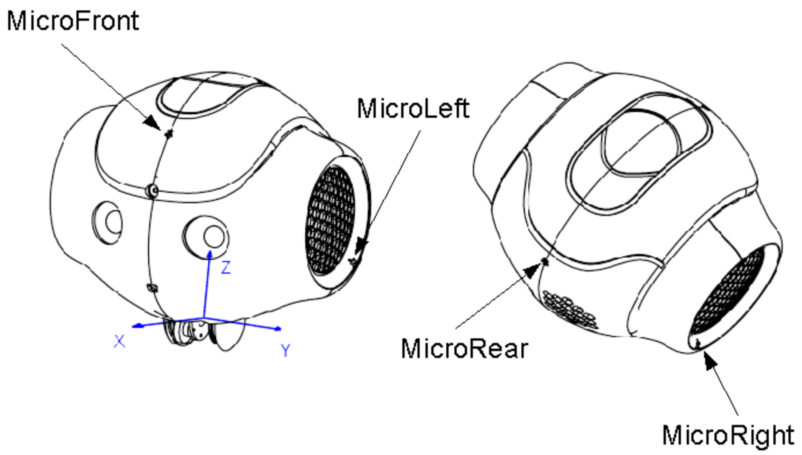
SPEECH RECOGNITION ON THE NAO

Humans frequently communicate through speech. For example, a common greeting when we meet someone is "hi" or "how are you?" We process speech automatically, and understand the meaning of the words we hear nearly instantaneously. On a robot, this process is more involved. The NAO humanoid robot has microphones on its head, which it uses to listen to sounds around it.



However, unlike our ears that listen for sounds all the time, the NAO has to be programmed to listen for sounds at specific times. After it hears human speech, the NAO performs speech recognition with an algorithm to convert what it hears into words that it knows.

To do so, the NAO requires a library of words that it expects to hear. For example, the library can contain two words, "yes" and "no". When the NAO processes the sounds it hears, it will classify it as either "yes", "no", or neither of the two. You may have had experience with a similar system when using automated phone services or voice control on your cell phone, where you are given a list of options that you can speak to select.

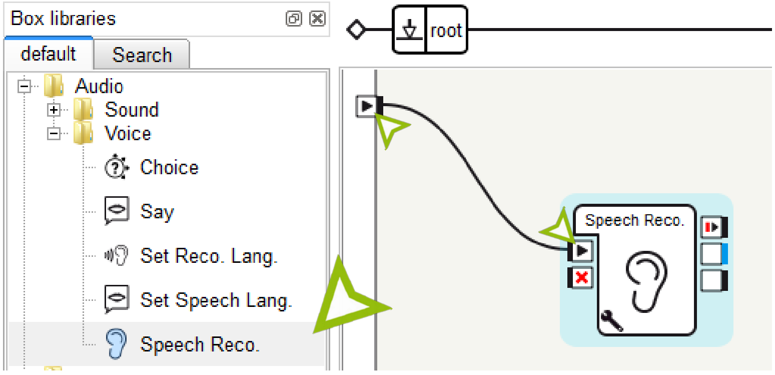
Once a word is recognized, the NAO can then be programmed to react in different ways. After hearing "yes", the NAO could reply with "I am happy" and after hearing "no", the NAO could say "I am sad". If the NAO doesn't understand the words (it did not sound like "yes" or "no") then the NAO could reply "I don't know." This is called a *conditional* in computer or robot programming, and we will go into more detail in the tasks below.

BASIC TASK
SPEECH RECOGNITION

IN THIS LESSON, WE'LL LEARN HOW TO USE SPEECH RECOGNITION ON THE NAO.
WE WILL PROGRAM THE NAO TO RECOGNIZE ITS OWN NAME AND TO GIVE A GREETING IN RESPONSE.

