```
// Copyright The League of Amazing Programmers, 2015

import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.InputStream;
import java.net.URL;

import javazoom.jl.decoder.JavaLayerException;
import javazoom.jl.player.advanced.AdvancedPlayer;

/* 1. Download the JavaZoom jar from here: http://bit.ly/javazoom
 * 2. Right click your project and add it as an External JAR (Under Java Build Path > Libraries).*/

public class IPodShuffle {

	public static void main(String[] args) throws IOException, JavaLayerException {
		// 3. Find an mp3 on your computer or on the Internet.

		// 4. Use the Song class below to instantiate a Song.

		// 5. Play the Song to test that it works.

	}

	/**
	 * 6. Congratulations on completing the sound check!
	 *
	 * Now we want to make an iPod Shuffle that plays random music.
	 *
	 * Create an ArrayList of Songs and a "Surprise Me!" button that will play a random song when it is clicked.
	 *
	 * If you're really cool, you can stop all the songs, before playing a new one on subsequent button clicks.
	 */
}

class Song {

	private int duration;
	private String songAddress;
	private AdvancedPlayer mp3Player;
```

```
	private InputStream songStream;

	/**
	 * Songs can be constructed from files on your computer or Internet addresses.
	 *
	 * Examples: <code>
	 *		new Song("everywhere.mp3");		//from default package
	 *		new Song("/Users/joonspoon/music/Vampire Weekend - Modern
Vampires of the City/03 Step.mp3");
	 *		new
Song("http://freedownloads.last.fm/download/569264057/Get%2BGot.mp3");
	 * </code>
	 */
	public Song(String songAddress) {
		this.songAddress = songAddress;
	}

	public void play() {
		loadFile();
		if (songStream != null) {
			loadPlayer();
			startSong();
		} else
			System.err.println("Unable to load file: " + songAddress);
	}

	public void setDuration(int seconds) {
		this.duration = seconds * 100;
	}

	public void stop() {
		if (mp3Player != null)
			mp3Player.close();
	}

	private void startSong() {
		Thread t = new Thread() {
			public void run() {
				try {
					if (duration > 0)
						mp3Player.play(duration);
					else
						mp3Player.play();
```

```
                        } catch (Exception e) {
                        }
                    }
                };
                t.start();
        }

        private void loadPlayer() {
                try {
                        this.mp3Player = new AdvancedPlayer(songStream);
                } catch (Exception e) {}
        }

        private void loadFile() {
                if (songAddress.contains("http"))
                        this.songStream = loadStreamFromInternet();
                else
                        this.songStream = loadStreamFromComputer();
        }

        private InputStream loadStreamFromInternet() {
                try {
                        return new URL(songAddress).openStream();
                } catch (Exception e) {
                        return null;
                }
        }

        private InputStream loadStreamFromComputer() {
                try {
                        return new FileInputStream(songAddress);
                } catch (FileNotFoundException e) {
                        return this.getClass().getResourceAsStream(songAddress);
                }
        }
}
```