НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

# Building the test framework: test automation patterns

Test automation basics with Selenium & Java

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

# PageObject

# Test class now

```java
@Test
public void searchByKeywordSeleniumHaveToFindSeleniumhqOrgInTop(){
    WebElement searchField = driver.findElement(By.id("lst-ib"));
    searchField.sendKeys("Selenium");
    searchField.sendKeys(Keys.RETURN);
    List<WebElement> resultURLs =

    driver.findElements(By.xpath("//cite[@class='iUh30']"));
    assertThat(resultURLs.get(0).getText())
            .as("seleniumhq.ord is not the first result!")
            .contains("https://www.seleniumhq.org/");
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Page Object

Patterns helps us to **resolve problems:**

- reduce code duplication
- improve readability
- improve maintainability

**Page Object** is needed for:

- code reuse - one page will be used in several tests
- defines allowed actions on the page
- separating elements and actions

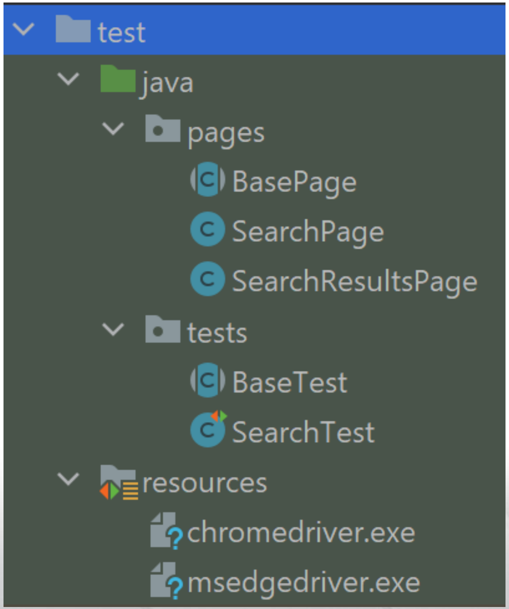
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## What is worth to use with page object:

1) @FindBy annotations
2) Page Factory pattern.

Let's refactor the framework and create package 'pages' with classes:

- `SearchPage`
- `SearchResultsPage`
- abstract `BasePage`



```
test
  java
    pages
      BasePage
      SearchPage
      SearchResultsPage
    tests
      BaseTest
      SearchTest
  resources
    chromedriver.exe
    msedgedriver.exe
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

# Page object pages

```java
package pages;

public class SearchPage { }
```

```java
package pages;

public class SearchResultsPage { }
```

```java
package pages;

public abstract class BasePage { }
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Provide access to WebDriver

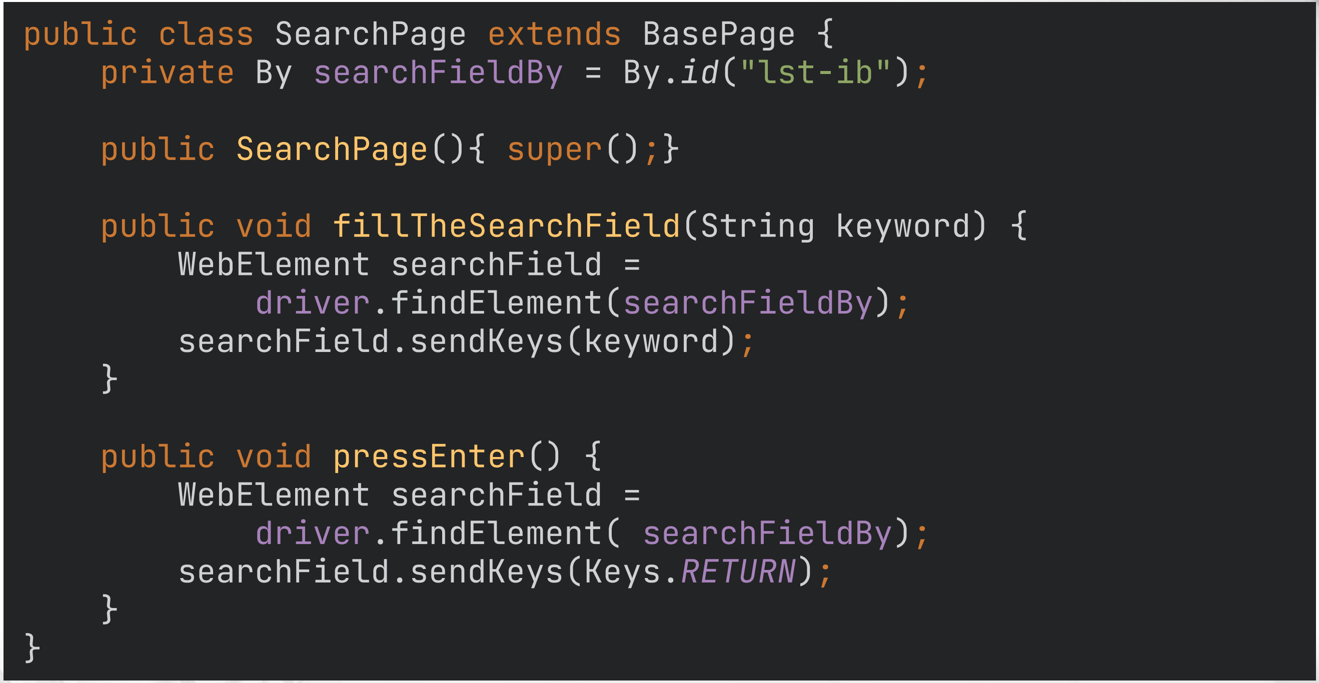
```java
public abstract class BaseTest {
    private static WebDriver driver;

