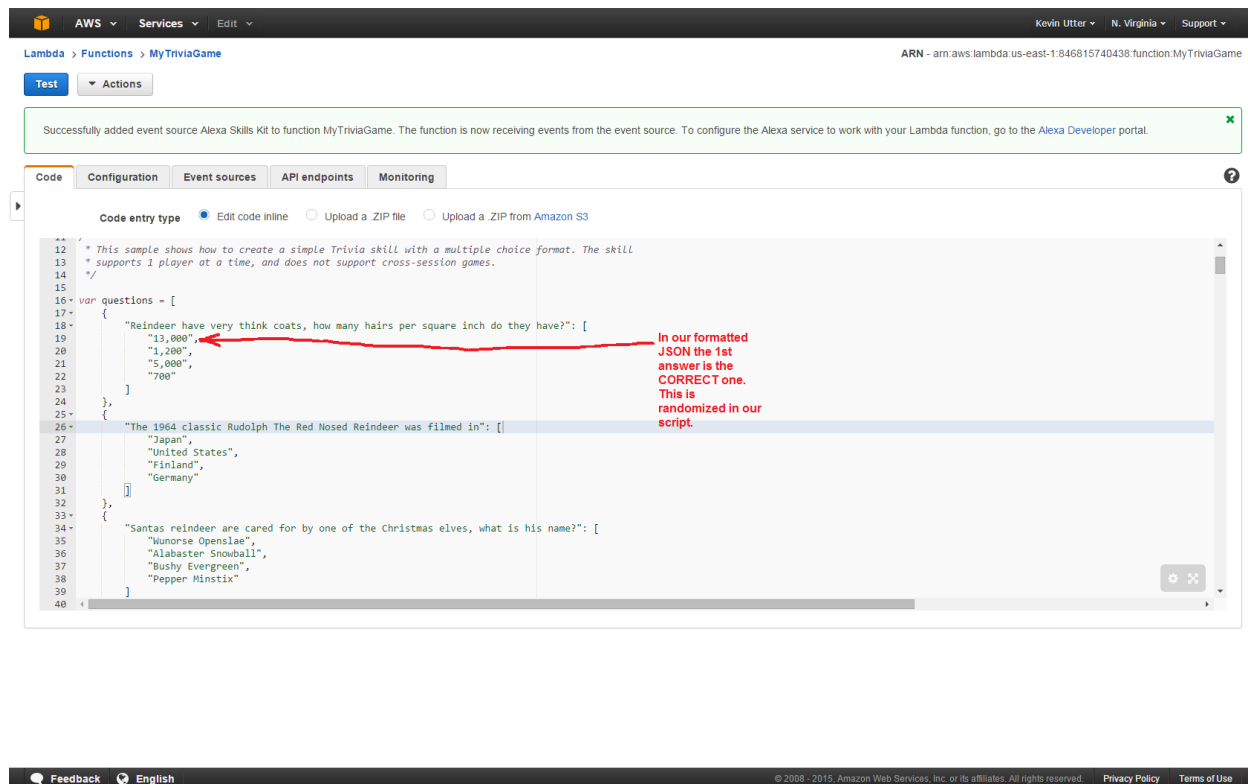
FAQs Contact Us App Distribution Agreement Trademark Guidelines Terms of Use Job Opportunities

* Not working (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 need to have a minimum of 7 questions/answers for the business logic to function.

Step #4 – Make it Yours

1. Edit the Skill Information to reflect your new trivia Game
 - a. A new **Name**
 - b. A cool **Invocation Name**
 - c. A fun **icon**
 - d. Everything else can stay as-is for now in the Developer Portal
2. Log back into your AWS console and edit the Trivia Game Function you have already created. We are going to copy in your new trivia game questions and answers.
 - a. As we have elected to edit our code in-line, you just need to edit the questions and answers JSON to reflect your trivia game. A few suggestions:
 - v. Note that the format calls for one question and 4 answers for each question. The first answer is the correct answer. The script logic takes care of randomizing the questions and answers for you.



The screenshot shows the AWS Lambda console interface. At the top, there's a navigation bar with 'AWS', 'Services', and 'Edit' tabs. Below that, the breadcrumb trail shows 'Lambda > Functions > MyTriviaGame'. The function's ARN is displayed as 'arn:aws:lambda:us-east-1:846815740438:function:MyTriviaGame'. A green notification banner at the top states: 'Successfully added event source Alexa Skills Kit to function MyTriviaGame. 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](#).' Below the notification, there are tabs for 'Code', 'Configuration', 'Event sources', 'API endpoints', and 'Monitoring'. The 'Code' tab is selected, and the 'Code entry type' is set to 'Edit code inline'. The code editor shows a JavaScript script with two questions and their corresponding answers. A red arrow points to the first answer in the first question, with a note: 'In our formatted JSON the 1st answer is the CORRECT one. This is randomized in our script.'

```
12 * This sample shows how to create a simple Trivia skill with a multiple choice format. The skill
13 * supports 1 player at a time, and does not support cross-session games.
14 */
15
16 var questions = [
17   {
18     "Reindeer have very thick coats, how many hairs per square inch do they have?": [
19       "13,000",
20       "1,200",
21       "5,000",
22       "700"
23     ],
24   },
25   {
26     "The 1964 classic Rudolph The Red Nosed Reindeer was filmed in": [
27       "Japan",
28       "United States",
29       "Finland",
30       "Germany"
31     ],
32   },
33   {
34     "Santas reindeer are cared for by one of the Christmas elves, what is his name?": [
35       "Wunorse Openslae",
36       "Alabaster Snowball",
37       "Bushy Evergreen",
38       "Pepper Minstix"
39     ],
40   }
41 ]
```

3. Back to your Developer Portal / Skill for a moment. Copy in your Application ID from the 'Skill Information' tab in your developer portal / skill into your Lambda script.

