

Cric info website-

Get text method-

If we apply get text on **td** and the text is not present, then it will get it from the child.

If the parent has the text and child also has the text, then get it from both of them. The child can be at any layer under the parent.

Capture the person who was responsible for the batsman getting out-

espnricinfo.com/series/icc-champions-trophy-2024-25-1459031/australia-vs-india-1st-semi-final-1466426/full-scorecard

ESPN cricinfo Live Scores Series Teams News Features Videos Stats

Australia Innings India Innings Match Flow Info

Batsman	Bowler	Runs	Wickets	Over	Dot
Steven Smith (c)	b Mohammed Shami	73	96	133	4
Marnus Labuschagne		29	36	49	2
Josh Inglis *		11	12	13	0
Alex Carey	run out (Iyer)	61	57	84	8
Glenn Maxwell	b Patel	7	5	3	0
Ben Dwarshuis	c Iyer b Varun	19	29	30	1
Adam Zampa	b Pandya	7	12	22	0
Nathan Ellis	c Kohli b Mohammed Shami	10	7	8	0
Tanveer Sangha	not out	1	1	2	0

Elements Console Sources Recorder Network Performance Memory Application Security Lighthouse

HTML Structure:

```

<td class="ds-w-0 ds-whitespace-nowrap ds-min-w-max">
  <div class="ds-flex ds-items-center"> flex
    <div class="ds-popover-wrapper ds-inline">
      <a href="/cricketers/alex-carey-326434" title="Alex Carey" class="ds-inline-flex ds-items-center ds-text-tight-s ds-font-medium ds-text-type ds-underline ds-decoration-ui-primary ds-block ds-cursor-pointer" aria-expanded="false">
        <span>
          "Alex Carey"
          &nbsp;
        </span>
      </a>
    </div>
  </td>
  <td class="ds-min-w-max !ds-pl-[100px]"> == $0
    <span class="ds-flex ds-cursor-pointer ds-items-center"> flex
      <i class="icon-expand_more-outlined ds-text-icon-primary ds-inline-flex ds-items-center" style="font-size: 1em; margin-right: 5px;">
        <span>run out (Iyer)</span>
      </i>
    </span>
  </td>

```

Locator: `//span(text()='Alex Carey')/ancestor::td/following-sibling::td`

```

1 package seleniumsessions;
2
3 import org.openqa.selenium.WebDriver;
4 import org.openqa.selenium.chrome.ChromeDriver;
5
6 public class CricinfoWebTable {
7
8     static WebDriver driver;
9
10    public static void main(String[] args) throws InterruptedException {
11
12        driver = new ChromeDriver();
13        driver.get("https://www.espnricinfo.com/series/icc-champions-trophy-2024-25-1459031/australia-vs-india-1st-semi-final-1466426/full-scorecard");
14        Thread.sleep(4000);
15    }
16 }

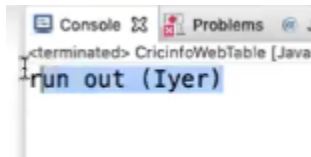
```

Locator for the person who made the batsman out-

```

10
17 String wktTakerName = driver.findElement(By.xpath("//span[text()='Alex Carey']/ancestor::td/following-sibling::td")).getText();
18 System.out.println(wktTakerName);

```



## Create utility-

```

20
21 public static void getWktTakerName(String playerName) {
22     String wktTakerName = driver.findElement(By.xpath("//span[text()=' + playerName + '']/ancestor::td/following-sibling::td")).getText();
23     System.out.println(wktTakerName);
24 }
25
26

```

## More clearer line 22-

```

47 public static String getWktTakerName(String playerName) {
48     String wktTakerName = driver
49         .findElement(By.xpath("//span[text()=' + playerName + '']/ancestor::td/following-sibling::td"))
50         .getText();
51     System.out.println(wktTakerName);
52     return wktTakerName;
53 }
54

```

## Calling utility-

From both the batting tables of australia and india the values are returned because the locators are same for both tables.

