

Exercise 1.1 Creating a Simple Bot with Lex

Rick's preamble

What we are ultimately building

Hi there.

So the goal of the exercises are to create a web based application that can be used to find the answer to the most pressing question in the world.

"Is it too cold for our cats to go outside?"

You all have cats right?

...yeh y'all have cats 🐱

This web application is designed to be 100% serverless. Hence the name of this course "Building serverless Applications".

Throughout these exercises we will leverage multiple AWS services to build out this application.

We will start off building a text based data driven web application, and then enhance it by turning it into a voice based data driven application using Amazon LEX.

Although we are going to focus on building the (non-LEX) text based application first, I would still like to introduce you to Amazon Lex early on for two reasons.

1) Lex is very easy to work with, and super easy to build, so you get to create something cool straight away, instead of waiting until week 3 for your first win.

2) The concepts covered in this first exercise (Creating a simple bot with LEX) will prepare you for enhancing your application later on. This way you won't have to context switch back to LEX fundamentals when we are full steam ahead in week 4. As you will already have the basics under your belt.

Let's jump straight into Exercise 1.1 and create a simple chatbot.

Exercise 1.1: Creating a simple BOT with LEX

The goal of this first lab is to create a simple chatbot in the LEX console that can gather basic information from a user, which we call "eliciting slots".

When we ask the chatbot for the weather it will not have enough information to provide an answer. It will need to further establish the city your cat lives in, and then confirm with you that the question you asked was understood correctly.

For example:

You	Weatherbot
Is it too cold for my cat?	
	What city?
Chicago	
	So you want to know if your cat can go outside today in Chicago, is that correct?
Yes	
	Fulfillment success - [i.e end conversation]

And that's it...for now at least!

We use this chatbot later in week 4 where we will get to enhance it significantly making it much smarter, so it provides the weather for the respective city.

Once built, we will integrate it with our web application that you will build next.

For now, let's just build that basic chatbot, and get the fundamentals out of the way.

Please follow all these steps below one by one in order very carefully.

It's very easy to accidentally miss a step, and it then not work as expected.

1. Steps for creating a Lex Bot

- Sign in to the AWS Management Console and in the **Find Services** search box type lex and choose **Amazon Lex**.
- Make sure you are in the **N. Virginia** region at the top right.
- If this is your first bot, choose **Get Started**; otherwise, on the **Bots** page, choose **Create**.
- Under **Create Your Own** choose **Custom Bot**.
 - Name the Bot: **WeatherCatBot**.
 - For **Output voice**. Leave it as: **None. This is only a text based application.**
 - Change the **Session timeout** to **1** minute.
 - Leave the IAM role as **AWSServiceRoleForLexBots**
 - For **COPPA** choose **No**.

CREATE YOUR OWN

TRY A SAMPLE

Custom bot

BookTrip

OrderFlowers

ScheduleAppointment

Bot name: WeatherCatBot

Language: English (US)

Output voice: None. This is only a text based application.

Session timeout: 1 min

IAM role: AWSServiceRoleForLexBots
Automatically created on your behalf

COPPA: Please indicate if your use of this bot is subject to the Children's Online Privacy Protection Act (COPPA). [Learn more](#)
☐ Yes ☒ No

Cancel Create

- Click **Create**.

Now we have created a bot we need to prepare it for the type of questions we might throw at it.

If we prepare LEX correctly with a good sample set of questions (that are basically saying the same thing) it can use it as "training data".

The idea is that over time, and the more it is used, it can figure out what you are trying to imply. Or what you mean when you talk to it. Pretty cool right?

Even if you ask it a question that is not in the sample questions you supply, it is usually smart enough to figure out your intent anyway.

So lets say you want to know if it is too cold for you cat, there are a gazillion edge case ways that you might phrase that conversationally right?

"Hey silly bot, cool enough for my cat yet?"

"Cat bot, I bet it is too hot for my cat in Arizona, isn't it?"

"Can my cat go outside yet? He's really bored..oh yeah we're in Texas."

...the list goes on.

They call these common question samples "utterances".

We obviously can't type every conceivable utterance, and we don't need to. LEX is pretty smart, and gets smarter over time.

We just need to provide a good sample of commonly phased questions that you think users would say. These are related to the INTENT of finding out if your cat should go out or not. And let LEX over time, figure out all the fancy humanized edge case intents.

Let's do that now.



2. Steps for creating an Intent.

- Click the **Create Intent** button.
- On the Add intent pop-up click **Create intent**.
- Name the Intent. **CatWeather** and click **Add**.
- The box will disappear
- Under **Sample utterances** open it up (there will be an small black arrow on the left to expand).

▼ Sample utterances ⓘ

Can my cat go outside



 NOTE: There is a  symbol at the far right of the text entry area. Either press that after adding text or press **Enter**, before entering the next utterance.

