“我们不能只靠少数的黄金妖精拯救世界。我们需要伟大的舰队与勇者的圣剑兵团！”

——最高武官 灰岩皮

各位无征种勇者们晚上早~！老朽名为艾瑟雅·麦杰·瓦尔卡利斯，二等武官。夜露死苦（请多指教）！

几个月前，地面上发现了两千或三千年前的传说之蟒蛇Python确实存在的证据。最不可思议的是，Python似乎会用魔力制作护符，并用护符量产各种各样的圣剑。过去的“人类”似乎就是从Python那里学到了不完全的圣剑制作方法。最近我们从古书和文物里解明了制造圣剑的深刻奥秘，并做出了少量试验产品，命名为大蛇圣剑。这些剑的威力当然不能与极位古圣剑相提并论，但即使是坐在轮椅上的普通生物也可以安全地使用这些剑。只要你有足够的精神力，而且用剑足够巧妙，你也可以大片大片地砍倒那些可恶的兽！假如你挥剑时出现不符合套路的失误（**比如咒语的语法不正确**），剑上附带的“**编译器**”通常会发现并阻止你做出错误动作。使用最基本套路攻击的话，你被兽击伤的可能性很低（是“很低”！）。但套路复杂起来以后，你可能会头昏脑胀，完全搞错攻击对象。**圣剑并不能阻止你用正确套路对错误的目标发起攻击，因为剑并不知道你要做什么。**如果你要正确使出一些更难的招数，可以使用“编译器”里附带的“**断点**”和“**调试**”来检查你挥剑动作的一切细节（之后再解释这两个东西）。大蛇圣剑唯一的副作用是，如果短时间内精神力消耗太多，你可能会疲倦、掉发或者秃头，还会有很小很小的概率猝死。嘛……这点牺牲相比曾经的那些可以说是微不足道啦！

**我们这次的目标只是尽快适应圣剑的基本用法。没必要浪费时间去理解每一个护符碎片。**我没有告诉你的事情，以后就让战场教你吧！

兽每过三五天就会袭击某个浮空岛，所以我们的练习时间非常有限。事实上有99.5%的东西需要各位在实战中习得。请各位无征种勇者勇敢参与演习，随时准备作战！

握剑之前，要准备好编译器及一些附属武装。**如果你的火控（操作）系统是32位的，请不要直接拿我准备的文件来用**！首先**下载Python（python-3.6.5.exe）**，**jdk（jdk-10.0.2\_windows-x64\_bin.exe）**和**eclipse（eclipse-java-photon-R-win32-x86\_64.rar）**（你可以暂时忽略python-2.7.12.msi）。Python是圣剑本体。eclipse可以听懂你的指令并驱动圣剑完成不可思议的攻击。jdk是eclipse运行的基础。下载完成后安装Python和jdk。特别注意！**安装Python不要一直点击“下一步”。要勾选pip和Add python.exe to Path**。

**目前还无法正确启动eclipse**。接下来我们要**配置系统变量（注意不是用户变量）**。请参照《Python爬虫开发与项目实战》的1.2.1节（pdf页码19页）。如果你使用Windows7火控系统，则系统变量各个路径之间要有分号(;)。而Windows10火控系统会直接列出每个系统变量的所有路径。

我可以提供一些我的系统变量。你可以把这些全部加进去。我的所有装备都在D盘。如果你装在C盘，把下面的D:\都改成C:\就好了。用快捷键Ctrl+H可立即把所有D:\替换为C:\（不要把所有D替换为C！）

JAVA\_HOME

D:\Program Files\Java\jdk-10.0.2

classpath

.;%JAVA\_HOME\lib;%JAVA\_HOME\lib\tools.jar

Path

D:\MinGW\bin;D:\Program Files (x86)\Tesseract-OCR\;D:\Program Files (x86)\Python36-32\Scripts\;D:\Program Files (x86)\Python36-32\;C:\Program Files\Mozilla Firefox; D:\Python27\;D:\Python27\Scripts;%JAVA\_HOME%\bin;%JAVA\_HOME%\jre\bin;D:\Program Files\nodejs\; C:\Windows\System32\;

如果以上都完成了，就可以打开eclipse。解压eclipse-java-photon-R-win32-x86\_64.rar后双击eclipse.exe。祝你启动愉快！

