








Difference between C46 1:1 & C46 1:2	<ol style="list-style-type: none"> 1. Included class dynamics. 2. Added emojis. 3. The activities have been restructured for app development 4. Removed the example for creation of context in teacher activity 5. Removed creation of many follow up intent and changed the responses accordingly. 6. Added knowledge FQ directly rather than adding to an intent.
Topic	HEALTH BOT
Class Description	Students will add the features to the health bot to include the follow up of previous intent.
Class	C46
Class time	50 mins
Goal	<ul style="list-style-type: none"> • Develop a Health Bot. • Learn to create a follow-up intent. • Understand the concept of contexts.
Resources Required	<ul style="list-style-type: none"> • Teacher Resources: <ul style="list-style-type: none"> ○ Dialogflow ○ laptop with internet connectivity ○ earphones with mic ○ notebook and pen • Student Resources: <ul style="list-style-type: none"> ○ Dialogflow ○ laptop with internet connectivity ○ earphones with mic ○ notebook and pen
Student Motivation and	<ul style="list-style-type: none"> • Hats-off: Specific instructions for giving hats-off will be provided in the lesson.

Engagement	<ul style="list-style-type: none">  Concept Magnifier: Used to highlight new concepts and connect them with real-life examples.  Knock-Knock!: To nudge the students to make sure they are attentive.  Thinking Caps: Used to engage the students for an activity or Q&A.  All types of Quizzes: Includes revision quizzes, riddles and pop-up quizzes.  Candy Boosters: Used to motivate the students to do better in the activities.  Important Points to Remember: To highlight important concepts. 	
Class structure	Warm-Up Teacher-led Activity Student-led Activity Wrap-Up	5 mins 15 min 25 min 5 mins
CONTEXT <ul style="list-style-type: none"> Introduce the concept of contexts and types of contexts. 		
Class Steps	Teacher Action	Student Action
Step 1: Warm-Up (5 mins)		<i>FYR: If out of the two students, one is an introvert/average learner, and the other is an extrovert/fast learner, then we refer to the introvert/average learner as</i>

	<p>Hello! Welcome back to your action-packed coding class!</p> <p>To start today's class - let's quickly review what we did in the last class.</p>	<p><Student name 1>and the extrovert/fast learner as <Student name 2>.</p> <p>The students review the code from the last class.</p>
	 <p>Refer Teacher Resources page/slide 1-2 for image.</p> <p>Q1) Who is an agent?</p> <p>Q2) What is the purpose of Small Talk?</p>	<p>ESRs: A Dialogflow agent is a virtual agent that handles conversations with your end-users.</p> <p>ESRs: Small talk is used to provide responses to casual conversation.</p>
	<p>In the last class, we learned about a new platform called Dialogflow, where we learned to create the intent and we have set an appointment by creating a new intent.</p> <p>In today's class, we will create a health bot by adding a few more features to the existing class.</p>	

Teacher Initiates Screen Share		
<p style="text-align: center;"><u>CHALLENGE</u></p> <ul style="list-style-type: none"> • Introduce the concept of contexts. • Create a customized welcome intent. 		
Step 2: Teacher-led Activity (15 min)	<p><i>The teacher opens Teacher Activity 2 and guide the student to open Student Activity 1.</i></p> <p><i>Note: The teacher should guide the student to open the agent created in the last class using the above link.</i></p>	<p><i>The students open Student Activity 1.</i></p>
	<p>Now, we will create a chatbot specific to health assistants by adding features to the existing creation.</p> <p>Let's create a welcome intent specific to our chatbot.</p> <p><Student name 1>, do you have an idea where we should go for welcome intent?</p>	<p>ESRs: Varied.</p>
<p>1. Click on the Default Welcome Intent:</p>		

Intents

CREATE INTENT

Search intents



Default Fallback Intent

Default Welcome Intent

2. Navigate to the **Responses** section:

Responses ?

DEFAULT +

Text Response



- 1 Hi! How are you doing?
- 2 Hello! How can I help you?
- 3 Good day! What can I do for you today?
- 4 Greetings! How can I assist?
- 5 Enter a text response variant

3. In the **Text Response** table, delete all the default responses.

4. Create a default **Text Response** under **Default Welcome Intent**:

DEFAULT GOOGLE ASSISTANT +

Text Response



1 Hi, I am your health assistant. Please say health checkup to start my service.

2 Enter a text response variant

5. Click on SAVE.

Great! We have created our welcome bot.

Similarly, we wanted the bot to reply with a closing conversation whenever we say some ending phrases.

To do that, let's create the intent first.

<Student name 2>, how will you create the intent?

Exactly! Let's create an intent for ending the conversation.

ESRs: Click the add button next to intents in the left sidebar menu.

