

Testowanie aplikacji internetowych z wykorzystaniem Selenium Webdriver oraz języka Python



Przypadek testowy nr 1

Tytuł: Przypadek testowy nr 1: Rejestracja nowego klienta używając niepoprawnego adresu e-mail.

Środowisko: Chrome wersja 80.0.3987.106, Ubuntu 18.04 3 LTS

Warunek wstępny: Uruchomiona przeglądarka – strona internetowa <http://douglas.pl/>.

Użytkownik nie jest zalogowany.

KROKI:

1. Po otwarciu przeglądarki zaakceptuj cookies
2. Wyszukaj przycisk "Mój Douglas" i kliknij
3. Wybierz płeć
4. Wpisz Imię i Nazwisko
5. Wpisz hasło
6. Powtórz hasło
7. Wpisz email (niepoprawny)
8. Kliknij Rejestracja

Oczekiwany rezultat: Rejestracja nie powodzi się Użytkownik dostaje informację, że wprowadzony email nie jest poprawny.

Przypadek testowy nr 2

Tytuł: Przypadek testowy nr 2: Założenie karty stałego klienta, używając niepoprawnego kodu captcha

Środowisko: Chrome wersja 80.0.3987.106, Ubuntu 18.04 3 LTS

Warunek wstępny: Uruchomiona przeglądarka – strona internetowa <http://douglas.pl/>.

Użytkownik nie jest zalogowany.

KROKI:

1. Po otwarciu przeglądarki zaakceptuj cookies
2. Znajdź na stronie ikonkę z napisem "Karta Douglas" i kliknij ją
3. Kliknij "Wypełnij już dziś"
4. Wpisz Imię
5. Wpisz Nazwisko
6. Wybierz płeć
7. Wpisz datę urodzenia
8. Wybierz z listy kraj
9. Wybierz województwo
10. Wpisz ulicę i nr domu
11. Wpisz kod pocztowy
12. Wpisz miasto
13. Wpisz nr telefonu komórkowego
14. Wpisz email
15. Powtórz email
16. Zaznacz zapoznanie się z regulaminem i zgodę na przetwarzanie danych
17. Wprowadź kod captcha (nieprawidłowy)
18. Kliknij WYŚLIJ

Oczekiwany rezultat: Karta nie zostaje założona. Użytkownik dostaje informację, że wprowadzony kod captcha nie jest poprawny.

