synthiam.com

# Pandorabot tutorial

This tutorial will cover the different uses with a Pandorabot chatbot and EZ-Builder. Pandorabots is a great option for having dynamic speech synthesis conversations with your robot, and you can tie in it's responses with Cheat Sheet commands to control sensors, media files, servos ect, all from the EZ-Builder Pandorabot control.
You can customize the Pandorabot with different and new responses, and there is quite a bit of documentation online describing how to use certain functions, such as using the  tag for variable response for different user names, and using  tags to keep conversations on topic, some of which will be covered in this tutorial.

Last Updated: 10/16/2015

# Step 1. Explaining the EZ-Robot Pandorabot Control.

Below is an explanation of the Pandorabot control option.
**Pause:**

When ticked, this pauses the voice recognition, and stops the control from hearing you.
**Configuration Menu Tab:**

Opens the control configuration menu.
**Speech Recognition Visualizer:**

Allows you to see a visual representation of your speech volume levels. The volume input wave line is red when control is paused, and green when unpaused.
**Text Input Field:**

Talk to your Pandorabot using your keyboard.
**Debug Window:**

A text display window that shows speech recognition results, the bots response, confidence levels, and possible control errors.
Below is an explanation of the Pandorabot configuration menu.
**Pandorabot Bot ID:**

Input field to add your Pandorabots unique Bot ID.
**Username:**

Your computers User Name.
**Response variable:**

The Pandorabots response variable that can be used in a script.
**Response Script:**

Insert a script that will run when the Pandorabot responds.
**Speak Response:**

When ticked, the Pandorabots response will be heard using speech synthesis.
**Confidence:**

A drop down menu that contains different values for adjusting the ability of the computer to better understand what is being said by the user.
**Show Unconfident Recognition:**

When ticked, any unconfident speech recognition phrases spoken by a user, will be displayed in the debug window.
**Use EZ-B Speaker**

When unticked, the Pandorabots verbal response will be heard through the computers speakers. When ticked, the response will be heard through the EZ-B v4's speaker.
**Save** and **Cancel:**

"Save" will save any recent setting changes made, and "Cancel" will cancel any recent changes made. Both options will close the configuration menu.

# Step 2. Creating a Pandorabot.

To get started, you will need to create your very own Pandorabot using their website. You can choose one which already has content so you can start a conversation straight away, or you can choose one with no content so you can build your bot from the ground up. Setting up your own Pandorabot is free to do (at the time this tutorial was written), and this step will explain how to do just that.
If you use the link below, you can sign up to a Pandorabot account and start creating a new bot.
Pandorabot sign up
**1.)** After clicking on the link above, on the sign up home page, click on "**Account Sign Up**".

**2.)** Enter the relevant information in to the text input fields, then Click on "**Submit Account Request**".

**3.)** You will now see a message saying that no Pandorabots have been defined yet. Click on the "**Create**" link in this message to create your new bot.

**4.)** Choose what content you want your new bot to start off with the categories are...
*Standard AIML - Sept 2001*
*Dr Wallace's A.L.I.C.E - March 2002*
*Annotated A.L.I.C.E. AIML - September 2003*
*Christian Drossmann's standalone German AIML - January 2002*
*Sandro Pons' Italian AIML set - April 2001*
and *No initial content, the Pandorabot starts with no knowledge*.

In this tutorial, we will use a bot with "*No Initial Content*".
**Easy way to train your bot.**
**1.)** To give your new bot the ability to hold a conversation, the more are a couple of options you can use. One of the ways is to use the "Train" interface. This is the easiest way to add or edit knowledge to your bot. To start, click on the "**Train**" tab.

**2.)** The text input field where it says "Human" is where you write what you want to say to the bot. Click on the field and write Hello", then press "Enter on your keyboard, or click on "**Ask**".

**3.)** When training a bot with no initial content, you will get the bots response "I have no answer for that" which is a default response. In the bots text input field, write "Hello. Nice to meet you.", then click "**Say instead**".

**4.)** Click on the "**Ask Again**" (which says your "Hello" response again) and the bot will now respond with it's new "Hello. Nice to meet you" response. As mentioned, this is the easiest way to train your bot, but more time consuming.

**Advanced way to train your bot.**
**1.)** Another way to train your bot, is by adding or editing it's AIML files directly. This is a more advanced option, but once you get the hang of it, it is much quicker to train you bot. To start with a simple example with the bot with no initial content, click on the "**AIML**" tab.

