

```
$ npm init
Package name: (byvoidmodule) byvoidmodule
Description: A module for learning perpose.
Package version: (0.0.0) 0.0.1
Project homepage: (none) http://www.byvoid.com/
Project git repository: (none)
Author name: BYVoid
Author email: (none) byvoid.kcp@gmail.com
Author url: (none) http://www.byvoid.com/
Main module/entry point: (none)
Test command: (none)
What versions of node does it run on? (~0.6.10)
About to write to /home/byvoid/byvoidmodule/package.json

{
  "author": "BYVoid <byvoid.kcp@gmail.com> (http://www.byvoid.com/)",
  "name": "byvoidmodule",
  "description": "A module for learning perpose.",
  "version": "0.0.1",
  "homepage": "http://www.byvoid.com/",
  "repository": {
    "url": ""
  },
  "engines": {
    "node": "~0.6.12"
  },
  "dependencies": {},
  "devDependencies": {}
}

Is this ok? (yes) yes
```

这样就在 `byvoidmodule` 目录中生成一个符合 `npm` 规范的 `package.json` 文件。创建一个 `index.js` 作为包的接口，一个简单的包就制作完成了。

在发布前，我们还需要获得一个账号用于今后维护自己的包，使用 `npm adduser` 根据提示输入用户名、密码、邮箱，等待账号创建完成。完成后可以使用 `npm whoami` 测验是否已经取得了账号。

接下来，在 `package.json` 所在目录下运行 `npm publish`，稍等片刻就可以完成发布了。打开浏览器，访问 <http://search.npmjs.org/> 就可以找到自己刚刚发布的包了。现在我们可以世界的任意一台计算机上使用 `npm install byvoidmodule` 命令来安装它。图3-6是 `npmjs.org` 上包的描述页面。

如果你的包将来有更新，只需要在 `package.json` 文件中修改 `version` 字段，然后重新使用 `npm publish` 命令就行了。如果你对已发布的包不满意（比如我们发布的这个毫无意义的包），可以使用 `npm unpublish` 命令来取消发布。