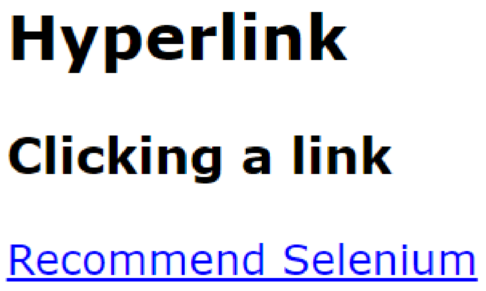
Web Elements

Hyperlink

- Hyperlinks (or links) are fundamental elements of web pages.
- As a matter of fact, it is hyperlinks that makes the World Wide Web possible.
- A sample link is provided below, along with the HTML source


Hyperlink

Clicking a link

Recommend Selenium

```
<a href="index.html" id="recommend_selenium_link"
   class="nav" data-id="123"
   style="font-size: 14px;">Recommend Selenium</a>
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Click a link by text

- Using text is probably the most direct way to click a link in Selenium, as it is what we see on the page

```
driver.get(TestHelper.siteUrl() + "link.html");
driver.findElement(By.linkText("Recommend Selenium"))
      .click();
```

Click a link by ID

```
driver.findElement(By.id("recommend_selenium_link"))
      .click();
```

- Furthermore, if you are testing a web site with multiple languages, using IDs is probably the only feasible option. You do not want to write test scripts like below:

```
if (is_ukrainian()) {
    driver.findElement(By.linkText("Увійти")).click();
} else if (is_english()) {
    driver.findElement(By.linkText("Sign in")).click();
} else {
    driver.findElement(By.linkText("Войти")).click();
}
```

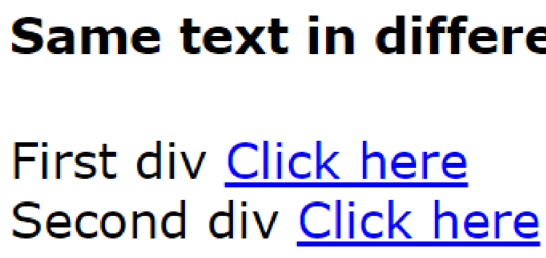
Click a link by partial text

```
driver.findElement(By.partialLinkText("Recommend Seleni"))
      .click();
```

Click a link by Xpath (1/2)

- The example below is finding a link with text ‘Recommend Selenium’ under a <p> tag

```
driver.findElement(By.xpath(
        "//p/a[text()='Recommend Selenium']"))
      .click();
```



Same text in different divs

First div Click here
Second div Click here

On this page, there are two ‘Click here’ links

```
<div>
    <h4>Same text in different divs</h4>
    <div>
        First div
        <a href="link-url.html">Click here</a>
    </div>
    <div id="second_div">
        Second div
        <a href="link-partial.html">Click here</a>
    </div>
    </p>
</div>
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Click a link by Xpath (2/2)

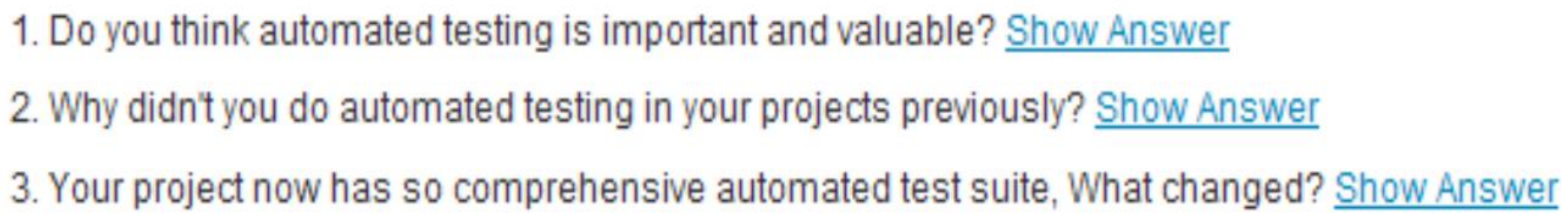
- If test case requires you to click the second ‘Click here’ link, the simple `findElement(By.linkText("Click here"))` won’t work (as it clicks the first one).
- Here is a way to accomplish using XPath:

```
driver.findElement(By.xpath(
  "//div[contains(text(), \"Second\")]/a[text()=\"Click here\"]"))
  .click();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Click Nth link with exact same label

- It is not uncommon that there are more than one link with exactly the same text.
- By default, Selenium will choose the first one.
- What if you want to click the second or Nth one?
- The web page below contains three 'Show Answer" links



1. Do you think automated testing is important and valuable? Show Answer
2. Why didn't you do automated testing in your projects previously? Show Answer
3. Your project now has so comprehensive automated test suite, What changed? Show Answer

```
assertThat(
        driver.findElements(By.linkText("Show Answer")).size())
        .as("Must be exactly two links")
        .isEqualTo(2);
// 2nd link
driver.findElements(By.linkText("Show Answer")).get(1).click();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Click Nth link by CSS Selector

