Difference between C48 1:1 & C48 1:2	1. Included class dynamics. 2. Added emojis. 3. The activities have been restructured for app development. 4. Removed creating Rich Response. 5. Removed Integration in Google Assistant. 6. Added importing zip file in the Dialogflow.
Topic	**CAPSTONE CLASS: FOOD ORDERING CHATBOT**
Class Description	**Students create a food ordering chatbot. Students will also learn to IMPORT the file in the Dialogflow.**
Class	**C48**
Class time	**50 mins**
Goal	• Develop a food ordering chatbot. • Learn to IMPORT and EXPORT the chatbot.

Resources Required	• Teacher Resources:
	  ○ Dialogflow
	  ○ Laptop with internet connectivity
	  ○ Earphones with mic
	  ○ Notebook and pen
	• Student Resources:
	  ○ Dialogflow
	  ○ Laptop with internet connectivity
	  ○ Earphones with mic
	  ○ Notebook and pen

Student Motivation and Engagement	• **Hats-off**: Specific instructions for giving hats-off will be provided in the lesson.

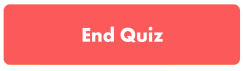
	•  **Concept Magnifier**: Used to highlight new concepts and connect them with real-life examples. •  **Knock-Knock!**: To nudge the students to make sure they are attentive. •  **Thinking Caps**: Used to engage the students for an activity or Q&A. •  **All types of Quizzes**: Includes revision quizzes, 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

- **Revising the previous class.**

Class Steps	Teacher Action	Student Action
Step 1: Warm-Up (5 mins)	Hello students! Welcome back to your action-packed coding class! To start today's class - let's first quickly revise what we did in the last class.	*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 <Student name 1> and the*

		extrovert/fast learner as a student name 2. *The students review the code from the last class.*
	I have an exciting quiz question for you! Are you both ready to answer this question? *Click on the*  *button on the bottom right corner of your screen to start the In-class Quiz.* *A quiz will be visible to both you and the students.* *Encourage the students to answer the quiz question.* *The students may choose the wrong option, help the students to think correctly about the question and then answer again.* *After the students select the correct option, the*  *button will start appearing on your screen.* *Click on End Quiz to close the quiz pop-up and continue the class.*	**ESRs:** Varied.

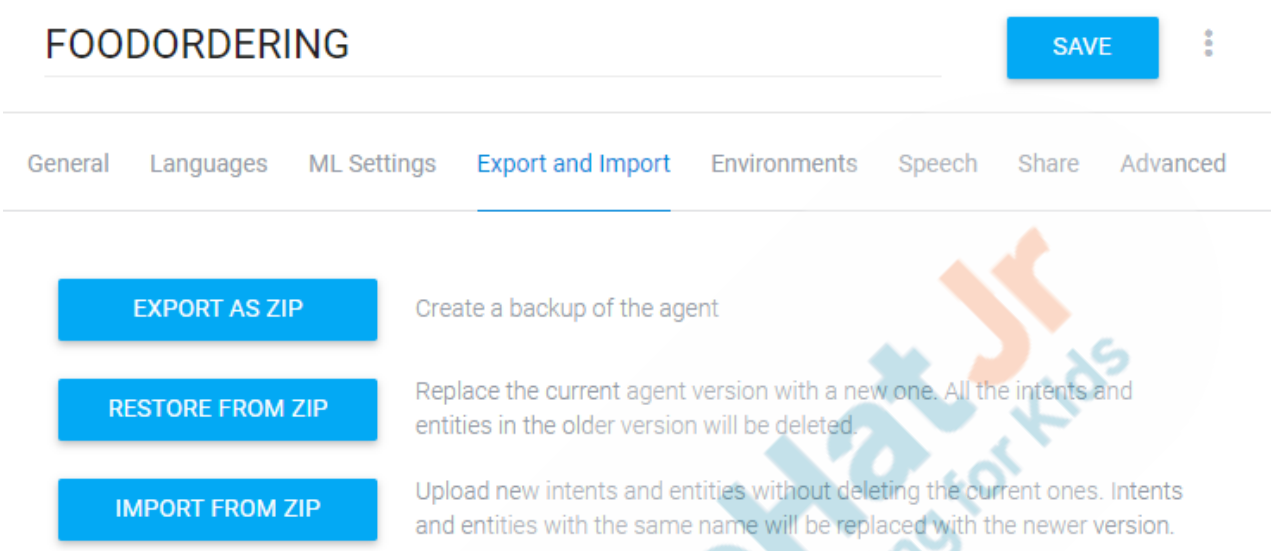
	 *Refer to Teacher Resources page/slides 1-2 for the visual aid.* Q1) What is an entity in Dialogflow? Q2) Which file format should be used for FAQ in the knowledge base?  *Give a candy booster to the student who has answered correctly.* Great! In today's class, we are going to revise the Dialogflow by creating a restaurant bot.	**ESRs:** Entities are a mechanism in Dialogflow for identifying and extracting useful data from natural-language inputs. **ESRs:** The document content contains question and answer pairs that are either HTML or in CSV format.
Teacher Initiates Screen Share		
CHALLENGE • **Learn to IMPORT the zip file**		
Step 2: Teacher-led Activity (15 mins)	*The teacher opens Teacher Activity 2, and guides the students to open Student Activity 1.*	*The students open Student Activity 1.*

	In the previous class, we learned to create an intent **parameter and action**. Finally, we ended the class by publishing the health chatbot as an app. In today's class, we will learn to create our own chatbot for restaurants and we will also learn how we can IMPORT the file in Dialogflow.	

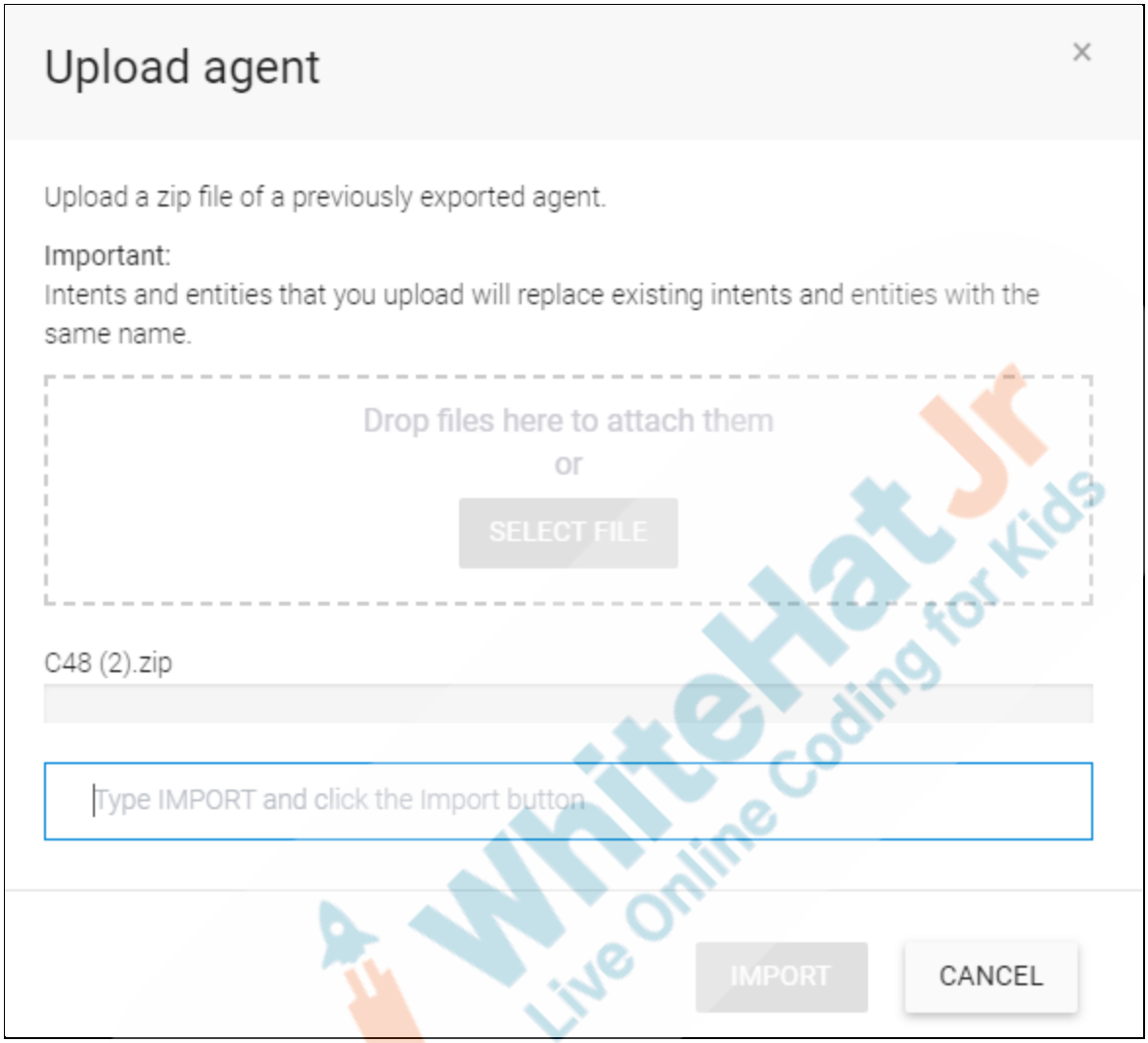
1. The teacher needs to download the zip file from [Teacher Activity 3.]

