

部署上线的最后步骤

在最后的章节中我们会向你展示如何部署你的应用到产品环境。你可以使用 Heroku 来免费托管和部署应用，在学习如何部署 React 应用的同时也可以了解更多 *create-react-app* 的相关特性。

弹出

接下来的步骤和知识对于部署产品环境来说**并不是必须的**，但我依然想要在这里讲解一下。*create-react-app* 提供了一个特性，既可以保持应用的可扩展性，又可以避免被第三方依赖绑架。被第三方依赖绑架通常意味着一旦我们采取了某项技术就没有退出机制的情况。幸运的是 *create-react-app* 为我们提供了一个逃生舱：“eject”。

在 *package.json* 中，你可以找到“start”、“test”和“build”这些命令，用来启动、测试和构建应用。最后的命令就是 *eject*。你可以试着去执行它，但是这个命令只能被执行一次并且不能撤回。**这是一个破坏性的命令，一旦执行就不能反悔**，如果你只是学习 React，那就没有理由离开 *create-react-app* 提供给你的便利环境。。

假设你运行了 `npm run eject`，它会复制所有的配置和依赖到 *package.json* 中，同时创建一个新的 *config/* 文件夹。你会将整个项目完全转换成带有 Babel 和 Webpack 等工具的自定义配置。最终，你将可以完全控制所有这些工具。

官方文档也说了 *create-react-app* 更适合中小型项目，所以你不用感到愧疚运行“eject”命令来移除掉 *create-react-app* 并拿回控制权。

练习

- 阅读更多关于 [eject](#) 的内容

部署你的App

说到底，没有哪个应用应该止步于本地环境（localhost），大家都想要有正式上线的一天。Heroku 是一个提供平台即服务（PaaS）的服务商，你可以在上面托管你的应用。他们提供了和 React 的无缝集成，确切地说，它可以

让你在几分钟之内部署一个通过`create-react-app`创建的应用。和`create-react-app`的理念一样，它可以做到零配置部署。

在部署应用到Heroku之前，你需要先满足两个条件：

- 安装 [Heroku命令行工具](#)
- 创建一个 [免费的Heroku账号](#)

如果你的Mac上已经安装了Homebrew，你可以用终端安装Heroku命令行：

```
{title="Command Line",lang="text"}
```

```
brew update  
brew install heroku-toolbelt
```

现在你可以使用git和Heroku命令行来部署你的应用了

```
{title="Command Line",lang="text"}
```

```
git init  
heroku create -b https://github.com/mars/create-react-app-buildp  
git add .  
git commit -m "react-create-app on Heroku"  
git push heroku master  
heroku open
```

搞定。我希望你的应用在线上运行一切正常。如果你遇到什么问题，这里有一些资源可以参考。

- [Git和GitHub入门](#)
- [零配置部署React应用](#)
- [create-react-app的Heroku部署套件](#)