# **Handling Alerts & Popups in Selenium**

## **⚠️ Types of Browser Popups**

| Popup Type | Description | Trigger Method |
| --- | --- | --- |
| JavaScript Alert | Simple OK dialog | alert("message") |
| Confirmation Box | OK/Cancel dialog | confirm("message") |
| Prompt | Input dialog with text field | prompt("message") |
| Browser Window Popup | Authentication/Download dialogs | System-level |

## **🔧 Alert Interface Methods**

Alert alert = driver.switchTo().alert();

| Method | Description | Example | Works With |
| --- | --- | --- | --- |
| accept() | Clicks OK/Confirm | alert.accept() | ✅ Alert,  ✅ Confirm,  ✅ Prompt |
| dismiss() | Clicks Cancel | alert.dismiss() | ❌ Alert,  ✅ Confirm,  ✅ Prompt |
| getText() | Gets popup message | String msg = alert.getText() | All types |
| sendKeys() | Enters text (prompts) | alert.sendKeys("John") | ❌ Alert,  ❌ Confirm,  ✅ Prompt |

## **🚀 Handling Different Popup Types**

### **1. Simple Alert (OK Button Only)**

java

// Trigger alert

driver.findElement(By.id("alert-btn")).click();

// Switch to alert

Alert alert = driver.switchTo().alert();

// Get text and accept

System.out.println("Alert text: " + alert.getText());

alert.accept(); // Clicks OK

### **2. Confirmation Dialog (OK/Cancel)**

java

driver.findElement(By.id("confirm-btn")).click();

Alert confirm = driver.switchTo().alert();

// Decision logic

if(confirm.getText().contains("Delete")) {

confirm.dismiss(); // Cancel

} else {

confirm.accept(); // OK

}

### **3. Prompt Dialog (Text Input)**

java

driver.findElement(By.id("prompt-btn")).click();

Alert prompt = driver.switchTo().alert();

prompt.sendKeys("Selenium User");

prompt.accept(); // Submit

// Verify input

String result = driver. findElement(By.id("output")).getText();

assert result.equals("Hello Selenium User!");

## **🛡️ Best Practices for Robust Handling**

### **1. Always Use Explicit Waits**

java

public void acceptAlertIfPresent() {

try {

WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(5));

Alert alert = wait.until(ExpectedConditions.alertIsPresent());

alert.accept();

} catch (TimeoutException e) {

// No alert found

}

}

### **2. Handle Unexpected Alerts**

try {

Alert unexpectedAlert = driver.switchTo().alert();

unexpectedAlert.accept();

} catch (NoAlertPresentException e) {

// Continue execution

}

### **3. Authentication Popups (Non-JavaScript)**

// Embed credentials in URL

String authUrl = "https://username:password@example.com";

driver.get(authUrl);

## **💡 Advanced Scenarios**

### **Multiple Alert Handling**

// First alert

driver.findElement(By.id("alert1")).click();

driver.switchTo().alert().accept();

// Second alert (appears after first is closed)

driver.switchTo().alert().accept();

### **1. Authenticating Basic Auth Popups**

java

// Handle browser-level auth (not JavaScript alert)

String url = "https://username:password@example.com";

driver.get(url);

### **2. Handling Unexpected Alerts**

java

try {

Alert unexpectedAlert = driver.switchTo().alert();

unexpectedAlert.accept();

} catch (NoAlertPresentException e) {

// No alert found

}

### **3. File Upload Popups (Non-Alert)**

java

// Not an alert - use sendKeys directly

driver.findElement(By.id("file-input"))

.sendKeys("/path/to/file.txt");

## 🛠️ Best Practices

1. Always switch to alert before interacting
2. Add waits for alert visibility:

java

1. new WebDriverWait(driver, Duration.ofSeconds(5))

.until(ExpectedConditions.alertIsPresent());

1. Never mix alert handling with frame/window switching
2. For modern apps, prefer WebDriver's Alert over JavaScript workarounds

## ❌ Common Mistakes

java

// WRONG: Forgetting to switch context

driver.findElement(By.id("ok-btn")).click(); // Fails!

// CORRECT:

Alert alert = driver.switchTo().alert();

alert.accept();