

# 《开源软件设计与开发》课程总结

姓名：李磊

学号：52195100006

## 1、开源理解

本文包含了一些粗浅的思考，关于“工业界开源和学术界开源的差异和一种设想的转化方式”。

### 一波 N 折只为了改 Angular.js 一行代码

我曾就职于一家传统企业的电商部门，从事前端开发。为了提升自己功力，也为了在工作中修 bug 更得心应手。于是我开始研究起项目所使用的 Angular.js(<https://github.com/angular/angular.js>)框架的源码。原本以为一个高星项目，有大量的开发者参与，又维护了很长时间。应该不会有肉眼可见的 BUG。结果还是让我碰上了一个问题，这个问题可大可小。往小处说，我遇到的问题并不是 BUG，只是一句在执行流程中执行了没有任何作用的代码，无伤大雅；往大了说，这句代码身处 Angular.js 框架比较核心的，被大量调用的函数中，聚沙成塔，势必会对框架整体性能造成影响。

当时我在工作中看似 Git 用得很溜，但是在企业内部的开发项目中，实际上就是“三板斧”：pull,commit,push。对于 Angular.js 这类大型开源项目如何协作，我并不熟悉，只能硬着头皮开始了这次“修 BUG”之旅。我首先到 Github 上提了一个 issue(<https://github.com/angular/angular.js/issues/14810>),然后就收到了项目维护者的“灵魂拷问”：

*Do you think this causes problems in the code? Or do you suggest we remove it? The first thing you can do is to check if all the tests pass if you remove the second statement. You can also look in the git history and see when / why this line was introduced and try to see if it is still needed ?*

翻译过来大概就是：那句代码会造成 BUG 么？你建议删掉它？测试用例跑过了没？看看谁写的？现在还有没有用？

我赶紧按要求检查了一遍代码提交历史，确认该代码确实是无用代码，试着跑了一下测试用例，但没跑通。把这些信息在 issue 中 comment 之后，过了几小时，收到了维护者的回复：确认该代码无效，测试用例在本地未跑通也没问题，PR 后官方会再跑一次，你是否愿意提交 PR 进行修复？

我满心欢喜，预想着鼠标再点个 3, 5 下，就可以在几万星的开源项目中留下我的用户名。殊不知，PR 的过程比 issue 还要繁琐。把项目 fork 到个人目录，pull,修改代码,commit,push, 点击 github 界面上的 pull request 之后。我又被 googlebot 要求签署开发者协议。

*Thanks for your pull request. It looks like this may be your first contribution to a Google open source project. Before we can look at your pull request, you'll need to sign a Contributor License Agreement (CLA).*

*Please visit <https://cla.developers.google.com/> to sign.*

*Once you've signed, please reply here (e.g. I signed it!) and we'll verify. Thanks.*

签了 CLA 之后，维护者又要求我按项目规范来重新填写 commit 信息。

*Can you please update the commit message according to our guidelines: <https://github.com/angular/angular.js/blob/master/CONTRIBUTING.md#commit-message-format>*

按规范填完信息之后，又被打回修改，要求我把代码修改的类型从修复改成重构。

*Not quite, sorry. In the commit message, you should write "refactor" instead of "fix".*

*And for the future, make sure that there are no merge commits on the branch (i.e. amend the commit instead)*

至此，这趟折腾的“修 BUG”之旅终于画上了句号。

## 工业界开源和学术界开源的差异和一种设想的转化方式

虽然文中记述的这次 PR 只改了区区一行代码，但是在工业界，确是有一定分量的，有助于升职或跳槽。因为虽然只改了一行代码，但是反映出了作为一个码农的四种能力：1.可与外国开发者准确交流的英语能力 2.开源工具的使用能力 3.Javascript 语言的熟练度 4.Angular.js 框架的熟练度。

有意思的是，我后来又辞职攻读计算机专业硕士，学术界（更具体指高校中评各类奖学金时）对开源行为的评估是另外一种完全不同的导向。高校也鼓励开源项目，但不鼓励学生去参与高星的知名项目，而是要求学生尽量以学校和实验室名义开发项目并自行推广求 star。换句话说，工业界看开源看的是开发者的个人实力，学术界看开源看的项目是否提升了学校或组织的影响力。

