# GitHub使用及相关说明

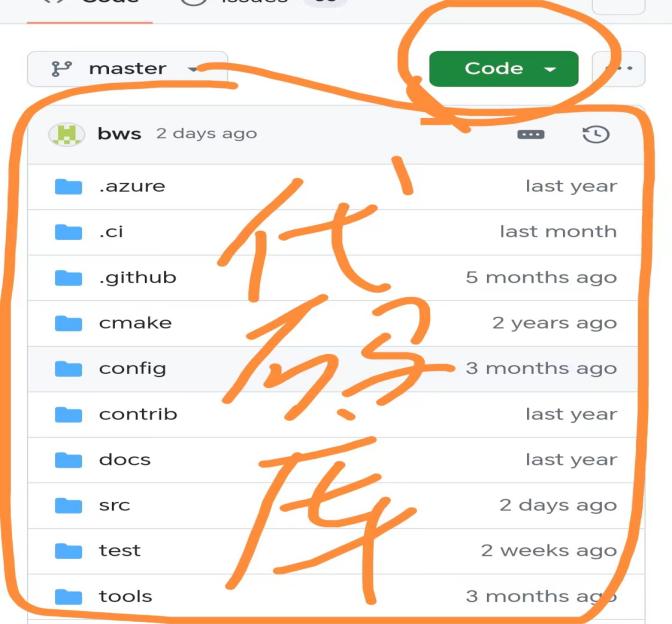
一．关于Github

Github是一个面向开源（开源一词最初是指开源软件（OSS）。开源软件是源代码可以任意获取的计算机软件，任何人都能查看、修改和分发他们认为合适的代码）及私有软件项目的托管平台，因为只支持Git作为唯一的版本库格式进行托管，故名GitHub。

1. Github功能

（1）下载软件,源代码

GitHub作为全球最大的代码托管与协作平台，最大的作用就是保存程序员编写的软件源代码。当我们需要相关资源时，可从顶部的Open Source 中的 Trend ing中找寻所需资源，或直接查找仓库，以https.//github.com+仓库主人+仓库本身名字（+git)的形式直接进入。