2. Create a new agent as **FOODORDERING:**

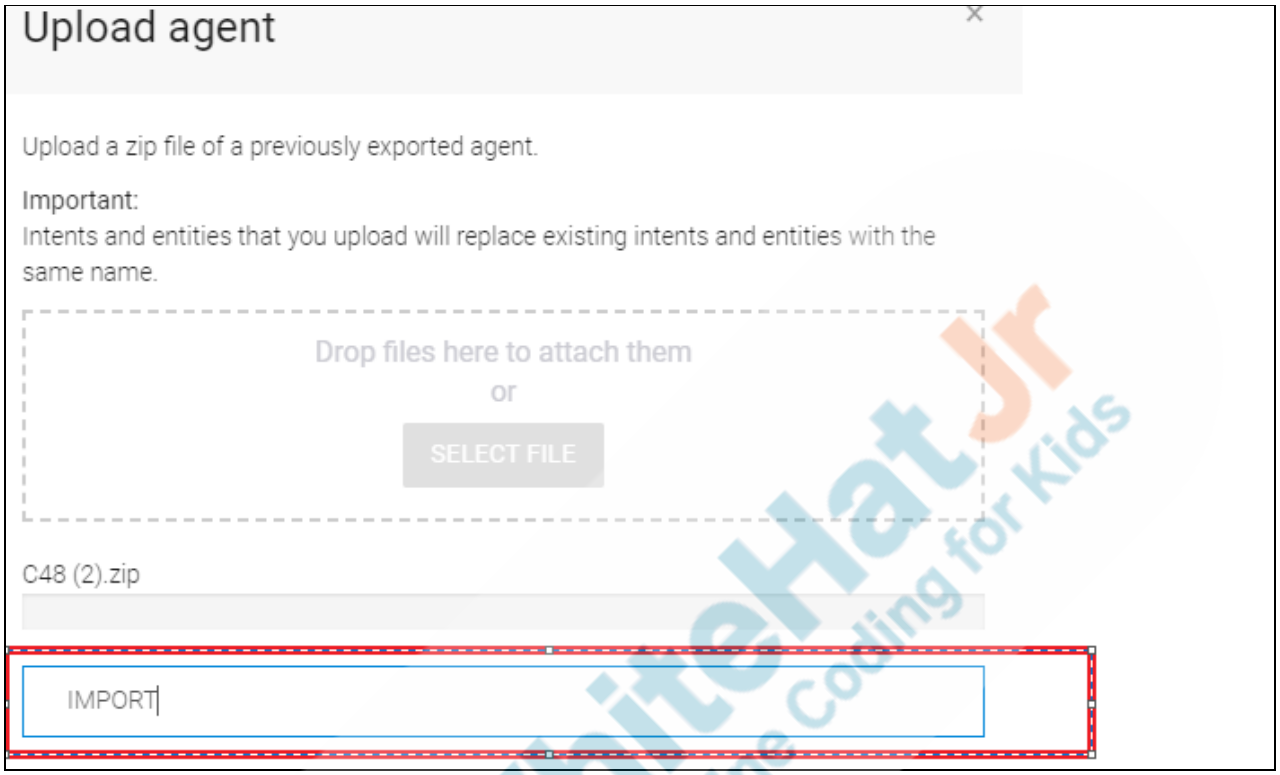
3. Click on the settings icon to select the tab, **Export and Import**. Thereafter, choose the **IMPORT FROM ZIP** option:



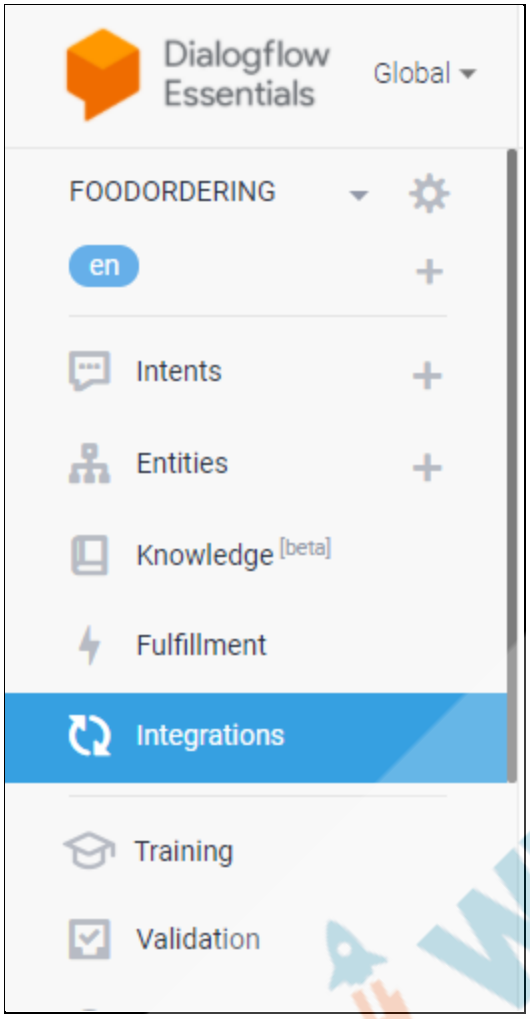
4. Add the zip file from folder **C48**:



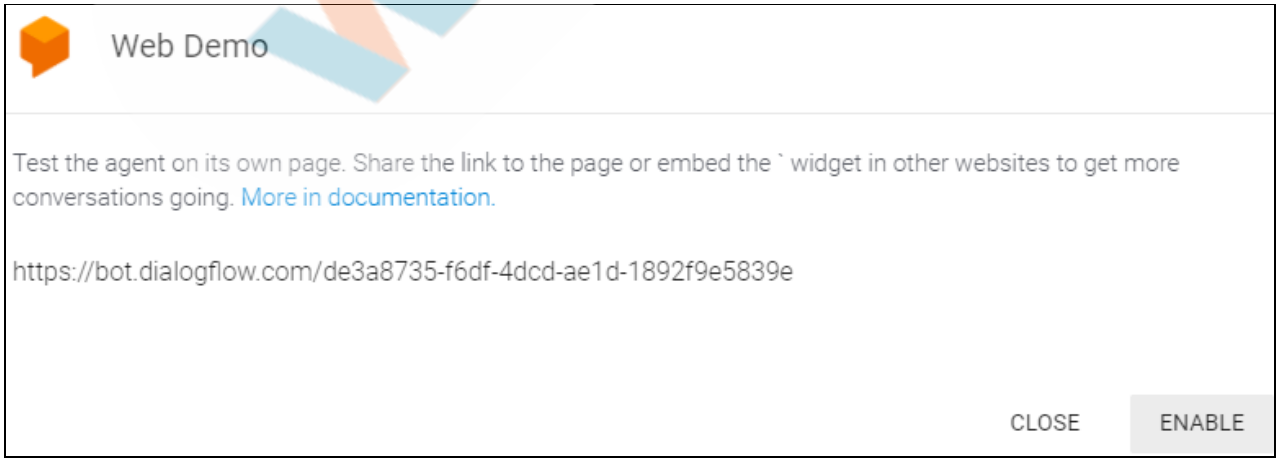
5. Type **IMPORT** within the tab provided and press the **IMPORT** button. Click **Done**:



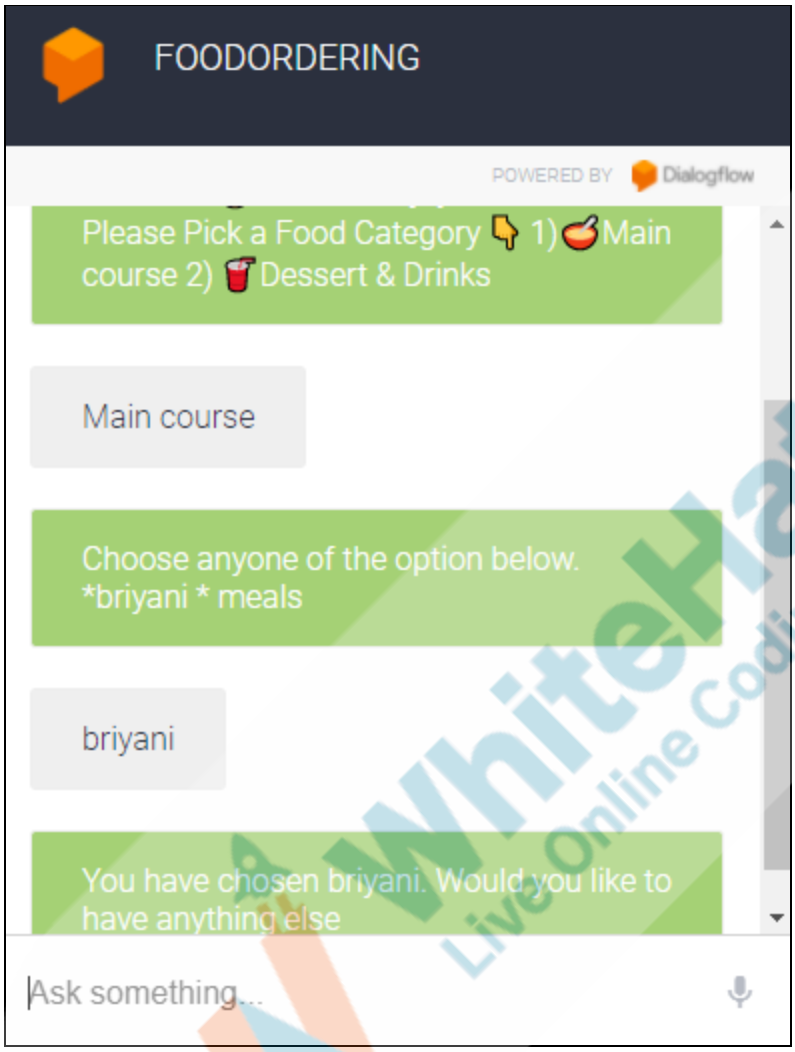
6. Now test the app by selecting the **integrations** option.



