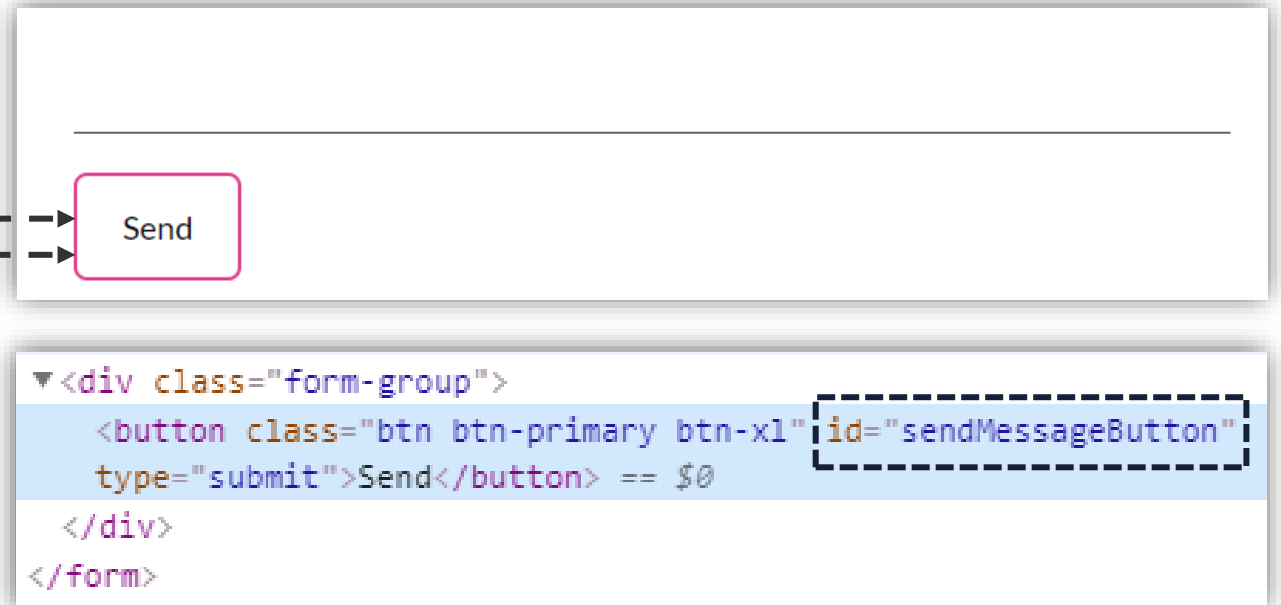# CSS vs Xpath

CSS vs XPATH

- **CSS** & **XPATH** are two methods which can be used to create selectors in order to locate web element(s) on a given webpage (DOM) in turn enabling us interact with them and perform specific actions.

CSS Selector:

button[id='sendMessageButton']

XPATH Selector:

//button[@id='sendMessageButton']

Send

```
▼<div class="form-group">
    <button class="btn btn-primary btn-xl" id="sendMessageButton"
    type="submit">Send</button> == $0
  </div>
</form>
```

CSS vs XPath - *Some* Differences - Part 1/3

- **CSS Selectors** only move forwards.
- **XPath Selectors** have the ability to traverse both forwards and backwards.

- It is noted that CSS Selectors perform better than XPath Selectors; refer to: http://elementalselenium.com/tips/34-xpath-vs-css-revisited-2 .

- XPath Selectors have some advantages and disadvantages over CSS Selectors.

XPATH *SOME* DISADVANTAGES:

- XPath engines are different in each browser, making them inconsistent; for example an XPath's index number may vary between Chrome, Firefox and IE browsers etc. In which case a given XPath Selector may work for Chrome browser but not work for Firefox browser etc.

- XPath Selectors can become complex; in turn making them hard to read; for example:

/html//section[@id='playground']//a[@role='button']/parent::div

XPATH *SOME* ADVANTAGES:

- XPath Selectors have the ability to search for Parent, Sibling elements etc.

- With XPath Selectors you can perform unique actions, for example you can create an XPath Selector which locates an element based upon text:

//*[text()='Send']