1. Python环境
   1. 测试框架依赖python3.x, 推荐使用python3.5及以上版本
   2. 依赖第三方包列表：
      * 1. Selenium
        2. Sqlite3（一般由python自带）
2. 配置说明
   1. 在根目录中config.json中进行配置
   2. 配置说明如下：
      1. options:
         1. hidden\_ui: 该值指示是否显示selenium浏览器界面，如果测试脚本较多则不建议开启
         2. timeout: 该值指示操作的最大超时时间
      2. test\_script\_dir: (本配置为列表结构，可同时存在多项)
         1. path: 测试目录, 可为绝对地址, 如为相对地址地址前不加”/”
         2. deep\_search: 该值指示是否进行深度搜索, 如果开启该项会对指定文件夹内的文件夹进行深层搜索, 关闭后只会执行指定目录的测试脚本
         3. run\_times: 运行次数, 通过设定该值可对指定文件夹下的脚本进行并发测试(具体并发数根据电脑性能决定)
      3. browser\_type:
         1. chrome: 该值控制是否开启chrome测试, 需要系统安装chrome浏览器
         2. firefox: 该值控制是否开启firefox测试, 需要系统安装firefox浏览器
         3. ie: 该值控制是否开启ie测试, 注意ie没有无界面模式, 即hidden\_ui功能不会对ie生效, 使用前需要进行ie权限配置, 参考文档: <https://blog.csdn.net/a5650892/article/details/78380289>
         4. chrome360: 该值控制360急速模式是否开启, 需要系统安装360浏览器, 理论上该项可以运行全部chrome内核浏览器(未验证)
      4. Path:
         1. 360: 用于配置360浏览器的应用程序(exe)路径, 注意粘贴可能会带有特殊符号, 尽量手打以保证正确。理论上讲该项配置为任一chrome内核的浏览器都可以生效(未验证)
3. 使用说明
   1. 在配置目录下编写脚本, 完成后运行start\_up.py开始测试
   2. 脚本书写规则如下：
      1. 框架会选择脚本中”test”类的”run”方法作为启动函数, 注意大小写, 同时注意run需要为实例方法
      2. 框架会在run方法第一个参数传入selenium封装后的实例, 具体api参照/lib/HttpHelper.py
      3. 框架会在run方法第二个参数(可选)传入数据库操作实例, 具体api参照/lib/DataBaseHelper.py
      4. 数据库操作结果保存在根目录database.db文件中, 本框架使用sqlite进行数据库保存, 建议使用navicat premium进行操作及查看
   3. 其他操作可参照demo文件夹下示例脚本