

Hello, humans.!

Objectives

In this tutorial, you'll create your first applet that lets **Pepper** say the phrase ‘Hello human!’.

Prerequisite

Make sure you have installed Android Studio, and the Robot SDK and other related tools.

For more details, see: [Installing the Pepper SDK plug-in](#).

Let's create a new project !

- Create a new project **HelloPepper**.
- Robotify This project and confirms the realisation**QiSDK & Robot Life Cycle**.

Setting up and running a Say

To make the robot talk, we will use the Say interface. This class represents one of many actions that can be performed by **Pepper**.

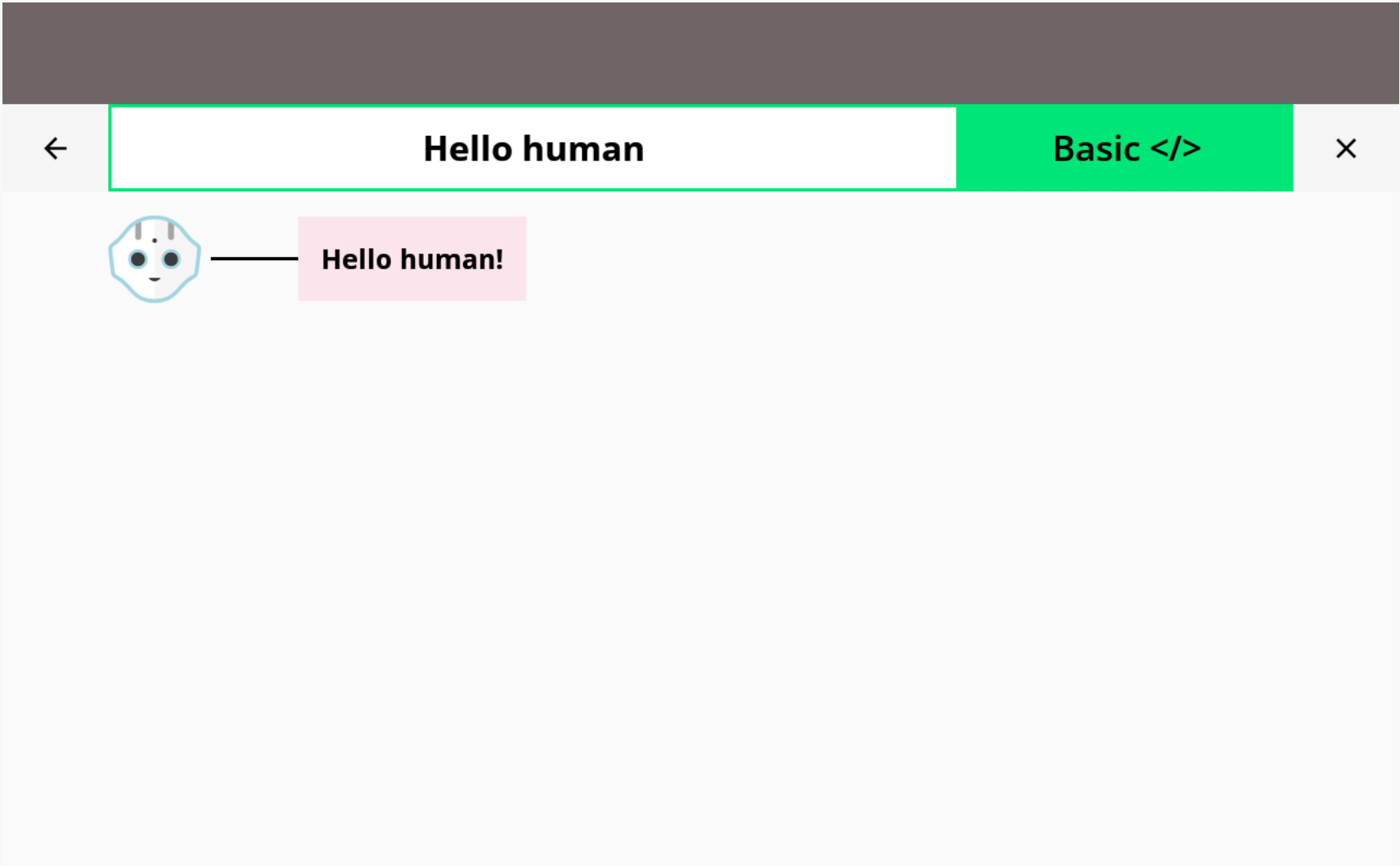
make use of [Say](#) interfaces:

Step	Action
	<div>be in <code>onRobotFocusGained</code> method with <code>SayBuilder</code> Create a new instance of Say:</div> <div><pre>// Create a new say action. Say say = SayBuilder.with(qiContext) // Create the builder with the context. .withText("Hello human!") // Set the text to say. .build(); // Build the say action.</pre></div>
	<div>run Say</div> <div>invocations run methodologies:</div> <div><pre>// Execute the action. say.run();</pre></div> <div>The full code should look something like this.</div> <div><pre>@Override public void onRobotFocusGained(QiContext qiContext) { // Create a new say action. Say say = SayBuilder.with(qiContext) // Create the builder with the context. .withText("Hello human!") // Set the text to say. .build(); // Build the say action. // Execute the action. say.run(); } @Override public void onRobotFocusLost() { // Nothing here. }</pre></div> <div>That's all that's needed to get Pepper talking!</div>

Let's try it.



The source code we put in [GitHub](#)

Step	Action
	<div>class="first">Install and run the application.</div>
	<div>option “Hello Human”.</div> <div></div> <div>Congratulations! You have a talking Pepper.</div>