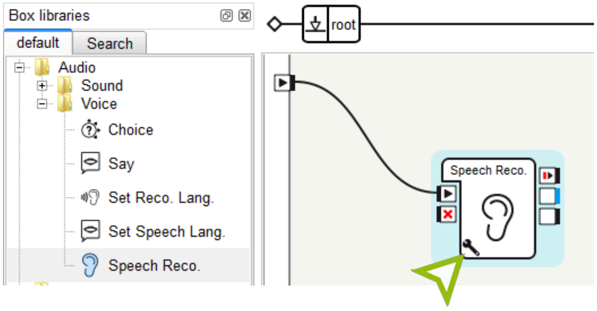
01/

First, drag a Speech Recognition box (found in Audio → Voice) to the workspace and link it to the start arrow.



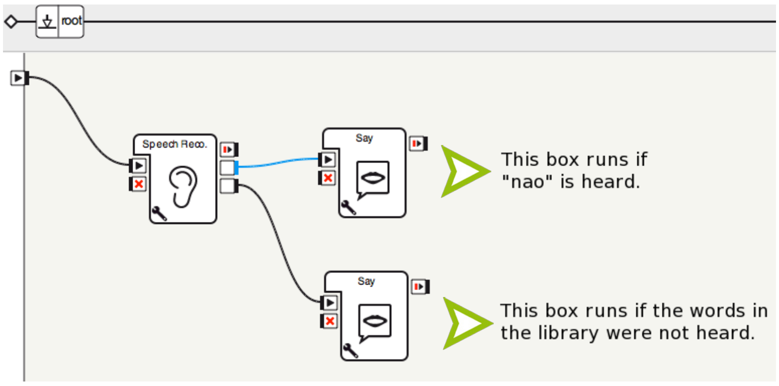
02/

Click on the wrench to examine the parameters for speech recognition. Set the word list to "nao". This defines the library of known words the NAO will listen for. Putting "nao" in the library will make the NAO listen for its own name. Set the threshold slider bar to 10%. The threshold controls how similar the sound has to be for the NAO to recognize it; we will discuss this in more detail later.



03/

Next, create two Say boxes and connect one to each of the two bottom right connectors on the speech recognition box. The top connector is triggered when a word you entered in the word list is heard, and the bottom box is triggered when it is not. This construct is called a *conditional*.

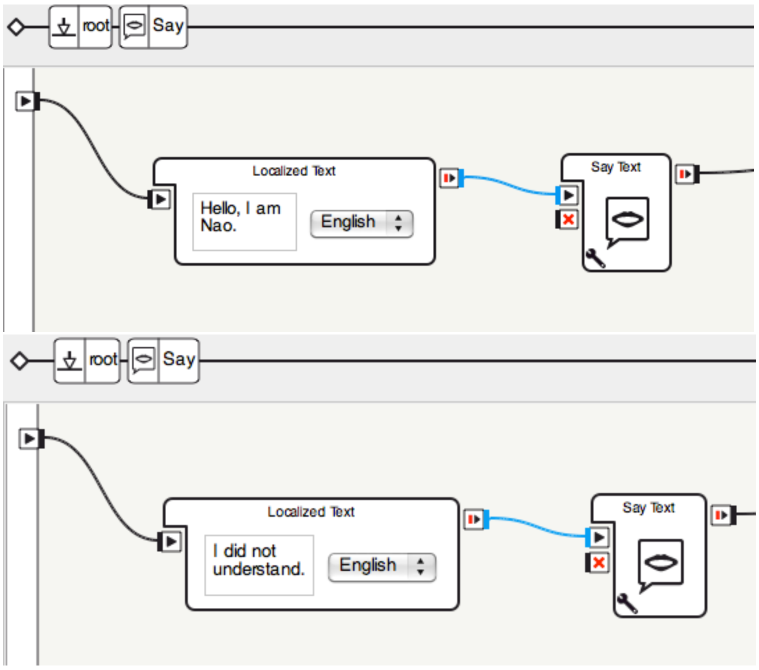


This box runs if "nao" is heard.

This box runs if the words in the library were not heard.

04/

Double click on each of the Say boxes to configure what the NAO will say. The top connector triggers when the robot hears the word “nao”. Change this Say box to say a greeting such as “Hello, I am Your Robot.” Configure the second box to make the NAO say that it didn’t understand.

