

做的东西

喜欢看知乎，做一个将指定问题及答案生成文本发送到指定邮箱的服务，这样可以发送到 kindle 邮箱，统一在终端上阅读。

UI

知乎答案生成服务

这是一个可以将知乎答案转成 kindle 电子书的服务。

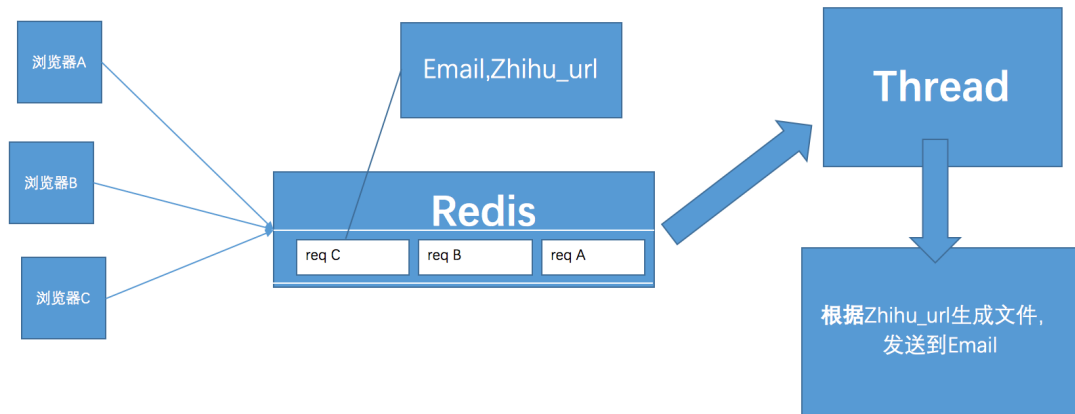
(1) 可以填普通邮箱

(2) 也可以填amazon的邮箱(**@kindle.cn)

邮件地址（例如：bo5509@kindle.cn）

知乎问题地址（例如：<http://www.zhihu.com/question/27191586>）

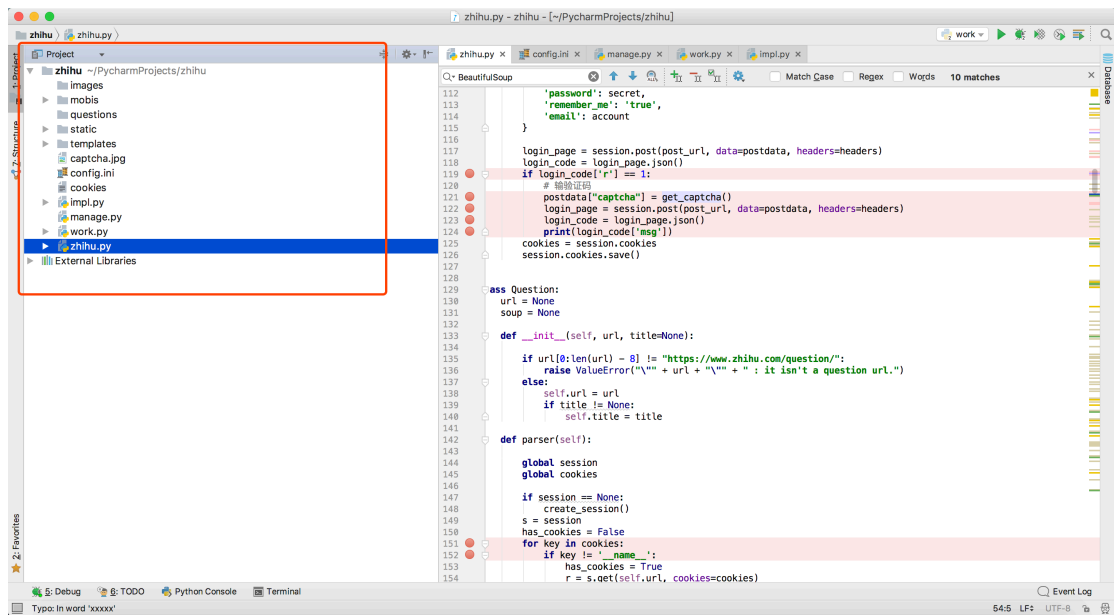
实现思路



用的东西

- ① flask, 提供 api, 接收客户端请求缓存到 redis。
- ② 线程脚本, 从 redis 循环取得用户请求, 根据 url 生成文本文件然后附件发送 mail。

用 BeautifulSoup 爬页面, 解析出想收集的内容, 转成 txt 或者 md
再用 book-convert 命令把文本转换成其他格式 (pdf, mobi),
smtplib 用 qq 邮箱发送邮件。
































impl.py 是 web 接口。

- ▼ impl.py
 - get_redis()
 - index()
 - send()

work.py 是循环线程，循环执行生成文本和发邮件两个方法

- ▼ work.py
 - generate(url=None, email=None)
 - get_redis()
 - ▼ MailToKindle(object)
 - __init__(self, mobi_path, email)
 - send_mail(self)
 - ▼ ZhihuToMobi(object)
 - __init__(self, question_id)
 - answer_to_md(self, answer)
 - md_to_mobi(self)
 - question(self, url)
 - question_to_md(self, url)

zhihu.py 是所有类定义，问题，回答，用户、

- ▼  zhihu.py
 - ▶  Answer
 -  create_session()
 -  get_captcha()
 -  get_xsrf()
 -  isLogin()
 - ▼  Question
 -  __init__(self, url, title=None)
 -  get_all_answers(self)
 -  get_answers_num(self)
 -  get_detail(self)
 -  get_followers_num(self)
 -  get_title(self)
 -  get_top_answer(self)
 -  get_top_i_answers(self, n)
 -  get_topics(self)
 -  parser(self)
 - ▼  User
 -  __init__(self, user_url, user_id=None)
 -  get_agree_num(self)
 -  get_answers(self)
 -  get_answers_num(self)
 -  get_asks(self)
 -  get_asks_num(self)
 -  get_followees_num(self)
 -  get_followers_num(self)
 -  get_thanks_num(self)
 -  get_user_id(self)
 -  parser(self)