

XPath contains the path of the element situated at the web page. Standard XPath syntax for creating XPath starts with //

There are two types of XPath:

1) Absolute XPath

2) Relative XPath

Абсолютный - It is the direct way to find the element, but the disadvantage of the absolute XPath is that if there are any changes made in the path of the element then that XPath gets failed. это путь от корня документа. Первым символом в нём должен стоять “/”.

/html/body/div/ul/li[1]

-----------------------------------------------------------------------------------------------------------

Relative XPath - Относительный - starts from the middle of HTML DOM structure. **It starts with double forward slash** (//). It can search elements anywhere on the web page, means no need to write a long xpath and you can start from the middle of the HTML DOM structure. Relative Xpath is always preferred as it is not a complete path from the root element. это путь от одного элемента (не обязательно от корневого) к другому, но этот элемент как мне кажется должен либо быть проиндексирован [...], либо быть уникальным

**//main/div[1]/div[1]/div[1]/div[1]/div[1]/div[1]/div[1]/a[1]/img[1]**-Здесь main типа уникальный поэтому и работает.

**//main//a[1]/img[1]**-тоже сработало,символ '//'озночает что учтены все родительские элементы данного тега

это называется типа рекурсивными спусками, но обратно-же первый тег(здесь-main)должен быть уникальным.

**//main//a**-тоже работает т.к. а только одна в main

Строка XPath — это фактически путь к элементу в дереве, где каждый уровень разделяется 1косой чертой «/».

//main/div[1]/div[1]/div[1]/div[1]/div[1]/div[1]/div[1]/a[1]/img[1]

**поиск дочернего элемента**

**//название\_элемента[@его\_тег=’значение’]/название\_дочернего\_элемента/…/искомый\_элемент** ←идёш от более явного элемента к вложенному через слэш

-----------------------------------------------------------------------------------------------------------

Поскольку результатом выполнения запроса XPath бывает множество элементов, то результат называется

коллекцией.//div данный XPath вернет 8 элементов (<div>). Для того, чтобы можно было работать с любым

элементом из коллекции нужно поместить в квадратные скобки номер элемента, с которым мы хотим работать.

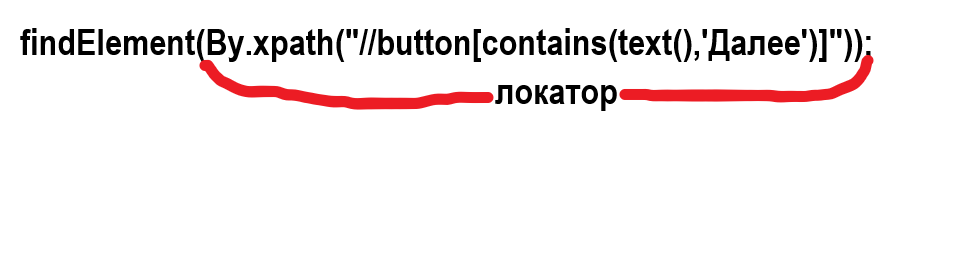
//div[2] данный XPath вернет 2-й элемент (<div>) из коллекции элементов.Данные, заключенные в квадратные

скобки называются предикатами.

Предикаты — это фильтры. После названия узла в квадратных скобках можно указать условие фильтрации.

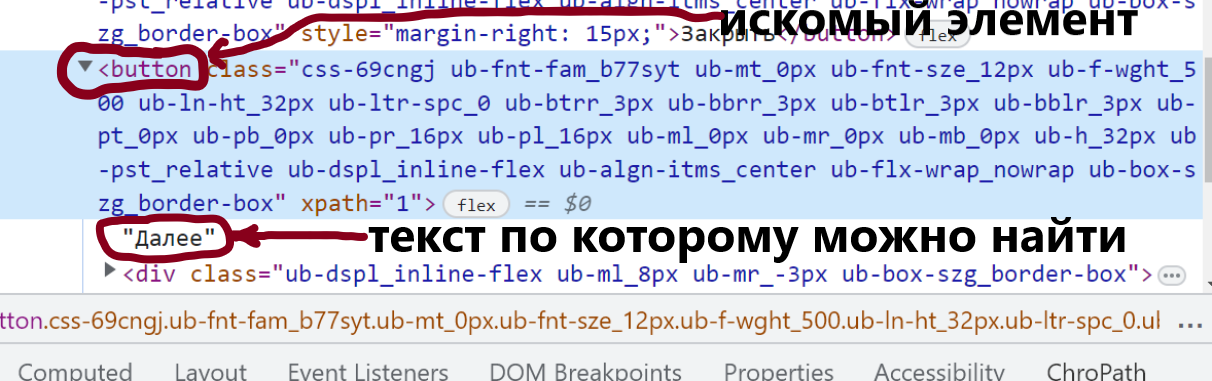
Попадают те узлы, которые соответствуют условию фильтрации

Пример://body/div[@id='\_\_next']/div[2]/div[3]

—------------------------------------------------- **Нахождение элемента по тексту** --------------------------------------------------------

Когда в элементе не за что зацепится можно указать текст, который возможно там есть

локатор - отвечает на вопросы: как искать? что искать?



Итак оказывается есть ещо один способ нахождения по тексту с совсем другим синтаксисом:

**WebElement element= driver.findElement(By.xpath("//button[contains(text(),'Далее')]"));** <-- способ №1

**WebElement element=driver.findElement(By.xpath("(//button[text()='Далее'])"));**  ←способ № 2

--------------------------------------------------------------------------------------------------------------------------------------------

\* -используется для ссылки на все элементы, которые являются дочерними для узла контекста.