那么这两种完全不同的导向是否可以相互转化呢？结合我自身的开发经历，如下三种开源行为的工作量大概是相近的。

- 1) 从头搭一个 1000 行代码的玩具项目，获得 10 星
- 2) 为一个 500 星的公开项目提交 20 行代码
- 3) 为一个 10000 星的知名公开项目提交 1 行代码

我们不妨设计一个单位，叫 星行，用来评估一个开发者的某次开源行为的价值，上述三个操作的价值均为 10000 星行=1000 行\*10 星=500 星\*20 行=10000 星\*1 行。

我想，当工业界和学术界对开源的评估标准趋同之后，更有利于开发者的协同和流动，造就一个更繁荣的开源环境。

## 2、开源贡献

此处记录了我参与 gitcourse 项目时，提交的一个 issue。对于该 issue，我发现了一个 bug，并查阅资料给出了该 bug 的来源和现状等信息，并给出了解决方案。并且就后续问题与开发团队进行了探讨。地址 <https://github.com/kfcoding/gitcourse/issues/45>

# lookbehind is not supported

## by IE, Edge and Safari #45

Open

[deploy](#) opened this issue Dec 8, 2019 · 4 comments

### Comments

**[deploy](#) commented Dec 8, 2019**

环境

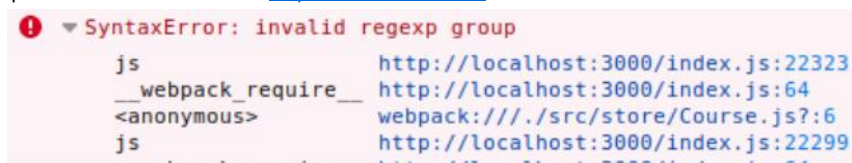
操作系统: Ubuntu18.04

浏览器: Firefox Quantum 68.0.1 64 位

Issue 类型: Bug

描述:

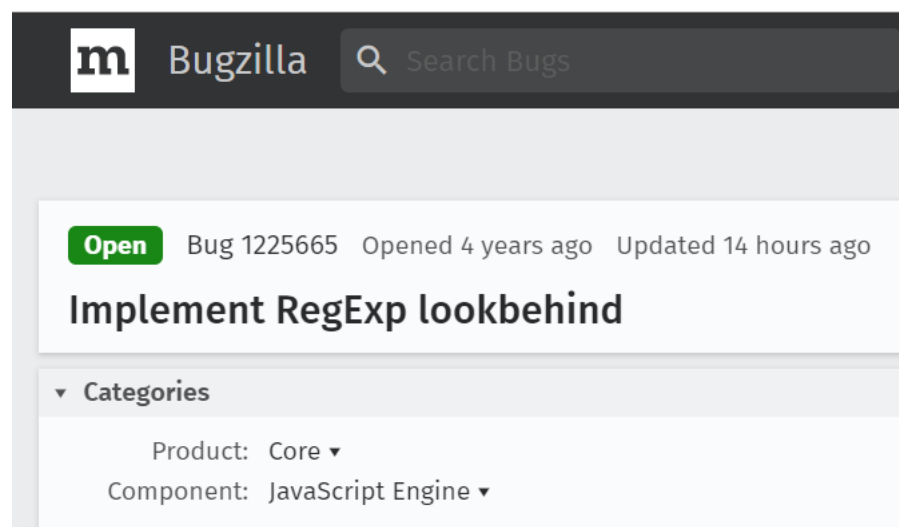
使用 Ubuntu18.04 系统按 <https://github.com/kfcoding/gitcourse> 所提供的步骤启动服务器后, 使用 Firefox quantum 68.0.1 64 位访问 <http://localhost:3000/> 发生错误, 如下图所示。



```
SyntaxError: invalid regexp group
    at http://localhost:3000/index.js:22323
    at __webpack_require__ http://localhost:3000/index.js:64
    at <anonymous> webpack:///./src/store/Course.js?:6
    at js http://localhost:3000/index.js:22299
```

报错原因来自 index.js, 这是由 webpack 打包而成的, 更具体的错误原因来自 gitcourse/src/store/Scenario.js 的第 53 行 `var matches = extraTab.match(/(?<=[:]).+?(?=))/mg);`。

在该行代码中使用了 js 正则中的 lookbehind 断言机制 (即 `?<=`) , 但该机制在 Firefox 中并未实现, 其对应的官方 issue 至今仍是 open 状态, [https://bugzilla.mozilla.org/show\\_bug.cgi?id=1225665](https://bugzilla.mozilla.org/show_bug.cgi?id=1225665)