7. Enable the **Web Demo** validation by selecting **ENABLE**:



8. Give a brief overview of what our application to order food will look like:

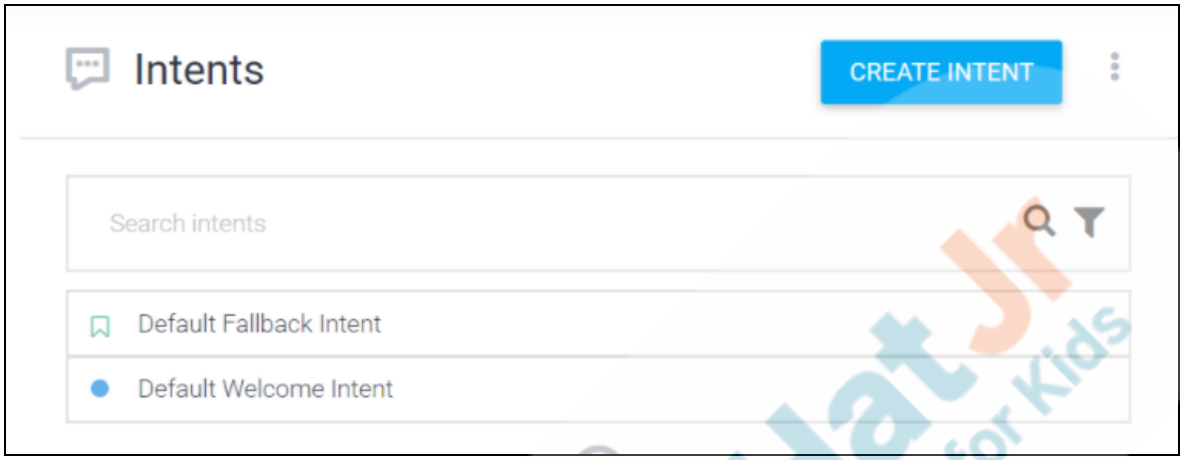


	Create excitement among the students about the upcoming activity. Now the most exciting part of today's class begins!	*The students listen.*

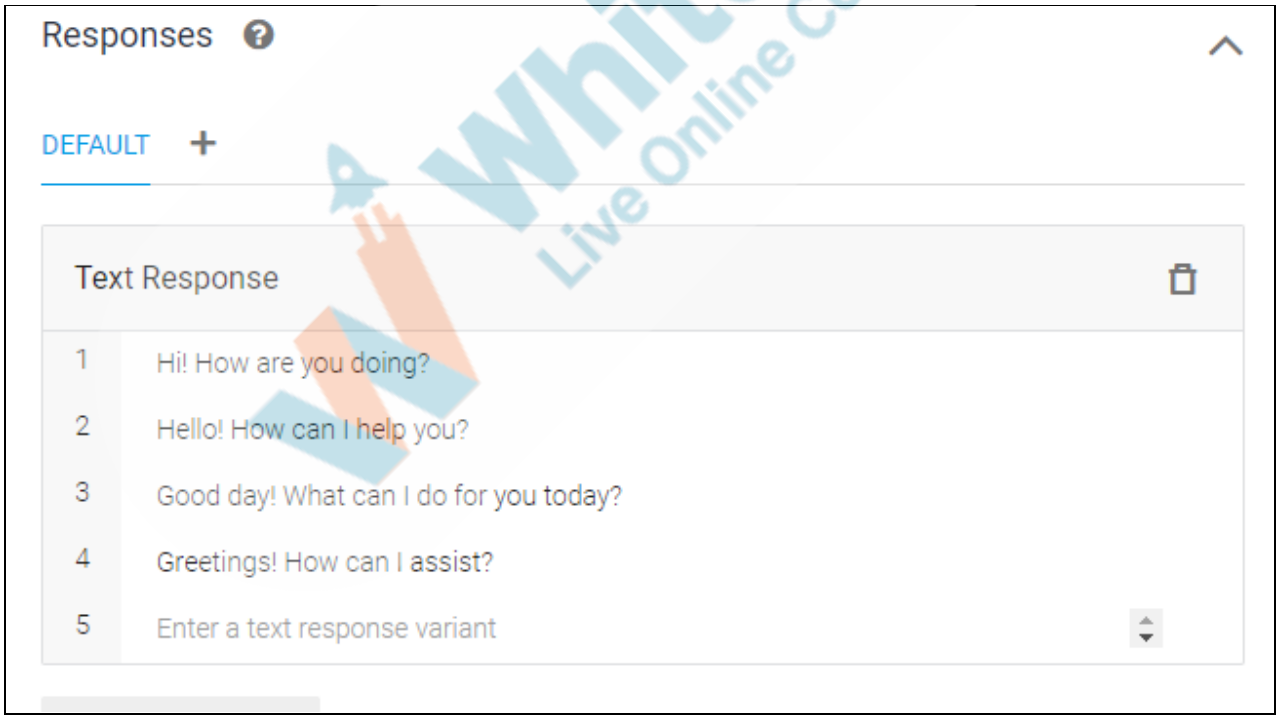
	Now, you both are going to make the restaurant app.	
• **Ask the students to press the ESC key to come back to the panel.** • **Guide the students to start a screen share.** • **Students get into fullscreen.**		
ACTIVITY • Create a Restaurant chatbot.		
Step 3: Student-Led Activity (25 mins)	*Guide the student to open Student Activity 1.* Now tell me how to get started to create our restaurant bot? Our bot should contain the following functionalities: • A welcome message that would greet the user. • Display the list of items to choose. • Provide the user with an option to either add more items after confirmation or proceed with checking out. • Get the user details to deliver the food.	*The students open Student Activity 1.* *Students are expected to build the Restaurant bot on their own with the teacher's guidance.* **ESRs:** Varied

Let's start creating the Chatbot for restaurants with the greeting message.