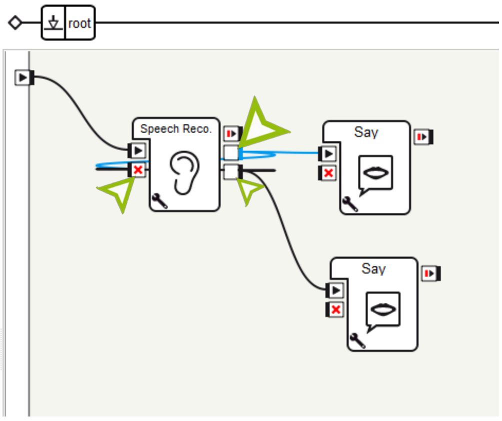


Hello, I am Nao.
English

I did not understand.
English

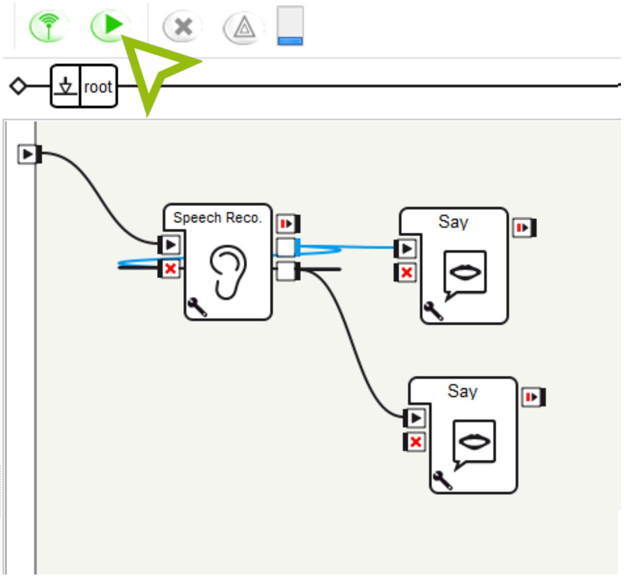
05/

Finally, connect both the boxes on the lower right of the Speech Recognition box to the X mark on the left side of the same box. This will make the robot stop listening for new words after hearing the first one.



06/

Now, hit the play button on Choregraphe. You will hear a sound on the NAO that indicates that it is listening. Its eyes will turn blue in color as well. Once the NAO hears a human speak, its eyes will turn yellow. If it understood the words it heard, the eyes will flash green, and if it did not understand, the eyes will flash red. When the NAO stops listening on its microphones, it makes another sound through its speakers.

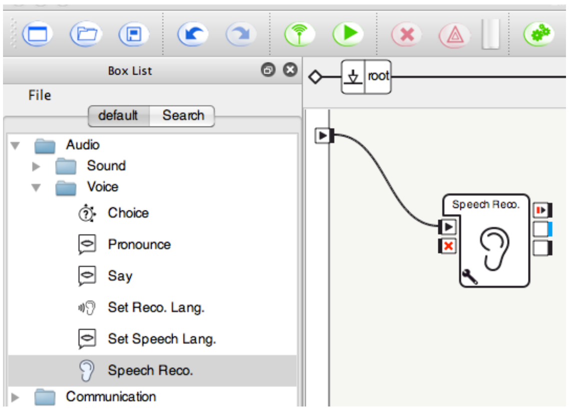


INTERMEDIATE TASK
DISTINGUISHING MULTIPLE NAMES

IN THIS NEXT EXERCISE, WE WILL PROGRAM THE ROBOT TO LISTEN FOR DIFFERENT NAMES AND RESPOND DIFFERENTLY AFTER HEARING EACH ONE.

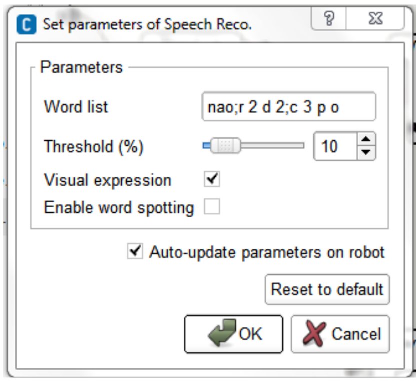
01/

Drag a Speech Recognition box to the workspace.



