**ЛАБОРАТОРНА РОБОТА № 3**

З курсу

«Автоматизація тестування»

**Студент:** Однокозов О. Д.

**Група:** 41ІПЗ

**Факультет:** Математики, інформатики та фізики

**Викладач:** Кархут В.

Київ, 2022

Репозіторій GitHub: <https://github.com/ElbruzOne/Automation-Testing>

Код програми:

const { Builder, By, until } = require('selenium-webdriver');

describe("Selenium Testing", () => {

  let driver;

  beforeAll(async () => {

    driver = await new Builder().forBrowser('firefox').build();

  })

  test("google page", async() => {

    await driver.get('https://www.google.com/');

    let gglTitle = await driver.getTitle();

    let gglLogo = await driver.findElement(By.css('img.lnXdpd'));

    let gglSearchLine = await driver.findElement(By.name('q'));

    let gglSearchButton = await driver.findElement(By.name('btnK'));

    let gglLink = await driver.findElement(By.xpath(".//a[contains(@href,'mail.google.com')]"));

    expect(gglTitle).toBe("Google");

    expect(gglLogo).toBeDefined();

    expect(gglSearchLine).toBeDefined();

    expect(gglSearchButton).toBeDefined();

    expect(gglLink).toBeDefined();

    console.log(gglTitle);

  },12000);

  test("wiki page", async() => {

    await driver.get('https://uk.wikipedia.org/');

    await driver.findElement(By.id("searchInput")).sendKeys("Київ");

    await driver.findElement(By.id("searchButton")).click();

    let kyivLogo = await driver.findElement(By.css('img[alt="COA of Kyiv Kurovskyi.svg"]'));

    let population = await driver.findElement(By.linkText("Населення"));

    let avgTemp = await driver.findElement(By.xpath("//\*[contains(text(),'Середня температура')]"));

    let aprilTemp = await driver.findElement(By.xpath("//\*[contains(text(),'Квіт.')]"));

    let covid = await driver.findElement(By.id("Епідемія\_коронавірусу"));

    let popDensity = await driver.findElement(By.xpath("//\*[contains(text(),'осіб/км')]"));

    const findArchitecture = await driver.findElement(By.xpath("/html/body/div[3]/div[3]/div[5]/div[1]/ul[11]"));

    let count = 0;

    await findArchitecture.findElements(By.css("li")).then(elements => {count = elements.length});;

    expect(kyivLogo).toBeDefined();

    expect(population).toBeDefined();

    expect(avgTemp).toBeDefined();

    expect(aprilTemp).toBeDefined();

    expect(covid).toBeDefined();

    expect(popDensity).toBeDefined();

    expect(count).toBeGreaterThan(20);

  },12000);

  afterAll(async () => {

    await driver.quit();

  })

})

Результат:

Text

Description automatically generated