

Telegram Bots

An introduction to python-telegram-bot & Azure Cognitive Services

Azeem Vasanwala | Jason Chee | Darren Tang | Samuel Huang



Azeem Vasanwala
Y3 CS



Jason Chee
Y3 CS



Samuel Huang
Y3 CS

Simple Ground Rules

- Question? Use the raise hand function on Zoom, or send your message in the chat
 - Keep the chat on topic, else it will be hard to pick out questions
- Stuck? Join a breakout room. Someone will be with you shortly
- Ask as many questions as you want, but keep it on topic

Relax. Slides are here:

<https://roc.ms/mlsateleslides>

Why Bots?

Key Benefits

- (Mostly) Navigationless interface
 - No menu bars to dig around to find information
 - Try finding the curriculum for an older batch on the SOC website without googling 😊
 - Try to find out what to do as an international student entering NUS now (SHN, student pass, etc) without googling

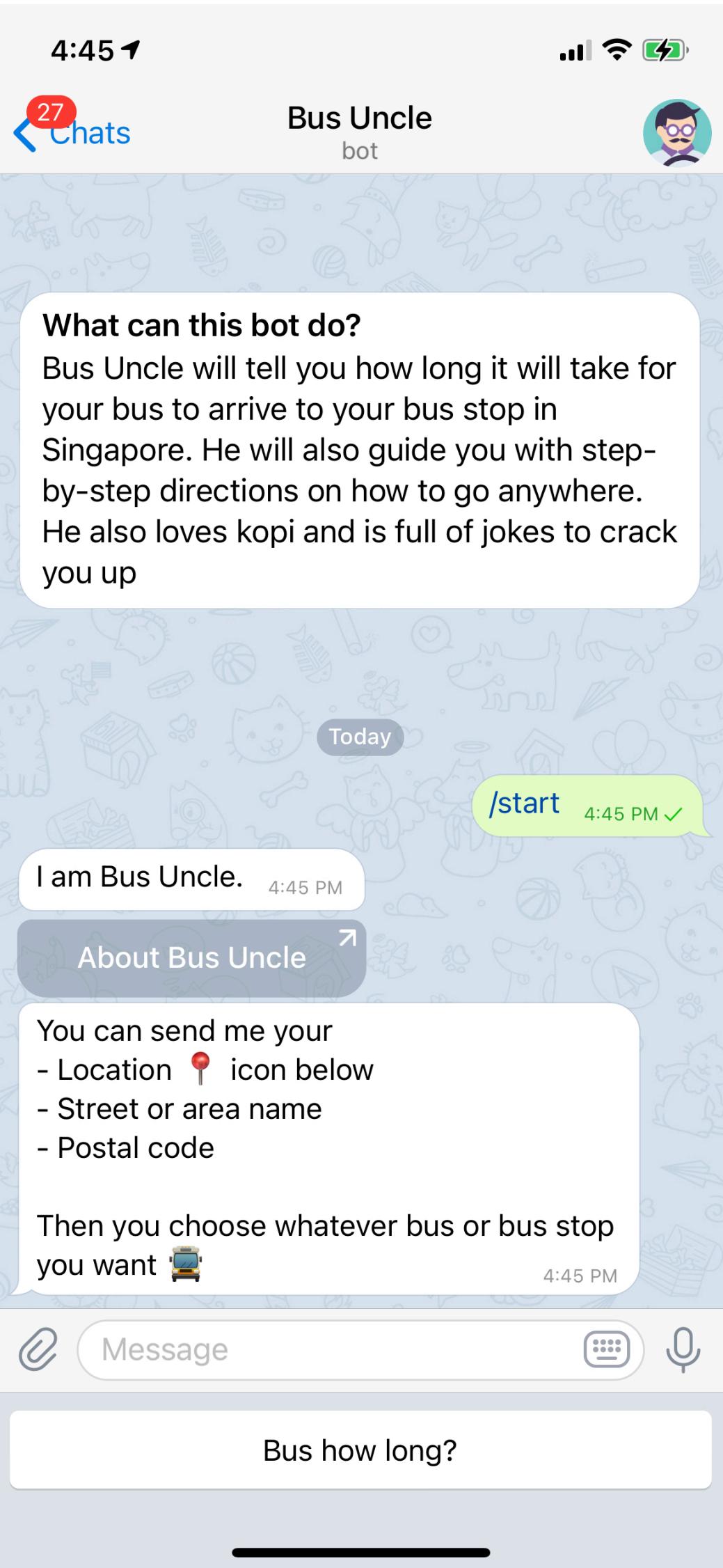
Key Benefits

- Easy Push alerts
 - Just like getting a notification for a text message
 - No complex integrations, device specific compatibility, etc
- Familiar UI
 - We all know how to use messaging apps
- Works on existing apps
 - Low bar of entry for new users
 - No need to download a new app, works with what you already have

