XPath Secrets Every Tester Misses

<xpath>

XML Path Language

1. Absolute XPath

- Full path from the root (html) to the element.

```
/html/body/div[1]/form/input[1]
```

2. Relative XPath

- Starts from anywhere in the DOM (not root).

```
//input[@id="username"]
```

3. Attribute-based XPath

- Uses attributes to locate an element.

```
//button[@name="login"]
```

4. Contains() XPath

- Matches part of an attribute value.

```
//div[contains(@class,"header")]
```

5. Text() XPath

- Locates element using visible text.

```
//button[text()="Submit"]
```

6. Starts-with() XPath

- Matches attribute values that start with specific text.

```
//input[starts-with(@id, "user")]
```


7. Index-based XPath

- When multiple elements match, pick by index.

```
(//input[@type="text"])[1]
```


8. Wildcard (*) XPath

- Selects elements without specifying tag name.

```
//*[@id="loginBtn"]
```


9. OR & AND Conditions

- Combine multiple conditions for precision.

```
//input[@type="text" and
@name="username"]
```

10. XPath with last()

- Selects the last matching element.

```
(//div[@class="item"])[last()]
```

11. XPath with position()

- Selects elements at a specific position.

```
(//tr/td)[position()=3]
```

12. XPath with not()

- Excludes elements with certain conditions.

```
//input[not(@type="hidden")]
```

13. XPath with normalize-space()

- Ignores leading/trailing spaces in text.

```
//button[normalize
space(text())="Login"]
```

14. XPath with ancestor / parent

- Navigate upward in the DOM.

```
//span[@id="child"]/ancestor::div
```

15. XPath with following / preceding

- Navigate sibling relationships.

```
//label[text()="Email"]/following
::input[1]
```

16. XPath with following-sibling / preceding-sibling

- Find elements at the same level in DOM.

```
//h2[text()="Section"]/following-
sibling::p
```

17. XPath with self

- Selects the current node itself.

```
//div[@class="container"]/self::d
iv
```

18. XPath with descendant

- Selects all children & nested elements.

```
//div[@id="main"]/descendant::a
```

19. XPath with child

- Selects direct child nodes.

```
//ul[@id="menu"]/child::li
```

20. XPath with starts-with() + contains() Combo

- For dynamic attributes that change partially.

```
//input[starts-with(@id, "user")
and contains(@name, "field")]
```

21. Matches with Regex (XPath 2.0)

- For flexible regex-based healing.

```
//input[matches(@id,'user.*')]
```