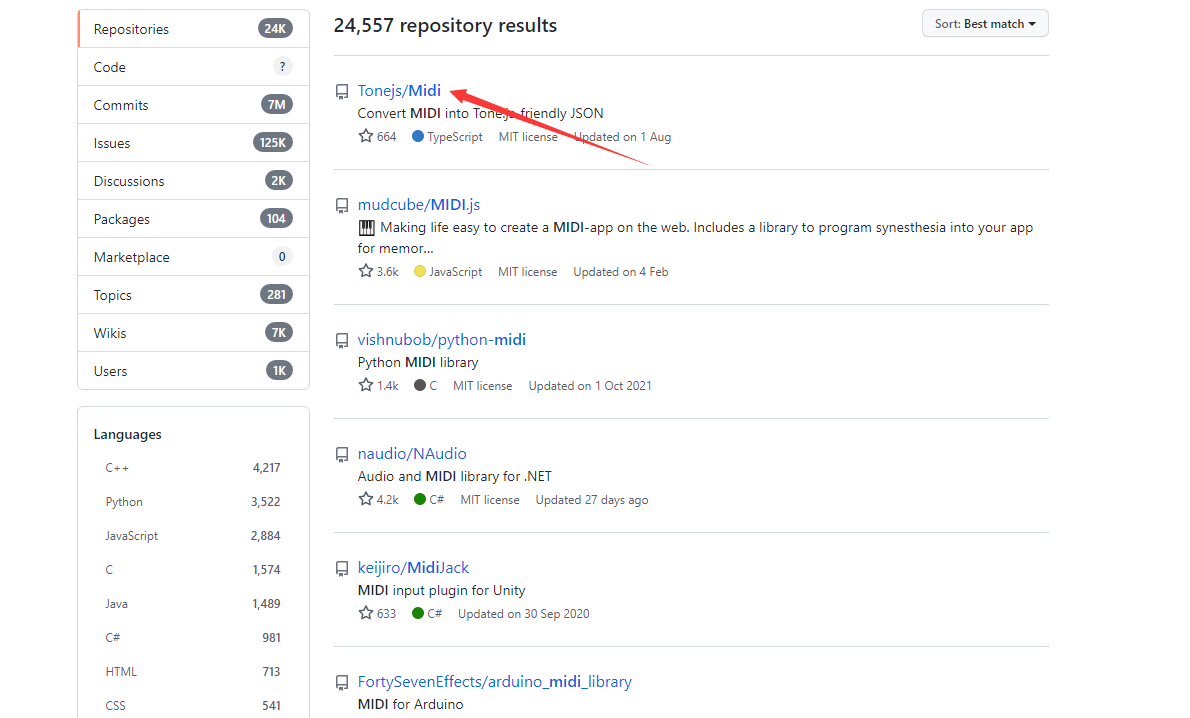
# Midi格式转换为json格式小教程

## 1.收集mid文件

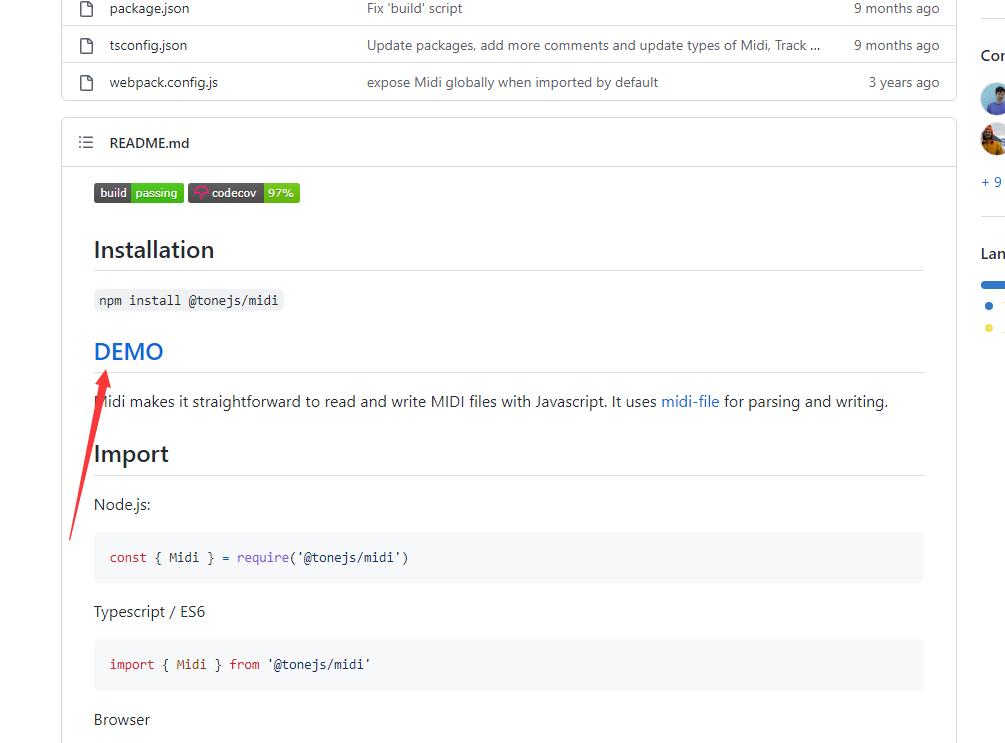
可以直接用搜索引擎搜索想要的歌曲的mid文件，下载下来。如果懂一些乐理知识甚至可以用专业软件原创音乐，生成一段自己音乐的mid文件

