Atitit 天气接口

<http://www.weather.com.cn/data/cityinfo/101250101.html>

返回

{"weatherinfo":{"city":"长沙","cityid":"101250101","temp1":"11℃","temp2":"22℃","weather":"多云","img1":"n1.gif","img2":"d1.gif","ptime":"18:00"}}

Promise，generator/yield，await/async 都是现在和未来 JS 解决异步的标准做法，可以完美搭配使用。这也是使用标准 Promise 一大好处。最近也把项目中使用第三方 Promise 库的代码全部转成标准 Promise，为以后全面使用 async/await 做准备。

另外，Fetch 也很适合做现在流行的同构应用，有人基于 Fetch 的语法，在 Node 端基于 http 库实现了 [node-fetch](https://github.com/bitinn/node-fetch" \t "https://segmentfault.com/a/_blank)，又有人封装了用于同构应用的 [isomorphic-fetch](https://github.com/matthew-andrews/isomorphic-fetch" \t "https://segmentfault.com/a/_blank)。

注：同构(isomorphic/universal)就是使****前后端运行同一套代码****的意思，后端一般是指 NodeJS 环境。

作者attilax 艾龙