6. Let's click on **Intents** and create the new intent named as **quit**:



Intents

CREATE INTENT

Search intents

Default Fallback Intent

Default Welcome Intent

No regular intents yet. [Create the first one.](#)

7. Create a **Training Phrase** with possible ending conversation.

Training phrases

goodbye|

Goodbye

cancel

exit

Quit

end

I got to go

Talk to you later

8. We can create a **Text Response** for the **Training Phrase**:

Text Response
🗑️

1 I am happy to assist you. I'll be around if you need me. See you then.

2 Enter a text response variant ⬆️⬆️

9. Click **"SAVE"**.

	<p>Great! We have created the intent to welcome and end the conversation.</p> <p>Now we are going to learn to include context into our Dialog Flow.</p> <p><Student name 1>, Do you know what context is?</p> <p>Contexts allow your agent to carry information from one intent to another.</p> <p>Let's understand in an easy way. If a person says to you "they are orange", you need to understand what "they" is referred to.</p>	<p>ESRs: Varied.</p>
	<p>Context is classified into 2 intents:</p> <ul style="list-style-type: none"> ● Output contexts ● Input contexts <p>Input and output contexts are applied to intents. They work together to control conversation flow.</p>	<p><i>The students listen.</i></p>

	<p>One thing to be noted while giving the context is lifespan.</p> <p>Each active context has a lifespan that defines the number of conversational turns for which the context remains active. The default value is 5, but we can give any value.</p>	
	<p>Let's create a bot that asks for the symptom and gives you the basic first aid, and it will also suggest fixing the appointment in the worst case. Now you can share the screen. I will guide you in creating the bot with context.</p>	
<ul style="list-style-type: none"> • Ask the students to press the ESC key to come back to the panel. • Guide the students to start screen share. • Teacher gets into fullscreen. 		
<p style="text-align: center;"><u>ACTIVITY</u></p> <ul style="list-style-type: none"> • Create a symptom checker context. • Learn the concept of follow up to the intent. 		
Step 3: Student-Led Activity (25 mins)	<p><i>Guide the student to open Student Activity 1.</i></p> <p><i>Note: Guide the students to create the welcome and closing conversations intent to their bot.</i></p>	<p><i>The students open Student Activity 1.</i></p>

	<p><Student name 1>, what could be the first step to train the bot?</p> <p>We need to create the intent for the symptom checker.</p> <p><Student name 2>, what will be the training phrase and response for that intent?</p> <p>If you could see when the user says health checkup, the bot will give options to choose from. The training phrase will be a health checkup and the option will be a response.</p> <p><Student name 1>, which concept will allow us to carry information from one intent to another.</p> <p>Great! Now, create a new input with these inputs.</p>	<p>ESRs: Varied.</p> <p>ESRs: Varied.</p> <p>ESRs: Contexts.</p>
10. Create a new intent as symptom checker:		

Intents

CREATE INTENT

Default Fallback Intent

Default Welcome Intent

quit

symptom checker

11. Open **symptom checker** and add the output **Contexts** as **await_choice**:

symptom checker

SAVE

Contexts ?

1


await_choice

Add output context

Note: we can give any name inside the output context, but the same name should be given inside the input contexts of intent.

12. Add the Training Phase as a health checkup:

Training phrases

Search training phrase  

” Add user expression


” health checkup

13. Add the **Text Response** to ask about the symptoms:

DEFAULT 

Text Response 

1 Can u choose any one option 1)Cold/flu 2)Tooth problem

2 Enter a text response variant 

14. Click SAVE:

Now let's create an intent for the symptom, frame the training phrase and the response accordingly.