Alex and glen from first table, virat from second table.

```

1 package seleniumsessions;
2
3 import org.openqa.selenium.By;
4 import org.openqa.selenium.WebDriver;
5 import org.openqa.selenium.chrome.ChromeDriver;
6
7 public class CricinfoWebTable {
8
9     static WebDriver driver;
10
11     public static void main(String[] args) throws InterruptedException {
12
13         driver = new ChromeDriver();
14         driver.get("https://www.espncricinfo.com/series/icc-champions-trophy-2024-25-1459831/australia-vs-india-1st-semi-final-1466426/full-scorecard");
15         Thread.sleep(4000);
16
17         getWktTakerName("Alex Carey");
18         getWktTakerName("Glenn Maxwell");
19         getWktTakerName("Virat Kohli");
20
21     }

```

```
Console Problems Javadoc
<terminated> CricinfoWebTable [Java Applicat
run out (Iyer)
b Patel
c Dwarshuis b Zampa
```

Capture everything to the right of virat from his score-

```
23
24
25 List<WebElement> scoreList
26     = driver.findElements
27       (By.xpath("//span[text()='Virat Kohli']/ancestor::td/following-sibling::td[contains(@class,'ds-text-right')]"));
28
29 System.out.println(scoreList.size());
30
31 for(WebElement e : scoreList) {
32     String text = e.getText();
33     System.out.println(text);
34 }
35
36
37
```



```
6
84
98
135
5
0
85.71
```

nte

Added return statement-

```
38
39 public static String getWktTakerName(String playerName) {
40     String wktTakerName = driver
41         .findElement(By.xpath("//span[text()='\" + playerName + \"]/ancestor::td/following-sibling::td"))
42         .getText();
43     System.out.println(wktTakerName);
44     return wktTakerName;
45 }
46
47
```

Utility for multiple elements-

```

30
31 public static List<String> getScoreCard(String playerName) {
32     List<WebElement> scoreList = driver.findElements(By.xpath(
33         "//span[text()='"+playerName+"']/ancestor::td/following-sibling::td[contains(@class,'ds-text-right')]"));
34
35     System.out.println(scoreList.size());
36     List<String> scoreCardList = new ArrayList<String>(); //[]
37     for (WebElement e : scoreList) {
38         String text = e.getText();
39         System.out.println(text);
40         scoreCardList.add(text);
41     }
42     return scoreCardList;
43 }
44

```

Call multiple elements utility-

```

26
27     System.out.println(getScoreCard("Virat Kohli"));
28     System.out.println(getScoreCard("Axar Patel"));
29

```

Note-

It will pick values from both table if the player is batsman as well as bowler. Axar batted for india and took wickets also, so returning from both tables.

```

C u w a r s n u i s o z a m p a
6
84
nt 98
135
5
s/ 0
85.71
[84, 98, 135, 5, 0, 85.71]
16
8
1
43
1
i" 5.37
") 25
3
2
0
0
pl 27
me 30
ce 27
1
1
St 90.00
[8, 1, 43, 1, 5.37, 25, 3, 2, 0, 0, 27, 30, 27, 1, 1, 90.00]

```

Performance issues-

Naveen website.