## 2.mid转为json

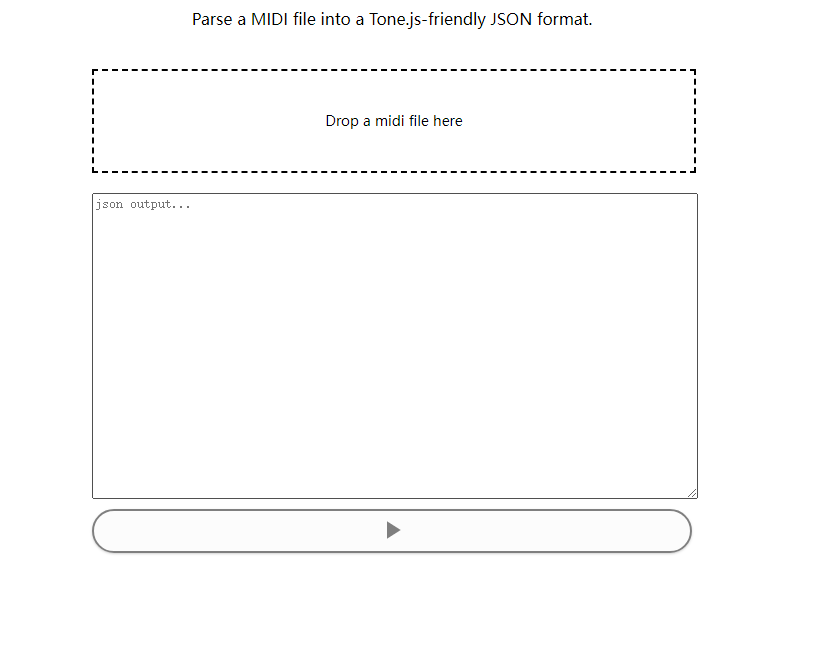
①进入此链接https://github.com/Tonejs/Midi，或直接登录github搜索midi,找到Tonejs/Midi,进入项目主页。(要是github登不上去,那么请科学上网或直接百度github为什么打不开)



②在项目主页就可以看到项目所有源代码，首先要做的是请在右上角为原作者点亮一下Star，支持原作者，谢谢。此项目主要是用TS写的，是一个运行在浏览器端的项目，有兴趣可以研究一下项目是如何写的，为什么可以把mid文件转为json,这里我们只谈使用。向下滑动找到DEMO，直接点击进去

