

# Java API Description

You can find here a full description of the functions included in the Reeti API.

## Constructors

Functions	Parameters	Return Value	Description
<b>Reeti</b>	<i>string_hostname</i> : IP adress of the reeti to connect to	<i>none</i>	the constructor used to instantiate a Reeti object

## Functions

Functions	Parameters	Return Value	Description
<b>subscribeToPosition</b>	<i>PositionCallback_callback</i> : a callback function of PositionCallback type	<i>void</i>	register a callback in order to get the current position of the Reeti
<b>subscribeToLedColor</b>	<i>LedColorCallback_callback</i> : a callback function of LedColorCallback type	<i>void</i>	register a callback in order to get the current led Color of the Reeti
<b>subscribeToIsSpeaking</b>	<i>SpeakingCallback_callback</i> : a callback function of SpeakingCallback type	<i>void</i>	register a callback in order to allow to know if the Reeti is currently speaking
<b>subscribeToBookmark</b>	<i>BookmarkCallback_callback</i> : a callback function of BookmarkCallback type	<i>void</i>	register a callback in order to allow to know when a bookmark is reached while the Reeti is talking
<b>subscribeToIsPlaying</b>	<i>PlayingCallback_callback</i> : a callback function of PlayingCallback type	<i>void</i>	register a callback in order to allow to know if the Reeti is currently playing a sequence
<b>unsubscribeToPosition</b>	<i>void</i>	<i>void</i>	unregister the position callback
<b>unsubscribeToLedColor</b>	<i>void</i>	<i>void</i>	unregister the led color callback
<b>unsubscribeToIsSpeaking</b>	<i>void</i>	<i>void</i>	unregister the callback used to know if the reeti is talking
<b>unsubscribeToBookmark</b>	<i>void</i>	<i>void</i>	unregister the callback to know when a bookmark is reached
<b>unsubscribeToIsPlaying</b>	<i>void</i>	<i>void</i>	unregister the callback used to know if the reeti is playing a sequence
<b>neckRotat</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the rotation of the Reeti's neck to the desired value
<b>neckPan</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the pan of the Reeti's neck to the desired value
<b>neckTilt</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the tilt of the Reeti's neck to the desired value
<b>rightLC</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the position of the right lips corner of the Reeti to the desired value
<b>leftLC</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the position of the left lips corner of the Reeti to the desired value
<b>topLip</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the position of the top lip of the Reeti to the desired value
<b>bottomLip</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the position of the bottom lip of the Reeti to the desired value
<b>rightEyePan</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the pan of the Reeti's right eye to the desired value
<b>rightEyeTilt</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the tilt of the Reeti's right eye to the desired value
<b>leftEyePan</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the pan of the Reeti's left eye to the desired value
<b>leftEyeTilt</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the tilt of the Reeti's right eye to the desired value
<b>rightEyeLid</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the position of the Reeti's right eye lid to the desired value
<b>leftEyeLid</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the position of the Reeti's left eye lid to the desired value
<b>rightEar</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the position of the Reeti's right ear to the desired value
<b>leftEar</b>	<i>float_pos</i> : a float between 0 and 100	<i>void</i>	set the position of the Reeti's left ear to the desired value
<b>setLedColor</b>	<i>Led_led</i> : the led to set between LEFTLED, RIGHTLED and BOTHLED  <i>String_color</i> : the desired color between red, green, light green, blue, dark blue, turquoise, yellow, violet, white, stop	<i>void</i>	set the color of the desired led