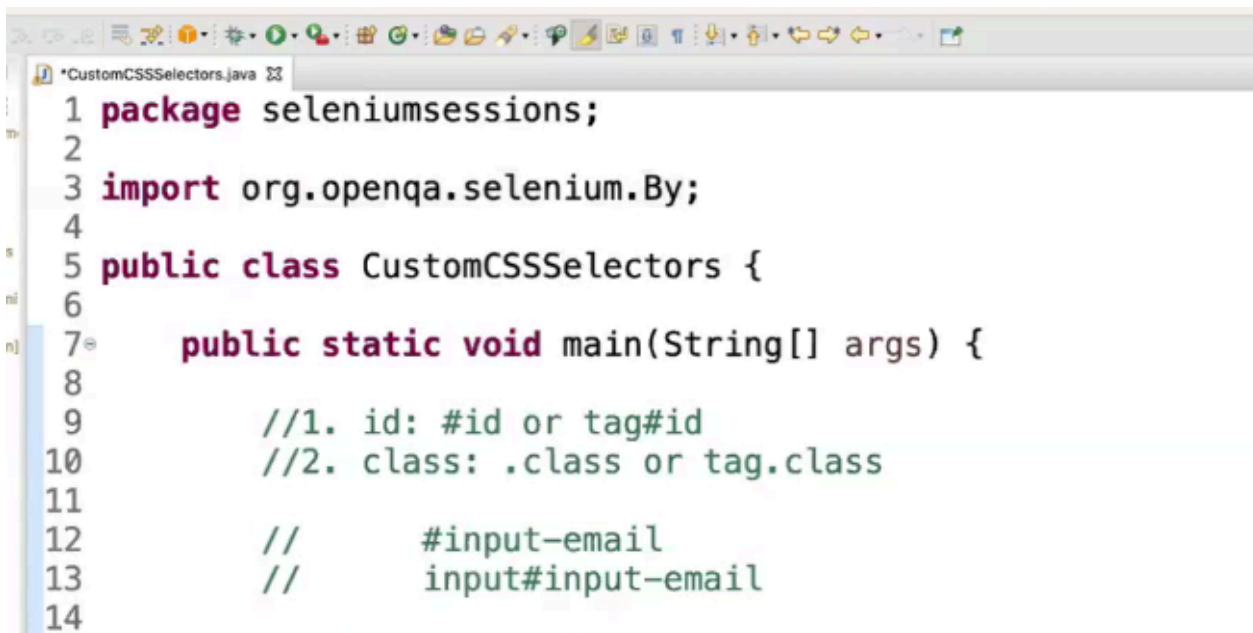
Email field.

Star tag name, has performance issues as all elements to be checked one by one.

Non star will only check for tags with name as input.

```
142
143
144      /*[@id="input-email"]: 200 --> 50th
145      //input[@id="input-email"] --> 20 -->
146
```

Css selector by id-



```
*CustomCSSSelectors.java
1 package seleniumsessions;
2
3 import org.openqa.selenium.By;
4
5 public class CustomCSSSelectors {
6
7     public static void main(String[] args) {
8
9         //1. id: #id or tag#id
10        //2. class: .class or tag.class
11
12        // #input-email
13        // input#input-email
14
```

By classname-

```
14
15 // .list-group-item
16 // a.list-group-item
17 // img.img-responsive
18 By image = By.cssSelector("img.img-responsive");
19
```

Combine two css selectors-  
Allowed.

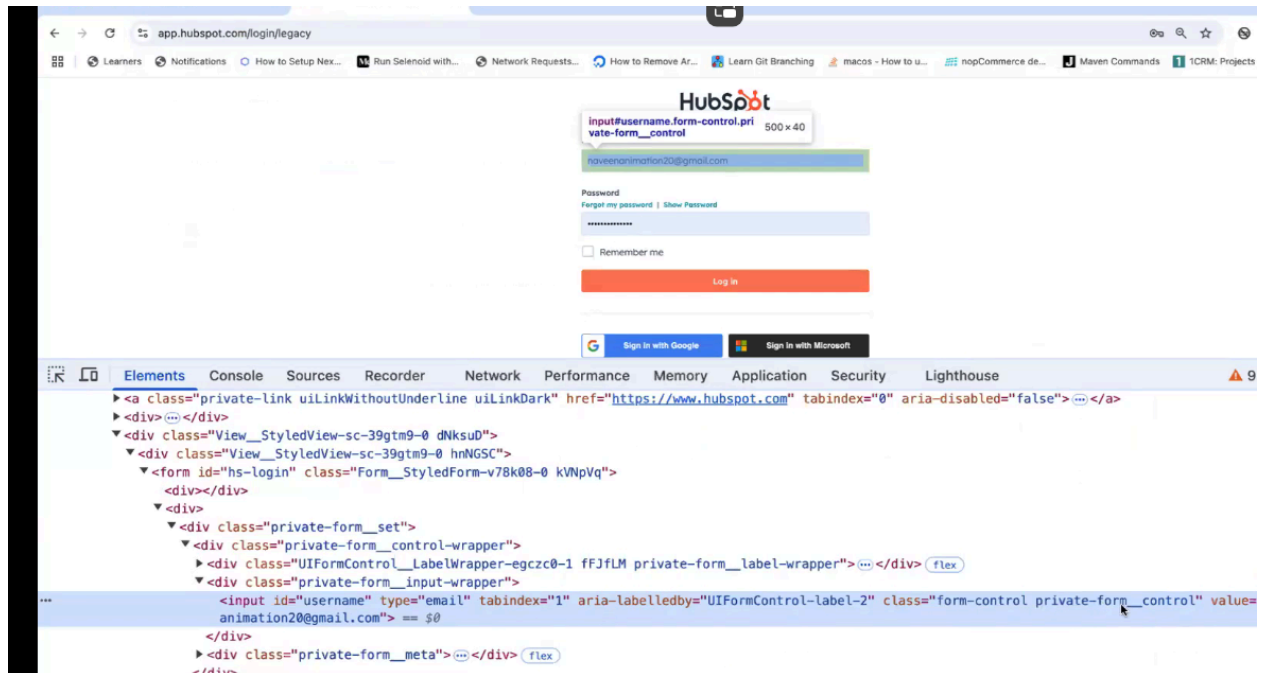
```
19
20 //3. tag#id.class
21 //input#input-email.form-control
22
23 //4. tag.class#id
24 // input.form-control#input-email
25
26 }
```