- You may also use CSS selector to locate a web element

```
// 3rd link
driver.findElement(By.cssSelector("p > a:nth-child(3)")).click();
```

➢ However, generally speaking, the stylesheet are more prone to changes

Verify a link present or not?

```
assertThat(driver.findElement(
           By.linkText("Recommend Selenium")).isDisplayed())
        .isTrue();
assertThat(driver.findElement(
           By.id("recommend_selenium_link")).isDisplayed())
        .isTrue();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Getting link data attributes

- Once a web control is identified, we can get its other attributes of the element.
- This is generally applicable to most of the controls

```
WebElement seleniumLink = driver
        .findElement(By.linkText("Recommend Selenium"));
assertThat(seleniumLink.getAttribute("href"))
        .isEqualTo(TestHelper.siteUrl() + "index.html");
assertThat(seleniumLink.getAttribute("id"))
        .isEqualTo("recommend_selenium_link");
assertThat(seleniumLink.getText())
        .isEqualTo("Recommend Selenium");
assertThat(seleniumLink.getTagName())
        .isEqualTo("a");
```

Getting link data attributes

- Also you can get the value of custom attributes of this element and its inline CSS style

```java
assertThat(driver.findElement(By.id("recommend_selenium_link"))
        .getAttribute("style"))
        .isEqualTo("font-size: 14px;");
// using attribute_value("style") won't work
assertThat(driver.findElement(By.id("recommend_selenium_link"))
        .getAttribute("data-id"))
        .isEqualTo("123");
```

Test links
open a new browser window

- Clicking the link below will open the linked URL in a new browser window or tab

```
<a href="https://eolymp.com/uk"
   target="_blank">Open EOlymp in new window</a>
```

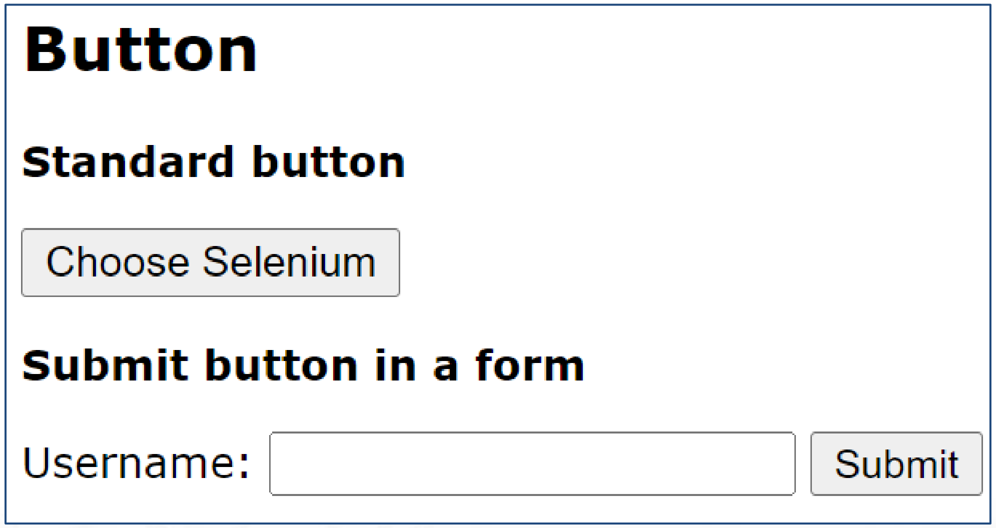
While we could use `switchTo()` method to find the new browser window, it will be easier to perform all testing within one browser window:

```
String currentUrl = driver.getCurrentUrl();
String newWindowUrl = driver.findElement(
    By.linkText("Open new window")).getAttribute("href");
driver.navigate().to(newWindowUrl);
driver.findElement(By.name("name")).sendKeys("sometext");
driver.navigate().to(currentUrl); // back
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Button

- Buttons can come in two forms - standard and submit buttons.
- Standard buttons are usually created by the ‘button’ tag,
- whereas submit buttons are created by the ‘input’ tag (normally within form controls)


Button

Standard button

Submit button in a form

Username:

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Button

- Buttons can come in two forms - standard and submit buttons.
- Standard buttons are usually created by the ‘button’ tag,
- whereas submit buttons are created by the ‘input’ tag (normally within form controls)

```
<h2>Button</h2>
<p><b>Standard button</b></p>

<p><button id="choose_selenium_btn" class="nav" data-id="123"
     style="font-size: 14px;">Choose Selenium</button></p>

<div>
    <p><b>Submit button in a form</b></p>
    <form name="input" action="index.html" method="get">
        Username: <input type="text" name="user">
        <input type="submit" name="submit_action" value="Submit"/>
    </form>
</div>
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Click a button by label

```
driver.get(TestHelper.siteUrl() + "button.html");
driver.findElement(
    By.xpath("//button[contains(text(),'ChooseSelenium')]"))
    .click();
```

- For an input button (in a HTML input tag) in a form, the text shown on the button is the ‘value’ attribute which might contain extra spaces or invisible characters