15. Let's create the intent for our first symptom **cold/flu**:

Intents

CREATE INTENT

Search intents

Q

cold/flu

Default Fallback Intent

Default Welcome Intent

quit

symptom checker

16. Open the **cold/flu** intent and add the input **Contexts** as **await_choice**:

cold/flu

SAVE

Contexts

await_choice

Add input context

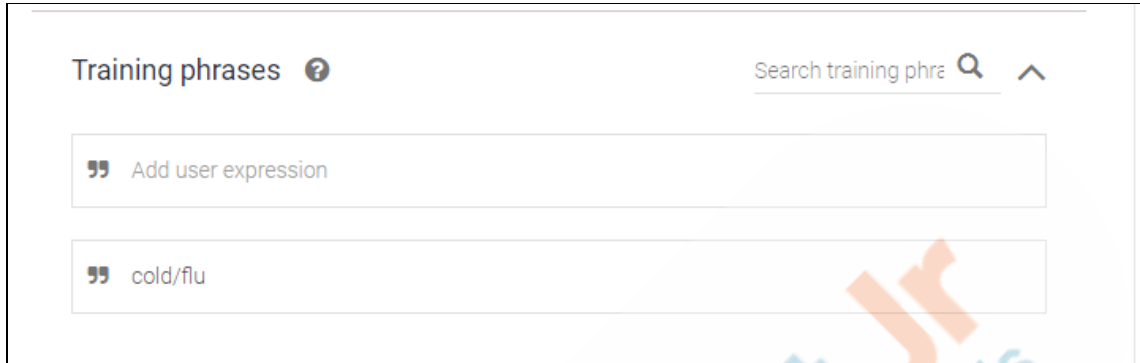
5 await_choice

Add output context

Note: We need not create output context, which is created automatically when follow-up intent is created.

17. Create the training phrase as **cold/flu**:

Note: This should be in the same name which is given as a response in the previous intent (symptom checker as choice 1).



18. Add the response to the intent as a question related to the symptom:

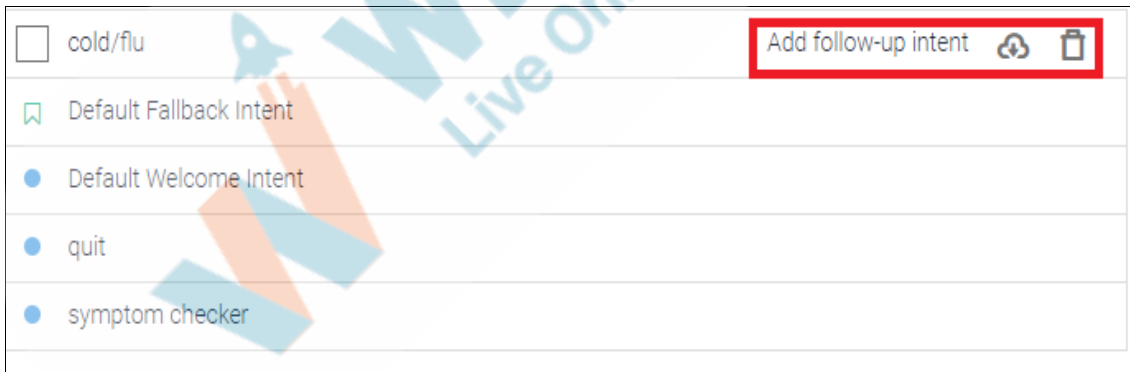


19. Click on **SAVE**.

For each symptom, we have two replies; either it should be yes or no.

	<p><Student name 2>, can you think about how we can create this follow-up?</p> <p>To do this, we can use the follow-up intents, which automatically set contexts for pairs of intents.</p> <p>Follow-up intent is a child of associated parent intent. To activate child intent the previous intent should be based on the parent.</p> <p><i>The teacher guides the students to create follow up intent.</i></p>	<p>ESRs: varied</p>
--	--	---------------------------------------

20. Create the child intent by clicking on **Add follow-up intent**, followed by yes/no. We can choose any option based upon the follow-up intent training phrase for which that intent was created:



The screenshot shows a list of intents in a table-like structure. The first row has a checkbox next to 'cold/flu'. To the right of this row, there is a button labeled 'Add follow-up intent' which is highlighted with a red rectangular box. Below this button are two small icons: a cloud and a trash can. The other rows in the list are 'Default Fallback Intent', 'Default Welcome Intent', 'quit', and 'symptom checker', each with a blue circular icon to its left.

- Select the **yes** option:

● cold/flu	custom
🔖 Default Fallback Intent	
● Default Welcome Intent	fallback
● quit	yes
● symptom checker	no
	later

21. Now you could see the child's intent for option yes is created for the **cold/flu**:

● cold/flu ^
● ↳ cold/flu - yes
🔖 Default Fallback Intent
● Default Welcome Intent
● quit
● symptom checker

22. You could see the input context is created automatically:

● cold/flu - yes
 SAVE

Contexts ⓘ

await_choice ⓘ coldflu-followup ⓘ Add input context

Add output context

23. Default **Training Phrases** for “yes” are created automatically. We can either make use of the same or can add any other intent needed.

Training phrases ⓘ
 Search training phrase 🔍

” Add user expression

” yes

” okay I will

” why not

” yes that's alright

” yes I do

” exactly

” of course

” yep that's ok

24. Add the response for the intent:

<div> <div>Responses ?</div> <div> <div>DEFAULT +</div> <div> <div>Text Response</div> <div> <div>1</div> <div>Diagnosis: Symptoms are related to COVID-19. Seek immediate medical attention if you have serious symptoms. Always call before visiting your doctor or health facility.</div> </div> <div> <div>2</div> <div>Enter a text response variant</div> </div> </div> </div> </div>		
	<p>Great! We have created the follow-up intent.</p> <p>Let's create more follow-up intent to make the bot understand the symptom.</p> <p><i>The teacher should guide the students to create more follow-up intents.</i></p>	<p><i>The students create the follow-up intents.</i></p>
<p>25. Create follow-up intent for the No:</p>		

● cold/flu ▾	custom
🔖 Default Fallback Intent	fallback
● Default Welcome Intent	yes
● quit	no
● symptom checker	later
	cancel

26. Add the response as there is no change in the Training Phrase and Contexts:

Responses ?