1. Click on **Default Welcome Intent**:

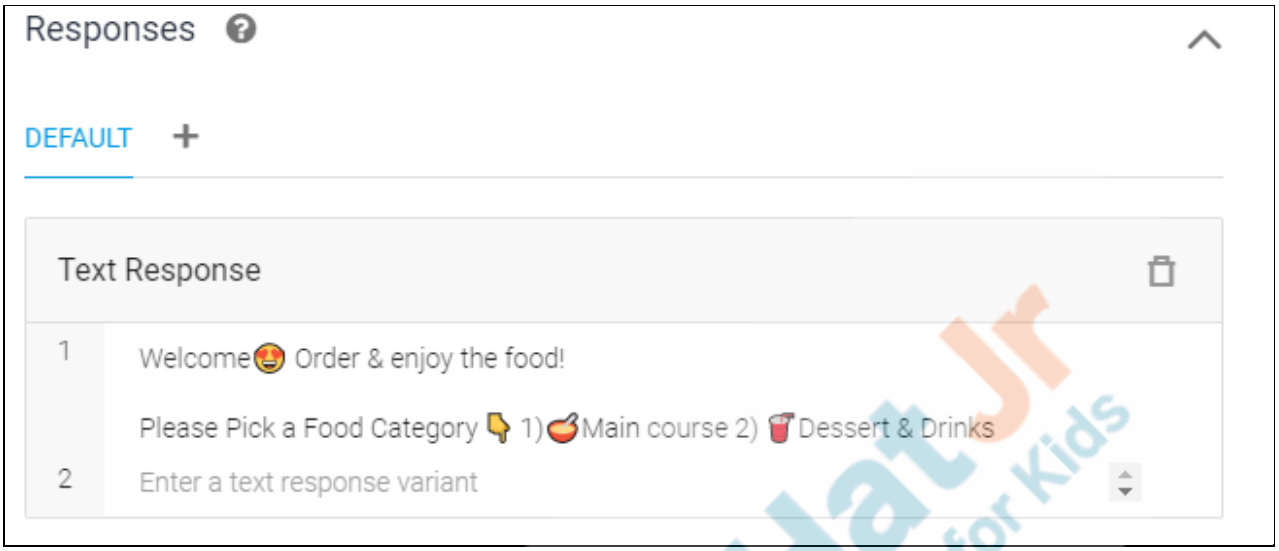


2. Navigate to the **Responses** section.

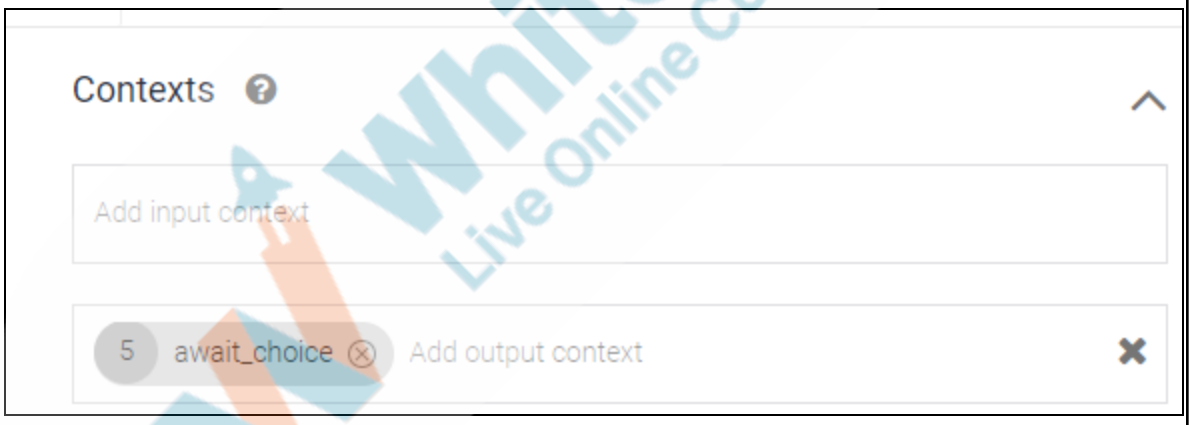


3. Delete all the default responses, in the **DEFAULT** -> **Text Response** table.

4. Create a custom response under the **DEFAULT** welcome intent:



5. Now, add the output **Contexts** as **await_choice**.



6. Click **SAVE**.

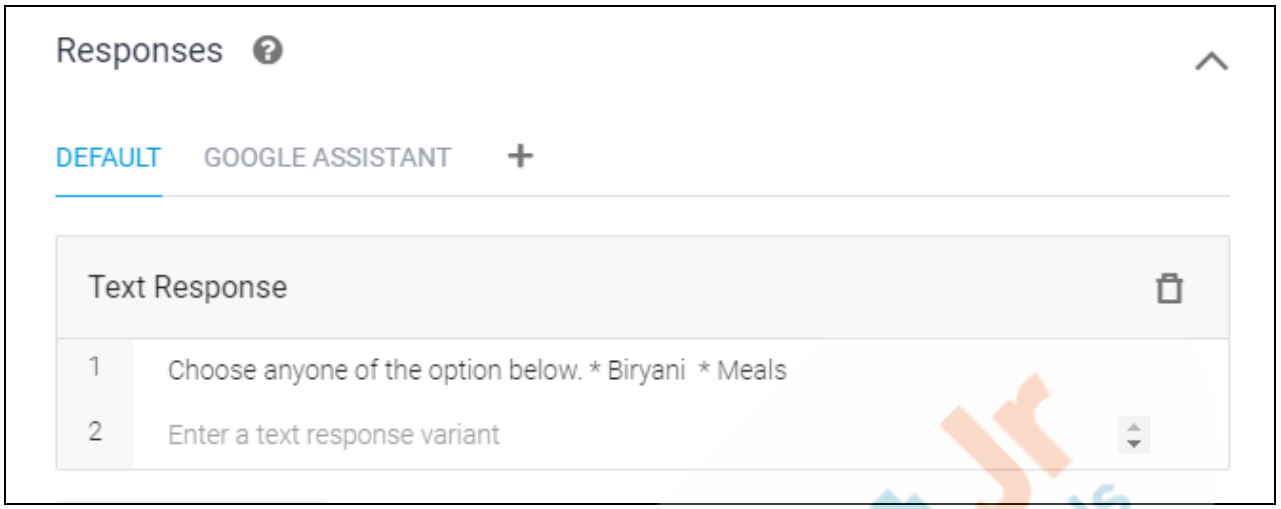
	Great! We have created a bot that displays a **Welcome** message to the user. *<Student name 1>*, why have we created output contexts for the Welcome intent?	**ESRs:** When the option is selected from the menus, the respective intent would be active.

		ESRs: Varied
	<Student name 2>, what will the next intent be? We need to create an intent for the menu options that we created during the **Welcome Intent**. *<Student name 1>*, which is the first option in the menu we added? *Note: Guide the student to look at the welcome intent where we have created a menu list along with the welcome message.* Exactly! Let's create an intent for our main course. *The teacher guides the students to create the **Main course** intent.*	**ESRs:** Main course. *Students create the **Main course** intent.*

1. Create a new intent named **Main course**.
2. Create input contexts for the corresponding output contexts. By adding **await_choice** in **Add input context** within the **Main course** intent.

3. Add the **Training phrases**.
    - The user can select the option, **Main Course**, Hence, we will be adding both to the **Training phrases**:

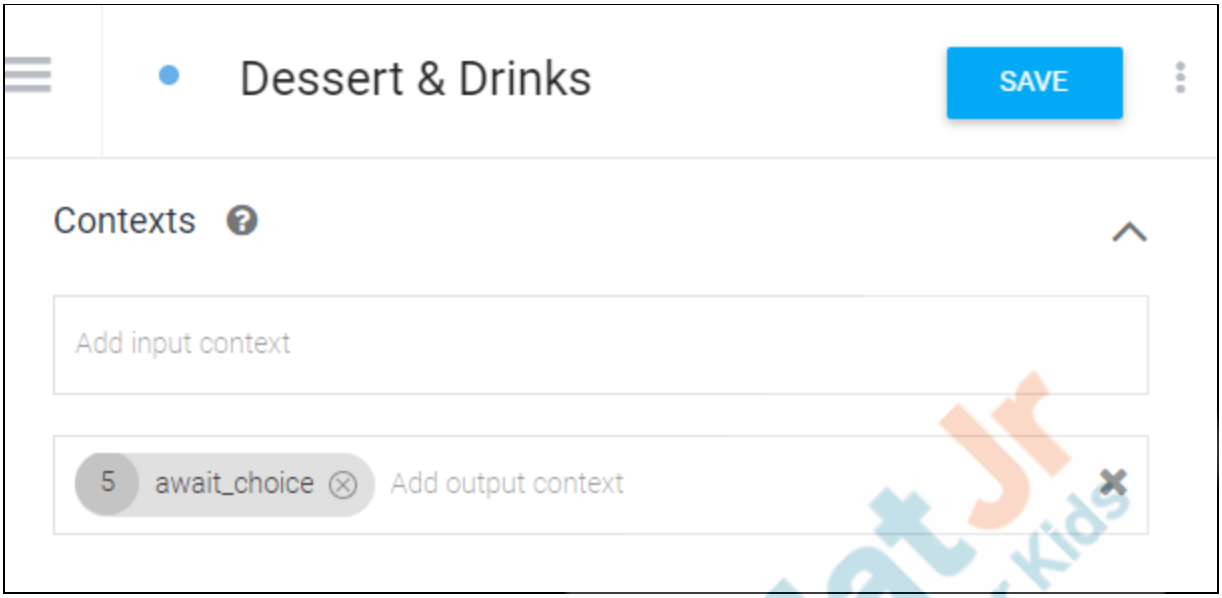
4. Now, head to the **Responses** section. Create a custom response under the **DEFAULT** response.