**m Bugzilla** Search Bugs

**Open** Bug 1225665 Opened 4 years ago Updated 14 hours ago

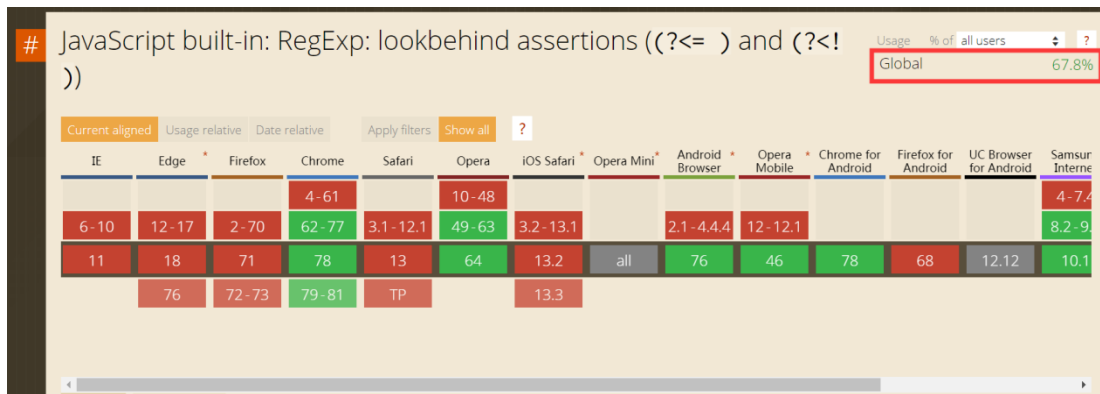
### Implement RegExp lookbehind

▼ Categories

Product: Core ▼

Component: JavaScript Engine ▼

更进一步, 据 <https://caniuse.com/#feat=mdn-javascript-builtins-regexp-lookbehind-assertion> 提供的数



该特性在全球浏览器中的支持度仅为 67.8%，尤其是 IE，Edge，Safari 对该功能均不支持。该 bug 对 kfcoding 的潜在用户（即高校师生）会造成很大的影响，主要有两个原因：1.高校为了正常使用一些老旧系统，还在使用 IE 浏览器 2.学生自行购置的 windows 笔记本一般预装 Edge，Macbook 和 iPad 等则使用 Safari。

经我搜索，lookbehind 断言在源码中只使用了一次，直接修复即可，并注意在以后的开发中不再使用或引入相应的 polyfill。一种可行的修改方案是把 gitcourse/src/store/Scenario.js 的第 53 行到 56 行的如下代码，仅供参考。

```
var matches = extraTab.match(/(?<=[:]).+(?=[=])/mg);
```

```
if (matches && matches.length > 0){
```

```
  exposed_ports[${matches[0]}/tcp]={};
```

```
}
```

改为

```
var matches = extraTab.match(/([:]).+(?=[=])/mg);
```

```
if (matches && matches.length > 0){
```

```
  matches[0]=matches[0].replace(/([:])/mg,"");
```

```
  exposed_ports[${matches[0]}/tcp]={};
```

```
}
```

Collaborator

**LiuChangFreeman** commented [Dec 9, 2019](#)

将采用您的修复建议

**WillWangx** commented [Dec 9, 2019](#)

能否用 PR 的形式来进行合并~

**deploy** commented [Dec 9, 2019](#)

我建议还是由开发团队处理此 issue 并确定具体修复方案，因为

1. 按我的方案修改代码后，在 Ubuntu18.04Firefox Quantum 68.0.1 64 位下仍有其他报错（应该不是因我的修改而产生的）需要处理，所以我建议开发团队使用我所述的环境跑一遍系统，处理

了所有问题后，一次性提交代码。

2. 该系统开源出来的部分并未包含自动化测试，所以我不能确定按我的方案修改后，该系统（而不仅是首页）的所有页面，所有功能是否和预期一致。

**WillWangx commented [Dec 9, 2019](#)**

好，有道理！今天讨论下，我们就开始做起来~

### 3、课程反馈

课程提供的项目都比较高大上，普通人很难有发挥的空间。以后的课程可以考虑设立几个从头开始的种子项目（比如做个网站，做个 APP，做个工具），自愿加入，自行设计项目要实现的功能，自行维护。每人都可以贡献代码，大家都是新手，bug 也会比较多，方便大家练习 git 等功能。

### 4、参考文献

无