# ЛАБОРАТОРНА РОБОТА № 2

## Особливості тестування мобільних застосунків

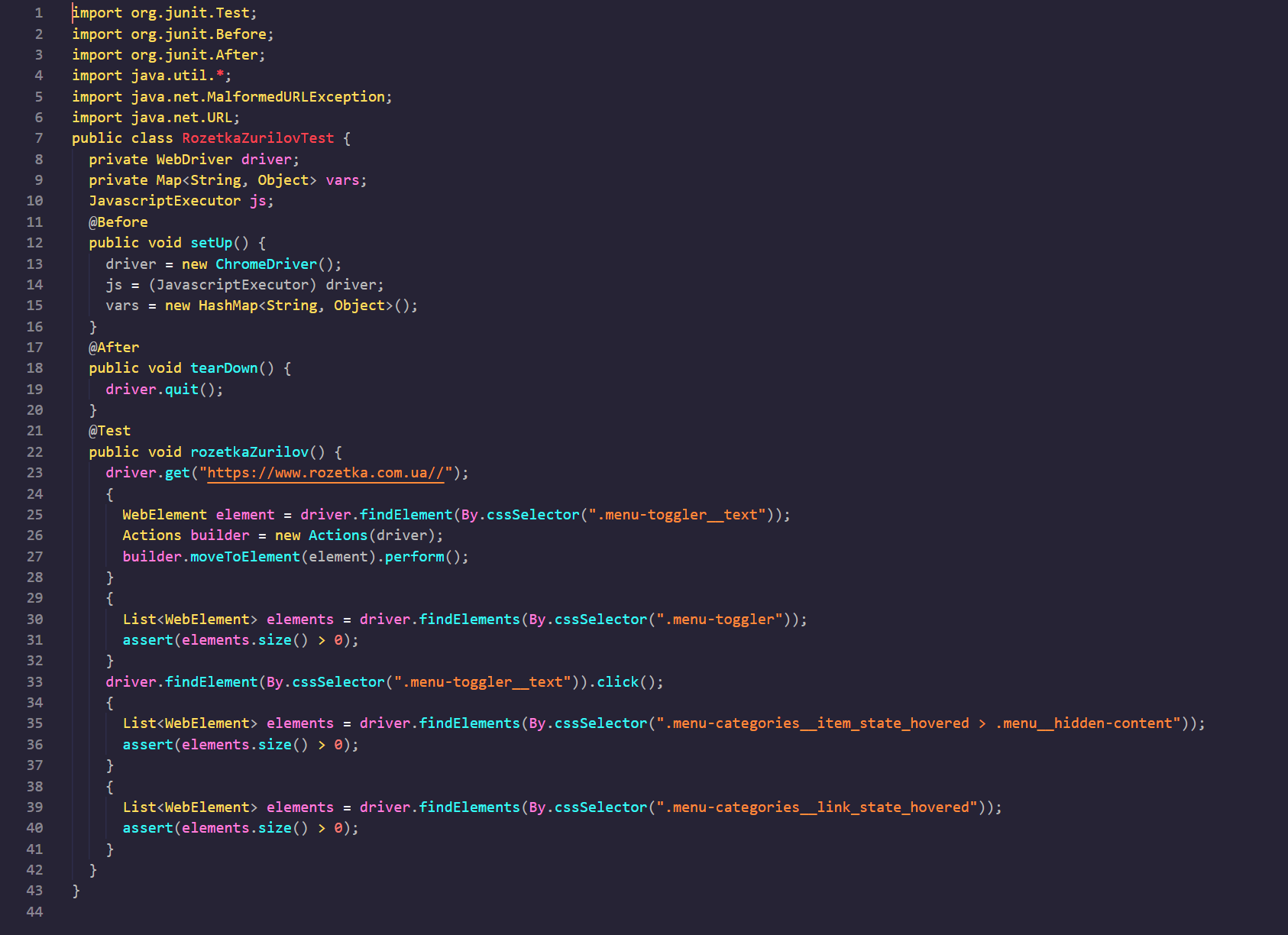
**Мета роботи:** отримати досвід тестування мобільних застосунків та навчитися проводити різні типи тестів для цього типу застосунків.

**Порядок виконання роботи:**

За допомогою JUnit створити функціональні тести, які б відповідали тестовим сценаріям з ЛР1. До звіту включити код автоматизованих тестів, які відповідають завданню.

*ЛР1 описувала систему інтернет-магазину rozetka.com.ua.*

Код тест-кейсу для тестового сценарію №1 з ЛР1.



**Теоретичні питання до захисту роботи :**

* Можливості JUnit для функціонального та навантажувального тестування:
  + Функціональне тестування:
    - JUnit має розширення JFunc яке дозволяє проводити functional test. Саме поняття юніт тестування є іншим видом тестування, тому на чистому JUnit не є доречним проведення функціональних тестів.
  + Навантажувальні тести:
    - JUnit зараз можно використовувати разом з Maven для проведення performance testing, але так само як і з функціональними тестами, юніт тести, для яких предназначений JUnit, це інший вид тестування.
* Особливості, які треба враховувати при тестуванні мобільних застосунків:
  + Ось деякі відмінності на які треба звернути увагу порівняно з десктопними застосунками:
    - Розмір екрану і touch-інтерфейс
    - Перевірка зображення на ретина екранах і різних версія OS
    - Перевірка роботи зворотного зв’язку
    - Перевірка реакції програми на зовнішні переривання (сповіщення)
    - Реклама в мобільному додатку
    - Перевірка енергоспоживання
* Техніки функціонального тестування:
  + Manual static design techniques:
    - Walk through
    - Informal reviews
    - Technical reviews
    - Audit
    - Inspection
    - Management review
  + Static design techniques using tool:
    - Static analysis of code — полягає в аналізі різних шляхів і потоків у додатку (анг. flow) та отриманих різних даних тестів. Приклади інструментів для Java: Checkstyle, FindBugs, PMD, для Javascript: JSLint.
    - Compliance to coding standard — визначають відповідність коду різним стандартам кодування.
    - Analysis of code metrics — інструмент, який використовується для статичного аналізу, необхідний для оцінки різних показників, таких як рядки коду, складності, покриття коду тощо.
  + Dynamic Test Design Techniques — це техніки дизайну, які передбачають тестування в процесі безпосереднього виконання програми.
    - Specification based — також називається тестуванням Blackbox Testing (Тестування Чорної Скриньки). Це техніки тестування, які включають тестування на основі володіння інформацією по специфікації існуючої системи, але без знання її внутрішньої архітектури.
    - Structure based — методика проектування тестів відома ще під назвою White Box Testing (Тестування Білої Скриньки). У цій техніці дизайну, на відміну від попередньої, необхідне знання коду або внутрішньої архітектури системи для проведення тестування.
    - Experienced based — це техніки дизайну тестів, які повністю базуються на досвіді або інтуїції тестера. Дві найбільш поширені форми тестування з досвіду — види Ad-hoc тестування та Exploratory testing.