Automatyzacja przypadków testowych nr 1 i nr 2:

```
# -*- coding: utf-8 -*-
import unittest
from selenium import webdriver
from time import sleep
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
```

```
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.support.ui import Select
```

```
# Dane testowe:
```

```
plec = "K"
imie = "Jolanta"
nazwisko = "Wesoła"
haslo = "Jolanta8@"
powtorz_haslo = "Jolanta8@"
niepoprawny_email = "JWesolagmail.com"
dzien_urodzenia = "02"
miesiac_urodzenia = "03"
rok_urodzenia = "1990"
ulica = "Poziomkowa"
nr_domu = "6"
kod_pocztowy = "43100"
miasto = "Tychy"
nr_telefonu = "123456789"
email = "JWesola@gmail.com"
captcha = "AbCd"
```

```
class DouglasRegistrationAndCard(unittest.TestCase):
```

```
    # Scenariusz testowy:
```

```
    # Test 1: Rejestracja nowego klienta
```

```
    # Test 2: Założenie karty stałego klienta
```

```
    def setUp(self):
```

```
        # Warunki wstępne:
```

```
        # Przeglądarka otwarta na stronie: http://douglas.pl/
```

```
        self.driver = webdriver.Chrome()
```

```
        self.driver.maximize_window()
```

```
        self.driver.get("http://douglas.pl/")
```

```
    def tearDown(self):
```

```
        self.driver.quit()
```

```
    def test_invalid_registration(self):
```

```
        driver = self.driver
```

```
        # Rejestracja nowego klienta
```

```
        # KROKI:
```

```
        # 1. Po otworzeniu przeglądarki zaakceptuj cookies
```

```
        cookies = WebDriverWait(driver, 10).until(EC.element_to_be_clickable((By.ID, "uc-  
btn-accept-banner")))
```

```
        cookies.click()
```

```
        # 2. Wyszukaj przycisk "Mój Douglas" i kliknij
```

```
        Moj_Douglas = driver.find_element_by_xpath('//*[@class="rd_nav-item rd_nav-item--  
80"])[contains(text(), "Mój Douglas")]')
```

```
        Moj_Douglas.click()
```

```
        # 3. Wybierz płeć
```

```
        if plec == 'K':
```

```
            # Wybierz PANI
```

```
            K = cookies = WebDriverWait(driver,
```

```
            10).until(EC.element_to_be_clickable((By.XPATH, '//label[@class="rd_form-  
field_label"])[text()="Pani"]'))
```

```
            K.click()
```

```
        else:
```

```
            # WYBIERZ PAN
```

```
            M = driver.find_element_by_xpath('//*[@class="rd_form-field_label"]  
[text()="Pan"]')
```

```
            M.click()
```

```
        # 4. Wpisz Imię i Nazwisko
```

```

Wpisz_imie = driver.find_element_by_name("my-douglas-register-prename")
Wpisz_imie.send_keys(imie)
Wpisz_nazwisko = driver.find_element_by_name("my-douglas-register-lastname")
Wpisz_nazwisko.send_keys(nazwisko)
# Sprawdź imię
Sprawdz_imie = driver.find_element_by_name("my-douglas-register-prename").get_attribute("value")
print("W polu jest wpisane imię", Sprawdz_imie)
assert Sprawdz_imie == imie
# Sprawdź nazwisko
Sprawdz_nazwisko = driver.find_element_by_name("my-douglas-register-lastname").get_attribute("value")
print("W polu jest wpisane nazwisko", Sprawdz_nazwisko)
assert Sprawdz_nazwisko == nazwisko
# 5. Wpisz hasło
Wpisz_haslo = driver.find_element_by_name("my-douglas-register-password")
Wpisz_haslo.send_keys(haslo)
# 6. Powtórz hasło
Wpisz_haslo_ponownie = driver.find_element_by_name("my-douglas-register-password-repeat")
Wpisz_haslo_ponownie.send_keys(powtorz_haslo)
# 7. Wpisz email (niepoprawny)
Wpisz_email = driver.find_element_by_name("my-douglas-register-email")
Wpisz_email.send_keys(niepoprawny_email)
# 8. Kliknij Rejestracja
Rejestracja = driver.find_element_by_xpath('//*[@name="RegisterForm|SubmitChanges"]')
Rejestracja.click()
# Sprawdź email
Sprawdz_email = driver.find_element_by_name("my-douglas-register-email").get_attribute("value")
print("W polu jest wpisany email", Sprawdz_email)
assert Sprawdz_email == email

def test_invalid_NewCard(self):
    driver = self.driver

    # Założenie karty stałego klienta
    # KROKI:
    # 1. Po otworzeniu przeglądarki zaakceptuj cookies
    cookies = cookies = WebDriverWait(driver,
    10).until(EC.element_to_be_clickable((By.ID, "uc-btn-accept-banner"))))
    cookies.click()
    # 2. Znajdź na stronie ikonkę z napisem "Karta Douglas" i kliknij ją
    Karta_Douglas = WebDriverWait(driver,
    20).until(EC.element_to_be_clickable((By.XPATH, '//a[contains(text(), "Karta Douglas")]'))))
    Karta_Douglas.click()
    # 3. Kliknij "Wypełnij już dziś"
    Wypełnij_karte = WebDriverWait(driver, 10).until(
    EC.element_to_be_clickable((By.XPATH, '//div[text()="WYPEŁNIJ JUŻ DZIŚ"]'))))
    Wypełnij_karte.click()
    # 4. Wpisz Imię
    Wpisz_imie2 = cookies = WebDriverWait(driver,
    10).until(EC.element_to_be_clickable((By.NAME, "address.firstname"))))
    Wpisz_imie2.send_keys(imie)
    # 5. Wpisz Nazwisko
    Wpisz_nazwisko2 = driver.find_element_by_name("address.lastname")
    Wpisz_nazwisko2.send_keys(nazwisko)
    # 6. Wybierz płeć
    if plec == 'K':
        # Wybierz PANI
        K = Select(driver.find_element_by_id("address.gender"))
        K.select_by_visible_text("Pani")