02/

Click the wrench to configure the word list. Set the word list to “nao;r 2 d 2;c 3 p o”. The semicolons separate different words in the library. Do not forget to add the spaces. Why are these spaces necessary? Once again, set the threshold to 10%.



Set parameters of Speech Reco.

Parameters

Word list: nao;r 2 d 2;c 3 p o

Threshold (%): 10

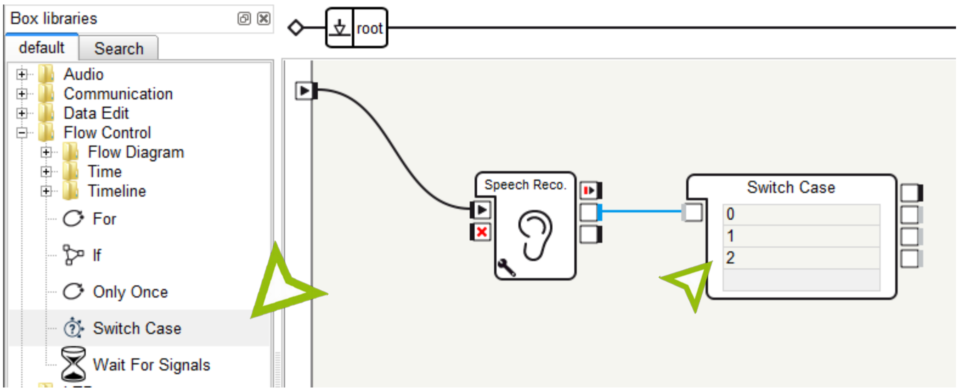
Visual expression ☑

Enable word spotting □

☑ Auto-update parameters on robot

03/

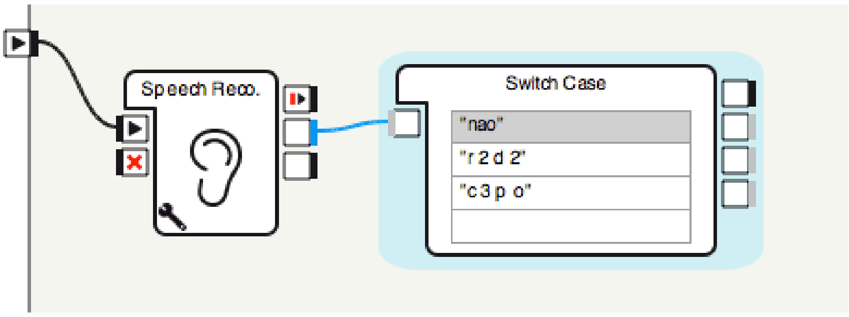
Next, add a Switch Case box. This is found in the list of flow control tools. The Switch Case box compares its input to a list of predefined values, and triggers one of the boxes to the right based on its input.



04/

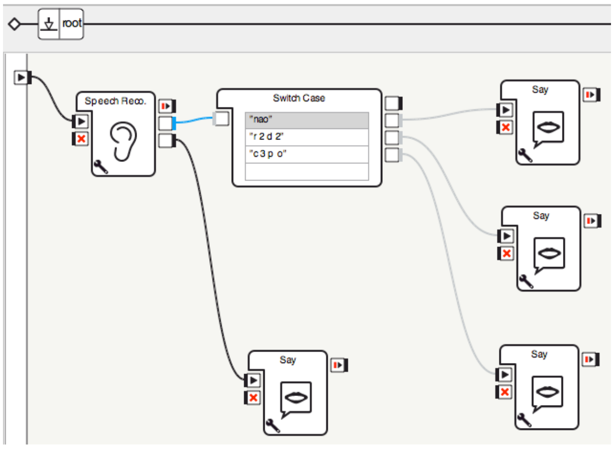
Change the predefined inputs of the Switch Case box to match the words in the speech recognition library. You need to add quotation marks around each word to indicate that it is a string, e.g., “nao”. Note also that the capitalization should be the same as in the speech recognition library—the strings must match exactly. When a word in the speech recognition library is heard, the Speech Recognition box sends that word to the Switch Case box.