    public static WebDriver getDriver() {
        return driver;
    }
...
```

## Use driver on BasePage

```
public abstract class BasePage {
    protected WebDriver driver;

    public BasePage() {
        driver = BaseTest.getDriver();
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Search page - version 1



```java
public class SearchPage extends BasePage {
    private By searchFieldBy = By.id("lst-ib");

    public SearchPage(){ super();}

    public void fillTheSearchField(String keyword) {
        WebElement searchField =
            driver.findElement(searchFieldBy);
        searchField.sendKeys(keyword);
    }

    public void pressEnter() {
        WebElement searchField =
            driver.findElement( searchFieldBy);
        searchField.sendKeys(Keys.RETURN);
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Search results page - version 1

```
public class SearchResultsPage extends BasePage {
    private By searchResultURLsBy =  By.xpath("//cite[@class='iUh30']");

    public SearchResultsPage(){
        super();
    }

    public void assertThatExpectedValueIsOnSearchTop(String expectedValue) {

        List<WebElement> searchResultURLs =
                driver.findElements(searchResultURLsBy);

        assertEquals(searchResultURLs.get(0).getText(),
                "expectedValue",
                expectedValue + " is not the first result!");
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## @FindBy annotations in Page Object

- allows to declare WebElement
- replace findElement/findElements

*Instead of*

```
WebElement clearButton = driver.findElement(By.id("clear"));
```

*there is*

```
@FindBy(id = "clear")
WebElement clearButton;
```

## Search page - version 2

```java
public class SearchPage extends BasePage {

    @FindBy(id = "lst-ib")
    private WebElement searchField;

    public SearchPage() {
        super();
    }

    public void fillTheSearchField(String keyword) {

        searchField.sendKeys(keyword);
    }

    public void pressEnter() {
        searchField.sendKeys(Keys.RETURN);
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Search results page - version 2

```java
public class SearchResultsPage extends BasePage {
    @FindBy(xpath = "//cite[@class='iUh30']")
    private List<WebElement> searchResultURLs;

    public SearchResultsPage() {
        super();
    }

    public void assertThatExpectedValueIsOnSearchTop(String expectedValue) {

        assertEquals(searchResultURLs.get(0).getText(),
                "expectedValue", expectedValue + " is not the first result!");
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## @FindBy problems

**WebElements** should be **initialized**
only **before** their **usage** to avoid NoSuchElementException.

Solution:

1) BaseTest extends PageFactory
2) PageFactory.initElements(driver, this);

