**selenium之chrome浏览器设置userAgent和代理ip**

**1.设置user-agent**

|  |
| --- |
| Python  option = ChromeOptions()  #设置无头模式  option.add\_argument("--headless")  option.add\_experimental\_option('excludeSwitches', ['enable-automation'])  option.add\_experimental\_option('useAutomationExtension', False)  #设置user-agent  option.add\_argument('user-agent=ywy')  driver = webdriver.Chrome(options=option) |

**2.通过user-agent池替换user-agent**

|  |
| --- |
| Python pip install fake\_useragent   from fake\_useragent import UserAgent   headers = { ‘User-Agent’: UserAgent().random, } |

**3.设置代理ip**

**3.1 连接无用户名密码认证的代理**

|  |
| --- |
| Python chromeOptions = webdriver.ChromeOptions() chromeOptions.add\_argument('--proxy-server=http://ip:port')  driver = webdriver.Chrome(chrome\_options=chromeOptions) |

**3.2有用户名和密码的连接**

|  |
| --- |
| Python from selenium import webdriver def create\_proxyauth\_extension(proxy\_host, proxy\_port,  proxy\_username, proxy\_password,  scheme='http', plugin\_path=None):  """Proxy Auth Extension   args:  proxy\_host (str): domain or ip address, ie proxy.domain.com  proxy\_port (int): port  proxy\_username (str): auth username  proxy\_password (str): auth password  kwargs:  scheme (str): proxy scheme, default http  plugin\_path (str): absolute path of the extension    return str -> plugin\_path  """  import string  import zipfile    if plugin\_path is None:  plugin\_path = 'd:/webdriver/vimm\_chrome\_proxyauth\_plugin.zip'    manifest\_json = """  {  "version": "1.0.0",  "manifest\_version": 2,  "name": "Chrome Proxy",  "permissions": [  "proxy",  "tabs",  "unlimitedStorage",  "storage",  "<all\_urls>",  "webRequest",  "webRequestBlocking"  ],  "background": {  "scripts": ["background.js"]  },  "minimum\_chrome\_version":"22.0.0"  }  """    background\_js = string.Template(  """  var config = {  mode: "fixed\_servers",  rules: {  singleProxy: {  scheme: "${scheme}",  host: "${host}",  port: parseInt(${port})  },  bypassList: ["foobar.com"]  }  };   chrome.proxy.settings.set({value: config, scope: "regular"}, function() {});   function callbackFn(details) {  return {  authCredentials: {  username: "${username}",  password: "${password}"  }  };  }   chrome.webRequest.onAuthRequired.addListener(  callbackFn,  {urls: ["<all\_urls>"]},  ['blocking']  );  """  ).substitute(  host=proxy\_host,  port=proxy\_port,  username=proxy\_username,  password=proxy\_password,  scheme=scheme,  )  with zipfile.ZipFile(plugin\_path, 'w') as zp:  zp.writestr("manifest.json", manifest\_json)  zp.writestr("background.js", background\_js)    return plugin\_path   proxyauth\_plugin\_path = create\_proxyauth\_extension(  proxy\_host="proxy.crawlera.com",  proxy\_port=8010,  proxy\_username="username",  proxy\_password="password" )     co = webdriver.ChromeOptions() co.add\_argument("--start-maximized") co.add\_extension(proxyauth\_plugin\_path)     driver = webdriver.Chrome(chrome\_options=co) driver.get("http://www.amazon.com/") |

**4.** [**Selenium chrome配置不加载图片**](https://www.cnblogs.com/roystime/p/6935585.html)

|  |
| --- |
| Python prefs = {"profile.managed\_default\_content\_settings.images":2} chrome\_options.add\_experimental\_option("prefs",prefs) |