

TokenShare

Talk n Share --> Tok n Share --> TokenShare







































This is a simple app for voice input. Press the mic button, say some thing, edit the words you said in edittext, then press the SHARE button. BOOM!

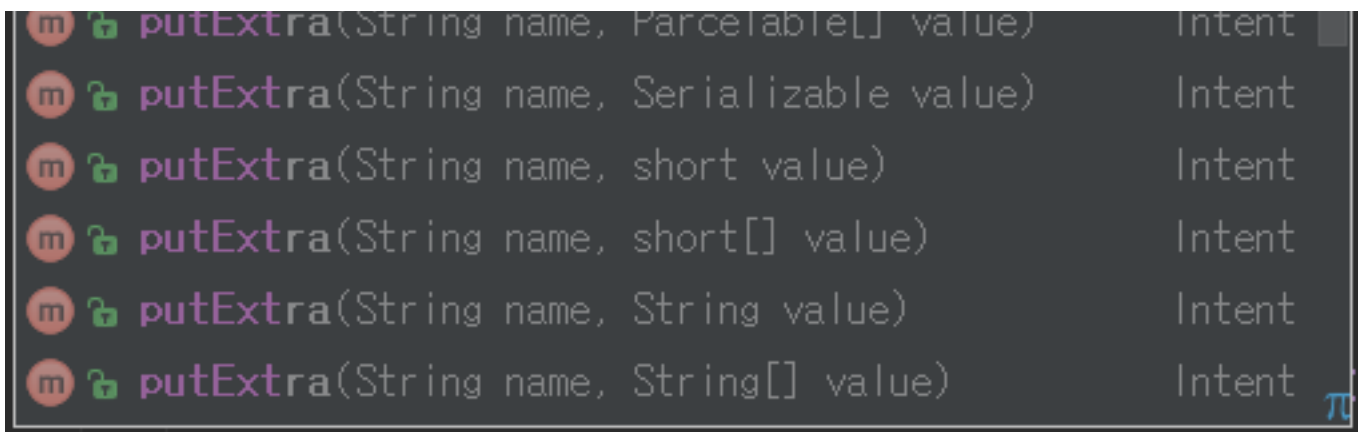
Important Imports

You must import `import android.speech.RecognizerIntent;` for voice recognition.

Implementing onClick Method to Mic Button

Implement onClick method to your mic button. In my case, `btnSpeak`. Then link a new custom method, such as `promptSpeechInput()`. Now in your new custom method, declare `ACTION_RECOGNIZE_SPEECH` intent and put some extra in `putExtra` parameters. You can look at the full list of parameters below.

		<code>putExtra(String name, boolean value)</code>	Intent
		<code>putExtra(String name, boolean[] value)</code>	Intent
		<code>putExtra(String name, Bundle value)</code>	Intent
		<code>putExtra(String name, byte value)</code>	Intent
		<code>putExtra(String name, byte[] value)</code>	Intent
		<code>putExtra(String name, char value)</code>	Intent
		<code>putExtra(String name, char[] value)</code>	Intent
		<code>putExtra(String name, CharSequence value)</code>	Intent
		<code>putExtra(String name, CharSequence[] value)</code>	Intent
		<code>putExtra(String name, double value)</code>	Intent
		<code>putExtra(String name, double[] value)</code>	Intent
		<code>putExtra(String name, float value)</code>	Intent
		<code>putExtra(String name, float[] value)</code>	Intent
		<code>putExtra(String name, int value)</code>	Intent
		<code>putExtra(String name, int[] value)</code>	Intent
		<code>putExtra(String name, long value)</code>	Intent
		<code>putExtra(String name, long[] value)</code>	Intent
		<code>putExtra(String name, Parcelable value)</code>	Intent
		<code>putExtra(String name, Parcelable[] value)</code>	Intent



Then add the following extras. `EXTRA_LANGUAGE_MODEL`, `LANGUAGE_MODEL_FREE_FORM`, `EXTRA_LANGUAGE`, `EXTRA_PROMPT`. You can add an optional try-catch method for error handling. I strongly recommend doing this.

Get Data from Voice Recognition Data

Declare `private final int REQ_CODE_SPEECH_INPUT = 100;`. Then make a temporary `ArrayList` and link the following : `.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS)`. Now you got the data from voice recognition!

Sharing Button

Note1 : I have implement the similar code in my previous project, LocaSend. Take a look there for detailed info.

Note2 : In my code there is something like `.setVisibility(View.VISIBLE)`. This is because I first made the button invisible. This is your choice.

Time to share, right? Phew, I am tired of onClick stuffs now. I believe you can do it on your own.

1. Set `onClick` listener and link to your button you previously declared as a variable.
 2. Set Intent type. In this case, `android.content.Intent.ACTION_SEND` and `text/plain`.
 3. Put some extra contents in `putExtra`. See parameters in above picture.
 4. Start `SharingIntent` Activity and set title.
-

License (You can find the same file in master/LICENSE)

Copyright 2017 cozyplanes

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.