## Changes on BasePage

```
public abstract class BasePage {
    protected WebDriver driver;

    public BasePage() {
        driver = BaseTest.getDriver();
        PageFactory.initElements(driver, this);
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

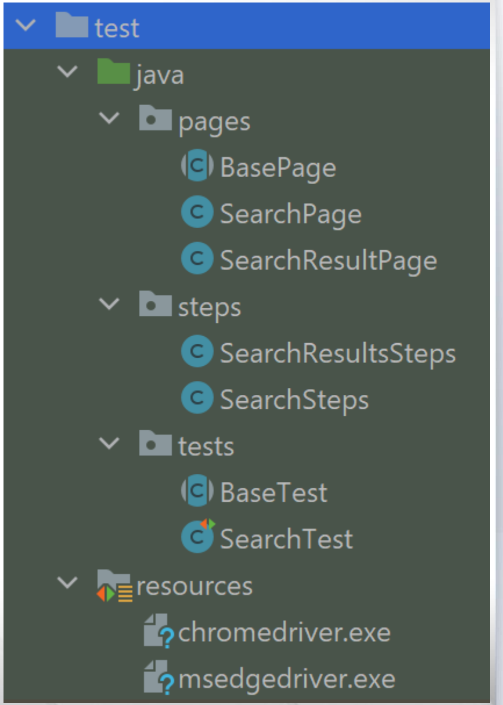
## Steps & Chain of  invocations

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

# Steps pattern

The additional layer: **business logic**

1. defines possible actions on each page
   - steps
   - verifications
2. encapsulates work on web page
3. binds the pages



```
test
  java
    pages
      BasePage
      SearchPage
      SearchResultPage
    steps
      SearchResultsSteps
      SearchSteps
    tests
      BaseTest
      SearchTest
  resources
    chromedriver.exe
    msedgedriver.exe
```

## Steps pattern implementation - SearchSteps

```java
public class SearchSteps {

    private SearchPage searchPage = new SearchPage();

    public SearchResultsSteps searchByKeyword(String keyword){

        searchPage.fillTheSearchField(keyword);
        searchPage.pressEnter();

        return new SearchResultsSteps();
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Steps pattern implementation - SearchResultsSteps

```
public class SearchResultsSteps {

    private SearchResultsPage searchResultsPage = new SearchResultsPage();


    public SearchResultsSteps verifyThatTopValueIsCorrect(String expectedValue) {

        searchResultsPage.assertThatExpectedValueIsOnSearchTop(expectedValue);
        return this;

    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Steps pattern implementation - BaseTest

```
SearchSteps steps;

//steps variable can be initialized in some @Before method, like below:

@BeforeClass
public void setUp() {

    // some code here

    steps = new SearchSteps();

}
```

## @Test method with Steps pattern

```
@Test
public void searchByKeywordSeleniumHaveToFindSeleniumhqOrgInTop() {

   steps.searchByKeyword("Selenium")
        .verifyThatTopValueIsCorrect("https://www.seleniumhq.org/");


}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Exercise

### Add new step

Extend test described in the presentation:

1) use the same scenario with `doSearchWithKeyword("Selenium")`
 - do search in Google
2) add new verification step:

Verify that all search results on the page actually contains keyword “Selenium”

Use List <WebElement> elements;

## Patterns: why and when?

**Why you should know them**

- > to understand different projects

**Use it when**

- > when it bring benefits: maintainability, readability, etc

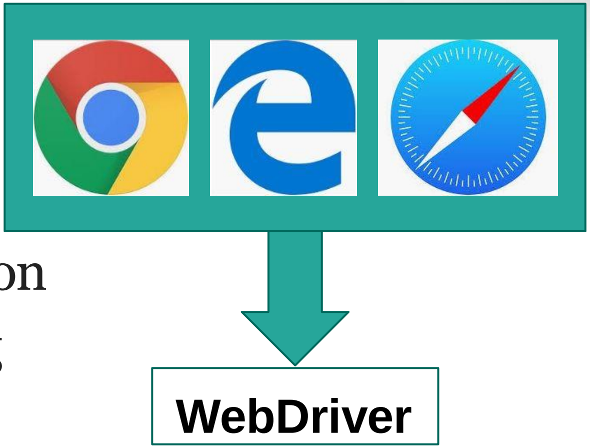
## Driver Factory method

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Driver factory method pattern

**It resolve problems:**

- encapsulates driver initialization and configuration
- handle cross browser testing in test framework