Enter these five utterances **one at a time**, in order:

- 1 Can my cat go outside
- 2 Is it warm enough for my cat
- 3 Can I let my cat out in {city_str}
- 4 Should my cat wear booties in {city_str}
- 5 Will my cat stay dry in {city_str}

We are *not* going to talk about synonyms in the course, but I recommend you check that concept out to make your bot super human. <https://aws.amazon.com/blogs/machine-learning/use-synonyms-and-slot-value-validation-in-your-amazon-lex-chatbots/>

Did you notice the {city_str}? Think of this as a variable, as the user can say any city they want.

This maps to what we call a **SLOT**.

The advantage in putting this in our utterances, is that if a slot is used in the initial user utterance the LEX bot has all it needs. It no longer needs to elicit that slot and ask the user what city, because it has just been volunteered ;).

Are you ready to create the slot that we used as a placeholder as city_str in some of our utterances?

Yeah, you're ready ;)

3. Steps for creating a Slot.

- Scroll down to the slots sections and expand it using the drop down arrow.

▼ Slots ⓘ

Priority	Required	Name	Slot type	Version	Prompt	Settings
		e.g. Location	e.g. AMAZON.US_CITY ▼		e.g. What city?	ⓘ

- Type `city_str` under **Name**.
- For **Slot type** start typing in `AMAZON.U` and choose `AMAZON.US_CITY`.
- Type `Which city?` under **Prompt** and click `+` or hit **Enter**.
- `{city_str}` will now be colored in the **Sample utterances** drop down.

▼ Sample utterances ⓘ

ⓘ

ⓘ

ⓘ

ⓘ

ⓘ

ⓘ

► Lambda initialization and validation ⓘ

▼ Slots ⓘ

Priority	Required	Name	Slot type	Version	Prompt	Settings
		e.g. Location	e.g. AMAZON.US_CITY ▼		e.g. What city?	ⓘ
1.	✓	<code>city_str</code>	AMAZON.US_CITY ▼	Built-in ▼	Which city?	ⓘ ⓘ

Before you save your intent, we want to humanize it a little more by configuring a confirmation step with "So you want to know if your cat can go out today in {city_str}?".

Let's add that now.

- Expand **Confirmation prompt**.
 - Check **Confirmation prompt**.
 - Under **Confirm** type in the following:

1 So you want to know if your cat can go out today in {city_str}?

- Under **Cancel** (if the user says "no", ie. that's wrong) type in:

1 Sorry, can you please repeat your initial question?

- Expand **Fulfillment** and verify **Return parameters to client** is selected.
- Leave Response as it is.
- Click **Save Intent** at the bottom.
- Click **Build** at the top right.
- At the Build your bot pop-up click **Build**.
- This can take a few minutes. *Usually less than 30 seconds, if your lucky.*

Believe it or not, that's it!

Let's test it in the helpful text console.

4. Steps for testing the bot in the Lex console.

- Expand the chatbot on the right. Where it says **Test Chatbot**.
- Try these conversations so you can see how LEX responds to you in different scenarios,

User 🧑	Chatbot 🤖
Will my cat stay dry in Seattle?	<editor note> likely not ;)
	So you want to know if your cat can go out today in Seattle?
Yes	
	Intent CatWeater is ReadyForFulfillment: city_str:Seattle
Can I let my cat out in Arizona?	
	So you want to know if your cat can go out today in Arizona?
Yes	
	Intent CatWeather is ReadyForFulfillment: city:Arizona
Is it warm enough for my cat?	
	Which city?
Denver	
	So you want to know if your cat can go out today in Denver?
Sure	
	Intent CatWeather is ReadyForFulfillment: city:Denver
Is it too cold for my cat?	
	Which city?
Las Vegas	
	So you want to know if your cat can go out today in Las Vegas?
Yes	
	Intent CatWeather is ReadyForFulfillment: city:Las Vegas

NOTE: Notice we didn't include "Is it too cold for my cat?" in the utterances. Yet our chat bot is intelligent enough to prompt us for the city.

I think that is so cool. LEX can actually figure out what you are trying to ask.

Awesome job!

Let's look at our course todo checklist.

Exercise goal checklist

1. ~~Create a simple chatbot using the lex console~~ :tick
2. Upload our website to S3.
3. Create a content delivery network and lock down S3.
4. Build an API gateway mock with CORS.
5. Build a Lambda mock, use IAM, push logs to CloudWatch.
6. Create and seed a database with weather data.
7. Enhance the lambda, so it can query the database.
8. Play with your new text based data driven application.
9. Create a LEX proxy using Lambda.
10. Enhance API gateway to use the LEX proxy.
11. Play with your new voice web application.

Well done. Please head back to the video content so we can discuss what to do next.