DEFAULT +

Text Response

1 Happy to hear that . Maintain school distancing to stay safe

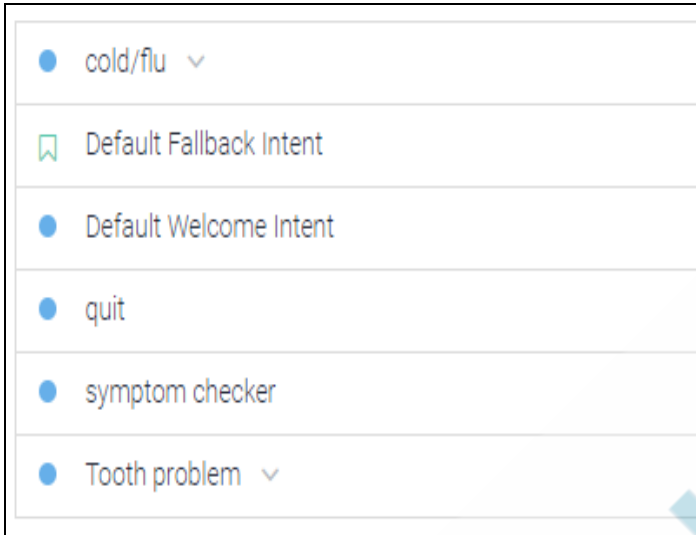
2 Enter a text response variant

<Student name 2>, do you remember the second option bot was providing?

Exactly!

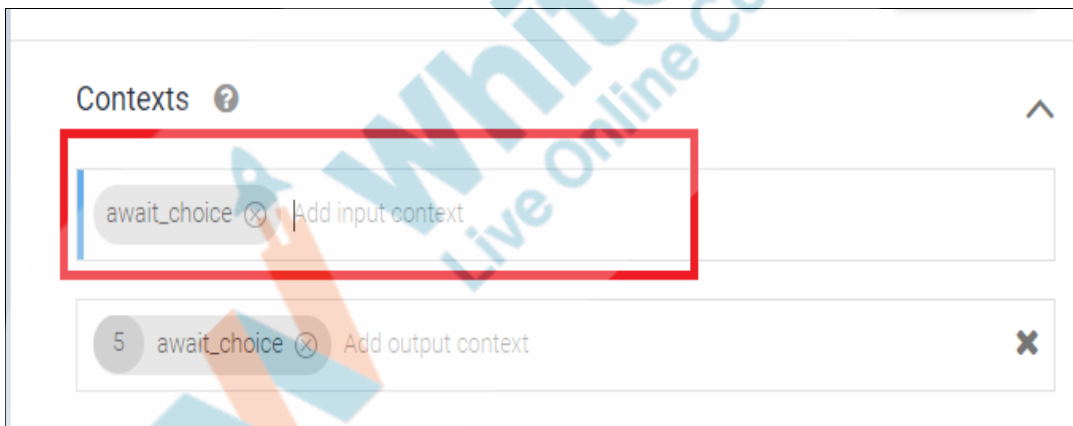
ESRs:
Tooth problem.

27. Let's create the intent as **Tooth problem**:



- cold/flu ▾
- 🔖 Default Fallback Intent
- Default Welcome Intent
- quit
- symptom checker
- Tooth problem ▾

28. Add the input contexts:



Contexts ?

- await_choice ⊗ Add input context
- 5 await_choice ⊗ Add output context

Note: We need not create output context, which is created automatically when follow-up intent is created.

29. Create the training phrase as **tooth problem**:

Note: This should be in the same name which is given as a response in the previous intent (symptom checker as choice 1).

Training phrases ?

Search training phrase

" Add user expression

" tooth problem

30. Add the response to the intent as a question related to the symptom:

Responses ?

DEFAULT +

Text Response

1 Do you have swelling and pain in mouth
2 Enter a text response variant

31. Click on SAVE.

32. Now let's create the follow-up intent for the **tooth problem** intent.

33. Create an intent with follow-up **yes**.

34. Add the response to the training phase:

Responses ?

DEFAULT +

Text Response

1

Can I book an appointment to see a dentist

2

Enter a text response variant

35. Similarly create a follow-up intent for **no** and add the following response:

Responses ?

