JavaScriptExecutor in Selenium (Java)

The **JavascriptExecutor** interface in Selenium is used to execute **JavaScript directly in the browser**, which is helpful when:

- Selenium WebDriver can't interact with elements normally (e.g., due to dynamic content, hidden elements).
- You want to do things faster or more precisely (like scrolling, clicking, setting values, etc.).

✅ How to Use It

```
JavascriptExecutor js = (JavascriptExecutor) driver;
```

🚀 Commonly Used JavascriptExecutor Methods (with Examples)

Action JavaScript Code Use Case

◆ 1. Click an Element

```
js.executeScript("arguments[0].click();", element);
```

Used when element.click() fails (e.g., due to overlay).

◆ 2. Scroll Down by Pixels

```
js.executeScript("window.scrollBy(0,500);");
```

Scrolls down 500 pixels vertically.

◆ 3. Scroll to an Element

```
js.executeScript("arguments[0].scrollIntoView(true);", element);
```

Brings the element into view.

◆ 4. Send Text to Input Field

```
js.executeScript("arguments[0].value='Ajay';", inputElement);
```

Sets value when sendKeys() fails.

◆ 5. Get Page Title

```
String title = (String) js.executeScript("return document.title;");
```

◆ 6. Get Current URL

```
String url = (String) js.executeScript("return document.URL;");
```

◆ 7. Get InnerText of Page

```
String text = (String) js.executeScript("return
document.documentElement.innerText;");
```

📜 3. Scroll to Bottom of Page Using JavaScript

```
JavascriptExecutor js = (JavascriptExecutor) driver;
js.executeScript("window.scrollTo(0, document.body.scrollHeight);");
```

📝 Scrolls the web page to the bottom using JavaScript.

◆ 8. Generate Alert

```
js.executeScript("alert('This is a test alert');");
```

◆ 9. Refresh the Page

```
js.executeScript("history.go(0);");
```

◆ 10. Set Element Attribute

```
js.executeScript("arguments[0].setAttribute('value', '12345')", element);
```

📌 Why Use JavascriptExecutor?

- **When normal WebDriver commands fail**
- **To interact with hidden/disabled elements**
- **To improve speed in some cases**
- **For advanced DOM manipulation**