别急着高兴！eclipse目前的状态只能驱动Java圣剑。在eclipse菜单栏点击Help->Install New Software。请参考《Python爬虫开发与项目实战》从图1-15开始的内容！（pdf页码29页）你会用到的链接是<http://pydev.org/updates>。

书中会让你安装Python2.7和PyCharm。我们不需安装PyCharm，也暂时（或者永远）不需要2.7。

如果一路顺利的话你会新建一个Pydev Project，并在Project里新建一个Pydeb Module。Module里会提供一个init.py文档。现在就可以在这个文档里写代码啦！试试在文档里写下这样两句话，向你的圣剑发出第一条指令：

#coding:utf-8

print(‘Chtholly Nota Seniorious’)

注意不要写到下面的Console里去。然后请尽力在上方菜单栏找到这样两个按钮：



点击“播放键”运行。一切顺利的话就会在下面的Console里见到珂朵莉前辈哦！

接下来把我提供的ISML crawling Mk.II所有内容复制粘贴进去。你可以新建一个项目，也可以把刚才写的#coding:utf-8和print(‘Chtholly Nota Seniorious’)覆盖掉。

恭喜你完成了50%的工作！之后我们真的要握起圣剑了！圣剑在大多数情况下会一行一行地严格按顺序执行你的咒语。当然你也可以命令圣剑循环地执行某些咒语，直到某一条件被满足（也可以让圣剑永不停歇地循环下去，不过大多数情况下我们不需要这样）。或者，你也可以在某个条件已被满足的情况下让圣剑执行这一批语句，而在这个条件未被满足的情况下让圣剑执行另一批语句。

首先我们认识一下“**注释**”：**如果你在某一行的开头写一个井号#，则不论你在井号后面说些什么，都会被圣剑无视。**这种功能看起来好像没啥意思，但请注意**你写下的东西不仅是给圣剑看的，更是给自己、给别人看的**。利用注释（以及其他种种方式）让大家容易理解你的意思，在团队作战中特别有利哦！此外你也可以把暂时不需要用的咒语给注释起来，以后想要使用就很方便了！**如果你想写中文（不管是写在注释还是咒语里），则必须在开头写上**#coding:utf-8

然后是“**空行**”。空行同样会被圣剑无视，只是让你看得更清楚。你可以在任意两句咒语之间空1行，10行，100行，或者不留空行。只要你看得清楚就好了。但**写完一句之前请不要随意换行。这会破坏原来的意思。**

接下来我们会看到一大堆的import，或者from 某某 import 某某。被import的这些内容是饱含着前辈们血泪、力量与经验的特殊圣剑护符！**如果你import了某个东西，你就可以直接使用某位前辈为你准备好的力量。**比如说我import time之后，就可以随时使用咒语time.sleep(10)，让程序休息10秒，什么都不做。而from scipy.fftpack import fft这样的语句，可以让你直接喊出fft(一个数列)（fft是对一个数列做快速傅里叶变换的意思。不明白也没关系）。假如你一开始说的是import scipy，则以后每次做fft都必须喊出scipy.fftpack.fft(一个数列)。这相当麻烦！

还有一种import语句的写法是import A as B，例如import numpy as np，以及import matplotlib.pyplot as plt。这样你就可以用plt代替matplotlib.pyplot这一长串！

不过还有一件很遗憾的事：**你大概还没有获得你想要import的特殊护符碎片，因为你还没有安装！**想要借用各位前辈的力量，就要先获得那块护符碎片。你可以按win（Windows的徽标键，是四个格子，可能在左边的Ctrl和Alt键之间）+R键，在弹出的窗口中输入cmd，然后输入

pip install 你想安装的护符碎片名称

比如

pip install scipy

按下回车即可开始安装！正常情况应该出现进度条和安装链接，不会出现红字，也不会出现“pip不是可执行的命令”之类。

或者你可以在eclipse的菜单中点击Window→Preferences→从左边一列展开Pydev下的内容→展开Interpreters→点击Python Interpreter→选中Python3.6编译器→点击Install/Uninstall with pip，然后install 你想安装的护符碎片名称。

import的名称与pip安装的名称不一定完全一样。如果在使用pip的时候遇到异常，且百度不能解决，可以找艾瑟雅哦！

**尽量安装上我import过的所有东西。**

接下来我们直接跳到程序的第219行。内容为

for j in range(2041,10001):