```
<input type="submit" name="submit_action" value="Space After "/>
```

```
// fail
driver.findElement(
        By.xpath("//input[@value='Space After']")).click();
```

```
// success
driver.findElement(
        By.xpath("//input[@value='Space After ']")).click();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Submit a form

- In the official Selenium tutorial, the operation of clicking a form submit button is done by calling *submit* function on an input element within a form.
- For example, the test script below is to test user sign in.

```
WebElement usernameElement = driver.findElement(By.name("user"));
usernameElement.sendKeys("eugeny");
WebElement passwordElement = driver.findElement(By.name("pwd"));
passwordElement.sendKeys("secret");
usernameElement.submit();
```

```
driver.findElement(By.name("user")).sendKeys("eugeny");
driver.findElement(By.name("pwd")).sendKeys("secret");
driver.findElement(
        By.xpath("//input[@value='Sign in']")).click();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Click a button by ID

- As always, a better way to identify a button is to use IDs.
- This applies to all controls, if there are IDs present

```
driver.findElement(By.id("choose_selenium_btn")).click();
```

- ➢ *For testers who work with the development team, rather than spending hours finding a way to identify a web control, just go to programmers and ask them to add IDs.*
- ➢ *It usually takes very little effort for programmers to do so.*

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Click a button by name

- In an input button, we can use a new generic attribute name to locate a control.

```
driver.findElement(By.name("submit_action")).click();
```

Click a image button

- There is also another type of ‘button’: an image that works like a submit button in a form

Image button

```
<div>
    <p><b>Image button</b></p>
    <form name="input2" action="assert.html" method="get">
        <input type="image" src="images/button_go.jpg"/>
    </form>
</div>
```

Besides using ID, the button can also be identified by using *src* attribute

```
driver.findElement(By.xpath(
        "//input[contains(@src, 'button_go.jpg')]"))
      .click();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Click a button via JavaScript

- You may also invoke clicking a button via JavaScript.
- Sometimes normal approaches didn’t click a button reliably on Firefox (for example), but this JavaScript way worked well

```
WebElement a_btn = driver.findElement(
      By.id("choose_selenium_btn"));
((JavascriptExecutor) driver).executeScript(
        "arguments[0].click();",
        a_btn);
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Assert a button present

- Just like hyperlinks, we can use isDisplayed() to check whether a control is present on a web page

```
assertThat(driver.findElement(By.id("choose_selenium_btn"))
    .isDisplayed()).isTrue();
driver.findElement(By.linkText("Hide")).click();
wait.withTimeout(Duration.ofMillis(500));
assertThat(driver.findElement(By.id("choose_selenium_btn"))
    .isDisplayed()).isFalse();
```

This check applies to most of the web controls, such as HyperLinks, Text Fields, .., etc

```
assertThat(driver.findElement(By.linkText("Show"))
    .isDisplayed()).isTrue();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

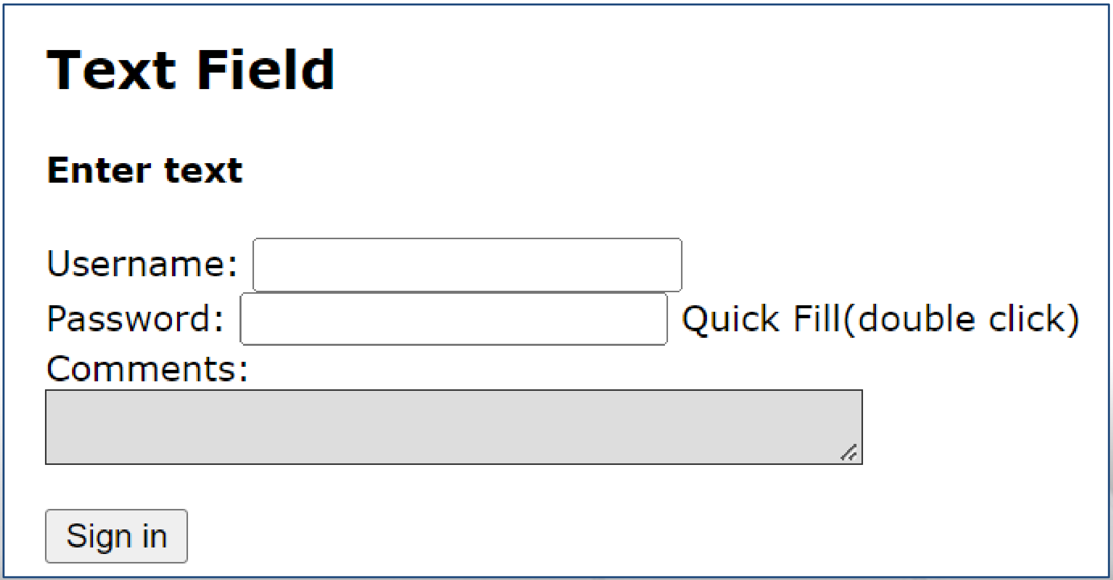
Assert a button enabled or disabled?

