

SELENIUM FAQ'S

Part-2

Ques.26. How can we clear a text written in a textbox?

- Ans. Using clear() method we can delete the text written in a textbox.

```
driver.findElement(By.id("elementLocator")).clear();
```


Ques.27. How to check a checkBox in selenium?

- Ans. The same `click()` method used for clicking buttons or radio buttons can be used for checking checkbox as well.

Ques.28. How can we submit a form in selenium?

- Ans. Using submit() method we can submit a form in selenium.

```
driver.findElement(By.id("form1")).submit();
```

- Also, the click() method can be used for the same purpose.

Ques.29. Explain the difference between close and quit command.

- Ans. *driver.close()* - Used to close the current browser having focus
driver.quit() - Used to close all the browser instances

Ques.30. How to switch between multiple windows in selenium?

- Ans. Selenium has `driver.getWindowHandles()` and `driver.switchTo().window("{windowHandleName}")` commands to work with multiple windows.
- The `getWindowHandles()` command returns a list of ids corresponding to each window and on passing a particular window handle to `driver.switchTo().window("{windowHandleName}")` command we can switch control/focus to that particular window

```
for (String windowHandle : driver.getWindowHandles())  
{  
    driver.switchTo().window(handle);  
}
```

Ques.31. What is the difference between driver.getWindowHandle() and driver.getWindowHandles() in selenium?

- Ans. driver.getWindowHandle() returns a handle of the current page (a unique identifier)
Whereas driver.getWindowHandles() returns a set of handles of the all the pages available.

Ques.32. How can we move to a particular frame in selenium?

- Ans. The *driver.switchTo()* commands can be used for switching to frames.

```
driver.switchTo().frame("{frameIndex/frameId/frameName}");
```

- For locating a frame we can either use the index (starting from 0), its name or Id.

Ques.33. Can we move back and forward in browser using selenium?

- Ans. Yes, using *driver.navigate().back()* and *driver.navigate().forward()* commands we can move backward and forward in a browser.

Ques.34. Is there a way to refresh browser using selenium?

- Ans. There are multiple ways to refresh a page in selenium-
- Using `driver.navigate().refresh()` command
- Using `sendKeys(Keys.F5)` on any textbox on the webpage

Ques.35. How can we maximize browser window in selenium?

- Ans. We can maximize browser window in selenium using following command-

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


Ques.36. How can we fetch a text written over an element?

- Ans. Using *getText()* method we can fetch the text over an element.

```
String text = driver.findElement("elementLocator").getText();
```

Ques.37. How can we find the value of different attributes like name, class, value of an element?

- Ans. Using `getAttribute("{attributeName}")` method we can find the value of different attributes of an element e.g.-

```
String valueAttribute = driver.findElement(By.id("elementLocator")).getAttribute("value");
```

Ques.38. How to delete cookies in selenium?

- Ans. Using deleteAllCookies() method-

```
driver.manage().deleteAllCookies();
```


Ques.39. What is an implicit wait in selenium?

- Ans. An implicit wait is a type of wait which waits for a specified time while locating an element before throwing NoSuchElementException. By default selenium tries to find elements immediately when required without any wait. So, it is good to use implicit wait. This wait is applied to all the elements of the current driver instance.

```
driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);
```

Ques.40. What is an explicit wait in selenium?

- Ans. An explicit wait is a type of wait which is applied to a particular web element until the expected condition specified is met.

```
WebDriverWait wait = new WebDriverWait(driver, 10);
```

```
WebElement element = wait.until(ExpectedConditions.elementToBeClickable(By.id("elementId")));
```

Ques.41. What are some expected conditions that can be used in Explicit waits?

- Ans. Some of the commonly used expected conditions of an element that can be used with explicit waits are-
 - `elementToBeClickable(WebElement element or By locator)`
 - `visibilityOfElementLocated(By locator)`
 - `attributeContains(WebElement element, String attribute, String value)`
 - `alertIsPresent()`
 - `titleContains(String title)`
 - `titleIs(String title)`
 - `textToBePresentInElementLocated(By, String)`

Ques.42. What is fluent wait in selenium?

- Ans. A fluent wait is a type of wait in which we can also specify polling interval(intervals after which driver will try to find the element) along with the maximum timeout value.

```
Wait wait = new FluentWait(driver)
.withTimeout(20, SECONDS)
.pollingEvery(5, SECONDS)
.ignoring(NoSuchElementException.class);
WebElement textBox = wait.until(new Function<webdriver,webElement>()
{
    public WebElement apply(WebDriver driver) {
        return driver.findElement(By.id("textBoxId"));
    }
});
```

Ques.43. What are the different keyboard operations that can be performed in selenium?

- Ans. The different keyboard operations that can be performed in selenium are-
- **.sendKeys("sequence of characters")** - Used for passing character sequence to an input or textbox element.
- **.pressKey("non-text keys")** - Used for keys like control, function keys etc that are non-text.
- **.releaseKey("non-text keys")** - Used in conjunction with keypress event to simulate releasing a key from keyboard event.
-

Ques.44. What are the different mouse actions that can be performed?

- Ans. The different mouse events supported in selenium are
 - `click(WebElement element)`
 - `doubleClick(WebElement element)`
 - `contextClick(WebElement element)`
 - `moveToElement(WebElement element)`
 - `dragAndDrop(source WebElement, target WebElement)`

Ques.45. Write the code to double click an element in selenium?

```
Actions action = new Actions(driver);  
WebElement element=driver.findElement(By.id("elementId"));  
action.doubleClick(element).build().perform();
```

Ques.46. Write the code to right click an element in selenium?

```
Actions action = new Actions(driver);  
WebElement element=driver.findElement(By.id("elementId"));  
action.contextClick(element). build().perform();
```

Ques.47. How to mouse hover an element in selenium?

```
Actions action = new Actions(driver);  
WebElement element=driver.findElement(By.id("elementId"));  
action.moveToElement(element).build().perform();
```


Ques.48. How to fetch the current page URL in selenium?

- Ans. Using `getCurrentURL()` command we can fetch the current page URL-

```
driver.getCurrentUrl();
```

Ques.49. How can we fetch title of the page in selenium?

- Ans. Using `driver.getTitle()`; we can fetch the page title in selenium. This method returns a string containing the title of the webpage.

Ques.50. How can we fetch the page source in selenium?

- Ans. Using `driver.getPageSource()`; we can fetch the page source in selenium. This method returns a string containing the page source.