```

```

else:
    # WYBIERZ PAN
    M = Select(driver.find_element_by_id("address.gender"))
    M.select_by_visible_text("Pan")
    # 7. Wpisz datę urodzenia
    Wpisz_dzien = driver.find_element_by_xpath('//*[@name="birthday"]')
    Wpisz_dzien.send_keys(dzien_urodzenia)
    Wpisz_miesiac = driver.find_element_by_xpath('//*[@name="birthmonth"]')
    Wpisz_miesiac.send_keys(miesiac_urodzenia)
    Wpisz_rok = driver.find_element_by_xpath('//*[@name="birthyear"]')
    Wpisz_rok.send_keys(rok_urodzenia)
    # 8. Wybierz z listy kraj
    Wybierz_kraj = Select(driver.find_element_by_name("country"))
    Wybierz_kraj.select_by_visible_text("Polska")
    # 9. Wybierz województwo
    Wybierz_województwo = Select(driver.find_element_by_name("province"))
    Wybierz_województwo.select_by_visible_text("śląskie")
    # 10. Wpisz ulicę i nr domu
    Wpisz_ulice = driver.find_element_by_name("address.street")
    Wpisz_ulice.send_keys(ulica)
    Wpisz_nrDomu = driver.find_element_by_name("address.number")
    Wpisz_nrDomu.send_keys(nr_domu)
    # 11. Wpisz kod pocztowy
    Wpisz_kod = driver.find_element_by_name("address.zip")
    Wpisz_kod.send_keys(kod_pocztowy)
    # 12. Wpisz miasto
    Wpisz_miasto = driver.find_element_by_name("address.city")
    Wpisz_miasto.send_keys(miasto)
    # 13. Wpisz nr telefonu komórkowego
    Wpisz_nrTelefonu = driver.find_element_by_name("cellular")
    Wpisz_nrTelefonu.send_keys(nr_telefonu)
    # 14. Wpisz email
    Wpisz_email2 = driver.find_element_by_id("email")
    Wpisz_email2.send_keys(email)
    # 14. Powtórz email
    Powtorz_email = driver.find_element_by_xpath('//*[@name="emailRepeated"]')
    Powtorz_email.send_keys(email)
    # Sprawdź email
    Sprawdz_email2 = driver.find_element_by_id("email").get_attribute("value")
    print("W polu jest wpisany email", Sprawdz_email2)
    assert Sprawdz_email2 == email
    # 15. Zaznacz zapoznanie się z regulaminem i zgodę na przetwarzanie danych
    Regulamin = driver.find_element_by_xpath('//*[@id="agbAccepted"]')
    Regulamin.click()
    Zgoda = driver.find_element_by_xpath('//*[@id="postAdvertising"]')
    Zgoda.click()
    # 16. Wprowadź kod captcha (nieprawidłowy)
    Captcha = driver.find_element_by_xpath('//*[@id="captcha"]')
    Captcha.send_keys(captcha)
    # 17. Kliknij WYŚLIJ
    Wyslij = driver.find_element_by_xpath(
        '//*[@alt="douglasscard.application.abschicken"]')
    Wyslij.click()

```

```

if __name__ == '__main__':
    unittest.main(verbosity=3)

```

Uwagi końcowe:

Automatyzacja przypadków testowych powiodła się. Nie jest wyświetlony error, ale nie można przejść do dalszego etapu.

Przypadek testowy nr 3

Tytuł: Przypadek testowy nr 3: Wyszukanie i dodanie do koszyka 1 produktu

Środowisko: Chrome wersja 80.0.3987.106, Ubuntu 18.04 3 LTS

Warunek wstępny: Uruchomiona przeglądarka – strona internetowa <http://sephora.pl/>.

Użytkownik nie jest zalogowany.

KROKI:

1. Po otwarciu przeglądarki wyłącz okno informujące o zniżce dla nowych klientów i akceptuj cookies
2. Znajdź pole wyszukiwania ("Jakiego produktu szukasz?")
3. Wpisz, jakiego produktu szukasz
4. Znajdź odpowiedni produkt
5. Wybierz odpowiednią pojemność
6. Dodaj produkt do koszyka
7. Otwórz koszyk, żeby sprawdzić czy zgadza się ilość produktów

Oczekiwany rezultat: Produkt zostaje wyszukany i dodany do koszyka.

W koszyku znajduje się 1 produkt.