Telegram Bots



- Chat messaging platform
 - Similar to WhatsApp, WeChat, etc
 - Cloud based, not P2P
- Bots are natively supported on the platform
- Texting a bot is similar to texting another person



- Chat messaging platform
 - Similar to WhatsApp, WeChat, etc
 - Cloud based, not P2P
- Bots are natively supported on the platform
- Texting a bot is similar to texting another person

So what is a “bot”?

Program that (typically) responds
to your input

What can I do with a bot?

Common Interactions

How do people *usually* use bots?

- Perform manipulation of input from user
 - “Get me the bus timings”
 - “Add these numbers up”
 - Answer a question from a database
 - Let’s build a bot to do this
- Pre-req: You have a telegram account and a gmail account
 - If you don’t, use the zoom raise hand function **NOW**

Our Environment

- Google CoLaboratory
 - Python development environment on Google's servers
 - No autocompletion, very sad 😞
 - Go to <https://roc.ms/mlsacolab>
 - Link in zoom chat
 - Duplicate this notebook: File > Save a copy in Drive
 - Name it whatever you want - This will be the notebook we will be working off

Creating a Bot

- Launch Telegram
- Search for the user @BotFather
- Create a Bot
 - Name it whatever you want
 - Give it whatever handle you want, but make sure it ends in bot
 - E.g azeem_bot
 - A token will be generated, copy that out and save it somewhere

Back to CoLab

Getting started

- Pre-req: Your packages are installed.
- Update the token with your own token
- Run the bot once
- Text the bot and say hello to it, it should echo “hello” back to you