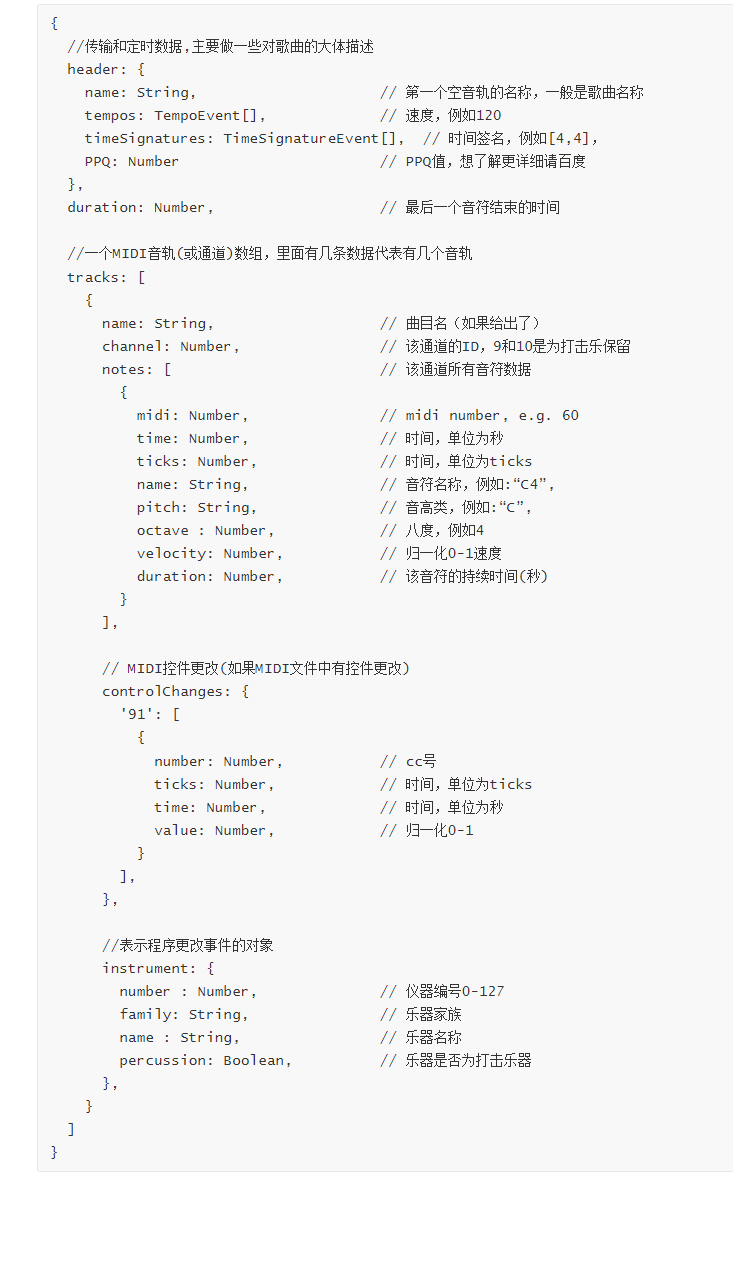


③此时打开的窗口就是项目此项目的界面了，点击上面的框可以上传一个mid的文件，然后相应的json会生成在下面的框里，还可以直接播放此mid文件(不过个人感觉在这里播放好像每个音都被调高了)



## 3.json文件结构介绍

项目生成的json文件应该是非常庞大的，因为它把一段音频用json格式描述了出来，而且十分详细，我们可以看看json里每个值都是什么意思(复制于项目主页，个人汉化，想看更详细的请移步至项目github主页，顺带点亮一下Star,支持原作者)



## 4.提取信息

生成的json结构内容实在是太多了，有一些对我们来说完全无用的信息，所以我们需要进行信息提取，一般我们只需要notes里的每一个name(音符名)，time(按下时间)，duration(持续时间)，我们可以用一段程序帮我们做这件事，当然也可以手动，不过时间成本就无法估量了，对于多个tracks,我们直接按time排序，合并为一个。这里我给出一段node端的js代码，可以完成此项工作，如果你的电脑安装了node环境，那么可以直接运行，当然，实现方法也不至这一种，你也可以用C或java来做到相同的事

const fs = require("fs");

fs.readFile("./1.txt", function (err, data) {

  const d = JSON.parse(data);

  const tracks = d.tracks;

  const arr = [];

  for(let i = 0 ;i<tracks.length;i++){

    const notes = tracks[i].notes;

    for(let j = 0 ;j<notes.length;j++){

      const name = notes[j].name;

      const time = Math.floor(notes[j].time\*1000);

      const duration = Math.floor(notes[j].duration\*1000);

      arr.push({name,time,duration});

    }

  }

  arr.sort((a,b) => {

    return a.time - b.time;

  })

  let str = "";

  arr.forEach((item) => {

      str += item.name;

      str += "\t";

      str += item.time;

      str += "\t";

      str += item.duration;

      str += "\n";

  })

  fs.writeFile("./2.txt", str, function(err){

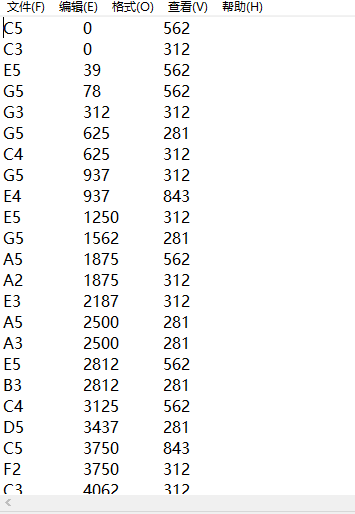
      if (err) {res.status(500).send('Server is error...')}

  })