Przypadek testowy nr 4

Tytuł: Przypadek testowy nr 4: Wyszukanie i dodanie do koszyka 2 produktów

Środowisko: Chrome wersja 80.0.3987.106, Ubuntu 18.04 3 LTS

Warunek wstępny: Uruchomiona przeglądarka – strona internetowa <http://sephora.pl/>.

Użytkownik nie jest zalogowany.

KROKI:

1. Po otwarciu przeglądarki wyłącz okno informujące o zniżce dla nowych klientów i akceptuj cookies
2. Znajdź pole wyszukiwania ("Jakiego produktu szukasz?")
3. Wpisz nazwę pierwszego produktu, którego szukasz
4. Znajdź pierwszy produkt, który chcesz dodać do koszyka
5. Dodaj pierwszy produkt do koszyka
6. Znajdź pole wyszukiwania ("Jakiego produktu szukasz?")
7. Wpisz nazwę drugiego produktu, którego szukasz
8. Znajdź drugi produkt, który chcesz dodać do koszyka
9. Otwórz paletę kolorów, w celu wybrania odpowiedniego
10. Wybierz odpowiedni kolor 1N2 - Ecru (30 ml)
11. Zatwierdź wybór koloru
12. Poczekaj, aż będzie możliwe dodanie do koszyka
13. Dodaj drugi produkt do koszyka
14. Otwórz koszyk, żeby sprawdzić czy zgadza się ilość produktów

Oczekiwany rezultat: Produkty zostają wyszukane i dodane do koszyka.

W koszyku znajdują się 2 produkty.

Automatyzacja przypadków testowych nr 3 i nr 4:

```
# -*- coding: utf-8 -*-
import unittest
from selenium import webdriver
from time import sleep
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.common.keys import Keys
import selenium.webdriver.common.alert

# Dane testowe:
wyszukiwanie1 = "RE NUTRIV KREM"
poprawny_produkt = "Re-Nutriv Ultimate Lift Regenerating Youth Creme - Krem do twarzy"
koszyk = 1
koszyk2 = 2
wyszukiwanie2 = "Advanced Night Repair"
```