The screenshot shows the Amazon Developer Console interface for configuring a skill named 'Reindeer Games'. The 'Skill Information' tab is active, displaying various fields for the skill's configuration. The Application ID is highlighted in blue. The Name field contains 'Reindeer Games', the Invocation Name field contains 'reindeer games', and the Version field contains '1.0'. The Endpoint field is set to 'HTTPS Lambda ARN (Amazon Resource Name)' with the value 'arn:aws:lambda:us-east-1:734610866264:function:Reindeer_Games'. At the bottom of the form, there are three buttons: 'Save', 'Submit for Certification', and 'Next'.

In your AWS lambda function find the applicationId section and copy YOUR Application ID into the section indicated (**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. amzn1.echo-sdk-ams.app.05aebcb3-1461-48fb-a008-8ddccd1e2b516") {
            context.fail("Invalid Application ID");
        }
    }
}
```

4. Be sure to remove any Reindeer questions (kind of goes without saying)

5. A minimum of 20 questions is needed to get started, about 100 is a good number to keep users engaged. The more the better.

6. **Be sure to select SAVE when you are all done. Note we test in the Developer Portal, not in our Lambda function (AWS).**

7. Now we need to go back to our Developer Portal to test and edit our skill and we will be GO for certification.
 - a. In your skills Test tab, enter your Utterances to make sure everything is working with your new questions and answers.
 - b. **Go ahead and test with your Alexa-enabled device to make sure everything sounds right, you may find a few words that need to be changed for a better user experience.**

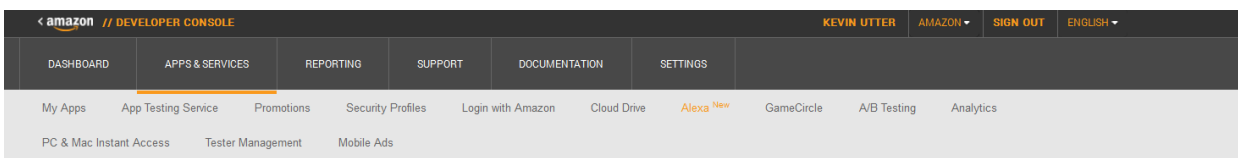
<IMPORTANT>

We recommend actually running your skill on an Echo device and testing every intent and question. We do this as part of the certification process and you will **FAIL certification** if there are any issues.


* Does every question and answer sound correct? Do you need to change any words to make them sound correct?

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

- c. Select the Description tab next
 - i. 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 Apps list of new skills available.
 - ii. 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 certification team will fail your submission.



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



Reindeer Games
CERTIFICATION
Version 1.1 | 12/3/15

[Getting started](#)

Skill Information ✓

Interaction Model ✓

Description ✓

Publishing Information ✓

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

Reindeer Games is a Trivia Game built as a template for developers to use in creating trivia games of their own. It is also fun to play!

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.

Think you can answer?
 "How far do reindeer travel when they migrate?"
 "Santa only has one female reindeer. Which one is it?"

This skill is fun to play alone, or compete with your family and friends to see who is the master of Reindeer trivia!

Example Phrases *

Phrases to teach users how to interact with this skill. The first phrase should be the easiest way to get started with the skill. It will be displayed with the short skill description in the main list of skills in the Alexa App. All three example phrases will be displayed on the skill's detail card. Maximum characters for each phrase: 200. NOTE: For certification, ALL THREE of the example phrases you define here must also be included in the Sample Utterances defined on the Interaction Model page.

Alexa, launch Reindeer Games

Repeat

Help

Put your Invocation Name here. Do not edit the other Example Phrases.

Category *

The general area of functionality of this skill.

Games

Keywords


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

Reindeer, games, trivia

Images

Small Icon *

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



Be sure you have the rights to any images you use.

9. Select Save and SUBMIT FOR CERTIFICATION.
 - a. On your Publishing section select 'No' for Account linking, spend money and personal information. Privacy and Terms URL's are optional.
 - b. Note in your testing instructions that you are using the Trivia Game Template.

c. 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.

More Information

Check out these additional Alexa developer resources:

[Intro to Alexa Skills On Demand](#)

[Voice Design 101 On Demand](#)

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

[Alexa Developer Forums](#)

- Kevin

Want the latest delivered to your inbox? [Subscribe here](#) to stay up to date with the latest Amazon Alexa news and IoT and voice industry trends.