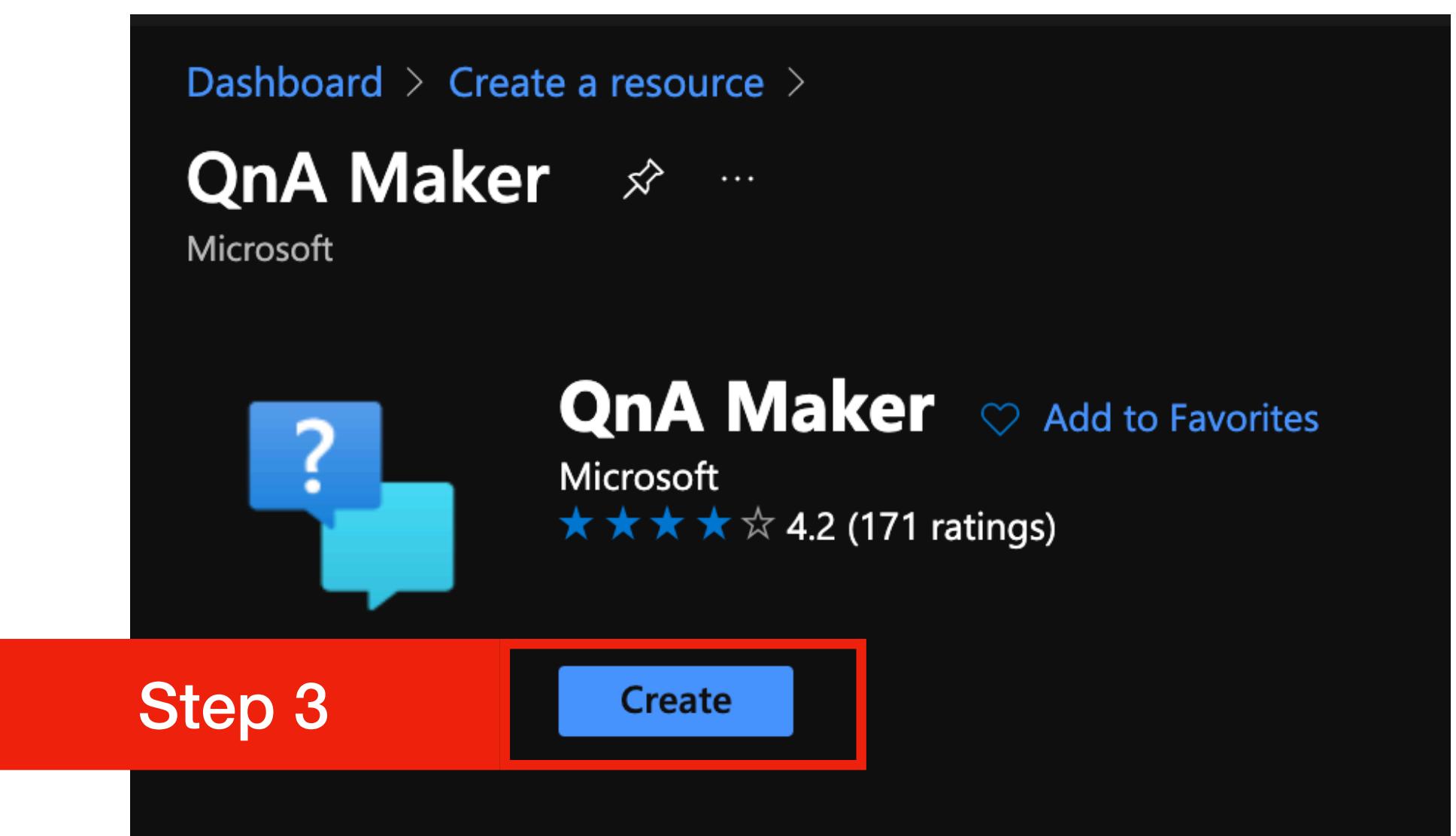
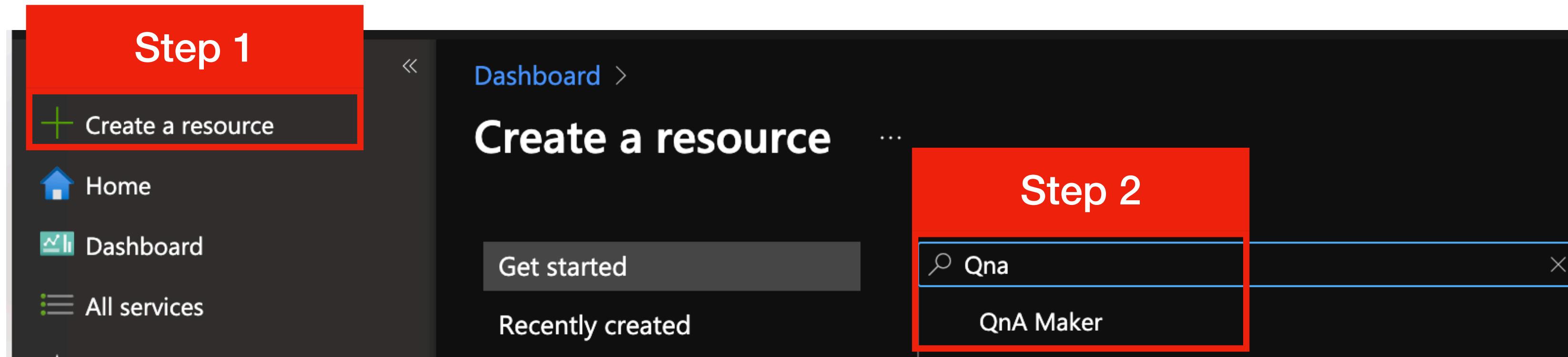
What's going on?

So now... Let's go beyond an echo

- Create a QnA Maker instance on Azure
- Send our user input to the bot to the QnA Maker instance
- Get a response from QnA Maker
- Send that response to the user
- Pre-req: You have created your Azure for Students account
 - If you don't, use the zoom raise hand function **NOW**

Creating a QnA Maker Instance

Step 1: Create the Azure resource: portal.azure.com



Creating a QnA Maker Instance

Step 1: Create the Azure resource: portal.azure.com

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ 6bc-4438-87fd-d03ad8c1258a

Resource group * ⓘ Create new

Name * ⓘ Enter a name

Pricing tier (Learn More) * ⓘ F0

Azure Search details - for data

When you create a QnAMaker resource, you host the data in your own Azure subscription. Azure Search is used to index your data.