wyszukiwanie3 = "Double Wear podkład"

```
class SephoraSearchAndBuyEL(unittest.TestCase):
```

```
    # Scenariusz testowy:
```

```
    # Test 3 i 4: Szukanie produktów i dodawanie do koszyka
```

```
    def setUp(self):
```

```
        # Warunki wstępne:
```

```
        # Przeglądarka otwarta na stronie: http://sephora.pl/
```

```
        self.driver = webdriver.Chrome()
```

```
        self.driver.maximize_window()
```

```
        self.driver.get("http://sephora.pl/")
```

```
    def tearDown(self):
```

```
        self.driver.quit()
```

```
    def test_search_and_buy(self):
```

```
        driver = self.driver
```

```
        # Wyszukanie i dodanie do koszyka 1 produktu
```

```
        # KROKI:
```

```
        # 1. Po otworzeniu przeglądarki wyłącz okno informujące o zniżce dla
```

```
        # nowych klientów i akceptuj cookies
```

```
        Wylacz_okienko = WebDriverWait(driver, 10).until(EC.element_to_be_clickable((By.CLASS_NAME, "abtasty-modal")))
        Wylacz_okienko.click()
```

```
        # 2. Znajdź pole wyszukiwania ("Jakiego produktu szukasz?")
        szukaj_ReNutriv = driver.find_element_by_class_name("fake-search-input-placeholder")
```

```
        szukaj_ReNutriv.click()
```

```
        # 3. Wpisz, jakiego produktu szukasz
```

```
        wyszukiwanie = driver.find_element_by_xpath('//*[@id="q"]')
```

```
        wyszukiwanie.send_keys(wyszukiwanie1)
```

```
        wyszukiwanie.send_keys(Keys.ENTER)
```

```
        # 4. Znajdź odpowiedni produkt
```

```
        znajdz_produkt = WebDriverWait(driver, 20).until
```

```
        (EC.element_to_be_clickable((By.XPATH, '//h3[contains(@title,"Re-Nutriv Ultimate Lift Regenerating Youth Creme - Krem do twarzy")]')))
```

```
        znajdz_produkt = driver.find_element_by_xpath('//h3[contains(@title,"Re-Nutriv Ultimate Lift Regenerating Youth Creme - Krem do twarzy")]')
```

```
        znajdz_produkt.click()
```

```
        # 5. Wybierz odpowiednią pojemność
```

```
        wybierz_pojemnosc = WebDriverWait(driver, 20).until
```

```
        (EC.element_to_be_clickable((By.XPATH, '//*[@id="product-content"]')))
```

```
        wybierz_pojemnosc.click()
```

```
        # 6. Dodaj produkt do koszyka
```

```
        dodaj_do_koszyka = driver.find_element_by_xpath('//button[@class="button important add-to-cart tag-commander-event"]').click()
```

```
        # 7. Otwórz koszyk, żeby sprawdzić czy zgadza się ilość produktów
```

```
        WebDriverWait(driver, 20).until(EC.presence_of_element_located((By.XPATH, '//*[@text()="Koszyk"]')))
```

```
        otworz_koszyk = WebDriverWait(driver, 30).until(EC.element_to_be_clickable((By.XPATH, '//*[a[@class="button mini-cart-link-cart button-important"]'])))
```

```
        otworz_koszyk.click()
```

```
        # 8. Sprawdź ilość produktów na ikonke koszyka
```

```
        sprawdzam_koszyk = WebDriverWait(driver, 20).until(EC.presence_of_element_located((By.XPATH, '//div[@id="header-item-minicart"]/span[@class="header-link-quantity"]')))
```

```
        sprawdzam_koszyk =
```

```
        driver.find_element_by_xpath('//div[@id="header-item-minicart"]/span[@class="header-link-quantity"]').get_attribute("innerText")
```

```
        self.assertEqual(str(koszyk), sprawdzam_koszyk)
```

```
        print("Ilość produktów na ikonke koszyka: ", sprawdzam_koszyk)
```

```
        # 9. Sprawdź, czy zgadza się ilość produktów w koszyku
```

```
        produkty = driver.find_elements_by_xpath('//div[@class="grid-item"]')
```

```
        ilosc_produktow = 0
```

```

for label in produkty:
    ilosc_produktow = (ilosc_produktow + 1)

self.assertEqual(koszyk, ilosc_produktow)
print("Ilość produktów w koszyku: ", ilosc_produktow)

def test_search_and_buy2(self):
    driver = self.driver
    # Wyszukanie i dodanie do koszyka 2 produktów

    # KROKI:
    # 1. Po otwarciu przeglądarki wyłącz okno informujące o zniżce dla nowych klientów i akceptuj
    cookies
    Wylacz_okienko = WebDriverWait(driver, 20).until(EC.element_to_be_clickable((By.CLASS_NAME,
    "abtasty-modal")))
    Wylacz_okienko.click()
    # 2. Znajdź pole wyszukiwania ("Jakiego produktu szukasz?")
    szukaj_ANR = driver.find_element_by_class_name("fake-search-input-placeholder")
    szukaj_ANR.click()
    # 3. Wpisz nazwę pierwszego produktu, którego szukasz
    wyszukiwanieNr2 = driver.find_element_by_xpath('//*[@id="q"]')
    wyszukiwanieNr2.send_keys(wyszukiwanie2)
    wyszukiwanieNr2.send_keys(Keys.ENTER)
    # 4. Znajdź pierwszy produkt, który chcesz dodać do koszyka
    znajdz_produkt2 = WebDriverWait(driver, 30).until(EC.element_to_be_clickable((By.XPATH,
    '//h3[@title="Advanced Night Repair - Serum naprawcze do twarzy na noc"]')))
    znajdz_produkt2 = driver.find_element_by_xpath('//h3[@title="Advanced Night Repair - Serum
    naprawcze do twarzy na noc"]')
    znajdz_produkt2.click()
    # 5. Dodaj pierwszy produkt do koszyka
    dodaj_do_koszyka2 = WebDriverWait(driver, 30).until(EC.element_to_be_clickable((By.XPATH,
    '//*[@class="button important add-to-cart tag-commander-event"]')))
    dodaj_do_koszyka2 = driver.find_element_by_xpath('//*[@class="button important add-to-cart tag-
    commander-event"]')
    dodaj_do_koszyka2.click()
    # 6. Znajdź pole wyszukiwania ("Jakiego produktu szukasz?")
    szukaj_DW = WebDriverWait(driver, 30).until(EC.element_to_be_clickable((By.CLASS_NAME, "fake-
    search-input-placeholder")))
    szukaj_DW = driver.find_element_by_class_name("fake-search-input-placeholder")
    szukaj_DW.click()
    # 7. Wpisz nazwę drugiego produktu, którego szukasz
    wyszukiwanieNr3 = WebDriverWait(driver, 30).until(EC.element_to_be_clickable((By.XPATH,
    '//input[@id="q"]')))
    wyszukiwanieNr3.send_keys(wyszukiwanie3)
    wyszukiwanieNr3.send_keys(Keys.ENTER)
    # 8. Znajdź drugi produkt, który chcesz dodać do koszyka.
    znajdz_produkt3 = WebDriverWait(driver, 20).until(EC.element_to_be_clickable((By.XPATH,
    '//h3[@title="Double Wear - Stay-in-Place Podkład SPF 10"]')))
    znajdz_produkt3.click()
    # 9. Otwórz paletę kolorów, w celu wybrania odpowiedniego
    otworz_palette_kolorow = WebDriverWait(driver, 20).until(EC.presence_of_element_located((By.XPATH, '//
    p[@class="dialog-link-wrap open-color-dialog-wrapper show-for-medium"]')))
    otworz_palette_kolorow.click()
    # 10. Wybierz odpowiedni kolor 1N2 - Ecru (30 ml)
    wybierz_odpowiedni_kolor = WebDriverWait(driver, 10).until(EC.presence_of_element_located((By.XPATH,
    '//button[@title="1N2 - Ecru (30 ml)"]')))
    driver.find_element_by_xpath('//button[@title="1N2 - Ecru (30 ml)"]')
    wybierz_odpowiedni_kolor.click()
    # 11. Zatwierdź wybór koloru
    zatwierdz_kolor = WebDriverWait(driver, 30).until(EC.element_to_be_clickable((By.XPATH,
    '//a[@class="js-colorguide-select button button-select fullwidth-for-msmall"]')))
    zatwierdz_kolor = driver.find_element_by_xpath('//a[@class="js-colorguide-select button button-select
    fullwidth-for-msmall"]')
    zatwierdz_kolor.click()

