**test\_add\_group**

**import** pytest  
**from** fixture.application **import** Application  
**from** model.group **import** Group  
  
  
@pytest.fixture  
**def** app(request):  
 fixture = Application()  
 request.addfinalizer(fixture.destroy)  
 **return** fixture  
  
**def** test\_test\_add\_group(app):  
  
  
  
 app.session.login(username=**"admin"**, password=**"secret"**)  
  
 app.group.create(Group(name=**"knlnölnlmlml"**, header=**"GUBJHB"**, footer=**"lknlknkln"**))  
  
 app.session.logout()  
  
  
**def** test\_\_add\_empty\_group(app):  
  
  
  
 app.session.login( username=**"admin"**, password=**"secret"**)  
  
 app.group.create(Group(name=**""**, header=**""**, footer=**""**))  
  
 app.session.logout()

application

**from** selenium.webdriver.firefox.webdriver **import** WebDriver  
**from** fixture.session **import** SessionHelper  
**from** fixture.group **import** GroupHelper  
  
**class** Application:  
  
 **def** \_\_init\_\_(self):  
 self.wd = WebDriver(firefox\_binary=**"C:\\Program Files\\Mozilla Firefox\\firefox.exe"**)  
 self.wd.implicitly\_wait(60)  
 self.session = SessionHelper(self)  
  
 *#driver wird ein einziges Mal inizilisiert bei der Erschaffung einer Fixture.  
 #ein Helper übernimmt einen Link auf ein Objekt der Klasse Application  
 #was uns die Möglichkeit gibt über einen Helper uns zu anderem wenden* self.group= GroupHelper(self)  
  
  
  
  
 **def** open\_home\_page(self):  
 wd = self.wd  
 wd.get(**"http://localhost/addressbook/group.php"**)  
  
 **def** destroy(self):  
 self.wd.quit()

session

*'''  
ein Class for login und logout session  
'''***class** SessionHelper:  
 **def** \_\_init\_\_(self, app):  
 self.app = app  
  
 **def** login(self, username, password):  
 wd = self.app.wd *# Zugang zu Driver wird benötigt* self.app.open\_home\_page()  
 wd.find\_element\_by\_name(**"user"**).click()  
 wd.find\_element\_by\_name(**"user"**).clear()  
 wd.find\_element\_by\_name(**"user"**).send\_keys(username)  
 wd.find\_element\_by\_name(**"pass"**).click()  
 wd.find\_element\_by\_name(**"pass"**).clear()  
 wd.find\_element\_by\_name(**"pass"**).send\_keys(password)  
 wd.find\_element\_by\_xpath(**"//form[@id='LoginForm']/input[3]"**).click()  
  
  
 **def** logout(self):  
 wd = self.app.wd  
 wd.find\_element\_by\_link\_text(**"Logout"**).click()

fixture/group

*'''  
ein Class für Hilfsmethoden  
'''***class** GroupHelper:  
  
 **def** \_\_init\_\_(self,app):  
 self.app=app  
  
  
  
 **def** return\_to\_group(self):  
 wd = self.app.wd  
 wd.find\_element\_by\_link\_text(**"group page"**).click()  
  
 **def** create(self, group):  
 wd = self.app.wd  
 self.open\_groups\_page()  
 *# init group creation* wd.find\_element\_by\_name(**"new"**).click()  
 *# fill group form* wd.find\_element\_by\_name(**"group\_name"**).click()  
 wd.find\_element\_by\_name(**"group\_name"**).clear()  
 wd.find\_element\_by\_name(**"group\_name"**).send\_keys(group.name)  
 wd.find\_element\_by\_name(**"group\_header"**).click()  
 wd.find\_element\_by\_name(**"group\_header"**).clear()  
 wd.find\_element\_by\_name(**"group\_header"**).send\_keys(group.header)  
 wd.find\_element\_by\_name(**"group\_footer"**).click()  
 wd.find\_element\_by\_name(**"group\_footer"**).clear()  
 wd.find\_element\_by\_name(**"group\_footer"**).send\_keys(group.footer)  
 *# submit group creation* wd.find\_element\_by\_name(**"submit"**).click()  
 self.return\_to\_group()  
  
 **def** open\_groups\_page(self):  
 wd = self.wd  
 wd.find\_element\_by\_link\_text(**"groups"**).click()

model/group

**class** Group:  
  
 **def** \_\_init\_\_(self, name, header, footer):  
 self.name = name  
 self.header = header  
 self.footer = footer

Я правильно понимаю- класс Application будет общим и для контактов и для групп, но кроме

self.session = SessionHelper(self)

self.group= GroupHelper(self)

туда будет входит еще и помощник для контактов ?