Azure Search location * (US) East US

Azure Search pricing tier * ⓘ Free

Step 4

Step 5

Step 6 - F0

App Service details - for runtime

When you create a QnAMaker resource, you host the data in your own Azure subscription. App Service is the compute engine that runs the QnA Maker service.

App name * ⓘ QnAMaker

Website location * (US) East US

Step 9

Step 10 - SEA

Step 11 - Disable

Step 12

Review + create **Next : Tags >**

Creating a QnA Maker Instance

Step 2: Link QnA Maker to the Azure Resource: www.qnamaker.ai

Step 1

Cognitive Services | QnA Maker My knowledge bases Create a knowledge base

STEP 1

Step 2

Create a QnA service in Microsoft Azure.

Create an Azure QnA service for your KB. If you already have an Azure QnA service for this KB, skip this step.

Learn more about Azure QnA service

Stable Preview ?

Create a QnA Service

STEP 2

Connect your QnA service to your KB.

After you create an Azure QnA service, refresh this page and then select your Azure service using the options below

Refresh

* Microsoft Azure Directory ID

azeemirongmail (Default Directory)

* Azure subscription name

DreamSpark

* Azure QnA service

Select service

* Language

English

Step 3
Only 1 option

Step 4
Only 1 option

Step 5
English

Creating a QnA Maker Instance

Step 2: Link QnA Maker to the Azure Resource: www.qnamaker.ai

STEP 3

Name your KB.

The knowledge base name is for your reference and you can change it at anytime.

* Name
Name your knowledge base

Step 6

STEP 4

Populate your KB.

Extract question-and-answer pairs from an online FAQ, product manuals, or other files. Supported formats are .tsv, .pdf, .doc, .docx, .xlsx, containing questions and answers in sequence. [Learn more about knowledge base sources.](#) Skip this step to add questions and answers manually after creation. The number of sources and file size you can add depends on the QnA service SKU you choose. [Learn more about QnA Maker SKUs.](#)

Enable multi-turn extraction from URLs, .pdf or .docx files. [Learn more.](#)

URL

http://

+ Add URL

File name

+ Add file

Step 7
Leave these blank

Creating a QnA Maker Instance

Step 2: Link QnA Maker to the Azure Resource: www.qnamaker.ai

Chit-chat

Give your bot the ability to answer thousands of small-talk questions in a voice that fits your brand. When you add chit-chat to your knowledge base by selecting a personality below, the questions and responses will be automatically added to your knowledge base, and you'll be able to edit them anytime you want. [Learn more about chit-chat.](#)

- None
- Professional
- Friendly
- Witty
- Caring
- Enthusiastic

Step 8
Pick any one

STEP 5

Create your KB

The tool will look through your documents and create a knowledge base for your service. If you are not using an existing document, the tool will create an empty knowledge base table which you can edit.