<b>setLedRGB</b>	<p><i>Led_led</i>: the led to set between LEFTLED, RIGHTLED and BOTHLED</p> <p><i>float_r</i>: red intensity (between 0 and 1023)</p> <p><i>float_g</i>: green intensity (between 0 and 1023)</p> <p><i>float_b</i>: blue intensity (between 0 and 1023)</p>	<i>void</i>	set the color of the desired led
<b>setPoseAsync</b>	<p><i>ReetiPosition_position</i>: an instance of ReetiPosition containing the 15 positions</p> <p><i>float_speed</i>: movement speed (from 10 to 300)</p> <p><i>ServicesCallback_callback</i>: a callback of the ServicesCallback type</p>	<i>void</i>	set the position of the 15 actuators of the Reeti
<b>setPoseAsync</b>	<p><i>float_neckRotat</i>: neck rotation position (from 0 to 100)</p> <p><i>float_neckPan</i>: neck pan position (from 0 to 100)</p> <p><i>float_neckTilt</i>: neck tilt position (from 0 to 100)</p> <p><i>float_rightLC</i>: right lips corner position (from 0 to 100)</p> <p><i>float_leftLC</i>: left lips corner position (from 0 to 100)</p> <p><i>float_topLip</i>: top lip position (from 0 to 100)</p> <p><i>float_bottomLip</i>: bottom lip position (from 0 to 100)</p> <p><i>float_rightEyePan</i>: right eye pan position (from 0 to 100)</p> <p><i>float_rightEyeTilt</i>: right eye tilt position (from 0 to 100)</p> <p><i>float_leftEyePan</i>: left eye pan position (from 0 to 100)</p> <p><i>float_leftEyeTilt</i>: left eye tilt position (from 0 to 100)</p> <p><i>float_rightEyeLid</i>: right eye lid position (from 0 to 100)</p> <p><i>float_leftEyeLid</i>: left eye lid position (from 0 to 100)</p> <p><i>float_rightEar</i>: right ear position (from 0 to 100)</p> <p><i>float_leftEar</i>: left ear position (from 0 to 100)</p> <p><i>float_speed</i>: movement speed (from 10 to 300)</p> <p><i>ServicesCallback_callback</i>: a callback function of ServicesCallback type</p>	<i>void</i>	set the position of the 15 actuators of the Reeti
<b>takePictureAsync</b>	<p><i>Camera_cam</i>: the camera to use , RIGHTCAM or LEFTCAM</p> <p><i>String_filename</i>: the output filename</p> <p><i>ServicesCallback_callback</i>: a callback function of ServicesCallback type</p>	<i>void</i>	take a picture with one of the reeti's camera and save it under /home/reeti/Pictures
<b>recordVideoAsync</b>	<p><i>Camera_cam</i>: the camera to use , RIGHTCAM or LEFTCAM</p> <p><i>String_filename</i>: the output filename</p> <p><i>ServicesCallback_callback</i>: a callback function of ServicesCallback type</p>	<i>void</i>	record a video with one of the reeti's camera and save it under /home/reeti/Videos
<b>stopRecordAsync</b>	<p><i>Camera_cam</i>: the camera to use , RIGHTCAM or LEFTCAM</p> <p><i>ServicesCallback_callback</i>: a callback function of ServicesCallback type</p>	<i>void</i>	stop the recording of the vidéo
<b>sayAsync</b>	<p><i>String_speech</i>: the text to say</p> <p><i>ServicesCallback_callback</i>: a callback function of ServicesCallback type</p>	<i>void</i>	make the reeti talk
<b>sayWithSynchronousAsync</b>	<p><i>String_speech</i>: the text to say</p> <p><i>ServicesCallback_callback</i>: a callback function of ServicesCallback type</p>	<i>void</i>	make the reeti talk and move its lips
<b>stopSpeechAsync</b>	<p><i>ServicesCallback_callback</i>: a callback function of ServicesCallback type</p>	<i>void</i>	make the reeti stop talking
<b>playSequenceAsync</b>	<p><i>String_sequence</i>: the sequence file to play</p> <p><i>ServicesCallback_callback</i>: a callback function of ServicesCallback type</p>	<i>void</i>	make the reeti play the chosen sequence

<b>playPoseAsync</b>	<i>String_pose</i> : the pose file to play <i>ServicesCallback_callback</i> : a callback function of ServicesCallback type	<i>void</i>	make the reeti play the chosen pose
<b>isConnected</b>	<i>void</i>	returns <i>true</i> if connected returns <i>false</i> if not	check if the connection is still available

See the [Samples](#) in order to get more information about using the C++ API.