Напиши наскрізний End-to-End тест для веб-додатку [TodoMVC](https://todomvc4tasj.herokuapp.com/) (версія [https://todomvc4tasj.herokuapp.com](https://todomvc4tasj.herokuapp.com/) , постарайся покрити основні частини функціоналу з точки зору «Smoke Acceptance Testing» вигляді набору [CRUD](https://en.wikipedia.org/wiki/Create,_read,_update_and_delete).

from selenium import webdriver

from selenium.webdriver.common.keys import Keys

import time

# запуск браузера

driver = webdriver.Chrome()

driver.get("https://todomvc4tasj.herokuapp.com")

# перевірка заголовку

assert "todos" in driver.title

# створення нового запису

elem = driver.find\_element\_by\_id("new-todo")

elem.clear()

elem.send\_keys("Test todo")

elem.send\_keys(Keys.RETURN)

time.sleep(1)

# перевірка, що новий запис відображається в списку завдань

todo\_list = driver.find\_element\_by\_id("todo-list")

assert "Test todo" in todo\_list.text

# створення ще одного запису

elem = driver.find\_element\_by\_id("new-todo")

elem.clear()

elem.send\_keys("Another test todo")

elem.send\_keys(Keys.RETURN)

time.sleep(1)

# перевірка, що новий запис відображається в списку завдань

assert "Another test todo" in todo\_list.text

# перегляд запису та оновлення тексту

todo\_item = driver.find\_element\_by\_xpath("//label[text()='Test todo']")

todo\_item.click()

edit\_field = driver.find\_element\_by\_xpath("//input[@value='Test todo']")

edit\_field.clear()

edit\_field.send\_keys("Updated todo")

edit\_field.send\_keys(Keys.RETURN)

time.sleep(1)

# перевірка, що текст запису змінився

assert "Updated todo" in todo\_list.text

# видалення запису

todo\_item = driver.find\_element\_by\_xpath("//label[text()='Test todo']")

todo\_item.click()

delete\_button = driver.find\_element\_by\_xpath("//button[@class='destroy']")

delete\_button.click()

time.sleep(1)

# перевірка, що запис видалено

assert "Test todo" not in todo\_list.text

# очищення списку завдань

toggle\_all = driver.find\_element\_by\_id("toggle-all")

toggle\_all.click()

clear\_completed = driver.find\_element\_by\_id("clear-completed")

clear\_completed.click()

time.sleep(1)

# перевірка, що список завдань порожній

assert not todo\_list.text

# закриття браузера

driver.quit()

З def

from selenium import webdriver

from selenium.webdriver.common.keys import Keys

import time

def test\_todomvc():

# запуск браузера

driver = webdriver.Chrome()

driver.get("https://todomvc4tasj.herokuapp.com")

# перевірка заголовку

assert "todos" in driver.title

# створення нового запису

elem = driver.find\_element\_by\_id("new-todo")

elem.clear()

elem.send\_keys("Test todo")

elem.send\_keys(Keys.RETURN)

time.sleep(1)

# перевірка, що новий запис відображається в списку завдань

todo\_list = driver.find\_element\_by\_id("todo-list")

assert "Test todo" in todo\_list.text

# створення ще одного запису

elem = driver.find\_element\_by\_id("new-todo")

elem.clear()

elem.send\_keys("Another test todo")

elem.send\_keys(Keys.RETURN)

time.sleep(1)

# перевірка, що новий запис відображається в списку завдань

assert "Another test todo" in todo\_list.text

# перегляд запису та оновлення тексту

todo\_item = driver.find\_element\_by\_xpath("//label[text()='Test todo']")

todo\_item.click()

edit\_field = driver.find\_element\_by\_xpath("//input[@value='Test todo']")

edit\_field.clear()

edit\_field.send\_keys("Updated todo")

edit\_field.send\_keys(Keys.RETURN)

time.sleep(1)