**for语句**特别有趣。这里它会首先让一个变量j等于2041，然后执行后面的东西。“后面的东西”执行完以后，for语句又会让j等于2042，再执行一边“后面的东西”。这样一直做到j=10000（不是j=10001！），后面的语句会最后一次被执行。

我们使用**缩进(Tab键)**来控制for语句的作用范围。要注意一件事：既然for语句会反复执行后面一堆语句，那么它反复执行的到底是哪一堆？比如说：

for i in range(1,11):

j=1

k=3.14159

L=2.71828

…

m=’我永远喜欢珂朵莉’

我们到底是循环地执行j=1，还是循环地执行j=1和k=3.14159，还是循环地执行j=1 k=3.14159 L=2.71828这三句话呢？还是说m=’我永远喜欢珂朵莉’也要被循环执行呢？事实上我们看到j,k,L三句咒语都被缩进了，m这句则没有缩进，其起始位置与for相同。这样我们就会循环执行j,k,L这三句话。每执行完这三句话，i就增加1，然后这三句话又会被执行。i等于10时，这三句话最后一次被执行。i变为11后，不再执行j,k,L这三句话，而执行一次m=’我永远喜欢珂朵莉’，并继续按顺序向后执行。你可以选中10行，按一下Tab，然后这10行全都会被缩进一格。按Shift+Tab可以让这10行全部减少一格缩进。

for j in range(2041,10001):的功能是，试图循环访问ISML网站很多次，截取一大堆验证码。

然后我们看第245行的

if(20>gdi32.GetPixel(hdc,380,431)):

这句**if语句**是说从屏幕左上角向右380像素点，再向下431像素点，这个像素的颜色**如果**小于20（足够偏黑色），那么执行allowed=1以及break。如果这个像素颜色不那么小（大于等于20），那么(**else:**)执行time.sleep(1)以及allowed=0。break是指立即退出一层循环，在此就是指for i in range (1,30)这层循环。此循环会不停检查上述像素点的颜色是否足够黑。够黑则说明ISML网站很可能加载出来了，我们立即允许程序做之后的工作。

接下来看第271行的while语句。while也可以实现循环。如果后面一批语句执行完成后，allowed<=6成立，则重新执行后面一批语句。while可以轻松实现无限循环。只要写while(True)就可以了（除非break掉）！刚使用圣剑的人很容易忘记在while循环里面改变被判断的变量的值。比如我们在这里应该要让allowed有所变化，才能保证循环会被退出。277行的allowed+=1是让allowed这个变量的值增加1。你也可以写allowed=allowed+1。

最后我们学习如何让圣剑完成具体的功能！这套咒语会用你的**默认浏览器**打开ISML投票网址，然后模拟鼠标键盘来“浏览”，最后对着验证码位置截图。以下是具体操作使用的咒语：

**鼠标移动到某个坐标：**

win32api.SetCursorPos([1898,12]) #可以把鼠标指针移动到坐标[1898,12]，这是窗口右上角的红叉，点击可以关闭窗口

**鼠标左键单击：**

win32api.mouse\_event(win32con.MOUSEEVENTF\_LEFTUP | win32con.MOUSEEVENTF\_LEFTDOWN, 0, 0, 0, 0)

#不用管这是什么意思。只要写这句话，就是单击

**按下PageDown键：**

shell.SendKeys(*"{PGDN}"*, 0)

**截图：（保存为filename。filename是装进了一些文字的变量。文字里要带有.png）**

window\_capture(filename)

**设定截图位置及横向宽度和纵向高度：**

看第110行。w是图片宽度，h是图片高度，x和y是图片左上角的坐标（到屏幕左上角的横向距离和纵向距离）。

程序里还会给出很多解释说明。如果是没有任何说明的语句，请不要随便修改。我们没有讨论过的，以def开头的咒语块，也请不要随便改（除了设定截图参数）。各位勇者的任务很简单：**设定适当的鼠标、键盘和截图动作，尽可能快速精准地截得验证码图片并保存下来。**你可以用浏览器或qq的截图工具测量屏幕上某一点到屏幕左上角的坐标。请各位勇者自行修改程序。

艾瑟雅祝大家活着回来！