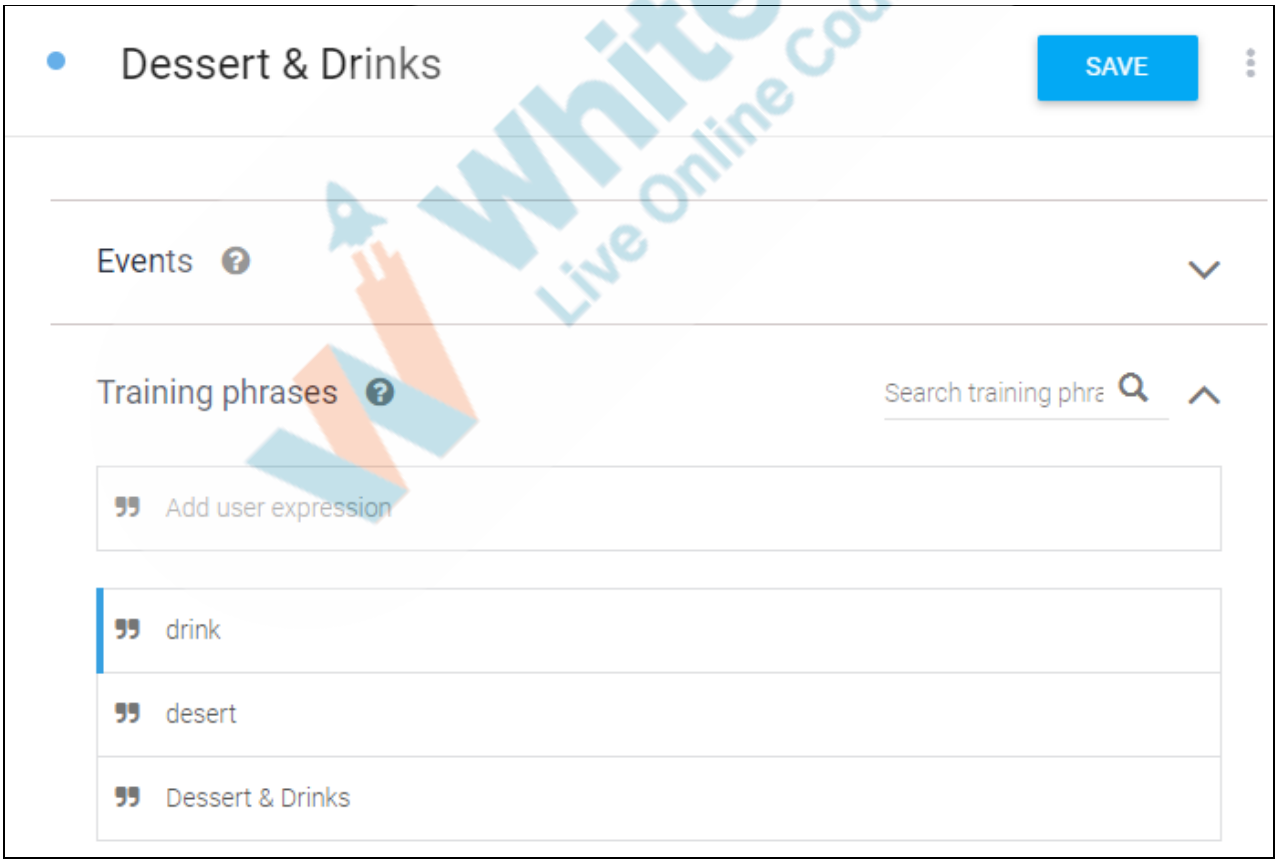
***Note**: We can use a number instead of *.*

	Great! We have added our first intent. Let's create our next intent. Can anyone tell me what our next intent will be? *The teacher guides the students to create the Dessert intent.*	**ESRs:** Dessert & Drinks.

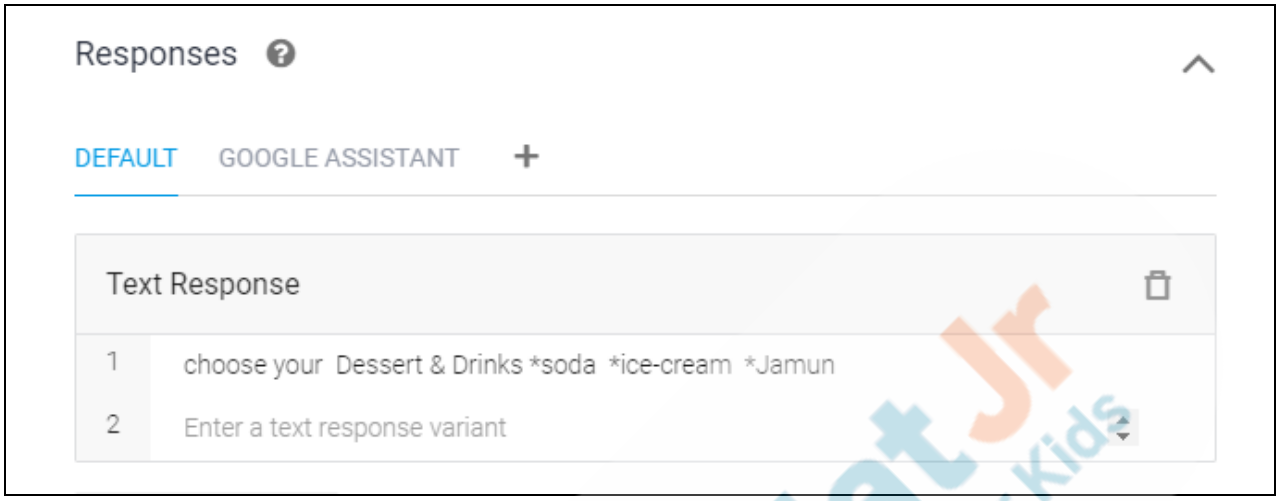
1. Create a new intent named **Dessert & Drinks:**



2. Add the **Training phrases**.



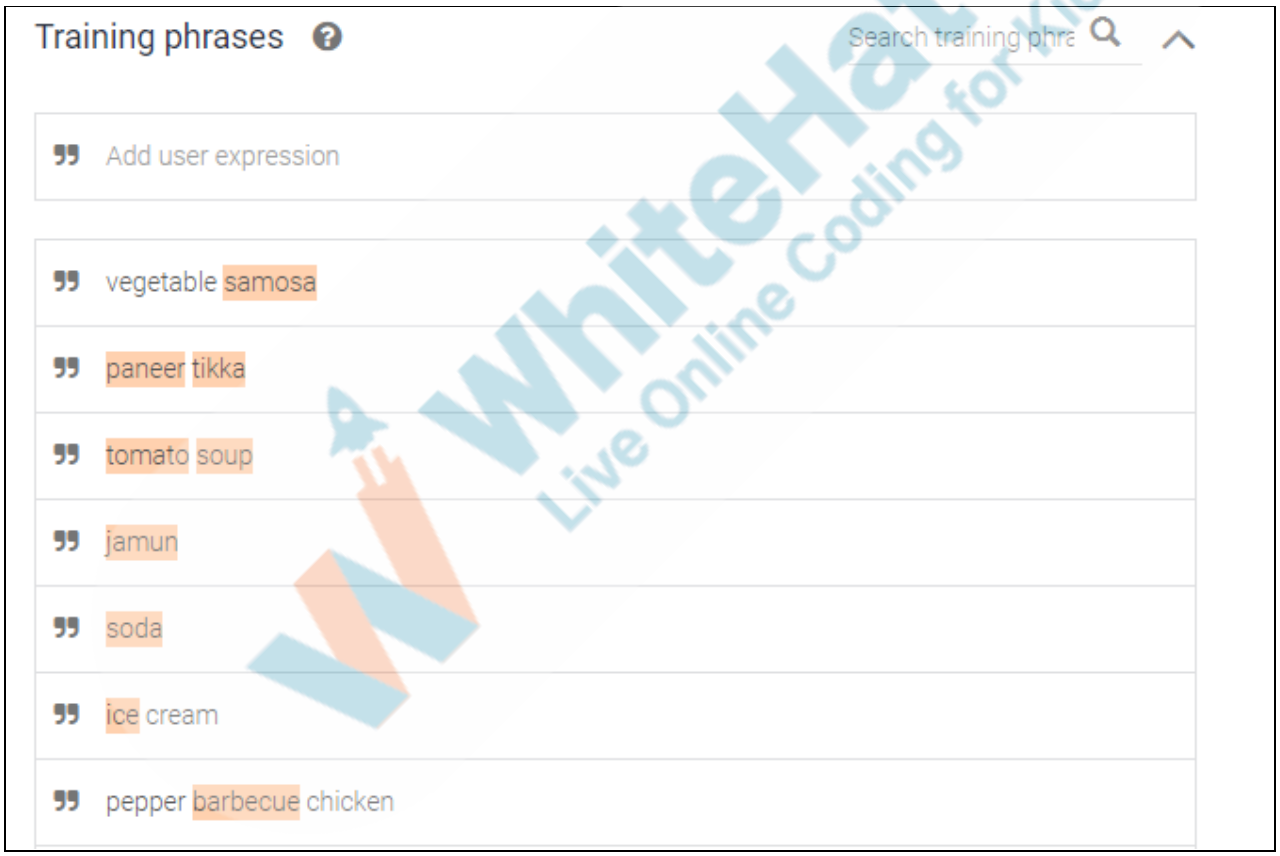
3. Add the **Responses** in the **DEFAULT** section:



	Great! We have created intent for all the menu options. Now when the user for instance selects the **Main course**, it will display the list of options related to the main course. *<Student name 1>*, what do you think the next step will be? Before confirming the user's order we need to ask the user if they might need any other item. Let's learn how we can do this.	**ESRs:** Varied.