Multiple classes-

Hubspot.

Username inspect.

Multiple classes - known as cascaded classes.36.00



For email address-  
Not necessary to give all names if one class identifies element uniquely.

```

26 //5. .c1.c2.c3.c4...cn
27 // .form-control.private-form__control
28
29 //6. tag.c1.c2.c3.c4...cn
30 // input.form-control.private-form__control
31 //button.uiButton.private-button.private-button--primary.private-button--default.private-button--block.private-button--non-link

```

Line 31 for login button not clear so pasting it again-

```

37 //button.uiButton.private-button.private-button--primary.private-button--default.private-button--block.private-button--non-link

```

Combine hash and class-  
And we can write in any sequence.

```

32
33 //7. input.form-control.private-form__control#username
34 //8. input#username.form-control.private-form__control

```



Any attribute can be used this way-

There is no special case given to them like id or classname. 43.00

```
35
36 //other attributes:
37 // htmltag[attr='value']
38 // input[name='email'] -- css
39 //input[@name='email'] -- xpath
40
41 // input[id='input-email']
42 // input[placeholder='E-Mail Address']
43
44 //
45
```

Combine multiple attributes- 46.00

For one element if we plan to use multiple attributes any sequence can be followed.

Email address of naveen website-

We can put any number of attributes.

```
42 // input[placeholder='E-Mail Address']
43
44 // htmltag[attr1='value'][attr2='value']
45 // input[name='email'][placeholder='E-Mail Address']
46 // input[name='email'][placeholder='E-Mail Address'][id='input-email']
47 // input[name='email'][placeholder='E-Mail Address']#input-email
48 // input[name='email'][placeholder='E-Mail Address']#input-email.form-control
49 // input#input-email.form-control[name='email'][placeholder='E-Mail Address']
50
```

Contains-

For email.

Naveen website.

```
51 //contains:
52 // tag[attr *= 'value']
53 // input[placeholder *= 'Address']
54 // input[placeholder *= 'Address']#input-email
55 // input[placeholder *= 'Address']#input-email[name *= 'email']
```

Starts with-  
Email, naveen website.

```
56
57 //starts-with:
58 // tag[attr ^= 'value']
59 // input[placeholder ^= 'E-Mail']
60
```

Ends with-  
Present in css.  
Not xpath.  
Email, naveen website.

```
61 //ends-with:
62 // tag[attr $= 'value']
63 // input[placeholder $= 'Address']
64
```

Text -  
Not there in css.

```
64
65 //text in css: NA
```

Parent to child-  
Direct child.  
Use > **59.00**

Capture all links from the right hand side section.  
Naveen website.

```
66
67 //parent to child:
68 //direct child: >
69 //parent tag > child tag
70 // div.list-group > a
71
```

To get direct and indirect child-  
Naveen website, give all direct and indirect inputs in  
register page.

If no direct child then we get zero.

```
71
72 //direct+indirect: space
73 //parenttag childtag
74
75 // div#content input -- 11 (direct + indirect child)
76 // div#content > input -- 0 (direct child)
77
```

Child to parent -  
Not possible in css.