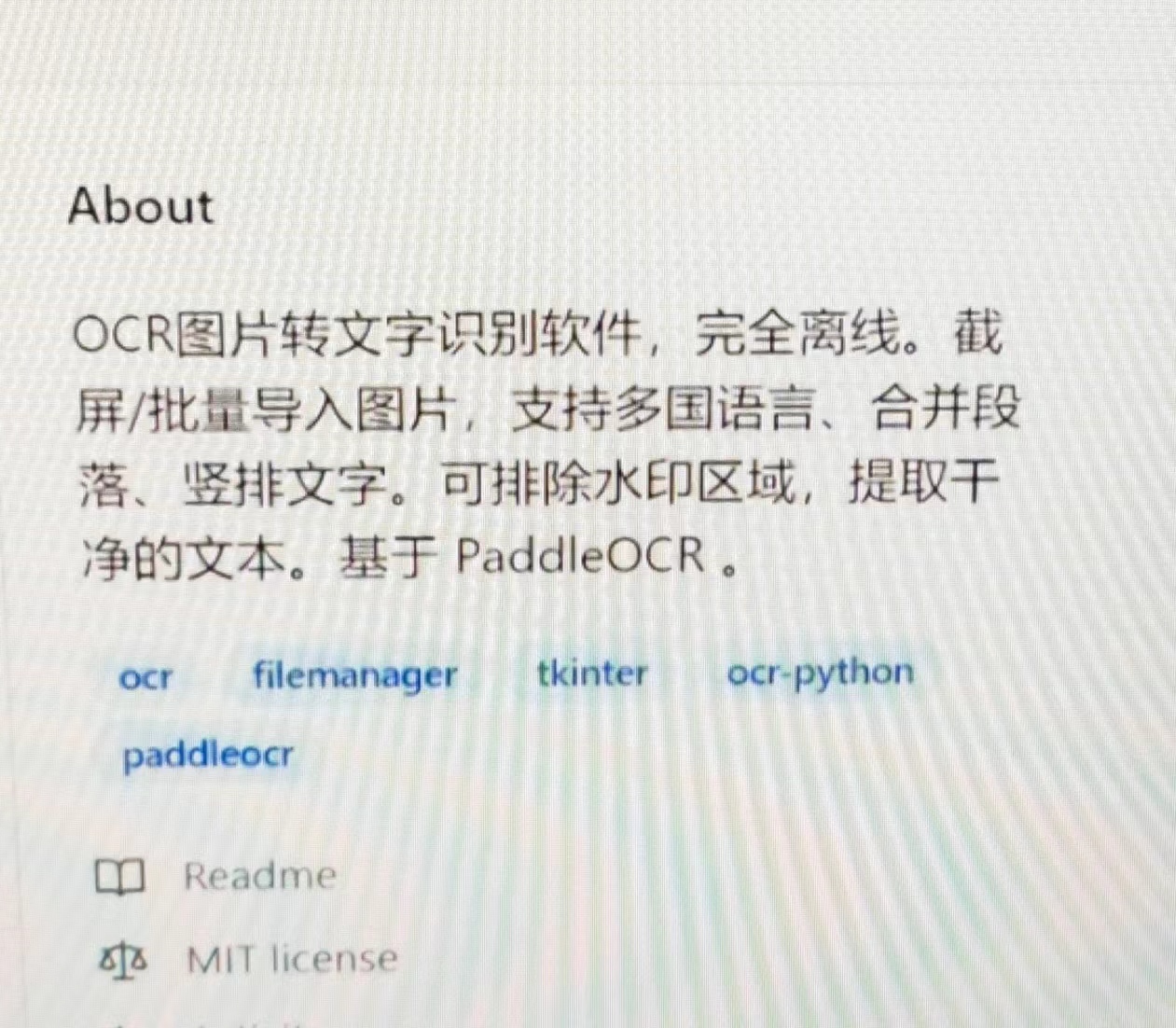
 进入以后，界面中最显眼的就是代码库，里面存放的是整个项目的源代码，可以点击右上角的code按键，点击Download Zip直接下载

代码库后都会有两条信息，第一条是Commit Message,即作者最后一次更新代码的备注，第二条是最后一次的更新时间。

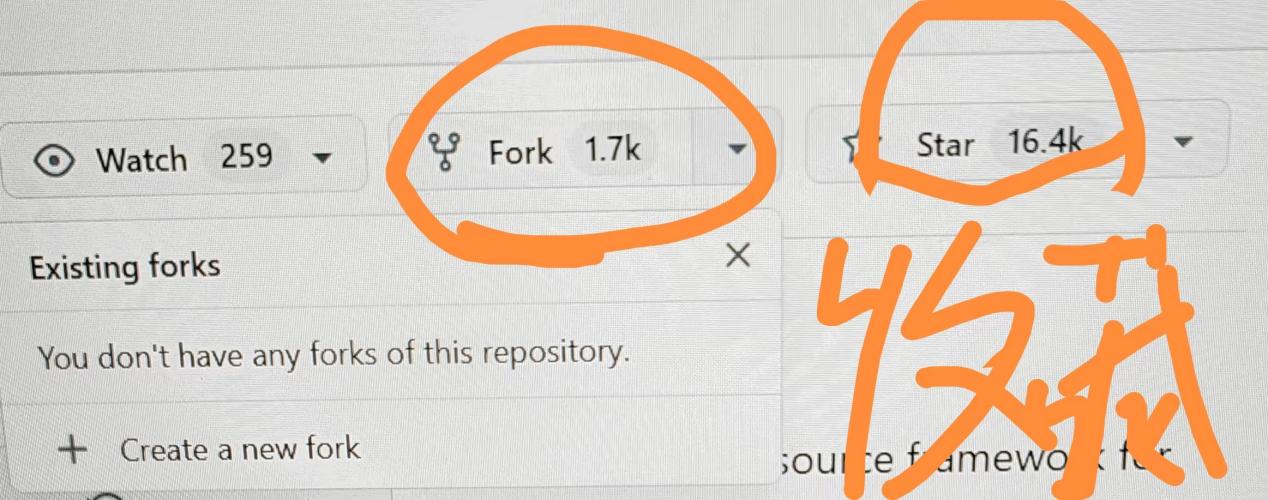
代码库下面是项目的介绍，即Readme部分。在Readme中一般可以看到该项目的作用和使用方法。（这部分比较关键，其详略程度取决于作者）