- A web control can be in a disabled state.
- A disabled button is un-clickable, and it is displayed differently.
- Normally enabling or disabling buttons (or other web controls) is triggered by JavaScripts.

```
assertThat(driver.findElement(By.id("choose_selenium_btn"))
    .isEnabled()).isTrue();
driver.findElement(By.linkText("Disable")).click();
wait.withTimeout(Duration.ofMillis(500));
assertThat(driver.findElement(By.id("choose_selenium_btn"))
    .isEnabled()).isFalse();
driver.findElement(By.linkText("Enable")).click();
wait.withTimeout(Duration.ofMillis(500));
assertThat(driver.findElement(By.id("choose_selenium_btn"))
    .isEnabled()).isTrue();
```

TextField and TextArea

- Text fields are commonly used in a form to pass user entered text data to the server. There are two variants (prior to HTML5): password fields and text areas.
- The characters in password fields are masked (shown as asterisks or circles).
- Text areas allows multiple lines of texts



Text Field

Enter text

Username:
Password: Quick Fill(double click)
Comments:

TextField and TextArea

```
<h2>Text Field</h2>

<h4>Enter text</h4>

<form action="index.html">
    <span style="width: 80px;">Username:</span>
    <input type="text" name="username" id="user"><br/>
    <span style="width: 80px;">Password:</span>
    <input type="password" name="password" id="pass">
    <span id="quickfill" ondblclick="quick_fill()">
    Quick Fill(double click)
    </span>
    <br/>
    Comments: <br/>
    <textarea id="comments" rows="2" cols="40" name="comments"/>
    <br/>
    <br/>
    <input type="submit" value="Sign in"/>
</form>
```

Enter text into a text field by name

- Selenium’s sendKeys ‘types’ characters (including invisible ones such Enter key) into a text box.

```
driver.get(TestHelper.siteUrl() + "text_field.html");
driver.findElement(By.name("username")).sendKeys("eugeny");
```

The ‘name’ attribute is the identification used by the programmers to process data. It applies to all the web controls in a standard web form.

Enter text into a text field by ID

```
driver.findElement(By.id("user")).sendKeys("eugeny");
```

In Selenium, password text fields are treated as normal text fields, except that the entered text is masked

```
driver.findElement(By.id("pass")).sendKeys("secret");
```

Calling sendKeys() to the same text field will concatenate the new text with the old text. So it is a good idea to clear a text field first, then send keys to it

```
driver.findElement(By.name("username")).sendKeys("test");
driver.findElement(By.name("username")).sendKeys(" 123");
// now => 'test 123'
driver.findElement(By.name("username")).clear();
driver.findElement(By.name("username")).sendKeys("eugeny");
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Enter text into a multi-line text area

- Selenium treats text areas the same as text fields

```
driver.findElement(By.id("comments"))
      .sendKeys("Automated testing is\r\nFun!");
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Assert value

```
driver.findElement(By.id("user")).sendKeys("eugeny");
assertThat(driver.findElement(By.id("user"))
     .getAttribute("value"))
     .isEqualTo("eugeny");
```

Focus on a control

- Once we identify one control, we can set the focus on it. There is no focus function on element in Selenium, we can achieve ‘focusing a control’ by sending empty keystrokes to it.

```
driver.findElement(By.id("pass")).sendKeys("");
```

Or using JavaScript

```
WebElement elem = driver.findElement(By.id("pass"));
((JavascriptExecutor) driver)
     .executeScript("arguments[0].focus();", elem);
```

This workaround can be quite useful. When testing a long web page and some controls are not visible, trying to click them might throw “Element is not visible” error. In that case, setting the focus on the element might make it a visible.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Set a value to a read-only or disabled text field

- ‘Read only’ and ‘disabled’ text fields are not editable and are shown differently in the browser (typically grayed out)

```
<!--     Read only text field:-->
<input type="text" name="readonly_text" readonly="true"/>
<br/>
<!--     Disabled text field:-->
<input type="text" name="disabled_text" disabled="true"/>
```

If a text box is set to be read-only, the following test step will not work

```
driver.findElement(By.name("readonly_text"))
      .sendKeys("new value");
```

Here is a workaround (with JS)

```
((JavascriptExecutor) driver)
    .executeScript("$('#readonly_text').val('bypass');");
assertThat(driver.findElement(By.id("readonly_text"))
    .getAttribute("value")).isEqualTo("bypass");
((JavascriptExecutor) driver)
    .executeScript("$('#disabled_text').val('anyuse');");
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Set and assert the value of a hidden field

- A hidden field is often used to store a default value.

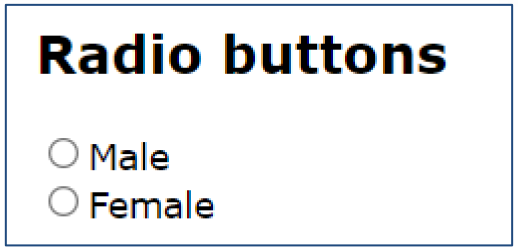
```
<input type="hidden" name="currency" value="USD"/>
```

The below test script asserts the value of the above hidden field and changes its value using JavaScript.