DEFAULT +

Text Response

1

Rinse your mouth with warm water to avoid tooth decay issues

2

Enter a text response variant



Now I want you both to put on your thinking caps.

Can anyone tell me how to give FAQ to the bot?

We can give this by the knowledge base. It is a set of documents.

ESRs: Varied.

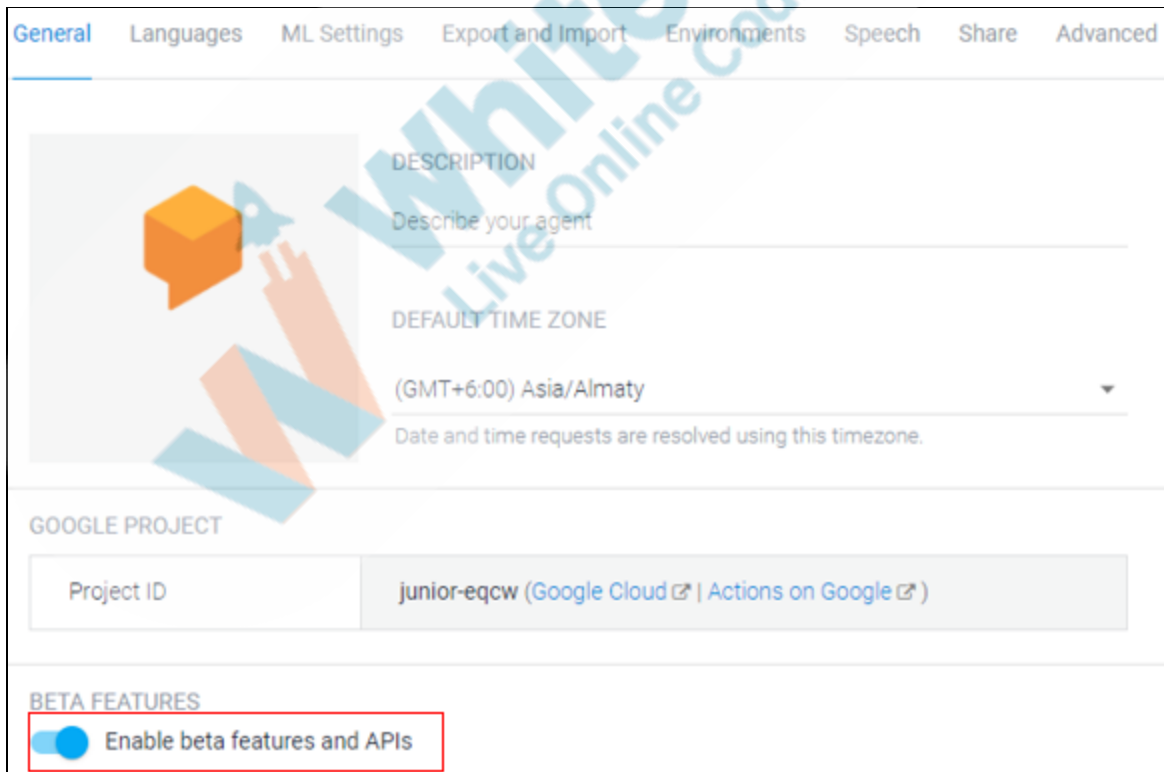
Dialogflow makes use of this concept of knowledge base when searching for the end-user expression response.

Let's see how to add a knowledge base to this bot.

To add the knowledge base we need to enable beta features.

Ensure that beta features are enabled:

