import javax.sound.sampled.\*;

import java.io.\*;

/\*\*

 \* A sample program is to demonstrate how to record sound in Java

 \* author: www.codejava.net

 \*/

public class JavaSoundRecorder {

    // record duration, in milliseconds

    static final long RECORD\_TIME = 60000;  // 1 minute

    // path of the wav file

    File wavFile = new File("E:/Test/RecordAudio.wav");

    // format of audio file

    AudioFileFormat.Type fileType = AudioFileFormat.Type.WAVE;

    // the line from which audio data is captured

    TargetDataLine line;

    /\*\*

     \* Defines an audio format

     \*/

    AudioFormat getAudioFormat() {

        float sampleRate = 16000;

        int sampleSizeInBits = 8;

        int channels = 2;

        boolean signed = true;

        boolean bigEndian = true;

        AudioFormat format = new AudioFormat(sampleRate, sampleSizeInBits,

                                             channels, signed, bigEndian);

        return format;

    }

    /\*\*

     \* Captures the sound and record into a WAV file

     \*/

    void start() {

        try {

            AudioFormat format = getAudioFormat();

            DataLine.Info info = new DataLine.Info(TargetDataLine.class, format);

            // checks if system supports the data line

            if (!AudioSystem.isLineSupported(info)) {

                System.out.println("Line not supported");

                System.exit(0);

            }

            line = (TargetDataLine) AudioSystem.getLine(info);

            line.open(format);

            line.start();   // start capturing

            System.out.println("Start capturing...");

            AudioInputStream ais = new AudioInputStream(line);

            System.out.println("Start recording...");

            // start recording

            AudioSystem.write(ais, fileType, wavFile);

        } catch (LineUnavailableException ex) {

            ex.printStackTrace();

        } catch (IOException ioe) {

            ioe.printStackTrace();

        }

    }

    /\*\*

     \* Closes the target data line to finish capturing and recording

     \*/

    void finish() {

        line.stop();

        line.close();

        System.out.println("Finished");

    }

    /\*\*

     \* Entry to run the program

     \*/

    public static void main(String[] args) {

        final JavaSoundRecorder recorder = new JavaSoundRecorder();

        // creates a new thread that waits for a specified

        // of time before stopping

        Thread stopper = new Thread(new Runnable() {

            public void run() {

                try {

                    Thread.sleep(RECORD\_TIME);

                } catch (InterruptedException ex) {

                    ex.printStackTrace();

                }

                recorder.finish();

            }

        });

        stopper.start();

        // start recording

        recorder.start();

    }

}