```
WebElement hiddenElem = driver.findElement(By.name("currency"));
assertThat(hiddenElem.getAttribute("value")).isEqualTo("USD");
((JavascriptExecutor) driver).executeScript(
     "arguments[0].value = 'AUD';", hiddenElem);
assertThat(hiddenElem.getAttribute("value")).isEqualTo("AUD");
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Radio button



Radio buttons

○ Male
○ Female

```
<form>
  <input type="radio" name="gender" value="male" id="radio_male">Male<br>
  <input type="radio" name="gender" value="female" id="radio_female">Female
</form>
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Select a radio button

```
driver.findElement(
      By.xpath("//input[@name='gender' and @value='female']"))
      .click();
wait.withTimeout(Duration.ofMillis(500));
driver.findElement(
      By.xpath("//input[@name='gender' and @value='male']"))
      .click();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Select a radio button

```
driver.findElement(
      By.xpath("//input[@name='gender' and @value='female']"))
      .click();
wait.withTimeout(Duration.ofMillis(500));
driver.findElement(
      By.xpath("//input[@name='gender' and @value='male']"))
      .click();
```

- The radio buttons in the same radio group have the same name. To click one radio option, the value needs to be specified.
- Note that the value is not the text shown next to the radio button, that is the label. To find out the value of a radio button, inspect the HTML source.
- As always, if there are IDs, using :id finder is easier.

```
driver.findElement(By.id("radio_female")).click();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Clear radio option selection

It is OK to click a radio button that is currently selected, however, it would not have any effect.

```
driver.findElement(By.id("radio_female")).click();
// already selected, no effect
driver.findElement(By.id("radio_female")).click();
```

- Once a radio button is selected, you cannot just clear the selection in Selenium.
- You need to select another radio button.
- The test script below will throw an error: “invalid element state: Element must be usereditable in order to clear it.“

```
driver.findElement(By.xpath("//input[@name='gender' and
@value='female']")).click();
try {
    driver.findElement(
        By.xpath("//input[@name='gender' and @value='female']")).clear();
} catch (Exception ex) {
// Selenium does not allow
    System.out.println("Selenium doesn’t allow clear currently selected" +
                       " radio button, just select another one");
    driver.findElement(
    By.xpath("//input[@name='gender' and @value='male']")).click();
}
```

Assert a radio option is selected

```
driver.findElement(
    By.xpath("//input[@name='gender' and @value='female']")).click();
assertTrue(driver.findElement(
    By.xpath("//input[@name='gender' and @value='female']")).isSelected());
assertFalse(driver.findElement(
    By.xpath("//input[@name='gender' and @value='male']")).isSelected());
```

Iterate radio buttons in a radio group

```
assertThat(driver.findElements(By.name("gender")).size()).isEqualTo(2);
for (WebElement rb : driver.findElements(By.name("gender"))) {
    if (rb.getAttribute("value").equals("female")) {
        rb.click();
    }
}
```

- Different from `findElement` which returns one matched control, `findElements` return a list of them (also known as an array) back.
- This can be quite handy especially when controls are hard to locate.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Click Nth radio button in a group

```
driver.findElements(
    By.name("gender")).get(1).click();
assertTrue(driver.findElement(
    By.xpath("//input[@name='gender' and @value='female']")).isSelected());
driver.findElements(
    By.name("gender")).get(0).click();
assertTrue(driver.findElement(
    By.xpath("//input[@name='gender' and @value='male']")).isSelected());
```

For example: Online calendar.

- There were many time-slots, and the HTML for each of these time-slots were exactly the same.
- Every time slot can be identified by using the index (as above) on one of these ‘plural’ locators

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Click radio button by the following label (1/2)

- Some code generators may generate poor quality HTML fragments like the one below:

```
<p>The radio buttons share the same ID and name, with value dynamically generated</p>
<div id="q1" class="question">
  <div class="question-answer col-lg-5">
    <div class="yes-no">
      <input id="QuestionViewModels_1__SelectedAnswerId“
             name="QuestionViewModels[1].SelectedAnswerId"
             type="radio" value="c225306e-8d8e-45b0-8261-22617d9796b5">
      <label for="QuestionViewModels_1__SelectedAnswerId">Yes</label>
    </div>
    <div class="yes-no">
      <input id="QuestionViewModels_1__SelectedAnswerId"
             name="QuestionViewModels[1].SelectedAnswerId"
             type="radio" value="85ff8db7-1c58-47a2-a978-581200fb7098">
      <label for="QuestionViewModels_1__SelectedAnswerId">No</label>
    </div>
  </div>
</div>
```

- The `id` attribute of the above two radio buttons are the same, and the `values` are meaningless to human.
- The only thing can be used to identify a radio button is the text in `label` elements.
- The solution is to use XPath locator.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Click radio button by the following label (2/2)

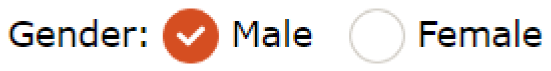
```
<p>The radio buttons share the same ID and name, with value dynamically generated</p>
<div id="q1" class="question">
  <div class="question-answer col-lg-5">
    <div class="yes-no">
      <input id="QuestionViewModels_1__SelectedAnswerId“
             name="QuestionViewModels[1].SelectedAnswerId"
             type="radio" value="c225306e-8d8e-45b0-8261-22617d9796b5">
      <label for="QuestionViewModels_1__SelectedAnswerId">Yes</label>
    </div>
    <div class="yes-no">
      <input id="QuestionViewModels_1__SelectedAnswerId"
             name="QuestionViewModels[1].SelectedAnswerId"
             type="radio" value="85ff8db7-1c58-47a2-a978-581200fb7098">
      <label for="QuestionViewModels_1__SelectedAnswerId">No</label>
    </div>
  </div>
</div>
```

- You might have noticed that input (radio button) and label are siblings in the HTML DOM tree.
- We can use this relation to come up a XPath that identifies the label text, then the radio button.