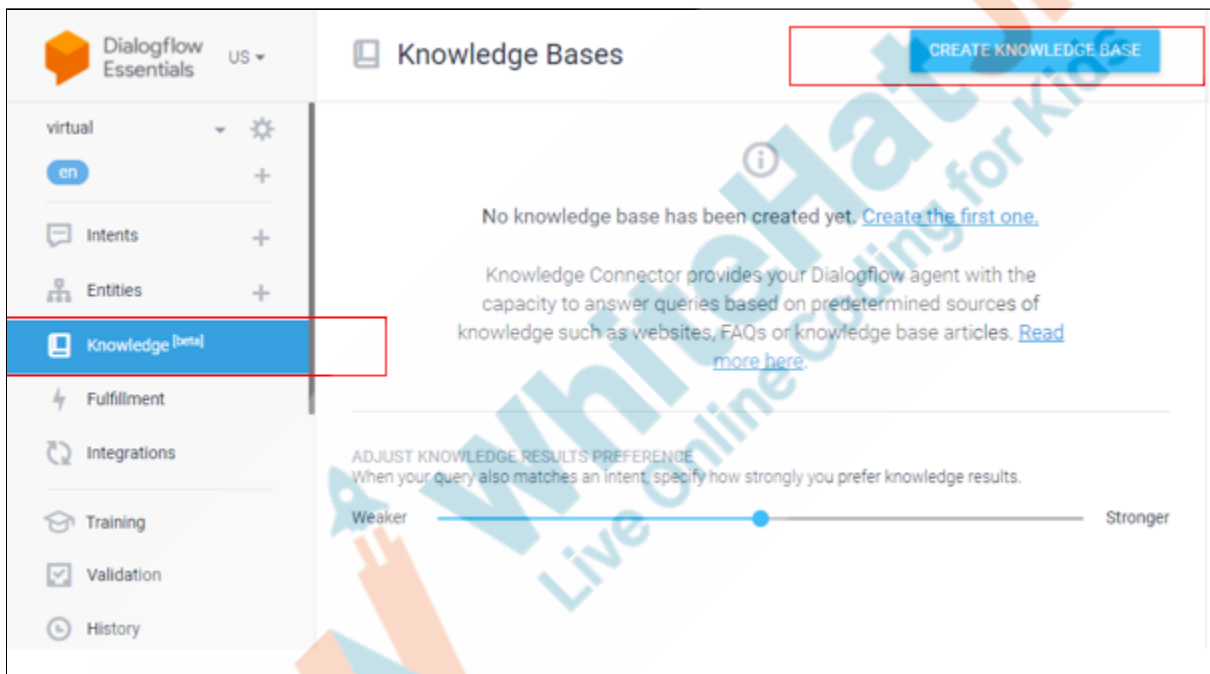
1. Go to the [Dialogflow ES Console](#).
2. Select an agent.
3. Click the settings button next to the agent's name.
4. Scroll down while on the General tab and ensure that Beta Features are enabled.
5. If you have made changes, click Save.



The screenshot shows the 'General' tab of the Dialogflow ES Console. The 'BETA FEATURES' section at the bottom has a toggle switch labeled 'Enable beta features and APIs' which is currently turned on. This section is highlighted with a red rectangular box.

We have enabled the knowledge base. Now, we need to create the knowledge intent.

1. First, go to the Dialogflow Console.
2. Then choose the agent.
3. Now, click on the option named **Knowledge**, which is present on the left side of the menu bar.
4. Next, click on the Create Knowledge Base.



5. Then enter the name of the knowledge base and then click on the Save button.

Let's see how to add the Document to the Knowledge Base

Currently, there is no document contained in the new knowledge base. So, we have to add the document in the knowledge base as per our needs.

There are various steps to add the document to the knowledge base:

1. First, go to the Dialogflow console.
2. Then, choose the agent.
3. Next, click on the option named Knowledge which is present on the left side of the menu bar.
4. Then, click on the knowledge base name in which we want to add the document.
5. Click on the option named **Create the First One** or New Document:



1. Then, enter the name of the document.
2. Next, for the Knowledge type, we have to select the option FAQ.
3. Now, select the MIME Type, which is text/HTML
4. Then, for the Data Source, choose the option as URL.
5. Now, in the URL field, we have to enter the URL:

<https://www.fda.gov/emergency-preparedness-and-response/coronavirus-disease-2019-covid-19/covid-19-frequently-asked-questions>.

Document Name *

Covid-19

Knowledge Type *

FAQ

Mime Type *

text/html

DATA SOURCE

☐ File on Cloud Storage

gs://bucket-name/object-name

☒ URL

<http://www.example.com/faq> *

<https://www.fda.gov/emergency-preparedness-and-response/coronavirus-disease-2019-covid-19/covid-19-frequently-asked-questions>

☐ Upload file from your computer

SELECT FILE

☐ Enable Automatic Reload ?


CREATE

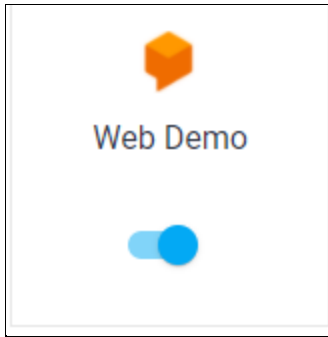
Note: For the link, the students can open [Student Activity 2](#) and the teacher can open [Teacher Activity 3](#).