Create your KB

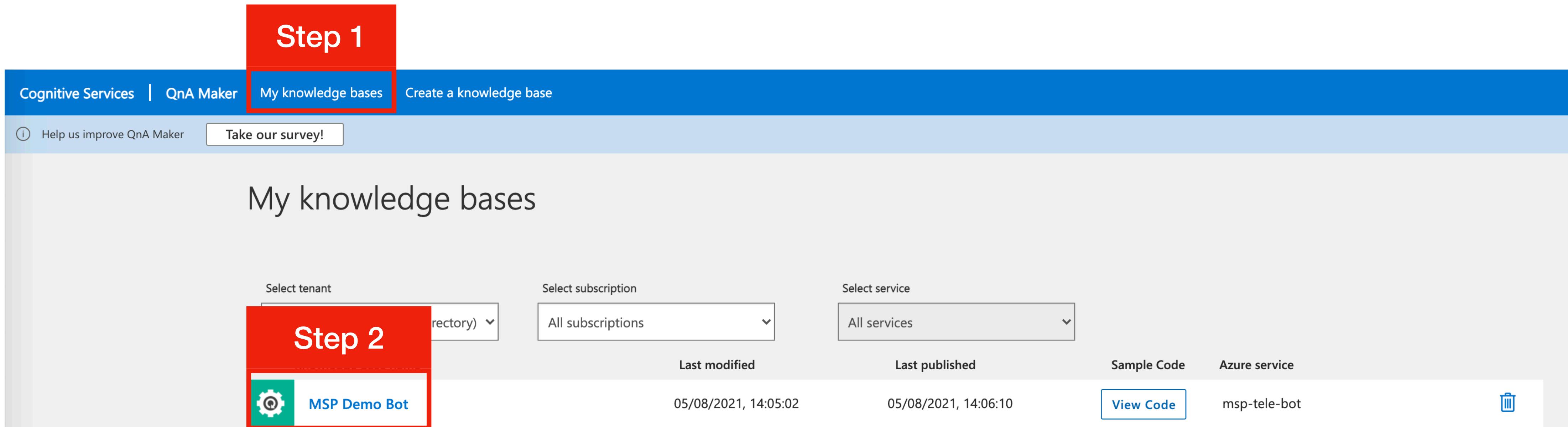
Step 9

Your QnA Maker Instance is ready!



Publishing your knowledgebase

Step 1



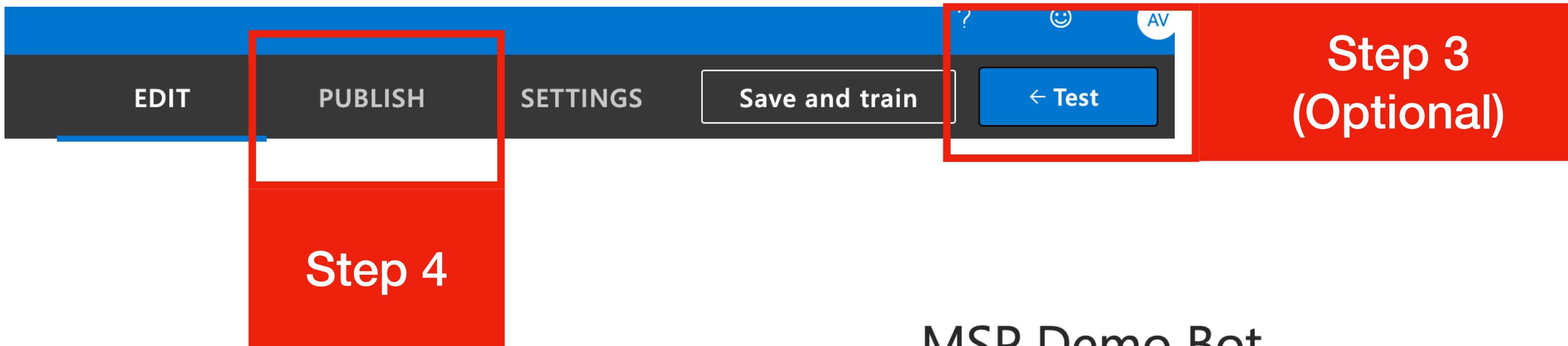
Cognitive Services | QnA Maker **My knowledge bases** Create a knowledge base

(i) Help us improve QnA Maker [Take our survey!](#)

My knowledge bases

Select tenant	Select subscription	Select service	Last modified	Last published	Sample Code	Azure service	
Step 2  MSP Demo Bot	All subscriptions	All services	05/08/2021, 14:05:02	05/08/2021, 14:06:10	View Code	msp-tele-bot	

Publishing your knowledgebase



MSP Demo Bot

Publishing your knowledge base moves your QnAs from the test index to the production index. Once you publish, the knowledge base endpoint becomes available for use in your Bot or App

This knowledge base will be published to the [msp-tele-bot](#) QnA Maker service.



Publishing your knowledgebase

Use the below HTTP request to call your Knowledgebase. [Learn more.](#)

Postman

Curl

```
POST /knowledgebases/2f422103-4f27-45e6-86d8-ae69f0f213a5/generateAnswer 1
Host: https://msp-tele-bot.azurewebsites.net/qnamaker 2
Authorization: EndpointKey 4c444c36-ed46-4815-87eb-35b710428129 3
Content-Type: application/json
{"question":"<Your question>"}
```

Step 6
Save this info

For later:
Url => 1 + 2
Key => 3

Side track: HTTP Verbs

What? Verbs? This isn't English class...

- HTTP: HyperText Transfer Protocol
 - Basically how the internet communicates
- 4 Main Verbs
 - GET [Retrieve]
 - POST [Create]
 - PUT [Update]
 - DELETE [Delete]

Back to colab

**Fix the answer_question
function**

Questions?