```
WebElement elem = driver.findElement(By.xpath(
     "//div[@id='q1']//label[contains(.,'Yes')]/../input[@type='radio']"));
elem.click();
```

Customized Radio buttons - iCheck

- There are a number of plugins that customize radio buttons into a more stylish form, like the one below (using iCheck).



- The iCheck JavaScript transforms the radio button HTML fragment

```
<input type="radio" name="sex" id="q2_1" value="male"> Male
```

to

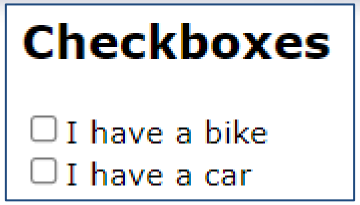
```
<div class="iradio_square-red" style="position: relative;">
  <input type="radio" name="sex" id="q2_1" value="male" style="..." />
  <ins class="iCheck-helper" style="..." />
</div>
```

Here are test scripts to drive iCheck radio buttons:

```
// Error: Element is not clickable
// driver.findElement(By.id("q2_1")).click();
driver.findElements(By.className("iradio_square-red")).get(0).click();
driver.findElements(By.className("iradio_square-red")).get(1).click();
// More precise with XPath
driver.findElement(By.xpath(
   "//div[contains(@class, " +
   "'iradio_square-red')]/input[@type='radio' and @value='male']/..")
   .click();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

CheckBox



Checkboxes

☐ I have a bike
☐ I have a car

```
<input type="checkbox" name="vehicle_bike" value="on" id="checkbox_bike">
   I have a bike<br>
<input type="checkbox" name="vehicle_car" id="checkbox_car">
   I have a car
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Check/Uncheck by name / id

```
driver.get(TestHelper.siteUrl() + "checkbox.html");
```

Check by name

```
driver.findElement(By.name("vehicle_bike")).click();
```

Check by id

```
WebElement the_checkbox =
driver.findElement(By.id("checkbox_car"));
if (!the_checkbox.isSelected()) {
    the_checkbox.click();
}
```

Uncheck a checkbox

```
WebElement the_checkbox =
driver.findElement(By.id("checkbox_car"));
the_checkbox.click();
if (the_checkbox.isSelected()) {
    the_checkbox.click();
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

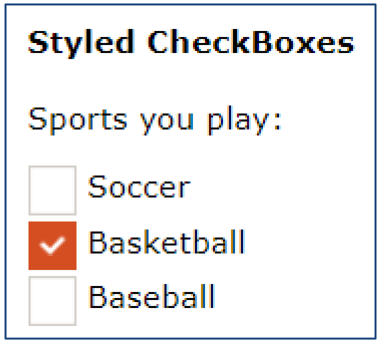
Assert a checkbox is checked (or not)

```
WebElement theCheckbox = driver.findElement(
    By.name("vehicle_bike"));

assertFalse(theCheckbox.isSelected());

the_checkbox.click();

assertTrue(theCheckbox.isSelected());
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Customized Checkboxes – iCheck (1/2)

- There are a number of plugins that customize radio buttons into a more stylish form, like the one below (using iCheck)



Styled CheckBoxes

Sports you play:

☐ Soccer
✓ Basketball
☐ Baseball

The iCheck JavaScript transforms the checkbox HTML fragment

```
<input type="checkbox" name="sports[]" value="Soccer"> Soccer <br/>
```

to

```
<div class="icheckbox_square-red" style="position: relative;">
  <input type="checkbox" name="sports[]" value="Soccer" style="..."/>
  <ins class="iCheck-helper" style="..."/>
</div>
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

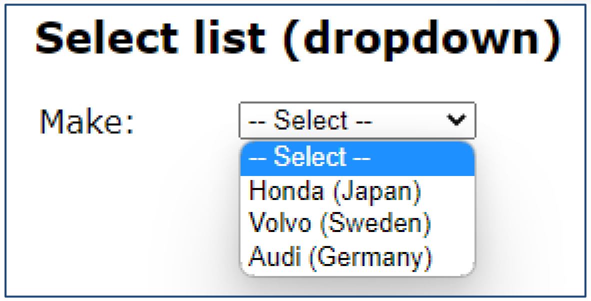
Customized Checkboxes – iCheck (2/2)

