



Av Erik
"Zkall"
Svensson
@esdev.se



Selenium aka Botar

Vad är selenium?

Det är en programvara för att göra browser bottar med. Eller också kallade automatiserade tester.

Hur fungerar selenium?

Du initierar en chromedriver som är ett msdos program som sedan "styr" webbläsaren på dina kommandon som du säger åt den att göra. Du kan komma åt html på en sida och substringa ut bildurler och ladda ner dom och spara i en databas. Du kan klicka på element som har x class namn eller y id, du kan skriva in text i textboxar, välja listviews etc. Du kan göra det mesta.

Exempel på användningsområden är en googlebot som googlar bilder från en lista och laddar ner och lägger dom i en databas för senare användning.

Ett annat exempel är en dejting bot som flirtar med tjejer och är driven utav selenium. Exempel om ovanstående kommer nedanför.

Happy Pancake Flirt Bot

OpenQA.Selenium.Chrome.ChromeDriver driver = new
OpenQA.Selenium.Chrome.ChromeDriver(); **Initiliserar**
och startar chromedriver.

```
public void Login(string username, string password)
{
    driver.Url = "http://happypancake.com/login";
    Thread.Sleep(1000);
var loginfield =
driver.FindElementById("ContentPlaceHolderDefault_mainContent_UserName");
var passwordfield =
driver.FindElementById("ContentPlaceHolderDefault_mainContent_Password");
var LoginButton =
driver.FindElementById("ContentPlaceHolderDefault_mainContent_lbLogin");

    loginfield.SendKeys(username);
passwordfield.SendKeys(password);
    LoginButton.Click();
    Thread.Sleep(1000);
}
```

Här har jag då kollat upp vad knapparn och fälten jag vill komma åt på happy pancake heter. Man skriver `driver.findElementById("id")` för att hitta ett element efter en viss id.

För att göra det mer läsbart kan du deklarerat fälten och knapparna som variabler och sedan kalla `variabel.Click()` eller `variabel.SendKeys(keys)`.

```
driver.Url = "http://www.happypancake.com/sok/";
Thread.Sleep(1000);
```

```
driver.FindElementById("ContentPlaceholderDefault_PageContent_MainContent_InkAdvancedSearch").Click();
Thread.Sleep(2000);
```

```
driver.FindElementById("ContentPlaceholderDefault_PageContent_MainContent_chkLookingForMe").Click();
driver.FindElementById("ContentPlaceholderDefault_PageContent_MainContent_drpMinAge").SendKeys(startage.ToString());
```

```
driver.FindElementById("ContentPlaceholderDefault_PageContent_MainContent_drpMaxAge").SendKeys(stopage.ToString());
```

```
driver.FindElementById("ContentPlaceholderDefault_PageContent_MainContent_drpProvince").SendKeys(län);
driver.FindElementById("ContentPlaceholderDefault_PageContent_MainContent_chkPic").Click();
```

```
driver.FindElementById("ContentPlaceholderDefault_PageContent_MainContent_chkOnline").Click();
Thread.Sleep(1000);
```

```
driver.FindElementById("ContentPlaceholderDefault_PageContent_MainContent_InkBtnSearch").Click();
Thread.Sleep(1000);
var html = driver.PageSource;
urllist = new List<string>();
```

Här sket jag i läsbarheten och skrev click på variablerna på igång. Samma som i förra exemplet i stort sett bara en annan sida, med `Thread.Sleep()` för att se till att botten inte försöker göra saker med en sida innan den laddat klart.

```
for (int f = 1; f < 10; f++)
{
    try
    {
        string[] split = Regex.Split(html, "/visa-profil/");
        for (int i = 1; i < split.Count(); i++)
        {
            string url = "http://www.happypancake.com/visa-profil/" +
split[i].Substring(0, split[i].IndexOf("\""));
            if (!url.Contains("onlinelist"))
urllist.Add(url);
        }
        driver.FindElementsByClassName("pager-row")[f].Click();
        Thread.Sleep(2000);
    }
    catch
```

```

    {
    }
}

```

Här gör jag en dubbel for loop som tar htmlen splitar den efter /visa-profil/ som indikerar att det är en länk till en profil. Sedan så kollar jag om den innehåller "onlinelist" då det är från menyn och visas på samma sida varje gång så dom filtreras bort. I slutet hittar den pager-row classen som är nästa sid klick. Och så här loopar den 10 gånger om tills den är klarläst 10 sidor. Addar dom slutligen i en list med string länkar.

```

var reallist = new List<string>();
var _happypancake = db.ProfilerSet.ToList();
for (int i = 0; i < urllist.Count; i++)
{
    bool itexists = false;
    foreach (var item in _happypancake)
    {
        if (item.profilurl == urllist[i])
        itexists = true;
    }
    foreach (var item in reallist)
    {
        if (item == urllist[i])
        itexists = true;
    }
    if (!itexists)
        reallist.Add(urllist[i]);
}

```

Här så skapar jag en ny lista som jag kallar reallist där jag filtrerar ut dom som finns i databasen där ja lagrar profil länkar jag besökt. Filtrerar även ut dubletter.

```

foreach (var item in reallist)
{
    db.ProfilerSet.Add(new ProfilerSet { profilurl = item });
} db.SaveChanges();

```

Här lägger jag till alla nya profiler i databasen. Sedan så:

```

for (int i = 0; i < reallist.Count; i++)
{
    driver.Url = reallist[i];
    Thread.Sleep(2000);
    driver.PageSource;
    {
        var _html =
        try
        {
            if (!_html.Contains("Ja</dd>"))
                driver.FindElementByLinkText("Flirta").Click();
            Thread.Sleep(1000);
        }
        catch
        {
        }
    }
}

```

Går jag till urlerna en för en, sover 2 sekunder sedan hittar jag flirtnappen och trycker den om jag inte haft någon tidigare interaktion med denna person. Sedan sover jag 1 sekund och klickar vidare på nästa.

Avslut

Väldigt praktiskt att använda, jag lyckades spara ner 12 000 profiler i en databas med bilder, kön, användarnamn när jag hade samlat på mig lite urler.