另一个找软件时需要关注的模块是右边的Releases,这里一般是作者已经打包好的软件和当前版本的介绍，可直接下载使用。

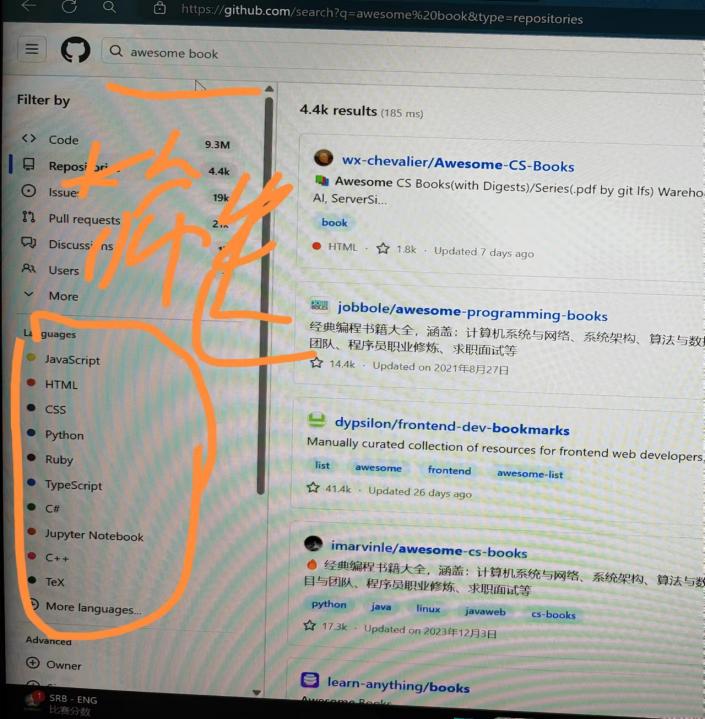
在Releases上方有一个About模块，这里会显示项目的整体概括，可以帮助快速了解当前项目。