The Switch Case box then compares its input with each of the cases (“nao”, “r 2 d 2”, and “c 3 p o”) and triggers the appropriate box to the right of the matching word.



05/

Now add four Say boxes, and connect the boxes as shown below.

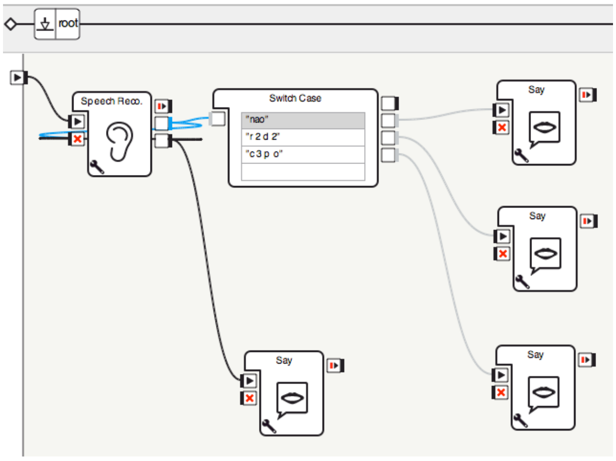


06/

Add messages to the Say boxes. The Say box connected to the Speech Recognition box will be what the robot says if it doesn't understand the human. The three boxes connected to the Switch Case box will be the robot's responses when it hears the corresponding messages. For example, the Say box on the top right will be triggered when "nao" is heard.

07/

Finally, connect the two bottom right boxes on the Speech Recognition box to the X mark on the same box, as we did in the previous exercise.



08/

Press play and try saying the different names.

09/

You may notice that the speech recognition is not perfect. For example, the robot may hear “c 3 p o” when you said “r 2 d 2”. Or, the robot may not understand what you said, even though you said “nao”. The level of recognition can be adjusted via the threshold in the Speech Recognition box.

Recall that we set the threshold to 10% in the Speech Recognition box. This means that the robot has to only be 10% sure of what it hears to recognize the word. Why not set the threshold to a much higher number like 90% then? If the threshold is too high, then the robot may not understand the words you say because it is unsure.

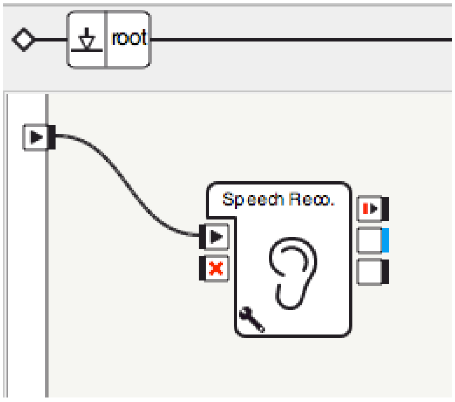
Try changing the value of the threshold, to get a better idea of what it does and how it affects the speech recognition. What value works best?

ADVANCED TASK
SELF-INTRODUCTIONS

IN THIS EXERCISE, WE WILL CREATE A BOX USING PYTHON THAT INTRODUCES THE NAO BASED ON WHICHEVER NAME A HUMAN SAYS.

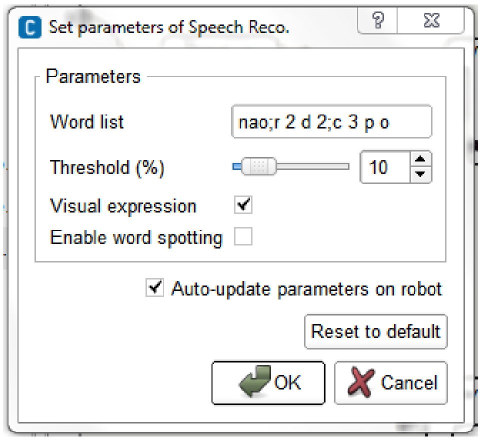
01/

Create a Speech Recognition box and connect it to the start arrow.