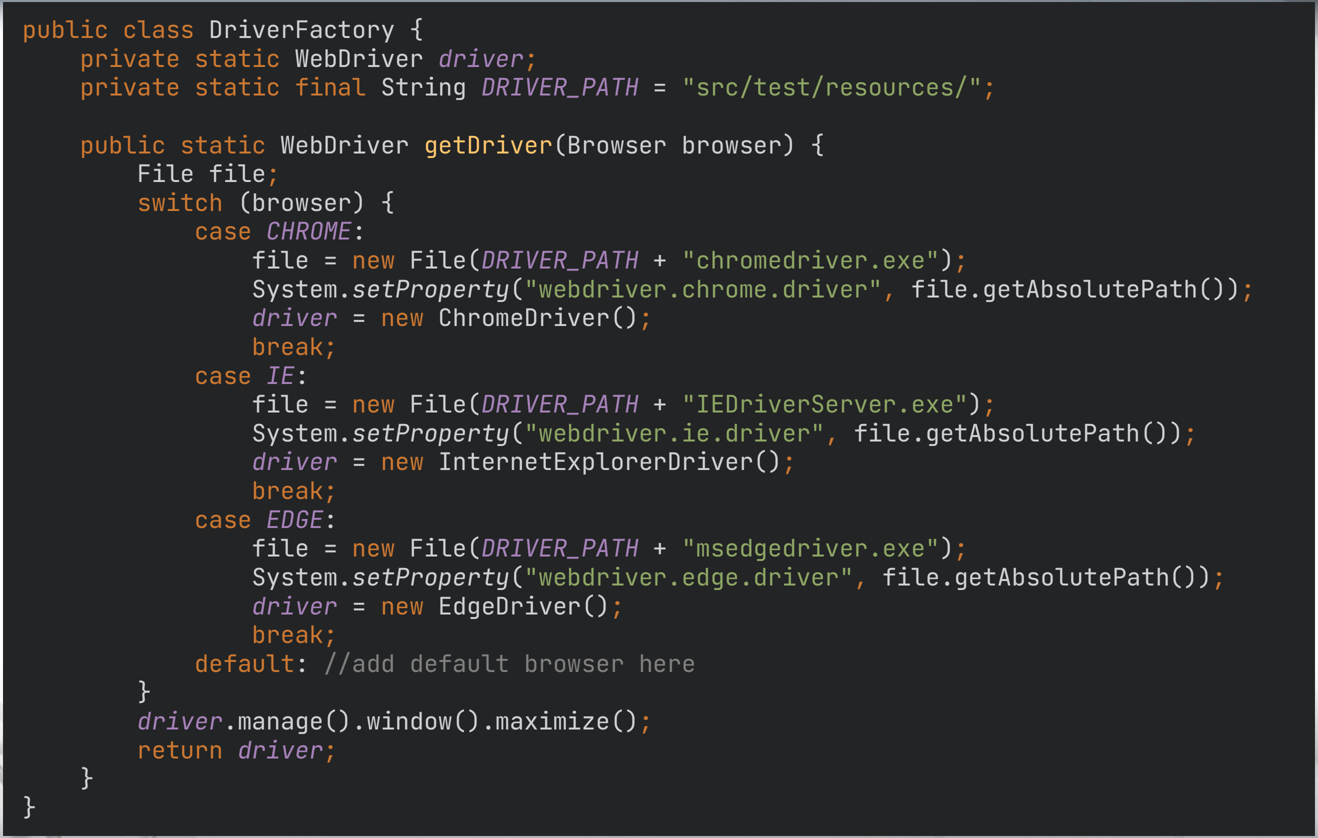
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Driver factory method: multiple browsers

```
public enum Browser {
    CHROME,
    IE,
    EDGE,
    FIREFOX,
    SAFARI
}
```

НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ
КОРАБЛЕБУДУВАННЯ
ІМЕНІ АДМІРАЛА МАКАРОВА

## Driver factory method implementation



```
public class DriverFactory {
    private static WebDriver driver;
    private static final String DRIVER_PATH = "src/test/resources/";

    public static WebDriver getDriver(Browser browser) {
        File file;
        switch (browser) {
            case CHROME:
                file = new File(DRIVER_PATH + "chromedriver.exe");
                System.setProperty("webdriver.chrome.driver", file.getAbsolutePath());
                driver = new ChromeDriver();
                break;
            case IE:
                file = new File(DRIVER_PATH + "IEDriverServer.exe");
                System.setProperty("webdriver.ie.driver", file.getAbsolutePath());
                driver = new InternetExplorerDriver();
                break;
            case EDGE:
                file = new File(DRIVER_PATH + "msedgedriver.exe");
                System.setProperty("webdriver.edge.driver", file.getAbsolutePath());
                driver = new EdgeDriver();
                break;
            default: //add default browser here
        }
        driver.manage().window().maximize();
        return driver;
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Driver factory method usage