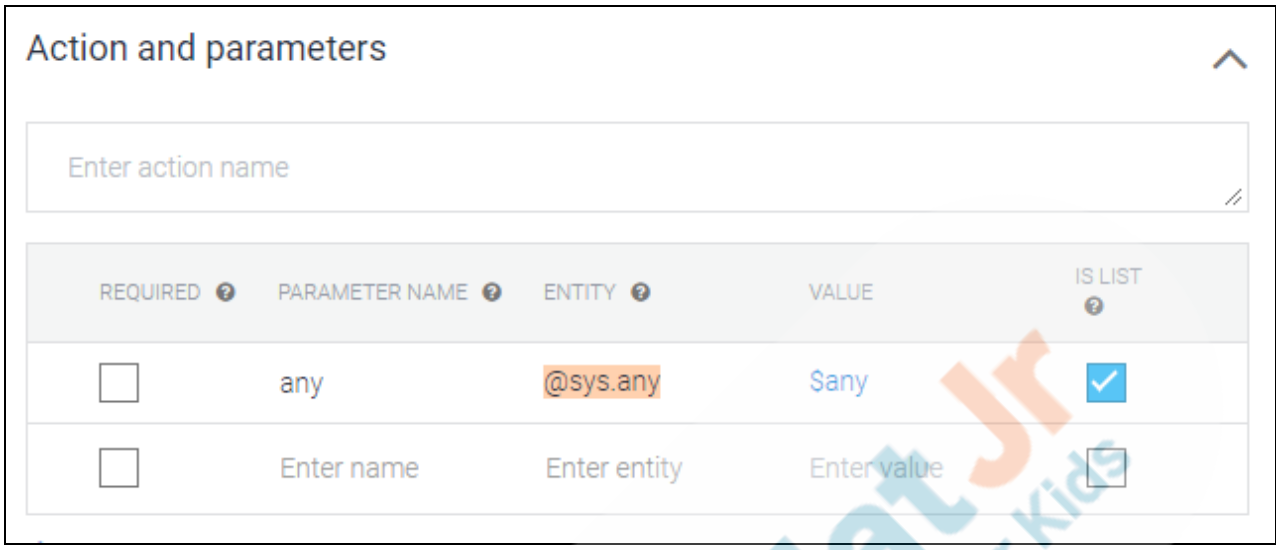
1. Create a new intent as **Item**:

2. Add the **Training phases** for the menu options.



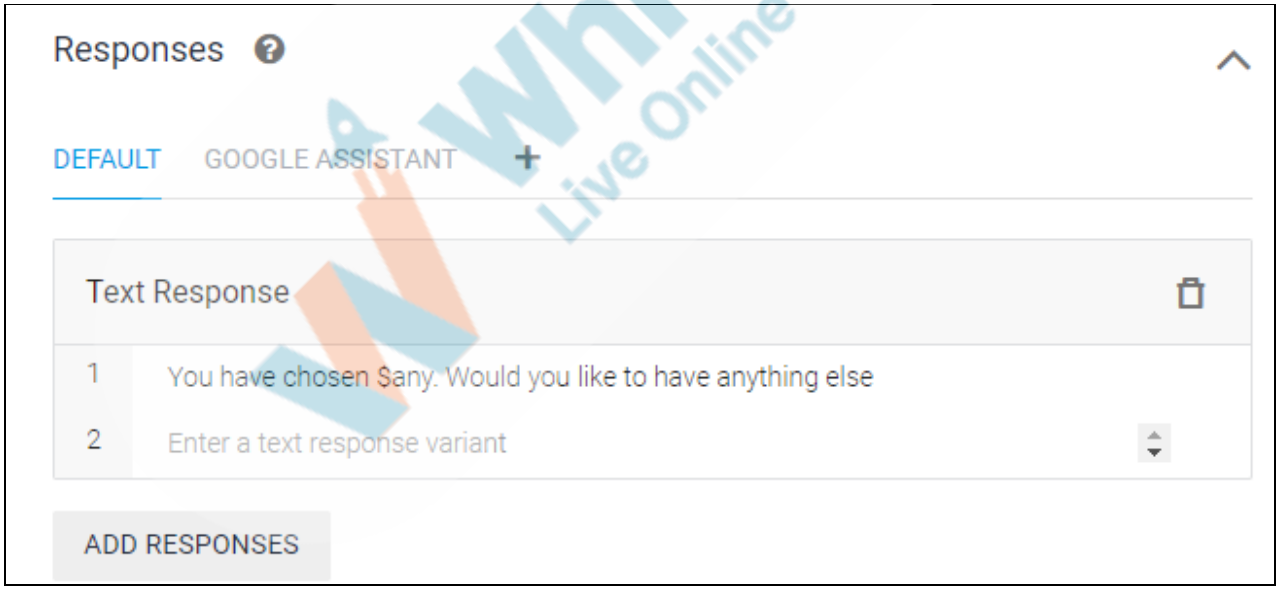
Note: *You need not add all names, as we are adding @sys.any as an action parameter. This would match all the inputs.*

3. Include **Action and parameters** as shown in the following section:



Here **@sys.any** is similar to the **date** action which we covered in previous **Lesson-C45** but in this case, **@sys.any** would match any non-empty input by the user.

4. Add the **DEFAULT** response with the action value:



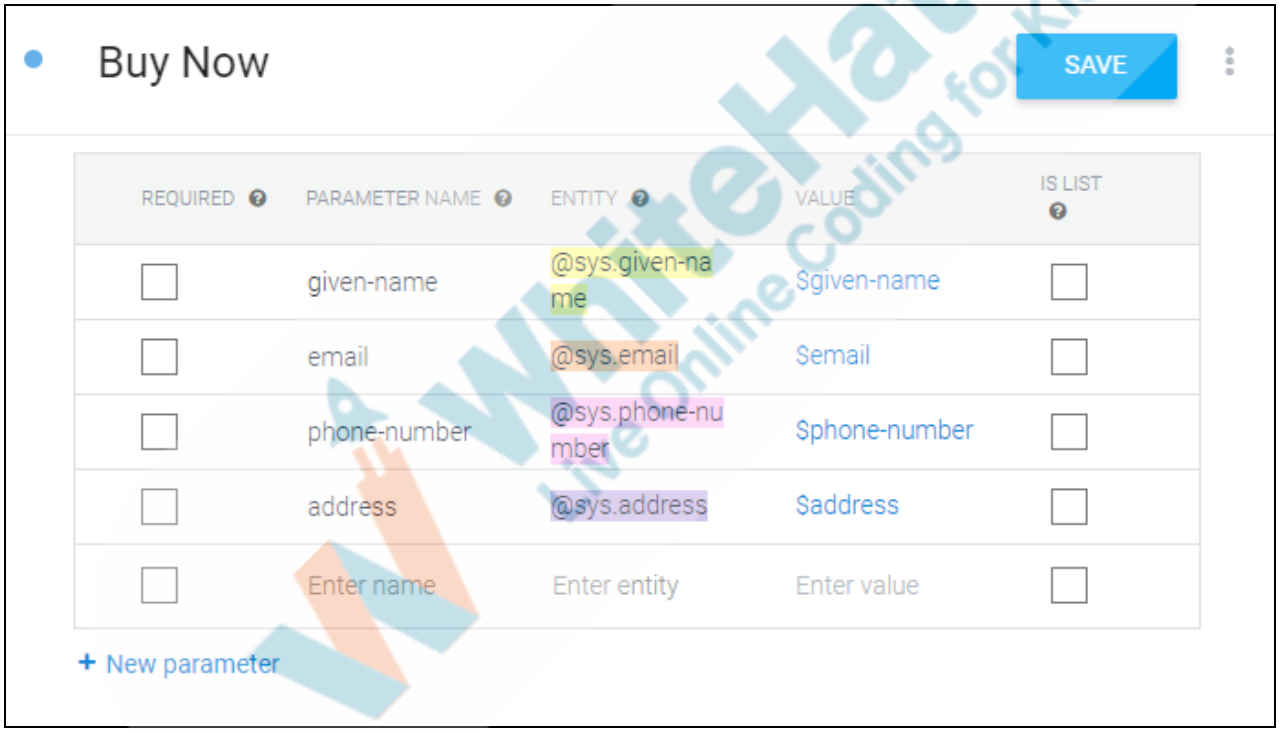
5. Click on **Save**.

	Amazing! Now, when a user is about to check out the cart, a message should display on their screen asking, **You have chosen the $any item.** *<Student name 2>*, what should we do if the user replies with no, he/she does not want to add any more items? The order confirmation message has to be displayed. Let's learn how to do it.	**ESRs:** Varied.