# перевірка, що текст запису змінився

assert "Updated todo" in todo\_list.text

# видалення запису

todo\_item = driver.find\_element\_by\_xpath("//label[text()='Test todo']")

todo\_item.click()

delete\_button = driver.find\_element\_by\_xpath("//button[@class='destroy']")

delete\_button.click()

time.sleep(1)

# перевірка, що запис видалено

assert "Test todo" not in todo\_list.text

# очищення списку завдань

toggle\_all = driver.find\_element\_by\_id("toggle-all")

toggle\_all.click()

clear\_completed = driver.find\_element\_by\_id("clear-completed")

clear\_completed.click()

time.sleep(1)

# перевірка, що список завдань порожній

assert not todo\_list.text

# закриття браузера

driver.quit()

# виклик функції для запуску тесту

test\_todomvc()

**ЗPytest**

import pytest

from selenium import webdriver

from selenium.webdriver.common.keys import Keys

import time

**@pytest.fixture()**

def browser():

# запуск браузера перед тестом

driver = webdriver.Chrome()

driver.get("https://todomvc4tasj.herokuapp.com")

yield driver

# закриття браузера після тесту

driver.quit()

def test\_todomvc(browser):

# перевірка заголовку

assert "todos" in browser.title

# створення нового запису

elem = browser.find\_element\_by\_id("new-todo")

elem.clear()

elem.send\_keys("Test todo")

elem.send\_keys(Keys.RETURN)

time.sleep(1)

# перевірка, що новий запис відображається в списку завдань

todo\_list = browser.find\_element\_by\_id("todo-list")

assert "Test todo" in todo\_list.text

# створення ще одного запису

elem = browser.find\_element\_by\_id("new-todo")

elem.clear()

elem.send\_keys("Another test todo")

elem.send\_keys(Keys.RETURN)

time.sleep(1)

# перевірка, що новий запис відображається в списку завдань

assert "Another test todo" in todo\_list.text

# перегляд запису та оновлення тексту

todo\_item = browser.find\_element\_by\_xpath("//label[text()='Test todo']")

todo\_item.click()

edit\_field = browser.find\_element\_by\_xpath("//input[@value='Test todo']")

edit\_field.clear()

edit\_field.send\_keys("Updated todo")

edit\_field.send\_keys(Keys.RETURN)

time.sleep(1)

# перевірка, що текст запису змінився

assert "Updated todo" in todo\_list.text

# видалення запису

todo\_item = browser.find\_element\_by\_xpath("//label[text()='Test todo']")

todo\_item.click()

delete\_button = browser.find\_element\_by\_xpath("//button[@class='destroy']")

delete\_button.click()

time.sleep(1)

# перевірка, що запис видалено

assert "Test todo" not in todo\_list.text

# очищення списку завдань

toggle\_all = browser.find\_element\_by\_id("toggle-all")

toggle\_all.click()

clear\_completed = browser.find\_element\_by\_id("clear-completed")

clear\_completed.click()

time.sleep(1)

# перевірка, що список завдань порожній

assert not todo\_list.text

**Другий варіант Pytest fixture**

from selenium import webdriver

from selenium.webdriver.common.keys import Keys

import pytest

@pytest.fixture

def browser():

# створення об'єкта webdriver

driver = webdriver.Chrome()

yield driver

# закриття браузера

driver.quit()

def test\_todomvc(browser):

# перехід на веб-сторінку

browser.get("https://todomvc4tasj.herokuapp.com")

# перевірка заголовку

assert "todos" in browser.title

# створення нового запису

elem = browser.find\_element\_by\_id("new-todo")

elem.clear()

elem.send\_keys("Test todo")

elem.send\_keys(Keys.RETURN)

browser.implicitly\_wait(1)

# перевірка, що новий запис відображається в списку завдань

todo\_list = browser.find\_element\_by\_id("todo-list")

assert "Test todo" in todo\_list.text

# створення ще одного запису

elem = browser.find\_element\_by\_id("new-todo")

elem.clear()

elem.send\_keys("Another test todo")

elem.send\_keys(Keys.RETURN)

browser.implicitly\_wait(1)

