**IntelliJ shortcuts**

|  |  |
| --- | --- |
| Alt + Enter | Generate code/solve issue |
| Ctrl + D | Duplicate line |
| Alt + Shift + cursor down/up | Move line up and down |
| Ctrl + Y | Delete complete line |
| Shift + F6 | Rename |
| Ctrl + Alt + L | Format code |
| Alt + drag mouse | Select text in block |
| Ctrl + / | Comment out/in row |

**CSS selectors**

|  |  |
| --- | --- |
| HTML | Selector |
| <div class="class\_name">  …  </div> | .class\_name  *or* [class=‘class\_name’] |
| <div class="class\_name button">  …  </div> | .class\_name.button |
| <div class="class\_name">  <div class="class\_child">  …  </div>  </div> | .class\_name .class\_child  space |
| <div class="class\_name">  <div class="class\_child">  <div class="child\_deep">  …  </div>  </div>  </div> | .class\_name .child\_deep  space |
| <div class="class\_name" id="id1">  …  </div> | #id1 *or* .class\_name#id1  *or* [id=‘id1’] |
| <div class="name" attr="specific">  …  </div> | [attr=‘specific’]  *works for any attribute, so attr can be name, for, data or anything else.* |
| <div class="name" attr="specific">  …  </div> | [attr\*=‘pecifi’]  *any attr that contains pecifi* |
| <div class="name" attr="specific">  …  </div> | [attr^=‘specif’]  *any attr that starts with specif* |
| <div class="name" attr="specific">  …  </div> | [attr$=‘cific’]  *any attr that ends with cific* |

Find elements:

browser.findElement(By.*cssSelector*("this is my selector, got from chrome"));

This returns a WebElement. On a web element you can perform most interactions:

browser.findElement(By.*cssSelector*("selectorString")).click();

or save first as WebElement;

WebElement myOwnGivenName = driver.findElement(By.*cssSelector*("selectorString"));

|  |  |
| --- | --- |
| myOwnGivenName.click() |  |
| myOwnGivenName.sendKeys("text to put in input field") |  |
| myOwnGivenName.clear() | To clear input field |
| myOwnGivenName.getText() | To get text from item |
| myOwnGivenName.getAttribute("value"); | Get text from input field (you need to use the term “value” here) |
| myOwnGivenName.isDisplayed() | Is true if WebElement is displayed |
| myOwnGivenName.findElement(By.*cssSelector*("selectorString")); | You can look into an existing element for a subelement. |

**import** org.openqa.selenium.Cookie;

Cookie myNiceCookieName = **new** Cookie("keyOfCookie", "valueOfCookie");  
browser.manage().addCookie(myNiceCookieName);

**Add cookie**

WebElement myIFrame = browser.findElement(By.*cssSelector*("selector of iframe"));

browser.switchTo().frame(myIFrame);

// and switch back  
browser.switchTo().defaultContent();

**Switch to iframe**

List<WebElement> myListOfElements = browser.findElements(By.*cssSelector*("selector that has more matching nodes"));

**for**(WebElement mySingleElement: myListOfElements){

**if**(mySingleElement.getText().equals("this text")){

mySingleElement.click();

**break**;

**List and for loop**

}

}