1. Click on new intent and name it as **Buy Now**:

2. Add the **Training phrases** as **no** which would state the user wishes to check out the items:

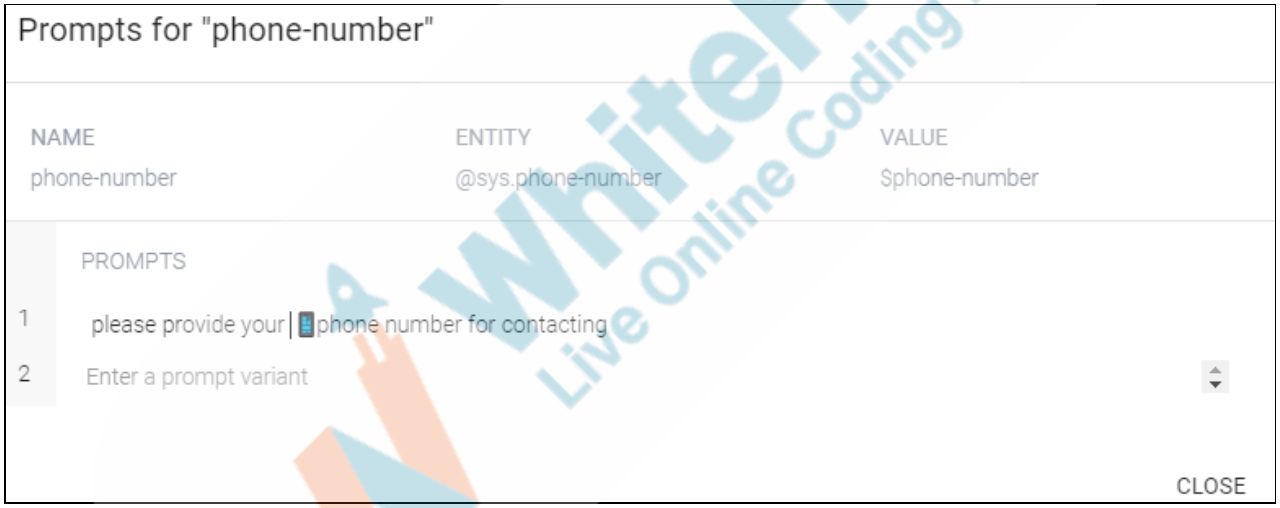
3. Include the action and parameters for the user's name, phone number, address, and e-mail:



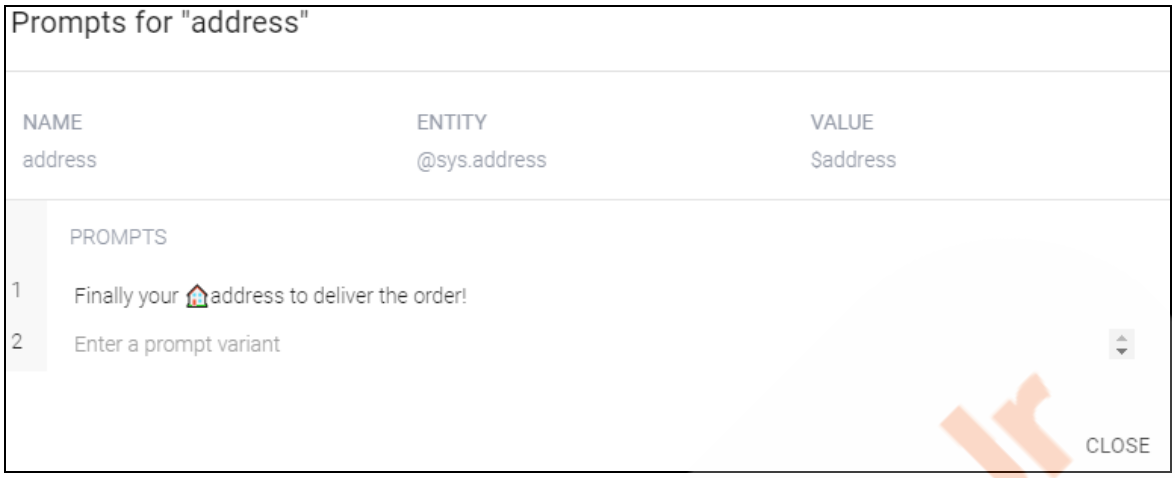
4. Add the prompt message for all the actions by choosing the checkbox. Similar to the **@sys.any**, each parameter has a separate entity.
5. Add the prompt message for the parameters you think it is mandatory for the user to complete.

    - Prompt a message for the username.

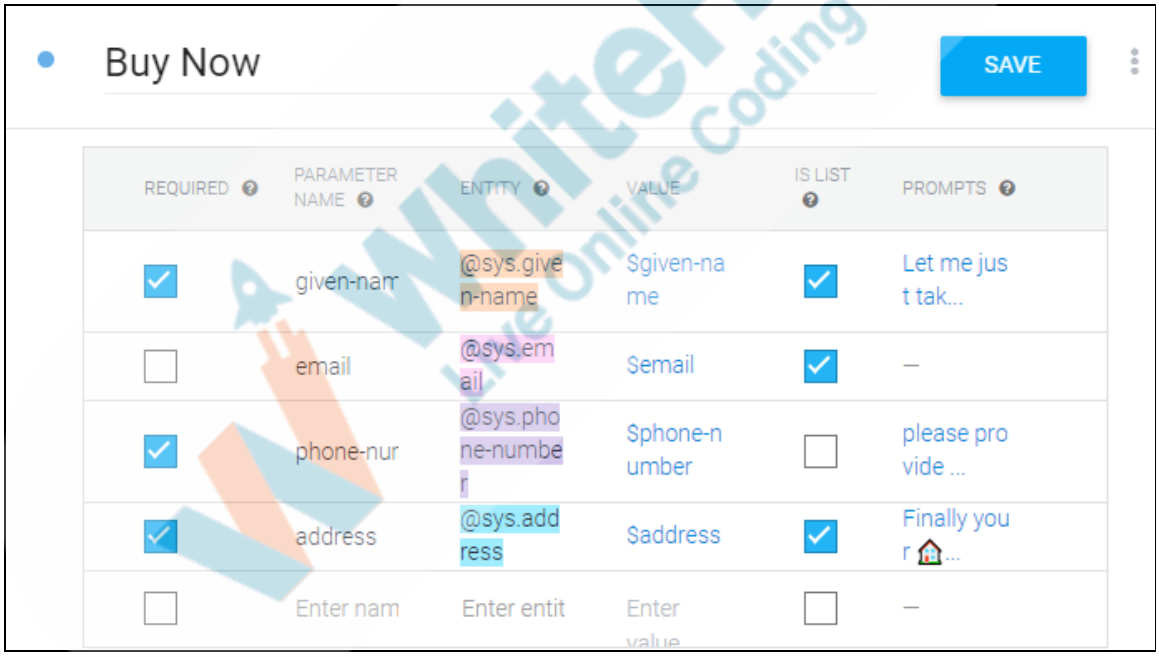
- Similarly, add the prompt message for the user's phone number.



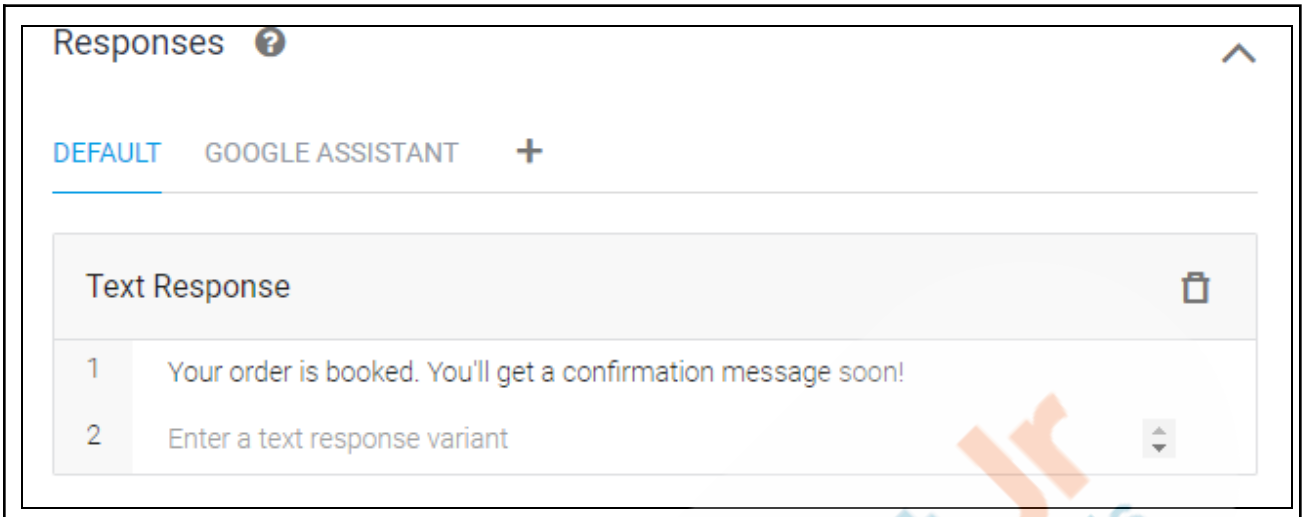
- Add the prompt message for the user's address.



- You can refer to the table below to know which entity is chosen for the prompt to display.

