przygotuj w jezyku python testy dla strony https://demoqa.com/ dla podstrony elements Text Box i Check Box

**ChatGPT powiedział:**

ChatGPT

Poniżej znajdziesz testy w Pythonie dla strony [demoqa.com](https://demoqa.com/) dla podstron **Elements > Text Box** i **Elements > Check Box**. Testy są napisane z użyciem biblioteki **unittest** i Selenium.

**Struktura projektu**

bash

Skopiuj kod

/tests

test\_text\_box.py

test\_check\_box.py

requirements.txt

**1. Testy dla podstrony Text Box**

**Plik: test\_text\_box.py**

python

Skopiuj kod

import unittest

from selenium import webdriver

from selenium.webdriver.common.by import By

class TestTextBox(unittest.TestCase):

@classmethod

def setUpClass(cls):

"""Uruchamianie przeglądarki przed rozpoczęciem testów."""

cls.driver = webdriver.Chrome()

cls.driver.maximize\_window()

cls.driver.get("https://demoqa.com/elements")

@classmethod

def tearDownClass(cls):

"""Zamykanie przeglądarki po zakończeniu testów."""

cls.driver.quit()

def test\_text\_box\_submission(self):

"""Test wypełnienia formularza Text Box."""

# Przejdź do sekcji Text Box

text\_box\_button = self.driver.find\_element(By.ID, "item-0")

text\_box\_button.click()

# Wypełnij formularz

self.driver.find\_element(By.ID, "userName").send\_keys("John Doe")

self.driver.find\_element(By.ID, "userEmail").send\_keys("johndoe@example.com")

self.driver.find\_element(By.ID, "currentAddress").send\_keys("123 Main Street")

self.driver.find\_element(By.ID, "permanentAddress").send\_keys("456 Another Street")

self.driver.find\_element(By.ID, "submit").click()

# Sprawdź wyniki

output = self.driver.find\_element(By.ID, "output").text

self.assertIn("John Doe", output)

self.assertIn("johndoe@example.com", output)

self.assertIn("123 Main Street", output)

self.assertIn("456 Another Street", output)

if \_\_name\_\_ == "\_\_main\_\_":

unittest.main()

**2. Testy dla podstrony Check Box**

**Plik: test\_check\_box.py**

python

Skopiuj kod

import unittest

from selenium import webdriver

from selenium.webdriver.common.by import By

class TestCheckBox(unittest.TestCase):

@classmethod

def setUpClass(cls):

"""Uruchamianie przeglądarki przed rozpoczęciem testów."""

cls.driver = webdriver.Chrome()

cls.driver.maximize\_window()

cls.driver.get("https://demoqa.com/elements")

@classmethod

def tearDownClass(cls):

"""Zamykanie przeglądarki po zakończeniu testów."""

cls.driver.quit()

def test\_check\_box\_selection(self):

"""Test zaznaczenia opcji Documents w Check Box."""

# Przejdź do sekcji Check Box

check\_box\_button = self.driver.find\_element(By.ID, "item-1")

check\_box\_button.click()

# Rozwiń katalog główny

toggle\_home = self.driver.find\_element(By.CSS\_SELECTOR, ".rct-icon.rct-icon-expand-close")

toggle\_home.click()

# Zaznacz opcję "Documents"

documents\_checkbox = self.driver.find\_element(By.XPATH, "//label[@for='tree-node-documents']//span[@class='rct-checkbox']")

documents\_checkbox.click()

# Sprawdź, czy opcja "Documents" jest zaznaczona

result\_output = self.driver.find\_element(By.ID, "result").text

self.assertIn("documents", result\_output.lower())

if \_\_name\_\_ == "\_\_main\_\_":

unittest.main()

**3. Plik requirements.txt**

Zależności projektu:

makefile

Skopiuj kod

selenium==4.10.0

**Uruchamianie testów**

1. **Zainstaluj Selenium:**

bash

Skopiuj kod

pip install -r requirements.txt

1. **Uruchom testy:**
   * Test dla Text Box:

bash

Skopiuj kod

python -m unittest tests/test\_text\_box.py

* + Test dla Check Box:

bash

Skopiuj kod

python -m unittest tests/test\_check\_box.py

**Opis testów**

1. **Test Text Box:**
   * Otwiera sekcję Text Box.
   * Wypełnia pola: pełne imię, e-mail, aktualny adres, stały adres.
   * Sprawdza, czy wyświetlone wyniki zawierają wprowadzone dane.
2. **Test Check Box:**
   * Otwiera sekcję Check Box.
   * Rozwija katalog główny.
   * Zaznacza opcję "Documents".
   * Sprawdza, czy opcja "Documents" pojawiła się w wynikach.

Testy zostały zaprojektowane tak, aby były łatwe w utrzymaniu i działały poprawnie na stronie demoqa.com.