

Locators

in list Locators

Notifications

👁 Watch



Description Edit

abstract class By {

```
1 public static By id(String idValue) {  
2     =====  
3     =====  
4 }  
5 public static By name(String nameValue) {  
6     =====  
7     =====  
8 }  
9 public static By className(String classValue) {  
10    =====  
11    =====  
12 }
```

Suggested

👤 Join

Add to card

👤 Members

🔖 Labels

☑ Checklist

🕒 Dates

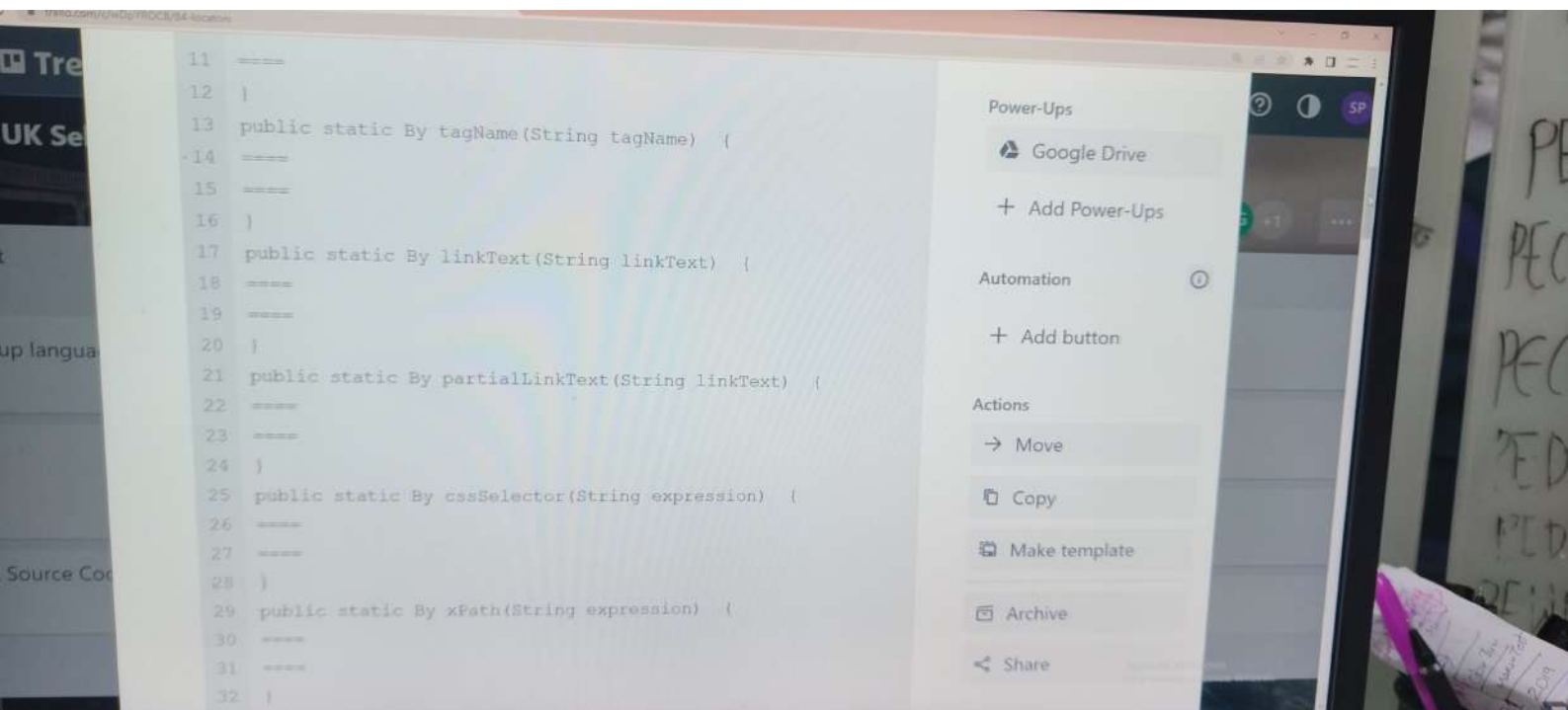
📎 Attachment

📁 Cover

📁 Custom Fields

Power-Ups

Activate Windows
Go to Settings to activate Windows.