```

```

# 12. Poczekaaj, aż będzie możliwe dodanie do koszyka
WebDriverWait(driver, 40).until(EC.invisibility_of_element_located((By.CLASS_NAME, "section-loader"))))
# 13. Dodaj drugi produkt do koszyka
dodaj_do_koszyka3 = WebDriverWait(driver, 40).until(EC.element_to_be_clickable((By.XPATH,
'//button[@id="add-to-cart"]'))))
dodaj_do_koszyka3 = driver.find_element_by_xpath('//button[@id="add-to-cart"]')
dodaj_do_koszyka3.click()
# 14. Otwórz koszyk, żeby sprawdzić czy zgadza się ilość produktów
WebDriverWait(driver, 20).until(EC.presence_of_element_located((By.XPATH, '//*[@text()= "Koszyk"]'))))
otworz_koszyk2 = WebDriverWait(driver, 20).until(EC.element_to_be_clickable((By.XPATH,
'//a[@class="button mini-cart-link-cart button-important"]'))))
otworz_koszyk2.click()
# 15. Sprawdź ilość produktów na ikonke koszyka
sprawdzam_koszyk2 =
driver.find_element_by_xpath('//div[@id="header-item-minicart"]//span[@class="header-link-
quantity"]').get_attribute("innerText")
self.assertEqual(str(koszyk2), sprawdzam_koszyk2)
print("Ilość produktów w koszyku: ", sprawdzam_koszyk2)
# 16. Sprawdź, czy zgadza się ilość produktów w koszyku
produkty = driver.find_elements_by_xpath('//div[@class="grid-item"]')
ilosc_produktow = 0
for label in produkty:
    ilosc_produktow = (ilosc_produktow + 1)

self.assertEqual(koszyk2, ilosc_produktow)
print("Ilość produktów w koszyku: ", ilosc_produktow)

if __name__ == '__main__':
    unittest.main(verbosity=3)

```

Uwagi końcowe:

Automatyzacja przypadków testowych powiodła się. Produkty zostały dodane do koszyka i ich ilość się zgadza.

Testowanie może przebiegać wolno – strona internetowa jest dynamiczna, często ładuje się ponownie podczas wczytywania podstron.