**2.)** Now click on the link that says "*Create a new AIML file*".

**3.)** You will be presented with an AIML script editor. In this editor the will already be a body of code that will run your AIML categories. The AIML categories  you wish to add (it could be

one, or a hundred) need to go where it says **[/b] (this message can be written over or deleted).**

[b]4.) An AIML category is made up of the following...
**Cetegory:** separates the  category  from others in your bots AIML files.
**Pattern:** is what you say.
**Template:** is the robots response.
Once you have added your new category, it should look like the screen shot below. When your category is finished, give your new AIML file a name and click "**Save As**". (NOTE: If you Aiml category contains any errors, you will receive a red error message. Carefully check over your code again and correct any errors).

**5.)** When the file is successfully saved, you will now see it listed in an AIML file list, like below...

**6.)** To talk to your new Pandorabot with EZ Builder, you first need to publish your bot. Click on your bots name tab, then click on the "**Publish**" link highlighted below...

**7.)** One final thing you need to do, is make a note of your new Pandorabots **Bot ID**. This will be needed to set up your bot in EZ Builder, which will be explained in step 4. The bot ID can be found in the URL on the publish screen which can be seen highlighted in "blue" in the screen shot, above.
That covers the basics of setting up a new Pandorabot and giving your new bot knowledge. There is a lot more extensive information that covers how to train your bot in the following links which will help.
www.pandorabots.com/pandora/pics/aimless/tutorial.htm
www.pandorabots.com/botmaster/en/tutorial?ch=1

# Step 4. Setting your Pandorabot up in EZ-Builder.

**1.)** Open up EZ-Builder on your computer, and click on "**Project**", then "**Audio**", then click on "**Pandorabot**" to add this control to your project.

**2.)** Click on the gear icon to open the Pandorabot control configuration menu.

**3.)** click on the text input field called "**Pandorabot Bot ID**". You need to enter the ID that is contained in your Pandorabots URL, like what is outlined in red along with your username seen in the following photo, and click "**Save**"...

The bot ID can be found in your browser on your bots training page, or on the bots "Publish" page like in the example photo above. It's not the whole Pandorabots URL you need, just the last set of numbers (also seen below).

**4.)** Now you can run a simple test by one of two ways. Typing in what you want to say to your bot in the text input field, or by connecting a good quality microphone or headset to your computer, then unpause the speech recognition check box, and speak your phrase. You should now get a response.
The following few steps will guide you through different examples of how to use scripts with your Pandorabots responses.

# Step 5. Ways of Using EZ-Script with Pandorabots.

In the next few steps of this tutorial, you will be guided through the various ways to use Pandorabot with EZ-Builder. This tutorial assumes that you are now familiar with Pandorabots, and assumes you have your own Pandorabot published with AIML brain files that your bot already uses.
Adding a simple EZ-Script.
Here, we will be writing a simple line of EZ-Script in to a Pandorabot category in the AIML bot brain files. This will make your robot say something, then use a single line of EZ-Script to carry out an action like moving a servo. Let's begin.
To add a script to a Pandorabots response, EZ-Builder will execute EZ-Script that is located within the **[** and **]** characters. Below is an example of this where one servo will move...
AIML category with EZ-Script example.

As you can see, the **[Servo(D0,1)]** part of the bots response goes at the end of the sentence. So when you say "*Move servo left*", your bot will say "*I will move my servo to the left.*" and then proceed to move the servo (see video demo below). Moving the **[Servo(D0,1)]** to the beginning of the sentence will not change the order of the bots speech and servo movement. It will still speak first, then move the servo.
You can only use a single line of script in your bots response, as adding multiple lines of script will not work. This could be an issue if you wanted to set different servo speeds for example. But there is a way around this.

# Step 6. Using multi line scripts.

As previously mentioned, using multiple script commands in a single Pandorabot response is not possible. Or is it?
In Fact there is a way to run multiple lines of script or multiple controls quite easily using "Control Commands" or "Cheat Sheet" commands as they are now called. If you're new to EZ Builder, or have had your EZ-B for a while and haven't yet fully explored its full potential and not sure what Cheat Sheet control commands are, these are ways of easily accessing existing scripts, controls, or other functions such as "Soundboards" and "Auto Positions", and puts them in to a single line of script. Basically, a Cheat Sheet is a short cut. This makes programming a robot to execute extensive or simultaneous actions very easy to do, and for this example, that's what we want for the Pandorabot control.
In this example, the servo will move to two positions with a three second sleep in between servo movements.
**1.)** What we do is add an EZ-Script control to an EZ Builder project. Then we will enter the following script...

```
#EZ-Script example.
Servo(D0,1)
Sleep(3000)
servo(D0,90)
```