Setting up text analysis

Creating a Text Analytics Instance

Create the Azure resource: portal.azure.com

Step 1

[Create a resource](#)

[Home](#)

[Dashboard](#)

[All services](#)

Step 2

text analytics

Step 3

MAQ Software
Text Analytics Playground - Discover Insights
MAQ Software
SaaS
Discover meaningful insights for text feedback generated from various customer channels
Software plan starts at Free
Set up + subscribe

Edge Module – Language Detection (Text Analytics)
Microsoft
IoT Edge Modules
Analyze text on the edge, on-premises and in the cloud using container support.
Create

Text Analytics
Microsoft
Azure Service
An AI service that enables you to unlock insights from natural language text using sentiment
Create

Arabic Speech to Text
Kanari AI
SaaS
Enterprise Focused Automatic Speech Recognition For Dialectal Arabic.
Software plan starts at \$30.00/month
Set up + subscribe

Analytics Zoo
Intel Software
Virtual Machine
A unified analytics + AI platform for distributed TensorFlow, Keras and BigDL on Apache Spark
Bring your own license
Create

Semeon Customer Feedback Analysis Platform
Semeon Analytics
SaaS
Ultra precise and flexible text analytics platform for customer feedback
Software plan starts at \$5,000.00/month
Set up + subscribe

Creating a Text Analytics Instance

Create the Azure resource: portal.azure.com

Step 4: On the **Text Analytics** service page, click the **Create** button.

Step 5: In the **Select additional features** step, review the available features and click **Continue to create your resource**.

Creating a Text Analytics Instance

Create the Azure resource: portal.azure.com

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ Visual Studio Enterprise (528b0e1e-76bc-4438-87fd-d03ad8c1258a) ▾

Resource group * ⓘ Create new ▾

Instance details

Region * ⓘ (US) East US ▾

Name * ⓘ Enter a name

Pricing tier (Learn More) * ⓘ Free F0 (5K Transactions per month) ▾

Responsible AI Notice

Microsoft provides technical documentation regarding the appropriate operation applicable to this Cognitive Service that is made available by Microsoft. Customer acknowledges and agrees that they have reviewed this documentation and will use this service in accordance with it.

[Responsible Use of AI documentation for Text Analytics for Health](#)

[Responsible Use of AI documentation for Text Analytics PII](#)

I certify that I have read and understood the Responsible AI Notice

Step 6
Pick prev one

Step 7
SEA

Step 8 & 9
9 - Free

Step 10

Review + create

Step 11

Creating a Text Analytics Instance

Navigate to the resource

Home > msp-tele-bot > msp-tele-bot-text-analytics

msp-tele-bot-text-analytics | Keys and Endpoint

Text analytics

Search (Cmd+/)

