反方向的座位

基于 Selenium 的网络爬虫基操

有个需求会用到爬虫, 于是使用 Selenium 库学习了一下大致的步骤, 基本实现了要爬取的功能后总结一下如何使用

安装包

首先在 nuget 中安装 `Selenium.WebDriver` 和 `Selenium.WebDriver.GeckoDriver`(火狐) , 咕咕噜用户选择 `Selenium.WebDriver.ChromeDriver`

开启浏览器

先根据自己需要设置浏览器启动参数

```
public static FirefoxOptions SetFirefoxOption()
{
    var options = new FirefoxOptions();
    options.AddArgument("--start-maximized");
    options.AddArgument("--ignore-certificate-errors");
    return options;
}
```

new 一个浏览器出来, IWebDriver 可以用来模拟用户在浏览器中的行为

```
using (IWebDriver driver = new FirefoxDriver(SetFirefoxOption()))
{
    try
    {
        // 浏览器要干的事
    }
    catch (Exception ex)
    {
        Console.WriteLine($"连接时发生错误: {ex.Message}");
    }
    finally
    {
        driver.Quit();
    }
}
```

导航到地址

使用 `driver.Navigate().GoToUrl()` 方法导航到目标地址

```
public static void OpenMainPage(IWebDriver driver, string url)
{
    // 打开主页
    driver.Navigate().GoToUrl(baseUrl);

    // 等待页面加载完成
    wait = new WebDriverWait(driver, TimeSpan.FromSeconds(10));
    wait.Until(d => d.FindElement(By.ClassName("content-box")));

    Console.WriteLine("页面加载完成");
}
```

查找控件

使用 `IWebDriver.FindElement()` 方法来查找页面中的元素, 常用的查找有 `ClassName`, `CssSelector`, `TagName` 等

我习惯的做法是

- 先在浏览器中选中目标, 这时候在 `DOM与样式检测器` 中会显示目标在 `DOM树` 中的层级
- 按照这个层级去 `主控台` 中使用 `querySelector()` 查找
- 不断缩减 `css selector` 中的内容达到最简
- 改为代码

例如

我想要获取一系列图片的地址, 于是先用 `css selector` 定位到了对应的 img

```
document.querySelector("html body.shop-style-02 div.main-box div.main div.main-top
div.major-function-box div.exhibition ul.sm-list li img")
```

缩减后

```
document.querySelectorAll("ul.sm-list li img")
```

然后使用代码查找, 并执行后续的相关逻辑

```
public static void GetProductImages(IWebDriver driver)
{
    try
    {
        var imgElements = driver.FindElements(By.CssSelector("ul.sm-list li img")).ToList();
```

```
        Console.WriteLine("找到的图片 URL:");

        var URLs = imgElements
            .Select(img => img.GetAttribute("src"))
            .Where(src => src != null && src.EndsWith("_s.jpg"))
            .Select(src => src.Replace("_s.jpg", "_n.jpg"))
            .Distinct()
            .ToList();

        URLs.ForEach(Console.WriteLine);

        imageURLs = URLs;
    }
    catch (Exception ex)
    {
        Console.WriteLine($"获取图片链接时发生错误: {ex.Message}");
    }
}
```

切换页面

有时候点击链接会打开一个新的标签页, 这时候 `IWebDriver` 依然会停留在原标签页, 直接使用会导致元素找不到, 需要手动切换

```
// 先获取当前窗口句柄
string currentWindowHandle = driver.CurrentWindowHandle;

// 获取所有打开的窗口句柄
var windowHandles = driver.WindowHandles;

// 切换到新标签页
windowHandles.ToList().ForEach(windowHandle =>
{
    if (windowHandle != currentWindowHandle) driver.SwitchTo().Window(windowHandle);
});
```

我始终保持最多两个页面所以上方代码可行, 如果存在较多页面请尝试将打开标签页前后的 `WindowHandles` 都存下来比较差异

Index

Index