1. 结构：

顶层：任意文件夹

---run.py //程序入口

---xml.xml //策略配置文件

---settings.xml //入口配置文件

---schema.xml //信息抽取配置文件  
 ---src //代码

---logs //日志

---网站1

---站名.txt

---company.txt

---job.txt

1. 启动：
   1. 填写、确认settings.xml中的参数
   2. 运行run.py
2. 配置文件：

Xml.xml：

框架：

project：标识某个网站的爬取策略

urlpattern：符合某种正则式的链接

pattern：链接地址的正则表达式

model：链接的处理方式

args：处理链接的参数

参数：

model为enter：

loc：标识获取网页文本的范围，从loc[0]到loc[1]

model为need：

ctg：表示类别，company为公司链接，job为职位链接

model为auto：

between：链接地址替换位置

replace：替换序列

submodel：系列链接的处理模式

其他：submodel为enter参照model enter，为need的参照model

need

规则：

1. project有两个属性，name为策略名，同时是其唯一标识；sleeptime为爬取该网址的休眠时间。
2. 每个project的第一个urlpattern的pattern为入口（种子）链接实际地址，不带参数。第二个为其正则式，带完整参数。
3. pattern要求符合正则表达式规范，原文中带“?”替换为“\?”。
4. model只识别“enter”、“need”、“auto”三种类型
5. args中参数要求符合json和xml规则，无“<”、“>”、单引号，双引号要求转义、“；”替换为“&amp;”
6. auto的args中，如果replace长度为2，则表示范围，如[2,200]表示从2到200,；否则，表示枚举，遍历替换。

Schema：

* 1. Field name一定要与solr中的field name一致
  2. Project 是网站句柄，标识一个网站
  3. 信息定位有两种方式：
     1. Tagname+attr+num：标签名+属性值+次序 定位---<td class=”aa”> tagname=td,attr={“class”:”aa”}
     2. 两个值之间定位：between={“left”:”right”}
        1. 支持一对多模式定位：between={“left”:[“right1”,”right2”….]}
        2. 一对多时选择长度最短的文本
  4. 支持最多二次定位，自动识别：
     1. 范例<field name=”name”……>

<sub between={…… />或者<sub tagname=…… />

</field>

* 1. 支持动态修改field：
     1. Field name 可以依据业务和网站情况自由添加和设置，只要确保和solr库属性一致即可。

Settings.xml：

List：内部是每个网站爬取的参数配置

Name：网站句柄。

Thread：分配的同时连接线程数量

Test：是否进行休眠时间检测

Solr：solr库路径配置

1. src文件
   1. strategy 爬取策略核心
   2. route 读取和处理策略配置文件内容（xml.xml）
   3. BloomFilter.py 查重模块
   4. block.py模块 sleeptime检测
   5. proxy 代理池操作模块
   6. deal 文本信息抽取和预处理模块（schema.xml）