☒ Introduction

Delete

0%

- ☐ Locators are the search criteria through which we can find the elements in the webpage.
- ☐ Locators are also called as Selectors.
- ☐ Selenium provides 8 Locators Strategies to find the element in the webpage they are:
 - ☐ 1) **id**
 - ☐ 2) **name**
 - ☐ 3) **className**
 - ☐ 4) **linkText**
 - ☐ 5) **partialLinkText**
 - ☐ 6) **tagName**
 - ☐ 7) **cssSelector**
 - ☐ 8) **xpath**

Activate Windows
Go to Settings to activate Windows.

- ☐ 8) **xpath**
- ☐ **NOTE 01** Here id, name and className are the attribute names in the source code of any WebElement.
- ☐ **NOTE 02** cssSelector and xpath are the Expressions which can be used to find the Element.
- ☐ Selenium developers provided a class called **By** in which all the Locators related static methods are Available.
- ☐ **By** is an abstract class which contains 8 Static methods which can be used to find the element in the web page.

Add an item

☒ **By - Abstract Class** Hide checked items Delete

10%

☒ See the Above Discription

☐ To use all the static methods we should use it in findElement() or findElements() as shown below:

☐ To use all the static methods we should use it in `findElement()` or `findElements()` as shown below:

- ☐ `By.id("idvalue");`
- ☐ `By.name("namevalue");`
- ☐ `By.className("classvalue");`
- ☐ `By.tagName("tagname");`
- ☐ `By.linkText("fulllinktext");`
- ☐ `By.partialLinkText("full or Partial Link Text");`
- ☐ `By.cssSelector("cssExpression");`
- ☐ `By.xpath("Xpath Expression");`

Add an item

☒ What is `findElement()` ...?

Delete

Activate Windows
Go to Settings to activate Windows.

☒ What is findElement() ...?

Delete

0%

- ☐ It is an abstract method of SearchContext interface which is used to search the element in the webpage.
- ☐ It accepts anyone of the 8 Locator strategies as an argument and based on that it will search the element in the webpage.
- ☐ It will start searching the element in the <html> from the beginning till the end </html> until the target element is found.
- ☐ If the element is found, it will return the address of that element or in other words if the element is found, it will return the WebElement interface type object using where we can perform some actions on it.
- ☐ `WebElement element = driver.findElement(By.id("LoginButton"));`
- ☐ If the element is not found with the given locator strategy then findElement() will throw NoSuchElementException.
- ☐ If the given locator strategy is matching with the multiple elements then findElement() will return the first matching element.
- ☐ Signature: `public WebElement findElement(By by)`

☐ Signature: public WebElement findElement(By by)

☐ Ex-01: -> Usage: WebElement element =
driver.findElement(By.id("username"));

☐ element.clear();

☐ element.sendKeys("admin");

☐ Ex-02: -> driver.findElement(By.id("LoginButton")).click();

Add an item

☒ **NoSuchElementException**

Delete

0%

☐ It's an unchecked exception of selenium which will be thrown:

☐ if the locator strategy is incorrect.

☐ If the element is not loaded (due to slow internet connection)

Add an item

☒ Points to be remembered:

Delete

0%

- ☐ NOTE: Whenever we upcast the browser object to WebDriver interface type reference then we can access only webdriver interface methods and it's parents i.e SearchContext interface methods.
- ☐ WebDriver driver = new ChromeDriver();
- ☐ We are achieving this by auto Upcasting.
- ☐ Other methods will be available in the browser object due to inheritance but not accessible through WebDriver reference.
- ☐ ChromeDriver driver = new ChromeDriver();
- ☐ So to access them we need to downcast to their interface type.

Add an item

☒ Priority of Locators while using findElement()

Delete

☒ Priority of Locators while using findElement()

Delete

0%

☐ id

☐ name

☐ linkText

☐ partialLinkText

☐ cssSelector

☐ xPath

☐ Note: Generally By.tagName() and By.className() are used to identify multiple elements because of this the chances of duplicates are more.

☐ Note: By.className() won't identify the element if the class attribute value is having a space and it gives InvalidSelectorException.

Add an item

Activate Windows
Go to Settings to activate Windows.

By.id()

in list [Locators](#)

Notifications

👁 Watch



Description

Edit



Attachments

LINK

Username textfield ↗

Added Feb 10 at 3:43 PM • [Remove](#) • [Edit](#)

Add an attachment

Suggested

👤 Join

Add to card

👤 Members

🏷 Labels

📋 Checklist

🕒 Dates

📎 Attachment

🖼 Cover

📁 Custom Fields