We will name this new script "Servo Test" and save the settings.
**2.)** Now reopen the script editor, and under the last line of script, right click the mouse button to bring up the cheat sheet commands. Find the "**ControlCommand("servo test", ScriptStart)**. Now click on it to add it to the script, highlight just the control command, and cut it using CTRL and X, or right mouse click. Re-save the project.
**3.)** Now make a new AIML category in your Pandorabot AIML editor and write a new category pasting the control command in the bots template. It should look like the following...
AIML category with multiline EZ-Script example.

Save this AIML category, and republish your Pandorabot.
**4.)** Once published, test the script in EZ Builder. Say "*Move your servo.*" and the bot should respond with "*I will move my servo to the left, then to the right.*", then EZ Builder will run the control command and tell the EZ-B to move the servo, as seen in the next video.
So far, the bot responds and then runs a script. The next step will show you how to have the bot speak while moving a servo at the same time.

# Step 7. Combining speech and movement.

So far, the examples provided have the EZ-B speaking and the carry out actions after it has stopped talking. But there may be times where you may want your robot to talk at the same time as it performs an action.
This is easy to do using a control command like mentioned in level 2, but with a slight difference. Instead of typing in the robots speech response followed by a [ContolCommand] in an AIML category, we only use the [ControlCommand] script, and put the robots spoken response in to an EZ Builder script using the **Say()** or **SayEZB()** script commands, and make a new or edited control command.
The video below shows two examples using this method. The first half shows using a Pandorabot response followed by a script command which doesn't do what we want. The second half shows using only a control command using the **Say("")** script command which executes the way we want it to.
Here is how it is done...
**1.)** Like in the previous step, what we do is add an EZ-Script control to an EZ Builder project. Then we will enter the following script...

```
#EZ-Script example.
Say("I am now moving a servo, while I speak at the same time.")
Servo(D0,1)
servospeed(D0,2)
Sleep(3000)
servo(D0,90)
```

We will name this new script "Speech Servo Test" and save the settings.
**2.)** Now reopen the script editor, and under the last line of script, right click the mouse button to bring up the cheat sheet commands. Find the "**ControlCommand("Speech servo test", ScriptStart)**, click on it to add it to the script, highlight just the control command and cut it using CTRL and X, or right mouse click. Re-save the project.
**3.)** Now make a new AIML category in your Pandorabot AIML editor and write a new category pasting the control command in the bots template. It should look like the following...
AIML category with speech EZ-Script example.

Save this AIML category and republish your Pandorabot. Have you noticed the difference between the AIML example above, and previous examples? There is no actual phrase for the bot to say in between the **Template** tags, only the control command. This is covered by using the **Say("")** command in the EZ-Script.
**4.)** Once published, test the script in EZ Builder. Say "*Move your servo.*", EZ Builder will run the control command and tell the EZ-B to move the servo while saying "*I am now moving a servo, while I speak at the same time.*".

# Step 8. Advanced response and actions.

Some of you may be aware that using "**That**" codes in a Pandorabot category, remembers your bots last response, or more specifically the last sentence of the response, which helps to carry on a conversation and keeps it on topic.
Now, if you put a control command at the end of a bots response as we have done so far, this will be classed as the last sentence spoken by your bot. This can cause problems keeping the conversation on topic and with other responses that uses the same control command at the end of them. But there is a very simple solution to avoid this. Simply put your EZ Script or control command "in front" of the bots response in your bots brain files, instead of at the end of it. Doing this works exactly the same way and avoids any problems using the  code.

If you have a bots response that only contains a script or control command (as mentioned in step 7) then you should avoid using "**That**" codes for these responses.

# Final Step. Now it's your turn.

With what you have learned from this tutorial, give it a go for yourself and find out what is possible using this interface. And why not share your ideas in the Community forum, with descriptions and/or videos of your projects using what you have now learned.
**NB: It should be noted that Pandorabots is a third party software platform, and EZ-Robot has no association with them, and are not responsible for Pandorabots functionality or issues. Any queries regarding Pandorabots should be submitted via their own website.**
But if you have any questions regarding the use of EZ-Script with Pandorabots, then feel free to ask in the forum, as there is always someone around in the community willing to help where possible.
Happy building.
**Tutorial created on 4th October 2015**