- Here are test scripts to drive iCheck checkboxes

```
driver.findElements(By.className("icheckbox_square-red")).get(0).click();
driver.findElements(By.className("icheckbox_square-red"))
      .get(1).click();
// More precise with XPath
driver.findElement(By.xpath(
    "//div[contains(@class, 'icheckbox_square-red')] " +
    " /input[@type='checkbox' and @value='Soccer']/.."))
    .click();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Select List



Select list (dropdown)

Make: -- Select --

- -- Select --
- Honda (Japan)
- Volvo (Sweden)
- Audi (Germany)

HTML Source

```
<select name="car_make" id="car_make_select">
  <option value="">-- Select --</option>
  <option value="honda">Honda (Japan)</option>
  <option value="volvo">Volvo (Sweden)</option>
  <option value="audi">Audi (Germany)</option>
</select>
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Select an option by text / value

- The label of a select list is what we can see in the browser.

```
driver.get(TestHelper.siteUrl() + "select_list.html");
```

by text

```
Select select = new Select(driver.findElement(By.name("car_make")));
select.selectByVisibleText("Volvo (Sweden)");
```

by value

```
Select select = new Select(driver.findElement(By.id("car_make_select")));
select.selectByValue("audi");
```

The value of a select list is what to be passed to the server

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Select an option by iterating all options

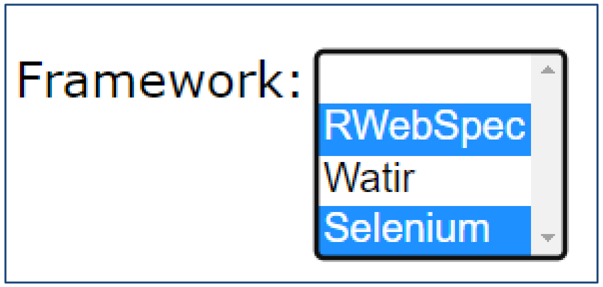
- A select list contains options, where each option itself is a valid control in Selenium

```
WebElement selectElem = driver.findElement(By.id("car_make_select"));
for (WebElement option : selectElem.findElements(By.tagName("option"))) {
    if (option.getText().equals("Volvo (Sweden)")) {
        option.click();
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Select multiple options

- A select list also supports multiple selections



HTML Source

```
<select id="framework_select" name="test_framework" multiple="multiple">
  <option></option>
  <option value="rwebspec">RWebSpec</option>
  <option value="watir">Watir</option>
  <option value="selenium">Selenium</option>
</select>
```

```
Select select = new Select(driver.findElement(By.name("test_framework")));
select.selectByVisibleText("Selenium");
select.selectByValue("rwebspec");
select.selectByIndex(2);
assertThat(select.getAllSelectedOptions().size()).isEqualTo(3);
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Clear one selection

```
Select select = new Select(driver.findElement(By.name("test_framework")));
select.selectByVisibleText("RWebSpec");
select.selectByVisibleText("Selenium");
select.deselectByVisibleText("RWebSpec");
select.deselectByValue("selenium");
// one more
// select.deselectByIndex(0);
assertThat(select.getAllSelectedOptions()).isEmpty();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Assert label or value in a select list

- To verify a label or value is in a select list

```java
WebElement selectElem = driver.findElement(By.id("car_make_select"));
List<String> selectLabels = new ArrayList<>();
List<String> selectValues = new ArrayList<>();
for (WebElement option : selectElem.findElements(By.tagName("option"))) {
    selectLabels.add(option.getText());
    selectValues.add(option.getAttribute("value"));
}
assertThat(selectLabels).contains("Audi (Germany)");
assertThat(selectValues).contains("audi");
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Assert selected option label

- To verify a particular option is currently selected in a select list

```
Select select = new Select(driver.findElement(By.id("car_make_select")));
select.selectByValue("audi");
assertThat(select.getFirstSelectedOption().getText())
          .isEqualTo("Audi (Germany)");
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Assert the value of a select list

- Another quick (and simple) way to check the current selected value of a select list:

```
Select select = new Select(driver.findElement(By.id("car_make_select")));
select.selectByVisibleText("Volvo (Sweden)");
assertThat(select.getFirstSelectedOption().getAttribute("value"))
        .isEqualTo("volvo");
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Assert multiple selections

- A multiple select list can have multiple options being selected

```
Select select = new Select(driver.findElement(By.name("test_framework")));
select.selectByVisibleText("Selenium");
select.selectByVisibleText("RWebSpec");

List<WebElement> selected = select.getAllSelectedOptions();
assert selected.size() == 2;
assert "RWebSpec".equals(selected.get(0).getText()); // display order
assert "Selenium".equals(selected.get(1).getText());
```

Note, even though the test script selected ‘Selenium’ first, when it comes to assertion, the first selected option is ‘RWebSpec’, not ‘Selenium’.