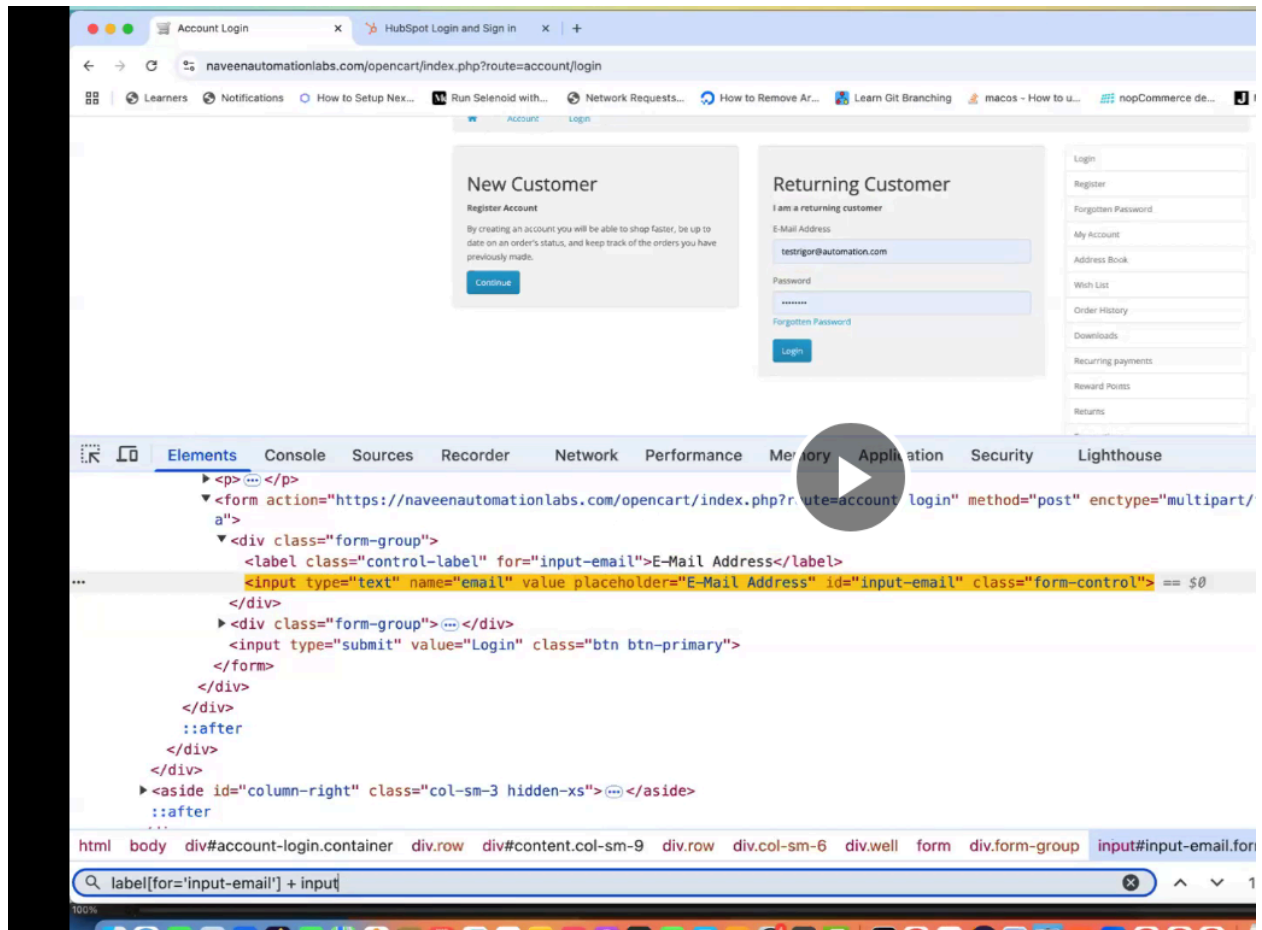
```
77
78 //child to parent: NA -- backward traversing is not supported with CSS
79
```

Siblings-

Only forward allowed. No concept of backward.

Naveen website.