# перевірка, що новий запис відображається в списку завдань

assert "Another test todo" in todo\_list.text

# перегляд запису та оновлення тексту

todo\_item = browser.find\_element\_by\_xpath("//label[text()='Test todo']")

todo\_item.click()

edit\_field = browser.find\_element\_by\_xpath("//input[@value='Test todo']")

edit\_field.clear()

edit\_field.send\_keys("Updated todo")

edit\_field.send\_keys(Keys.RETURN)

browser.implicitly\_wait(1)

# перевірка, що текст запису змінився

assert "Updated todo" in todo\_list.text

# видалення запису

todo\_item = browser.find\_element\_by\_xpath("//label[text()='Test todo']")

todo\_item.click()

delete\_button = browser.find\_element\_by\_xpath("//button[@class='destroy']")

delete\_button.click()

browser.implicitly\_wait(1)

# перевірка, що запис видалено

assert "Test todo" not in todo\_list.text

# очищення списку завдань

toggle\_all = browser.find\_element\_by\_id("toggle-all")

toggle\_all.click()

clear\_completed = browser.find\_element\_by\_id("clear-completed")

clear\_completed.click()

browser.implicitly\_wait(1)

# перевірка, що список завдань порожній

assert not todo\_list.text

**З pytest, def, class, \_init\_**

import pytest

from selenium import webdriver

from selenium.webdriver.common.keys import Keys

class TestTodoMVC:

def \_\_init\_\_(self):

# створення об'єкта webdriver

self.driver = webdriver.Chrome()

def setup(self):

pass

@pytest.fixture(scope="function", autouse=True)

def setup\_teardown(self):

# перехід на веб-сторінку

self.driver.get("https://todomvc4tasj.herokuapp.com")

# дочірні тести будуть виконані тільки після закінчення методу

yield

# закриття браузера

self.driver.quit()

def test\_create\_todo\_item(self):

# створення нового запису

elem = self.driver.find\_element\_by\_id("new-todo")

elem.clear()

elem.send\_keys("Test todo")

elem.send\_keys(Keys.RETURN)

self.driver.implicitly\_wait(1)

# перевірка, що новий запис відображається в списку завдань

todo\_list = self.driver.find\_element\_by\_id("todo-list")

assert "Test todo" in todo\_list.text

def test\_update\_todo\_item(self):

# створення нового запису

elem = self.driver.find\_element\_by\_id("new-todo")

elem.clear()

elem.send\_keys("Test todo")

elem.send\_keys(Keys.RETURN)

self.driver.implicitly\_wait(1)

# перегляд запису та оновлення тексту

todo\_item = self.driver.find\_element\_by\_xpath("//label[text()='Test todo']")

todo\_item.click()

edit\_field = self.driver.find\_element\_by\_xpath("//input[@value='Test todo']")

edit\_field.clear()

edit\_field.send\_keys("Updated todo")

edit\_field.send\_keys(Keys.RETURN)

self.driver.implicitly\_wait(1)

# перевірка, що текст запису змінився

todo\_list = self.driver.find\_element\_by\_id("todo-list")

assert "Updated todo" in todo\_list.text

def test\_delete\_todo\_item(self):

# створення нового запису

elem = self.driver.find\_element\_by\_id("new-todo")

elem.clear()

elem.send\_keys("Test todo")

elem.send\_keys(Keys.RETURN)

self.driver.implicitly\_wait(1)

# видалення запису

todo\_item = self.driver.find\_element\_by\_xpath("//label[text()='Test todo']")

todo\_item.click()

delete\_button = self.driver.find\_element\_by\_xpath("//button[@class='destroy']")

delete\_button.click()

self.driver.implicitly\_wait(1)

# перевірка, що запис був видалений

todo\_list = self.driver.find\_element\_by\_id("todo-list")

assert "Test todo" not in todo\_list.text