Например: //\*/p ищет элемент (<p>) во всех элементах которые есть на странице.

В примере выше //div/p мы искали p только в элементе (<div>).

—--------------------—----------------------------------**поиск по цепочке**—---------------------------------------------

//nazvavie\_tega[@atribyt=’znachenie’]//nazvavie\_tega[@atribyt=’znachenie’]

первый элемент указал и к примеру последний, это благодаря **//**

------------------------------------------------------------**Поиск по атрибуту**-------------------------------------------------

**@ - используется для ссылки на любые атрибуты.**

Например: //div[@class='mainWrap'] ищет все элементы (<div>), в которых есть атрибут class

со значением mainWrap. А в css такого атрибута не требуется, там просто div[class='mainWrap']

@\* -используется для ссылки на все атрибуты узла контекста.

Например: //div[@\*] ищет все элементы div в которых есть любые аттрибуты (id, name, style...)

-------------------------------------**Поиск элемента с несколькими атрибутами**—-----------------------------------

**//a[@class=’awdsas’][@id=’dasd x’]**

—---------------------------------------------------------------**contains**—--------------------------------------------------------------

/**/nazvanie\_tega[contains(@nazvanie\_atribyta, ’znachenie’)][contains(@nazvanie\_atribyta, ‘znachenie’)]** ←ищет тег который содержит в себе указанный атрибут с значением (может указыватся часть значения - уникальная)

------------------------------------------------------- **Поиск нескольких элементов** ----------------------------------------------

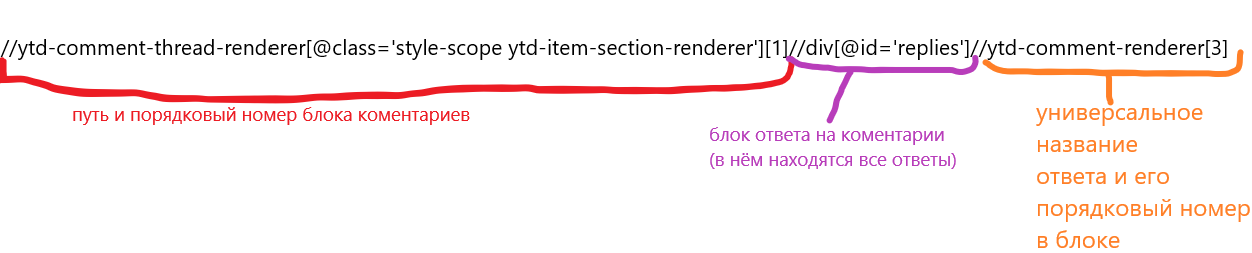
Ловил ошибку при нахождении, так как если ищем вложенный элемент/элементы, то записываем путь до этого данного элемента, но самый верхний элемент **div,** находится по его атрибуту **class**, именно значение этого атрибута берётся в квадратные скобки, а далее просто через слэш.

List <WebElement> elements=driver.findElements(By.xpath("//**div[@class='data-container']/ul/li**"));

—--------------**выбрать один элемент из нескольких присутствующих на стронице**--------------------------------

//h[3] ← выведется 3-ий элемент попадающий под это описание

—----------------------------------------**После написания botFindera**--------------------------------------------------------



На рисунке выше показан путь построения до уникального элемента, очень важно отметить что пользуясь инструментом разработчика (f12) при нахождении элемента нужно иметь ввиду, что если ты кликнул по элементу и получил его тэг с атрибутами и значениями, то это не факт что данный элемент один на странице и возможно проставив порядковый номер в конце xPath-а ничего не даст. В этом случае нужно вручную возвращатся к более старшему элементу который также будет иметь несколько копий, что надо учитывать пользуясь порядковым номером. **В целом помни, что при ненахождении нужного элемента (например их много) надо откатить к более старшим и крупным и возможно на их уровне и выстроить порядковый номер, что приведёт тебя к желаемому элементу дочернему**

—-------------------------------------------------------------------------------------------------------------------------------------------------------------

Ссылка на родительский узел (..) используется для ссылки на родительский узел узла контекста.

//div/ul[@id="longNumber"]/li[1] данный XPath дойдет до первого li[1]

//div/ul[@id="longNumber"]/li[1]/.. вернется на один уровень выше. Этот XPath равен XPath //div/ul[@id="longNumber"]

//div/ul[@id="longNumber"]/li[2] зайдет во второй li[2]

-----------------------------------------------------------------------------------------------------------

Группировка операций '()'

Группирует операции для явного задания порядка их выполнения.

Например: //div[4] вернет все (<div>) которые находятся на 4-ой позиции

(//div)[4] ввернет 4-ый элемент из коллекции элементов, т.е. выражение (//div) создаст коллекцию из всех элементов (<div>).

-----------------------------------------------------------------------------------------------------------

ancestor:: — возвращает множество предков.

//ul[@id="number"]/li/ancestor::\* вернет всех предков тега (<li>)

-----------------------------------------------------------------------------------------------------------

ancestor-or-self:: — возвращает множество предков и текущий элемент.

//ul[@id="number"]/li/ancestor-or-self::\* вернет всех предков тега (<li>) и сам тег (<li>)

-----------------------------------------------------------------------------------------------------------

attribute:: можно заменить на '@' — возвращает множество атрибутов текущего элемента,

//attribute::class вернет все элементы в которых содержится атрибут class

----------------------------------------------------------------------------------------------------------

Use Multiple attributes to locate an element.

Xpath- //button[starts-with(@id, 'save') and contains(@class,'publish')]

—---------------------------------------------------------------------------------------------------------------------------------------------------------------