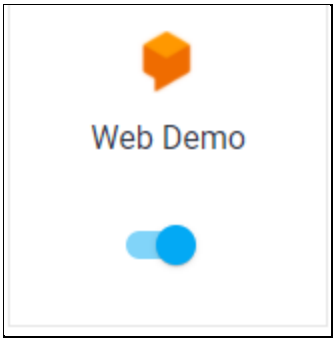


6. Finally, add the **DEFAULT** response confirmation message:

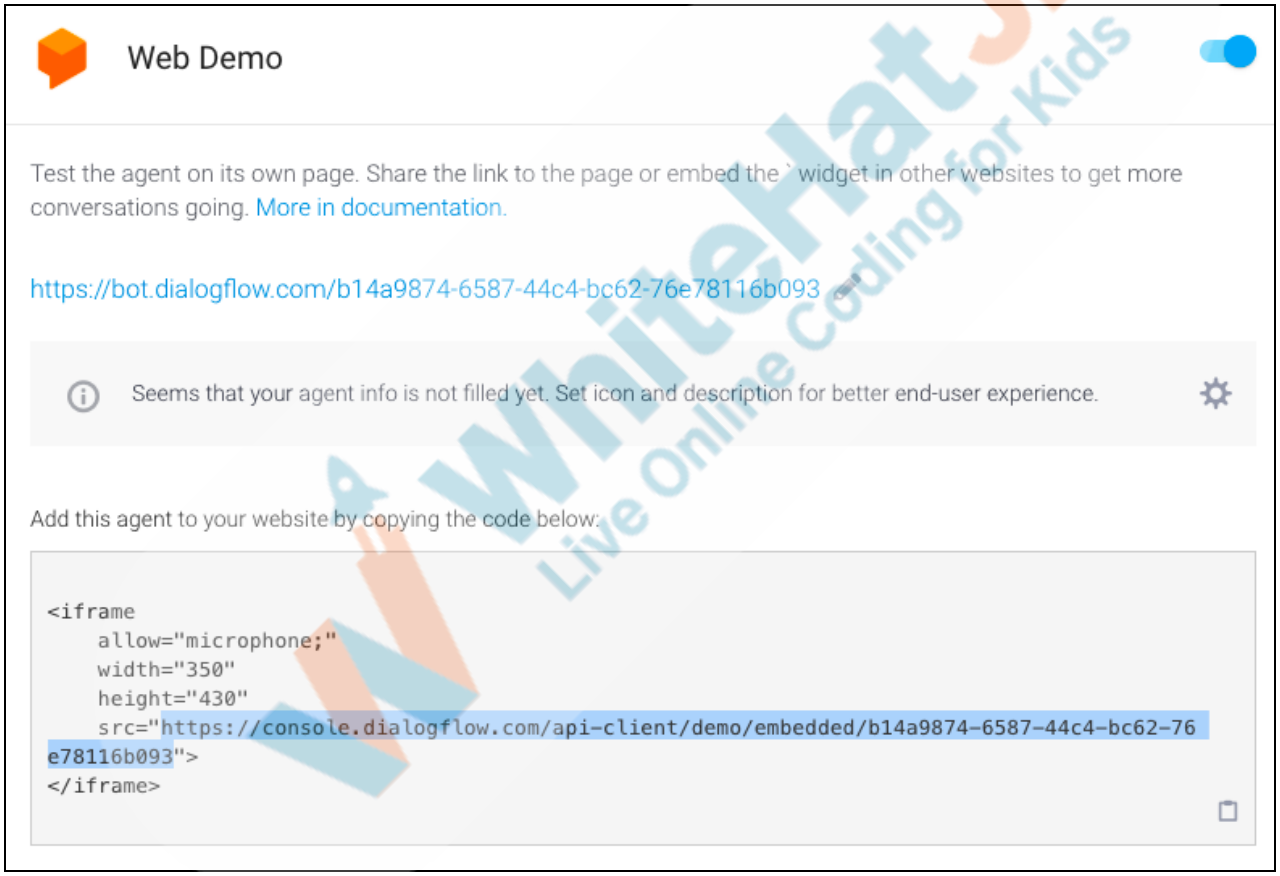


	We have now added the confirmation message. We are at the end of our bot creation. *<Student name1>*, what is the one last thing we need to do with our application? Exactly! Let's now integrate and check our bot.	**ESRs:** Integration.

1. Enable **Web Demo** by clicking on the **Integrations** button on the left-hand side tab:

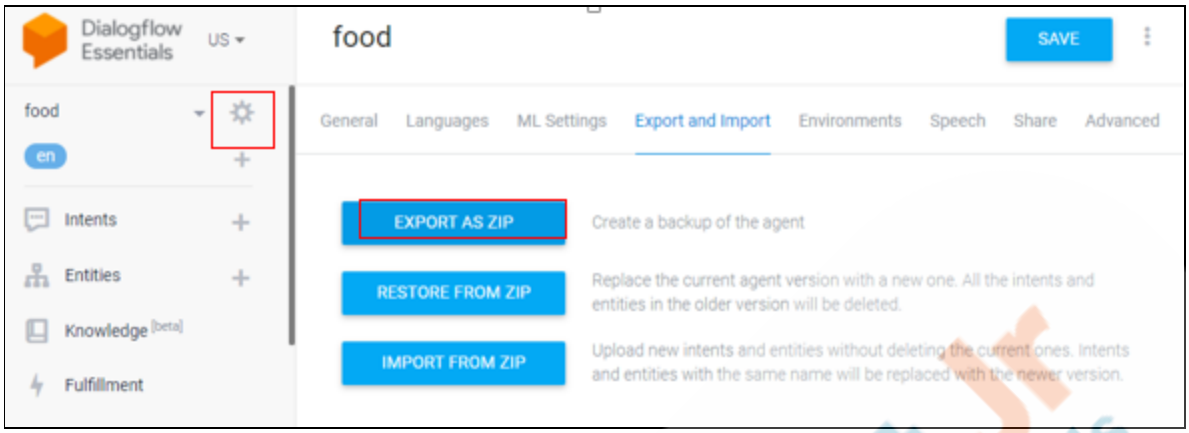


You will get a popup showing the link to our bot:



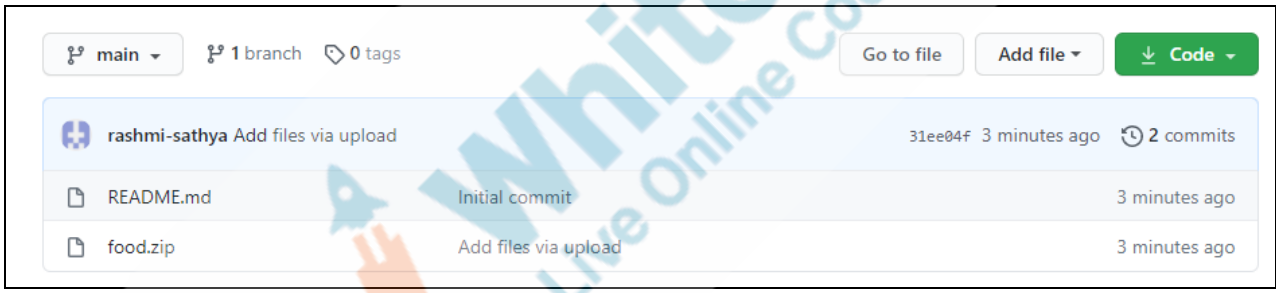
***Note**: We can only test our app. To publish we need the Dialogflow enterprise(paid) edition. To get the URL link for the chatbot, we can have this downloaded as a **zip** file and import it in GitHub.*

2. Choose settings and select **EXPORT AS ZIP**:



***Note**: The zip file will begin downloading. Once you have downloaded the zip file, unzip the folder and upload it in GitHub as we did in **C17 -Setting up a local environment and hosting game online**).*

3. Upload the downloaded zip file to GitHub.



Wow! Great!

Now, we have learned to develop our chatbot and add features to the chatbot by creating the entities, knowledge base document, adding rich responses, and we also learned to integrate this using the Google Assistant and publish it as APK.

Teacher Guides Students to Stop Screen Share		
FEEDBACK • **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 min)	Have you learned to create your own chatbot?	**ESRs:** Yes!
	<Ask both the students alternatively.>  *Refer to **Teacher Resources** page/slide 3-4 for visual aid* Q1) When do we receive the responses from Dialogflow? Q2) What is the use of **@sys.any**?	**ESRs:** Intents have a built-in response handler that can return responses after it matches with an intent. **ESRs:** Matches any non-empty input.
	You get Hats Off for your excellent work!	*Make sure you have given at least 2 Hats Off during the class for:* 

		 
	Congratulations! You have set a new benchmark. Brace yourself! Your new challenge is ready.	
Project Overview	**YOUR OWN CHATBOT.** **Goal of the Project:** In Class C48, we created a chatbot using Google Assistant. We have learned to use rich responses. In this project, you will apply what you have learned in the class to achieve the following goals. **Main Goal:** • You will be designing your own chatbot. **Additional Goal I** • Integrate the chatbot in Thunkable. **Additional Goal II** • Include rich responses using Google Assistant.	

	Story: Every bot has its own purpose. Create a chatbot that analyzes the user's request and provides relevant intent. In this project, you need to develop a chatbot to create a certain awareness in your neighborhood. I am very excited to see your project solution and I know you both will do really well. Bye Bye!	
Teacher Clicks		

Additional Activities I	*Encourage the students to write reflection notes in their reflection journals using Markdown.*
	
	Use these as guiding questions:
	• What happened today?
	  ○ 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?
		The students use the markdown editor to write their reflections in a reflection journal.

Activity	Activity Name	Links
Teacher Activity 1	Teacher Resource	https://s3-whjr-curriculum-uploads.whjr.online/c51552c2-152f-4473-94e2-cee7ea36cd57.pptx
Teacher Activity 2	Dialogflow	https://dialogflow.cloud.google.com/
Teacher Activity 3	Teacher Reference	https://github.com/rashmi-sathya/C48-food
Student Activity 1	Dialogflow	https://dialogflow.cloud.google.com/