在About上方显而易见有个星星图标，类似于收藏

在starred边上有个fork功能，点击fork可以直接把当前项目的所有代码拷贝到自己的主页。

1. .搜索功能

虽然将搜索单独列出来，但归根到底，搜索是为了用户更好的查找自己所需的软件或资源等等。

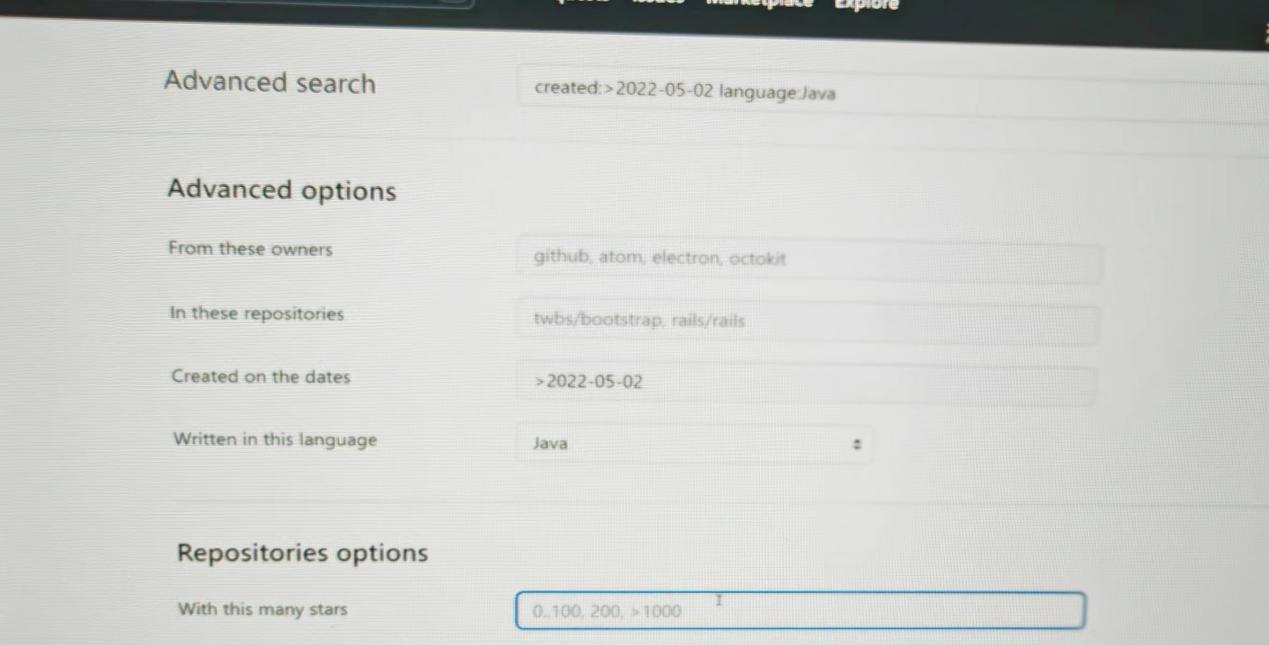
如果你想搜索远程工作的信息 remote job

如果你想找面试题 awesome interview

同时可以通过左侧Language来进一步筛选。

免费书籍 awesome book ......

总之，在搜索时awesome起手是个很实用的技巧

但对大部分老手来说，利用github的高级搜索功能所提供的限定符更为便利（在搜索框中输入限定符）。除此之外，还可以进入高级搜索界面（advance search)用可视化表单 搜索。

由于表单搜索条件有限，我们可以用更多的搜索限定语法，如按代码库名称，描述搜索等等。

1. Issue

GitHub issue 是参与和提升开源项目必经过程。提issue是为了给出建议性的意见。

Issue大体上分为两类，一类是bug report,即这个项目号称这么干，但没这么干；另外一类叫feature request,即你希望这个项目这么干。但他暂时还没这么干。（记得在发布issue之前，先用关键词找一下是否有人已经发过相同或相似issue了）

* 1. Bug report 最重要的事情就是表达清楚，你认为他怎么工作，实际又是怎么工作的，并给出一个复现方式，如果要想提出一个好的issue,一定要给出一个minimum demo,（"demo" 通常指的是一个演示或者样本程序。它是一个软件或应用程序的实例，通常包含了一个完整的功能或者特性，用于向用户展示产品的使用方式或功能。"demo" 在计算机科学和软件开发中是一个非常重要的概念，用于演示、测试和验证软件或应用程序的功能和性能。）无法复现bug的优先级比较低。对于bug report而言，复现就是基础，所以解释的越清楚越好。
* 注意：若所提的issue有模板，它应会列出来，如果没有，那自己应该想着办法把这个版本加上去，因为你提的bug很有可能在某些特定的版本上才能复现。

1. Feature request 的核心有两点：
2. 解释清楚你要加什么功能
3. 说服这个开发者为什么要加这个功能

注：解决完且无其他问题后，记得把这个issue给close掉。

除了这些，GitHub还有能够自动帮助你写代码的AL-Copilot插件

和可用于自动执行生成、测试和部署管道的Github Action等等，这些内容太细就不一一讲解。

（资料来源b站，以上图片均实际操作得出）

谢谢观看