Login.



```

79
80 //sibling:
81 //immediate sibling: +
82 // label[for='input-email'] + input
83

```

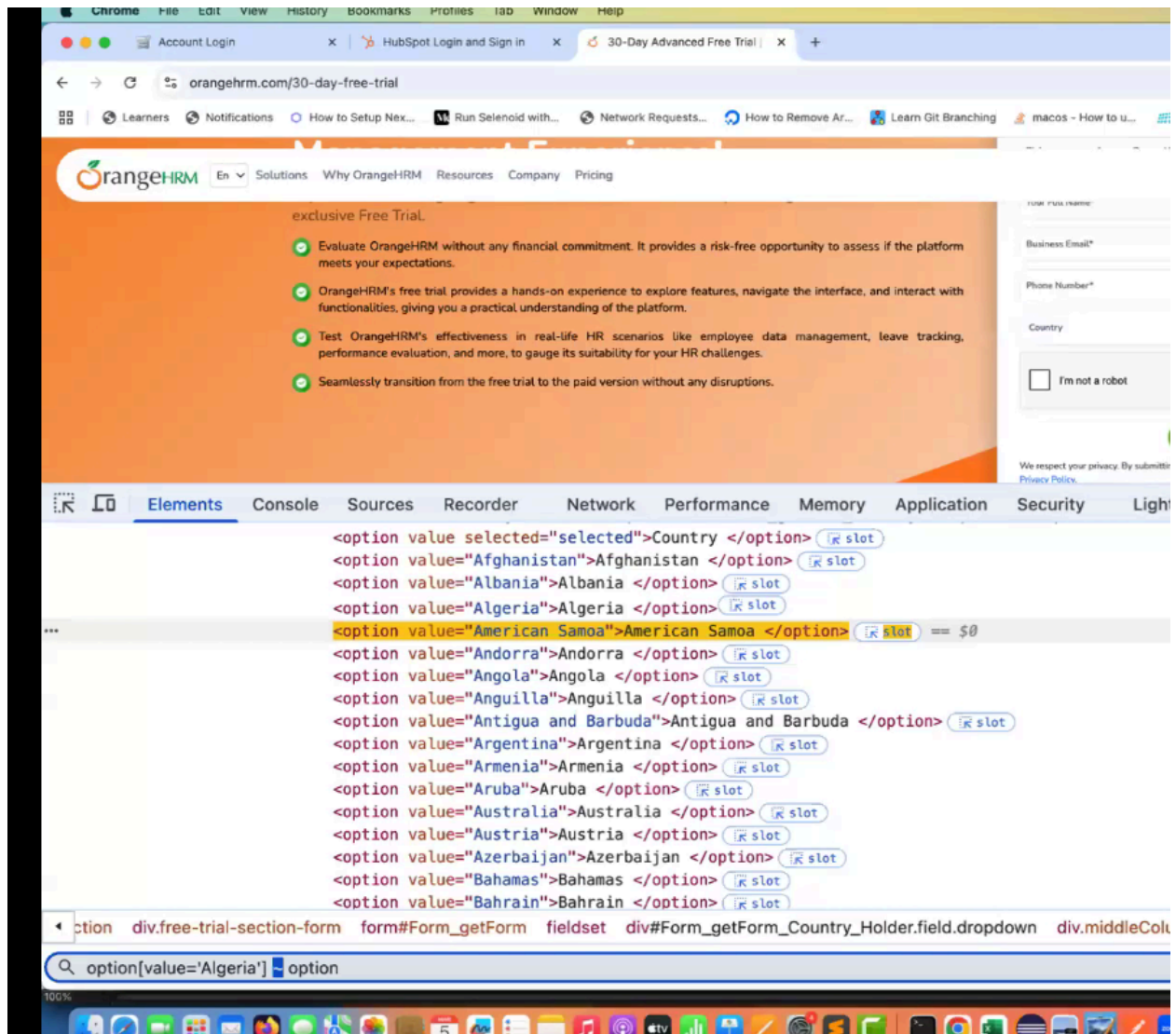
Gets only immediate following sibling.

```

79
80 //sibling:
81 //immediate sibling: +
82 // label[for='input-email'] + input
83 // option[value='Algeria'] + option

```

Get all following siblings-  
Orange hrm country dropdown.



```

84
85 //all following siblings:
86 // option[value='Algeria'] ~ option
87

```