02/

Set the word list and threshold as in the previous exercises.



Set parameters of Speech Reco.

Parameters

Word list: nao;r 2 d 2;c 3 p o

Threshold (%): 10

Visual expression ☑

Enable word spotting □

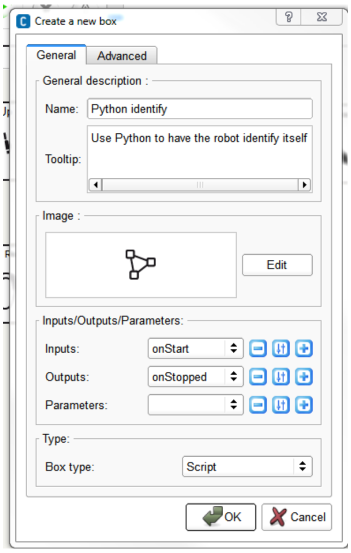
☑ Auto-update parameters on robot

03/

Create a new box

04/

(right click, choose “Add a New Box”). Choose an appropriate (for example Speech box) name, tooltip and image. In the Inputs / Outputs / Parameters section, click the center button to the right of “Inputs: onStart”, the first line. This allows us to set the properties of the onStart input.



Create a new box

General | Advanced

General description :

Name: Python identify

Tooltip: Use Python to have the robot identify itself

Image :

Inputs/Outputs/Parameters:

Inputs: onStart

Outputs: onStopped

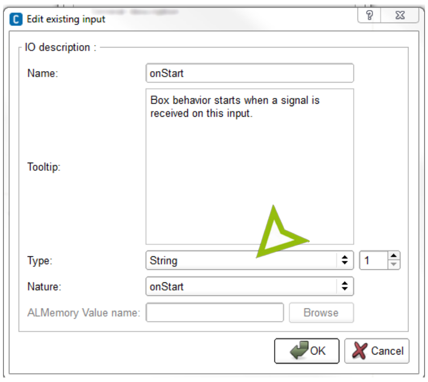
Parameters:

Type:

Box type: Script

05/

A dialog box appears. Change the type to string, and click OK. This makes the input to the box take a string (a sequence of characters).



Edit existing input

IO description :

Name: onStart

Tooltip: Box behavior starts when a signal is received on this input.

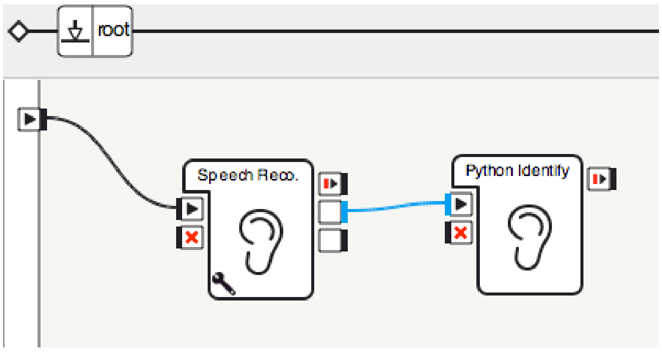
Type: String 1

Nature: onStart

ALMemory Value name:

06/

The Speech Recognition box outputs a string, the word that has been recognized. Now we can link the two boxes together since the `onStart input` accepts a string.



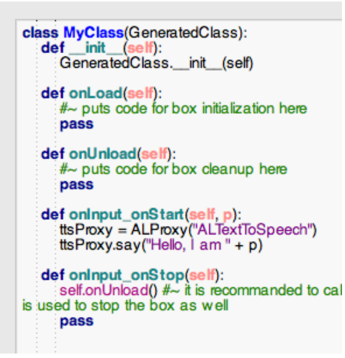
07/

Double-click on the new box to edit the Python source code. Add the following two lines to the onInput_onStart method.

```
ttsProxy = ALProxy(“ALTextToSpeech”)
ttsProxy.say(“Hello, I am ” + p)
```

In the earlier exercises, we created an ALTextToSpeech proxy and called the say method. What’s new is the addition of `“Hello, I am ” + p`. This is called string concatenation, appending two strings together. The `p` is a parameter to the `onInput_onStart` function. This parameter is set to the value arriving at the onStart input, in this case sent by the Speech Recognition box.

