

# Class Notes-13

## List of WebElements

`driver.findElements();`

we were trying to locate one element until now. We will learn how to locate multiple elements and interact with it.

1. Find common locators for the elements that you want to locate
2. pass your locator to findElements method
3. `List<WebElement> buttons = driver.findElements("yourLocator")`

after this point, buttons keep list of webelements, we have to reach them to apply selenium methods either with loop or `.get()` list method

`findElements()` method does not throw `NoSuchElementException`, if you locator is wrong it will return you empty list.

=====  
DROPDOWNS

Dropdown menu is a menu that offers a list of options. Most of the time we only allowed to choose one option.

select --> html tag

Select class --> class from selenium

SELECT class only works with the webelement which has select tag.

How to use SELECT class

1. make sure the dropdown has select tag.
2. locate the dropdown just like any other element
3. create object using Select class by passing the element that we locate step 2 as a constructor

## SELECT methods

`getOptions()` --> returns all the available options from the dropdown list.

`getFirstSelectedOption()` --> return webelement, first selected option

## HOW TO SELECT OPTION FROM DROPDOWN

### 1. Using Visible Text

`selectByVisibleText("text")` --> selecting based on the visible text of options

### 2. Using index Number

`selectByIndex(indexNumber)` --> select based on index of options (starts from 0)

### 3. using value attribute

`selectByValue(option value)` --> select based on value of value attribute

## NO SELECT DROPDOWN

When we try to use Select class with the dropdown which does NOT have select tag, we got UnexpectedTagNameException: Element should have been "select" but was "a"

If we don't have select tag, we locate dropdown just like any other web elements and to see options with clicking or hovering over to it.

To get all options from dropdown, we can use common locator between them(exp:classname, tagname etc)  
then we can use List methods to get the size and expected options from list.

we use click() method to click different options, we can either locate the option separately or use index of the option from the list.

=====

Popups, alerts, iframes, Windows

=====

Popups, alerts, modal, dialog, overlay --> anything pop ups on the webpage.

How to handle pop ups

we have 2 different ways of handling pop ups, based on their structure

1. HTML POP UPS

if the pop up is from html, we deal with it as if it is another web element.

2. JAVASCRIPT ALERTS

second types of pop up comes from javascript, mostly known as js alerts. we CANNOT right click on those alerts to see html(inspect)

To deal with non html alerts, we have special class from selenium which is Alert

how to point the alert ?

```
Alert alert = driver.switchTo().alert();
```

and we have accept, dismiss, sendKeys, getText method for alerts.

If there is no alert on the webpage, and we try to switch,  
NoAlertPresentException: no such alert