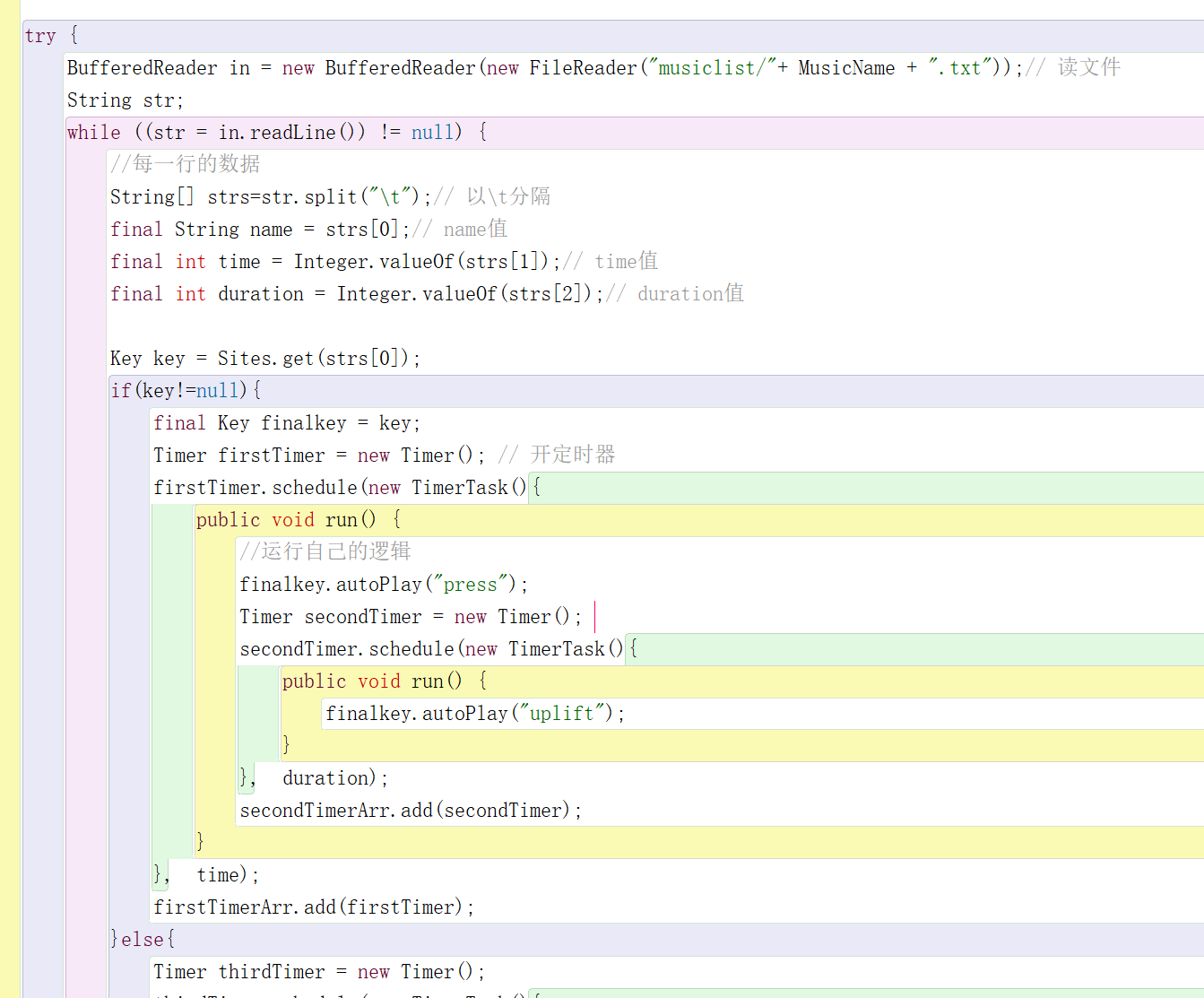
});

## 5.最终成品

创建一个1.txt文件，里面直接复制进去刚才在网页端生成的json，再创建一个js文件，里面放入这段程序，把1.txt放在与这段程序的js文件同一目录下，运行它(前提是你有node环境)，它会读入1.txt，自动生成2.txt，2.txt的结构类似这样



第一列是name(音符名)，第二列是time(按下时间,特别注意，用上面所给程序跑出的time单位为毫秒，而不是一开始生成json的单位秒)，第三列是duration(持续时间，特别注意，用上面所给程序跑出的duration单位为毫秒，而不是一开始生成json的单位秒)，中间都是用一个tab键进行分隔，也就是一个’\t’，此时，就可以在Greenfoot里用java一行一行的读入，再根据定时器时间批量生成定时器，再在定时器里加入自己的逻辑，运行程序，便可以听到歌曲的声音传来了，这里还是给一段示例代码



目标完成，最后别忘了在github上给原作者点亮star,支持原作者！！！