XPATH Cheat Sheet

Every element does not have an id -> static id, unique name, unique link text. For those elements we need to build xpath to find and then perform actions on them.

Whatever we use to find an element, id, name, xpath -> It should always be unique.
It should only find one matching node unless we want to capture a list of elements.

Difference between single '/' or double '//'

Single slash '/' anywhere in xpath signifies to look for the element immediately inside the parent element.
Double slash '//' signifies to look for any child or nested-child element inside the parent element.

Syntax:

//tag[@attribute='value']

Relative xpath using single '/' for Login link
//div[@id='navbar']/div/div/div/ul/li[2]/a

Relative xpath using double '//' for Login link.
//div[@id='navbar']//ul/li[2]/a

Don't use "*", always use the tag name.

Using Text of the element to build xpath

Finding Login link:
//div[@class='homepage-hero']//a[text()='Enroll now']

Using Contains to find the elements:
Syntax: //tag[contains(attribute, 'value')]

Finding Login link:
//div[@id='navbar']//a[contains(text(),'Login')]
//div[@id='navbar']//a[contains(@class,'navbar-link') and contains(@href,'sign_in')]

Using Starts-With to find the elements:
Syntax: //tag[starts-with(attribute, 'value')]

Finding Login link:
//div[@id='navbar']//a[starts-with(@class,'navbar-link')]

Parent
Syntax: xpath-to-some-element//parent::<tag>

Preceding Sibling
Syntax: xpath-to-some-element//preceding-sibling::<tag>

Following Sibling
Syntax: xpath-to-some-element//following-sibling::<tag>

Exercise:
http://letskodeit.teachable.com/pages/practice
Find the price of the course "Python Programming Language"
Solution:
//table[@id='product']//td[text()='Python Programming Language']//following-sibling::td

http://dhtmlx.com/docs/products/dhtmlxGrid/
Find Author of the book "The Green Mile"
Solution:
//div[@id='gridbox']//a[text()='The Green Mile']//parent::td//following-sibling::td[1]