

mechanize

# Create a browser object

Create a browser object and give it some optional settings.

```
import mechanize  
br = mechanize.Browser()  
br.set_all_readonly(False) # allow everything to be written to  
br.set_handle_robots(False) # ignore robots  
br.set_handle_refresh(False) # can sometimes hang without this  
br.addheaders = [('User-agent', 'Firefox')]
```

# Open a webpage

Open a webpage and inspect its contents

```
response = br.open(url)
print response.read()    # the text of the page
response1 = br.response() # get the response again
print response1.read()   # can apply lxml.html.fromstring()
```

# Using forms

List the forms that are in the page

```
for form in br.forms():  
    print "Form name:", form.name  
    print form
```

To go on the mechanize browser object must have a form selected

```
br.select_form("form1")    # works when form has a name  
br.form = list(br.forms())[0] # use when form is unnamed
```

# Using Controls

Iterate through the controls in the form.

```
for control in br.form.controls:  
    print control  
    print "type=%s, name=%s value=%s" % (  
        control.type,  
        control.name,  
        br[control.name])
```

Controls can be found by name

```
control = br.form.find_control("controlname")
```

# Using Controls

Having a select control tells you what values can be selected

```
if control.type == "select":  
    for item in control.items:  
        print " name=%s values=%s" % (  
            item.name, str([label.text  
                for label in item.get_labels()])))
```

Because 'Select' type controls can have multiple selections, they must be set with a list, even if it is one element.

```
control.value = ["ItemName"]  
br[control.name] = ["ItemName"]
```



# Using Controls

Text controls can be set as a string

```
if control.type == "text": # means it's class ClientForm.TextControl
    control.value = "stuff here"
br["controlname"] = "stuff here" # equivalent
```

Controls can be set to readonly and disabled.

```
control.readonly = False
control.disabled = True
```

OR disable all of them like so

```
for control in br.form.controls:
    if control.type == "submit":
        control.disabled = True
```

# Submit the form

When your form is complete you can submit

.

```
response = br.submit()  
print response.read()  
br.back() # go back
```



# Finding Links

Following links in mechanize is a hassle because you need to have the link object. Sometimes it is easier to get them all and find the link you want from the text.

```
for link in br.links():  
    print link.text, link.url
```

Follow link and click links is the same as submit and click

```
request = br.click_link(link)  
response = br.follow_link(link)  
print response.geturl()
```