```
@BeforeClass
public void setUp() {
    driver = DriverFactory.getDriver(Browser.CHROME);
    driver.get("https://www.google.com/");
    steps = new SearchSteps();
}
```

The factory method, which accepts desired browser and initializes the driver

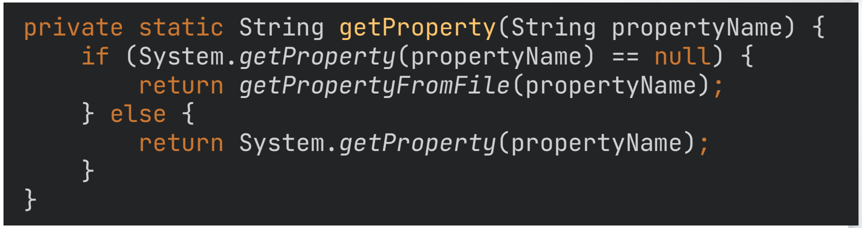
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Property reader

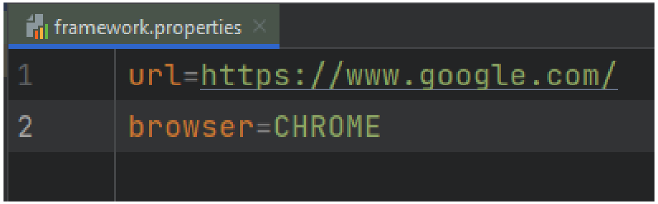
# Property reader

**It resolve problems:**

- avoiding hard coded properties
- reading system variables
- reading variables from file



```java
private static String getProperty(String propertyName) {
    if (System.getProperty(propertyName) == null) {
        return getPropertyFromFile(propertyName);
    } else {
        return System.getProperty(propertyName);
    }
}
```



framework.properties

```
url=https://www.google.com/
browser=CHROME
```

## Property reader implementation

```java
public class PropertyReader {

    public static String getBaseUrl() {
        return getProperty("url");
    }

    public static Browser getBrowser() {
        return Browser.valueOf(getProperty("browser"));
    }

    private static String getProperty(String propertyName) {
        if (System.getProperty(propertyName) == null) {
            return getPropertyFromFile(propertyName);
        } else {
            return System.getProperty(propertyName);
        }
    }

    // ...
```

Continues in next slide…

# Property reader implementation - continuation

```java
    // ...
    private static String getPropertyFromFile(String propertyName) {
        Properties prop = new Properties();
        try (InputStream input =
            new FileInputStream("src/test/resources/framework.properties")) {
            prop.load(input);
        } catch (IOException ex) {
            System.out.println("Cannot read property value for " +
                                        propertyName);
            ex.printStackTrace();
        }
        return prop.getProperty(propertyName);
    }

}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Property reader usage

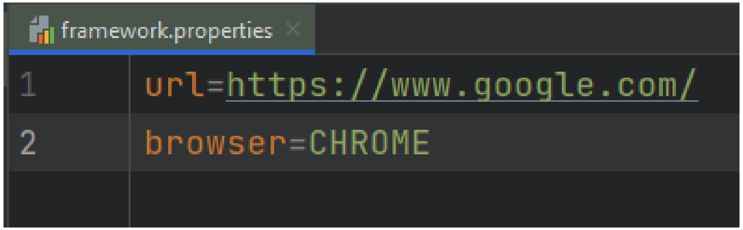
```
@BeforeClass
public void setUp() {

    driver = DriverFactory.getDriver(PropertyReader.getBrowser());

    driver.get(PropertyReader.getBaseUrl());

    steps = new SearchSteps();
}
```

Getting strings from properties using PropertyReader

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

## Property reader usage: sources of properties

- read from property file



```
url=https://www.google.com/
browser=CHROME
```

- read system variables

Run tests in console via maven:

```
mvn clean test -Dbrowser=CHROME -Durl=https://www.google.com.ua/
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

# Exercise

## Add new property

Add new property to your test framework

1. Add possibility to regulate the option below from properties

```
driver.manage().window().maximize();
```

3. This property should contains just “true” or “false”.

For example: maximize=true

so if maximize is true, the browser will be opened in the maximized mode.