6. Then click on the CREATE button.
7. The knowledge is created, and you can note that the response is automatically added as shown below:

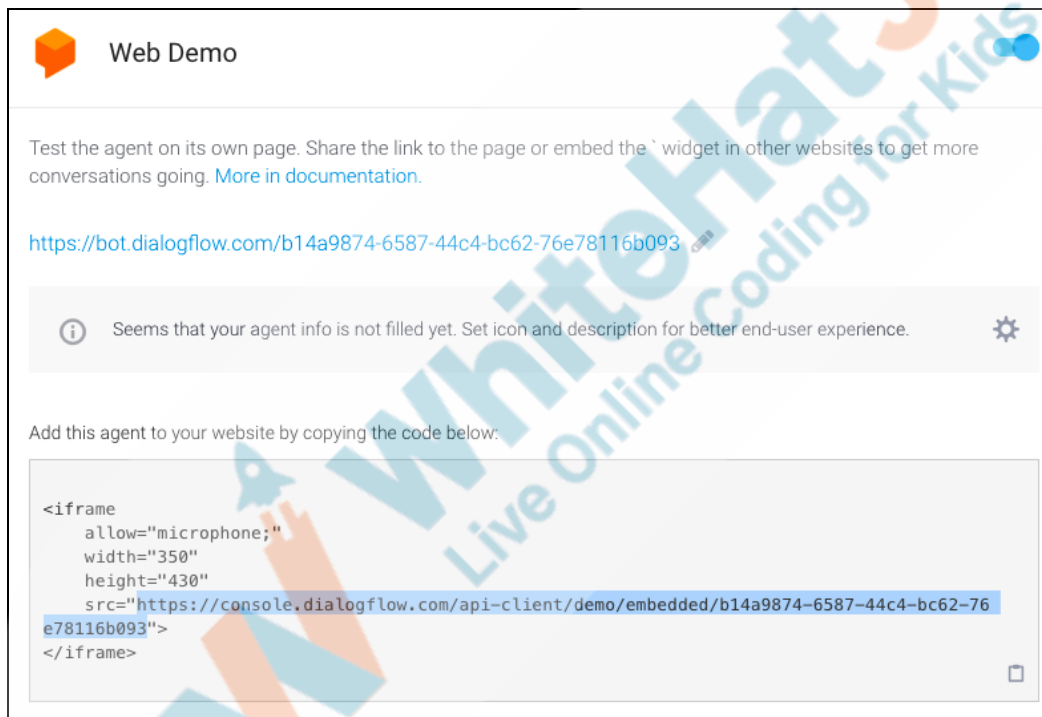


8. Click on SAVE.

	<p>Feeling interested?</p> <p>Now, let's integrate our chatbot into the website.</p>	ESRs: Varied.
	<p>To do that, we need to click on "Integrations" from the menu, and enable Web Demo.</p> <p><i>The teacher guides the students to enable integration.</i></p>	<p><i>The students enable the web demo.</i></p>
		

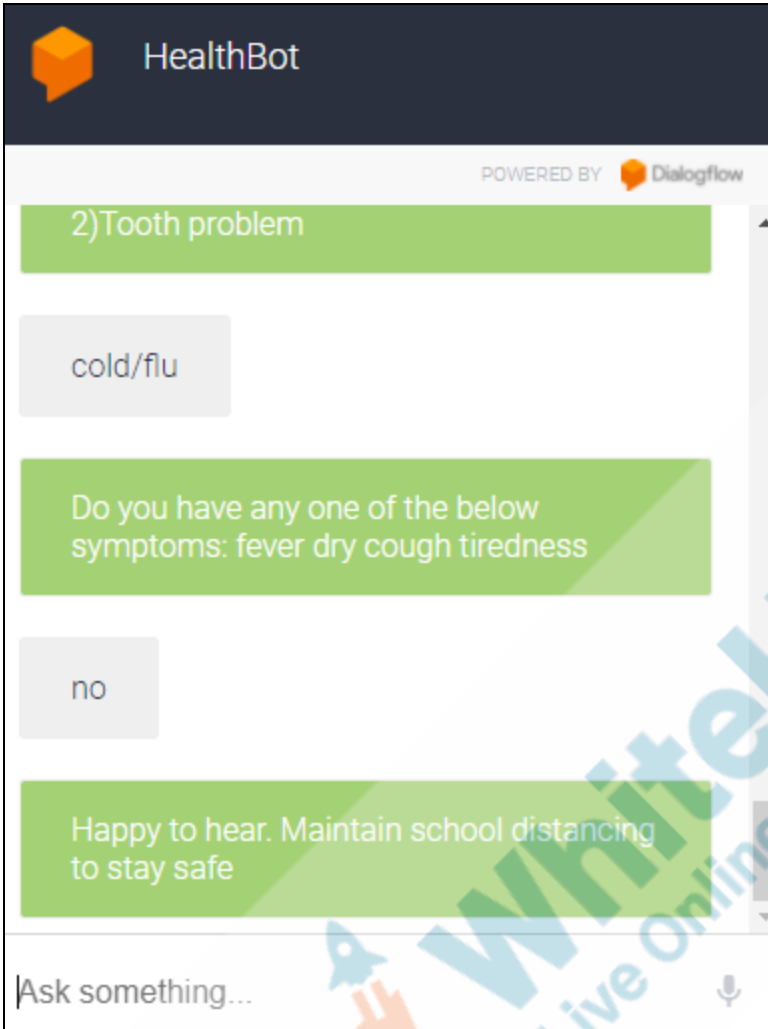





You will get a pop-up showing the bot link:





The link in the src or source is the bot link. You copy this link, paste it into the panel, and press submit.