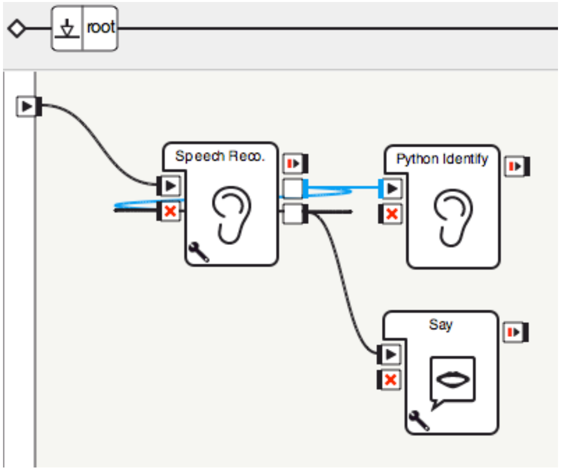
For example, if the Speech Recognition box heard “nao”, then the expression `“Hello, I am ” + p` would evaluate to “Hello, I am nao”.



```
class MyClass(GeneratedClass):
    def __init__(self):
        GeneratedClass.__init__(self)

    def onLoad(self):
        #~ puts code for box initialization here
        pass

    def onUnload(self):
        #~ puts code for box cleanup here
        pass

    def onInput_onStart(self, p):
        ttsProxy = ALProxy("ALTextToSpeech")
        ttsProxy.say("Hello, I am " + p)

    def onInput_onStop(self):
        self.onUnload() #~ it is recommanded to cal
is used to stop the box as well
        pass
```

08/

Next, add a Say box which says “I did not understand,” and connect it to the bottom output of the Speech Recognition box. Link the two bottom right outputs of the Speech Recognition box to the X mark on the same box, as we have done before.



09/

Hit play and try speaking the different names.

ADVANCED TASK
SPECIALIZED INTRODUCTIONS WITH *IF* STATEMENTS

UNTIL NOW, WE HAVE ALWAYS EXECUTED
THE SAME CODE IN EACH PROGRAM.
BUT WE CAN ALSO USE *CONDITIONALS* TO EXECUTE
CODE ONLY IF SOME CONDITION IS SATISFIED.
WE WILL MODIFY THE PREVIOUS EXERCISE TO GIVE
SPECIALIZED GREETINGS FOR EACH ROBOT NAME.

01/

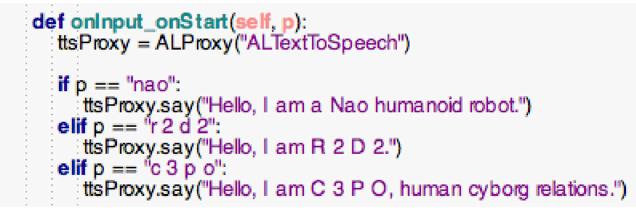
Begin with the result of the previous exercise.

02/

Double click on the custom box to edit the Python source code.

03/

Enter the code shown below in the `onInput_onStart` method.



```
def onInput_onStart(self, p):
    ttsProxy = ALProxy("ALTextToSpeech")

    if p == "nao":
        ttsProxy.say("Hello, I am a Nao humanoid robot.")
    elif p == "r 2 d 2":
        ttsProxy.say("Hello, I am R 2 D 2.")
    elif p == "c 3 p o":
        ttsProxy.say("Hello, I am C 3 P O, human cyborg relations.")
```

What does this code do? You should already be familiar with the ALTextToSpeech proxy and its `say` method. The `if` statement checks if the following condition is satisfied, and if so, executes the code after the colon. The `p == “nao”` condition is satisfied if and only if the string `p` is “nao”. The two `elifs` (short for “else if”) are the same as the `if` statement, except they are executed only if the previous `if` statement was not satisfied (this is the “else” part). So, this code speaks a specific message based on the output of the speech recognition box.

04/

Run the program and try speaking the different names.