

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <TextView
        android:id="@+id/fstTxt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="100dp"
        android:layout_marginTop="150dp"
        android:text="Enter Text to Speak" />

    <EditText
        android:id="@+id/txtSpeak"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="100dp"
        android:ems="10" />

    <Button
        android:id="@+id/btnSpeech"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="100dp"
        android:text="Speak" />
</LinearLayout>
```

MainActivity.java

```
package com.example.textttospeech;

import android.os.Build;
import android.speech.tts.TextToSpeech;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;

import java.util.Locale;

public class MainActivity extends AppCompatActivity implements
TextToSpeech.OnInitListener {
    Button speakBtn;
    EditText speakText;
    TextToSpeech;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        speakText = (EditText) findViewById(R.id.txtSpeak);
        speakBtn = (Button) findViewById(R.id.btnSpeech);
    }
}
```

```

        textToSpeech = new TextToSpeech(this, this);
        speakBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                texttoSpeak();
            }
        });
    }

    @Override
    public void onInit(int status) {
        if (status == TextToSpeech.SUCCESS) {
            int result =
                textToSpeech.setLanguage(Locale.FRENCH);
            if (result == TextToSpeech.LANG_MISSING_DATA || result ==
TextToSpeech.LANG_NOT_SUPPORTED) {
                Log.e("error", "This Language is not supported");
            } else {
                texttoSpeak();
            }
        } else {
            Log.e("error", "Failed to Initialize");
        }
    }

    @Override
    public void onDestroy() {
        if (textToSpeech != null) {
            textToSpeech.stop();
            textToSpeech.shutdown();
        }
        super.onDestroy();
    }

    private void texttoSpeak() {
        String text = speakText.getText().toString();
        if ("".equals(text)) {
            text = "Please enter some text to speak.";
        }
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
            textToSpeech.speak(text, TextToSpeech.QUEUE_FLUSH, null, null);
        } else {
            textToSpeech.speak(text, TextToSpeech.QUEUE_FLUSH, null);
        }
    }
}

```