Navigation and Browser

- **Go to a URL**

```
driver.get("http://eolymp.com");
driver.navigate().to("https://google.com");
```

- **Visit pages within a site**

`driver.navigate().to()` takes a full URL. Most of time, testers test against a single site and specifying a full URL (such as `http://…`) is not necessary.
Let’s create a reusable function to simplify its usage.

```
String site_root_url = "http://eolymp.com";
// ...
public void visit(String path) {
    driver.navigate().to(site_root_url + path);
}

@Test
public void testGoToPageWithinSiteUsingFunction() {
    visit("/uk");
    visit("/en");
    visit("/"); // default
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Navigation and Browser

Perform actions from right click context menu such as ‘Back’, ‘Forward’ or ‘Refresh’

- Operations with right click context menu are commonly page navigations, such as “Back to previous page”. We can achieve the same by calling the test framework’s navigation operations directly

```
driver.navigate().back();
driver.navigate().refresh();
driver.navigate().forward();
```

Open browser in certain size

```
driver.manage().window().setSize(new Dimension(1024, 768));
```

Move browser window

```
driver.manage().window().setPosition(new Point(100, 100));
Thread.sleep(1000);
driver.manage().window().setPosition(new Point(0, 0));
```

Navigation and Browser

- **Maximize browser window**

```
driver.manage().window().maximize();
Thread.sleep(1000); // wait 1 second to see the effect
driver.manage().window().setSize(new Dimension(1024, 768));
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Navigation and Browser

- **Maximize browser window**

```
driver.manage().window().maximize();
Thread.sleep(1000); // wait 1 second to see the effect
driver.manage().window().setSize(new Dimension(1024, 768));
```

- **Minimize browser window**

Surprisingly, there is no `minimize` window function in Selenium. The hack below achieves the same:

```
driver.manage().window().setPosition(new Point(-2000, 0)); // hide
driver.findElement(By.linkText("Hyperlink")).click(); // still can use
Thread.sleep(2000);
driver.manage().window().setPosition(new Point(0, 0));
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Scroll focus to control

- For certain controls are not viewable in a web page (due to JavaScript), WebDriver is unable to click on them by returning an error like “Element is not clickable at point (1180, 43)”.
- The solution is to scroll the browser view to the control

```
WebElement elem = driver.findElement(By.name("submit_action_2"));
Integer elemPos = elem.getLocation().getY();
((JavascriptExecutor)driver).executeScript("window.scroll(0, "+elemPos+");");
Thread.sleep(500);
elem.click();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Switch between browser windows or tabs

- A “`target='_blank'`” hyperlink opens a page in another browser window or tab (depending on the browser setting).
- Selenium drives the browser within a scope of one browser window.
- However, we can use Selenium’s `switchTo` function to change the target browser

```
driver.findElement(By.linkText("Open new window")).click();
Set<String> windowHandles = driver.getWindowHandles();
String firstTab = windowHandles.iterator().next();
String lastTab = null;
for (String windowHandle : windowHandles) {
    lastTab = windowHandle;
}
driver.switchTo().window(lastTab);
assertThat(driver.findElement(By.tagName("body")).getText())
    .contains("This is url link page");
driver.switchTo().window(firstTab); // back to first tab/window
assertTrue(driver.findElement(By.linkText("Open new window"))
    .isDisplayed());
```

Open new and close existing browser Tabs (1/2)

- To open a browser Tab, we should use the JavaScript code:

```
((JavascriptExecutor) driver).executeScript(
          "window.open('http://facebook.com', '_blank’)”
);
```

To close a browser Tab, make sure it is ‘focused’ (using `switchTo().window()`) and call `driver.close() => (next slide)`

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Open new and close existing browser Tabs (2/2)

```java
String newPageUrl = "https://eolymp.com";
JavascriptExecutor jse = (JavascriptExecutor) driver;
jse.executeScript("window.open('" + newPageUrl + "', '_blank')");
Set<String> winHandles = driver.getWindowHandles();
int tabCount = winHandles.size();
driver.switchTo().window((String) winHandles.toArray()[tabCount - 1]);
Thread.sleep(1000);
//driver.findElement(By.linkText("PRICING")).click(); // on new tab
driver.findElement(By.xpath("/html/body/header/nav[1]/a[7]")).click(); // блог
// Now try to close first tab
driver.switchTo().window((String) winHandles.toArray()[0]);
Thread.sleep(1000);
driver.close();
winHandles = driver.getWindowHandles();
assertThat(winHandles.size()).isEqualTo(tabCount - 1);
// Refocus
int lastTabIndex = winHandles.size() - 1;
driver.switchTo().window((String) winHandles.toArray()[lastTabIndex]);
Thread.sleep(1000);
driver.findElement(By.xpath("//*[@id=\"content\"]/div/section/article[1]/a")).click();
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Remember current web page URL, then come back to it later

```
String url; // instance variable

@Before
public void before() throws Exception {
    url = driver.getCurrentUrl();
}

//...
@Test
public void testGoBackUrlSetElsewhere() throws Exception {
    driver.findElement(By.linkText("Button")).click();
//...
    driver.navigate().to(url);
}
```