第9步：合并PR

点击绿色的“合并请求”按钮。将您的更改合并到master分支中。

完成后，建议您删除分支（太多分支会变得凌乱），因此请点击灰色的“删除分支”按钮。

您可以通过单击新存储库首页上的“提交”来再次检查提交是否已合并。

这将显示该分支中所有提交的列表。您可以看到刚刚合并的那个（合并请求2）。

您还可以在右侧看到提交的哈希码。哈希码是该特定提交的唯一标识符。这对于引用特定的提交以及撤消更改很有用（使用git revert <hash code number>命令撤销）。

第10步：将GitHub上的更改返回到您的计算机

现在，GitHub上的仓库看起来与本地计算机上的仓库有所不同。例如，您在分支中进行的提交并合并到master分支中的提交在本地计算机上的master分支中不存在。

为了获得您或其他人在GitHub上合并的最新更改，请使用git pull origin master命令（在master分支上操作）。

这将向您显示所有已更改的文件及其更改方式。

现在我们可以再次使用git log命令查看所有新提交。

（您可能需要将分支切换回master分支。您可以使用git checkout master命令来完成。）

第11步：沉浸在git的荣耀中

您已成功进行PR，并将您的代码合并到master分支中。恭喜你！如果您想更深入一点，请查看此Git101文件夹中的文件，以获取有关使用git和GitHub的更多提示和技巧。

建议您花一些时间与您的团队一起模拟一个较小的小组项目，就像在这里操作的那样。让您的团队用您的团队名称创建一个新文件夹，并向其中添加一些文本文件。然后，尝试将这些更改推送到此远程仓库中。这样，您的团队可以开始对最初没有创建的文件进行更改并使用PR功能进行练习。然后，使用GitHub上的git blame和git历史记录工具熟悉跟踪文件中进行了哪些更改以及谁进行了这些更改。

您使用git的次数越多，您将越感到舒适。 （我抗拒不了。）

由梅根·尼尔森（Meghan Nelson）撰写