Regenerate Key1 Regenerate Key2

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Resource Management

Quick start

Keys and Endpoint

Pricing tier

Networking

Identity

Billing By Subscription

Properties

KEY 1

KEY 2

Location/Region

southeastasia

Endpoint

<https://msp-tele-bot-text-analytics.cognitiveservices.azure.com/>

Show Keys

Save this

Save this

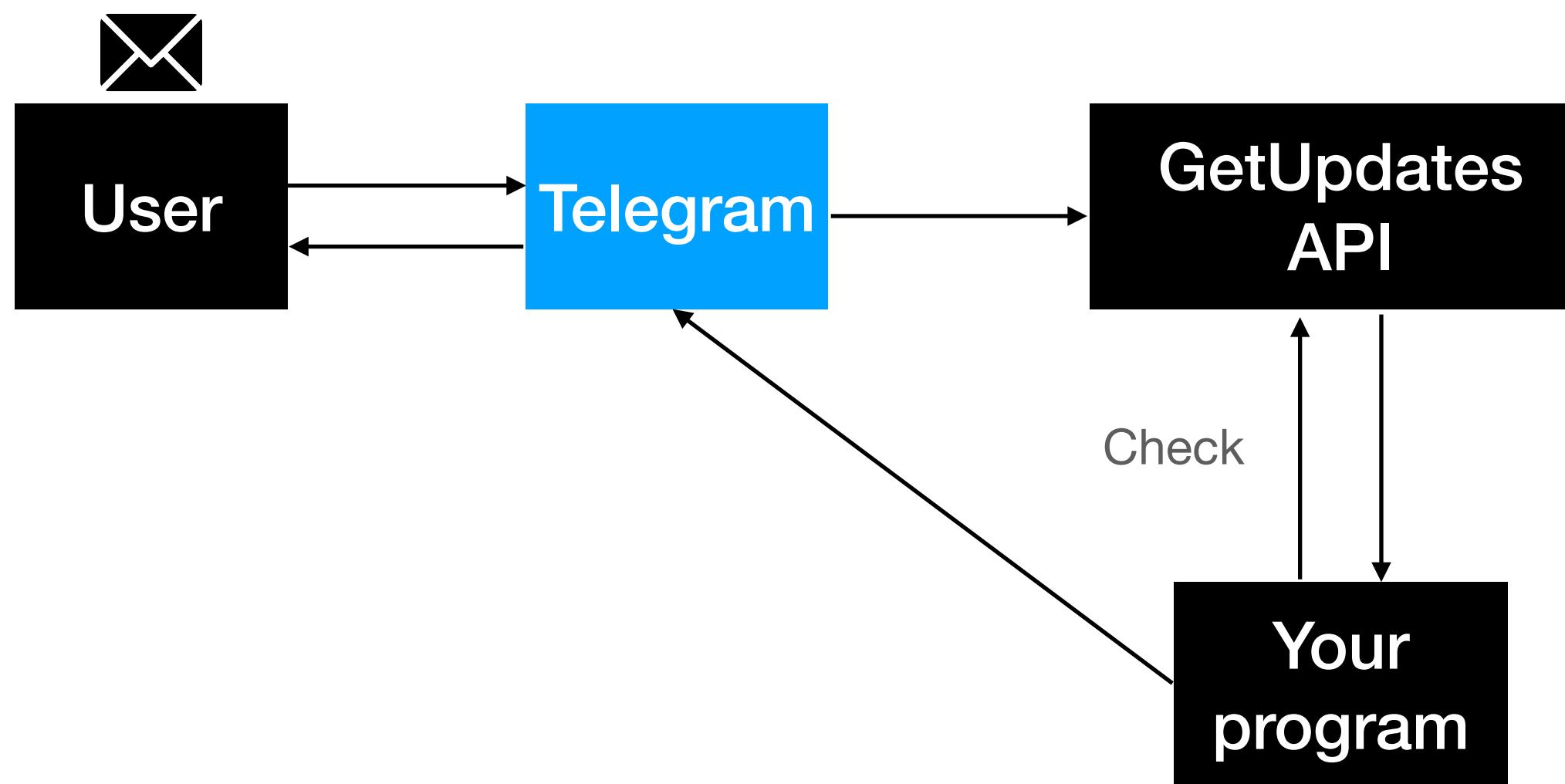
Fix the `authenticate_client`
and `analyse_text` function

All done! 😊

Extra: Webhook vs Polling

- Remember `updater.start_polling()`?
- What does that mean?

Polling



Webhook



Extra: Webhook vs Polling

- Sample implementation of a webhook: <https://towardsdatascience.com/bring-your-telegram-chatbot-to-the-next-level-c771ec7d31e4>

Does it run on Colab forever?

- No.
- You need to host your bot somewhere
 - Could be your computer
 - Could be on the cloud somewhere (someone else's computer)
 - Visit <https://roc.ms/mlsatelepy> to get a copy of the same bot we made and run it on your own computer instead
 - Google “deploy python project to Azure” to start
 - Caveat: Name your file “app.py” for it to work on Azure

Further Reading & Resources

Telegram's Bot API

- Telegram
 - <https://core.telegram.org/bots/api>
 - Other things you can do
 - Accept payments (<https://core.telegram.org/bots/payments>)
 - Talk to @ShopBot
 - Log In with Telegram (<https://core.telegram.org/api/passport>)
 - Create Games (<https://core.telegram.org/bots/games>)

Further Reading & Resources

SDKs

- <https://github.com/python-telegram-bot/python-telegram-bot>
 - Learning By Example
 - Documentation
- Want to use nodejs?
 - <https://github.com/yagop/node-telegram-bot-api>
- Other SDKs for other languages: <https://core.telegram.org/bots/samples>

Good Practices

- Your token is precious
 - Don't commit it to Git
 - People can do nasty things with it
 - Inject it into your code via an environment variable
 - Your username and user_id is precious too
 - Manage your packages properly
 - Use pip, npm, maven, whatever

What we did today

- Build a simple telegram bot
- Connect it to Azure QnA Maker and Azure Text Analytics
- Fully functional demo of what we did: <https://roc.ms/mlsatelepy>
- Slides: <https://roc.ms/mlsateleslides>