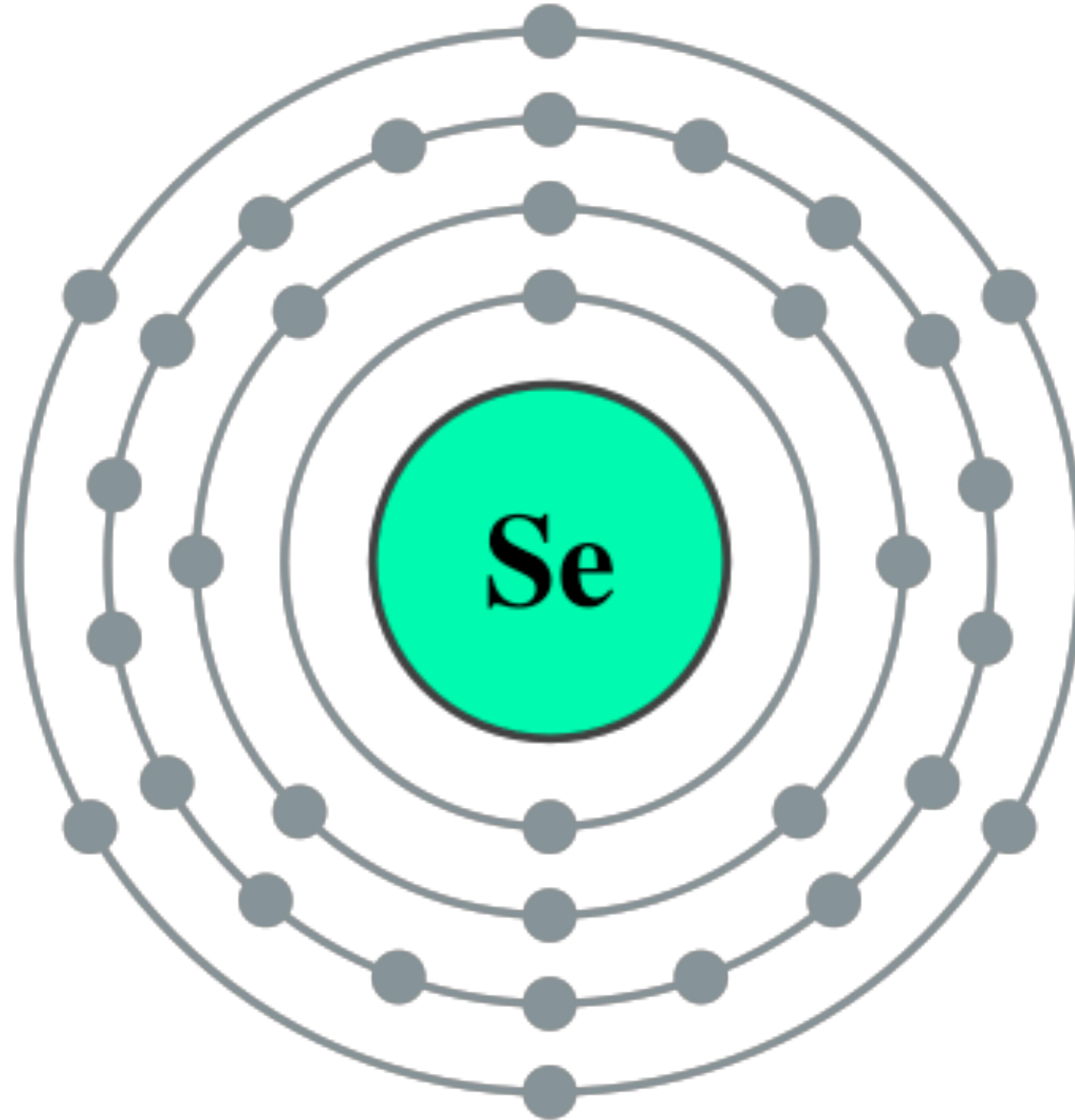


selenium



```
require "test_helper"
```

```
class LoginScreenTest < Test::Unit::TestCase
```

```
  context "When going to the Slicehost Manage website" do
```

```
    setup do
```

```
      @login_screen = LoginScreen.new(initiate_selenium)
```

```
      @login_screen.go
```

```
    end
```

```
    should "be able to login" do
```

```
      @login_screen.email.text = "myemail@mydomain.com"
```

```
      @login_screen.password.text = "s3cr3t"
```

```
      @slices_screen = @login_screen.login_button.click
```

```
      assert_equal @slices_screen.title, "my title"
```

```
    end
```

```
  teardown do
```

```
    terminate_selenium
```

```
  end
```

```
end
```

```
end
```

```

require "test/unit"
require "rubygems"
gem "selenium-client", ">=1.2.9"
require "selenium/client"
require "shoulda"

require File.expand_path(File.dirname(__FILE__) + "../screen_models/manage/login_screen")
require File.expand_path(File.dirname(__FILE__) + "../screen_models/manage/on_slice")
require File.expand_path(File.dirname(__FILE__) + "../screen_models/manage/backup_screen")
require File.expand_path(File.dirname(__FILE__) + "../screen_models/manage/new_slice_screen")
require File.expand_path(File.dirname(__FILE__) + "../screen_models/manage/slices_screen")
require File.expand_path(File.dirname(__FILE__) + "../screen_models/manage/slice_screen")

# eventually we can use this:
# $: << "Folder path"
# this will load all ruby files in that folder
# to the "framework" path

def initiate_selenium()
  if $selenium
    @selenium = $selenium
  else
    @selenium = Selenium::Client::Driver.new("localhost", 4444, "*chrome", "https://
manage.slicehost.com/", 10000);
    @selenium.start
  end
  return
end

def terminate_selenium()
  @selenium.stop unless
end

```

```
require File.expand_path(File.dirname(__FILE__) + '/../base/base')

module LoginScreenHelper
  def common_login
    self.go
    self.email.text = "myemail@mydomain.com"
    self.password.text = "s3cr3t"
    return self.login_button.click
  end
end

class LoginScreen
  include Base
  include LoginScreenHelper

  def go
    page.open "/login"
  end

  def email
    return Textbox.new("email", page)
  end

  def password
    return Textbox.new("password", page)
  end

  def login_button
    return Button.new("commit", page, "Slices")
  end

  def logout
    click_link "Logout"
  end
end
```

```
require File.expand_path(File.dirname(__FILE__) + "/slice_ssh")
require File.expand_path(File.dirname(__FILE__) + "/"
common_basic_actions")
require File.expand_path(File.dirname(__FILE__) + "/elements")
```

```
module Base
  include SliceSSH
  include CommonBasicActions

  attr_reader :page

  def initialize(selenium)
    page = selenium
  end
end
```