You can also paste this link in another Chrome tab and see how your bot looks and works.

		
	<p>Wow! Great!</p> <p>Now we have learned to create a chatbot API. Good work!</p>	
<p>Teacher Guides Student to Stop Screen Share</p>		
<p style="text-align: center;"><u>FEEDBACK</u></p> <ul style="list-style-type: none"> ● Encourage the students to make reflection notes in markdown format. ● Compliment the students for their effort in the class. ● Review the content of the lesson. 		

Step 4: Wrap-Up (5 mins)	Do you see how easy it is to add new functionalities on top of properly structured code?	ESR: Yes!
	<p><i><Ask both the students alternatively.></i></p>  <p>Refer Teacher Resources page/slide 3-5 for image.</p> <p>Q1) What do you mean by context?</p> <p>Q2) What is follow-up intent?</p> <p>Q3) What is a knowledge base in Dialogflow?</p>	<p>ESRs: Contexts represent the current state of a user's request and allow your agent to carry information from one intent to another.</p> <p>ESRs: Follow-up intent is a child of associated parent intent.</p> <p>ESRs: A knowledge base represents a collection of knowledge documents that you provide to Dialogflow.</p>
	<p>You get Hats-Off for your excellent work!</p> <p>In the next class, we will add how to add the appointment to our google calendar, and how to publish it in different messengers.</p>	<p><i>Make sure you have given at least 2 Hats-Off during the class for:</i></p> <div data-bbox="1019 1539 1312 1644">  </div> <div data-bbox="1019 1703 1312 1801">  </div>

		<div> <div>Strong Concentration</div> <div>  <div>+10</div> </div> </div>
	<p>Congratulations! You have set a new benchmark.</p> <p>Brace yourself! Your new challenge is ready.</p> <p>You have to apply the programming constructs learned during the past few classes to create the Fruit Collector App.</p>	
Project Overview	<p>eFRIENDBOT</p> <p>Goal of the Project:</p> <p>In Class C46, you created a chatbot using Google Dialogflow and integrated the knowledge base and follow-up intent.</p> <p>In this project, you will apply what you have learned in the class to achieve the following goals.</p> <p>Main Goal</p> <ul style="list-style-type: none"> Create a chatbot. <p>Additional Goal 1</p> <ul style="list-style-type: none"> Make the chatbot more interactive. <p>Story:</p>	

	<p>Ron is an immigrant who has recently shifted to a new place and is missing his younger brother around. As being good at technology, he has decided to learn and create a chatbot to give him company.</p> <p>I am very excited to see your project solution, and I know you both will do really well.</p> <p>Bye Bye!</p>	
<p style="text-align: center;">Teacher Clicks</p> <div style="text-align: center;">  </div>		
Additional Activities I	<p><i>Encourage the students to write reflection notes in their reflection journal using markdown.</i></p> <p>Use these as guiding questions:</p> <ul style="list-style-type: none"> • What happened today? <ul style="list-style-type: none"> ○ Describe what happened. ○ The code I wrote. • How did I feel after the class? • What have I learned about programming and developing games? • What aspects of the class helped me? What did I find difficult? 	<p><i>The students use the markdown editor to write their reflections in a reflection journal.</i></p>

Activity	Activity Name	Links
Teacher Activity 1	Teacher Resource	https://s3-whjr-curriculum-uploads.whjr.online/29055bd0-b5f5-447b-93f2-2adc1c4c7b5e.pptx
Teacher Activity 2	Dialogflow	https://Dialogflow.cloud.google.com/
Teacher Activity 3	COVID-19 FAQ	https://www.fda.gov/emergency-prep-aredness-and-response/coronavirus-disease-2019-covid-19/covid-19-frequently-asked-questions .
Teacher Activity 4	Teacher Reference	https://console.dialogflow.com/api-client/demo/embedded/efe7b039-495f-4337-8a65-91f7517a74b4
Student Activity 1	Dialogflow	https://Dialogflow.cloud.google.com/
Student Activity 2	COVID-19 FAQ	https://www.fda.gov/emergency-prep-aredness-and-response/coronavirus-disease-2019-covid-19/covid-19-frequently-asked-questions .