



Java Testing with Selenium

A Comprehensive Syntax Guide for
Automation

—

Sujay Raghavendra

Apress®

Java Testing with Selenium

**A Comprehensive Syntax
Guide for Automation**

Sujay Raghavendra

Apress®

Java Testing with Selenium: A Comprehensive Syntax Guide for Automation

Sujay Raghavendra
Dharwad, Karnataka, India

ISBN-13 (pbk): 979-8-8688-0290-4

ISBN-13 (electronic): 979-8-8688-0291-1

<https://doi.org/10.1007/979-8-8688-0291-1>

Copyright © 2024 by The Editor(s) (if applicable) and The Author(s),
under exclusive license to APress Media, LLC, part of Springer Nature

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Melissa Duffy
Development Editor: Laura Berendson
Coordinating Editor: Gryffin Winkler
Copyeditor: Kim Burton

Cover designed by eStudioCalamar

Cover image by Wirestock on Freepik (www.freepik.com)

Distributed to the book trade worldwide by Apress Media, LLC, 1 New York Plaza, New York, NY 10004, U.S.A. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail booktranslations@springernature.com; for reprint, paperback, or audio rights, please e-mail bookpermissions@springernature.com.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at <http://www.apress.com/bulk-sales>.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub (<https://github.com/Apress>). For more detailed information, please visit <https://www.apress.com/gp/services/source-code>.

If disposing of this product, please recycle the paper

Forever in my heart, Grandma

Table of Contents

- About the Author xix**
- About the Technical Reviewer xxi**
- Acknowledgments xxiii**
- Introduction xxv**

- Chapter 1: Introduction to Selenium: Java Automation..... 1**
 - Introduction..... 1
 - The Need for Testing Application 2
 - What Is Selenium? 3
 - Development and History of Selenium..... 3
 - The Genesis and Early Development (2004–2006)..... 4
 - Expanding Horizons (2006–2011)..... 5
 - Maturing and Expanding (2011–2018) 6
 - The Modern Era of Selenium (2018–Present)..... 6
 - Why Selenium? Unraveling the Strengths of Selenium in Web Testing 7
 - Selenium Architecture..... 10
 - Core Components 10
 - Automation Tool Comparison: Selenium and Alternatives..... 12
 - Java: Selenium’s Preferred Language..... 15
 - Summary..... 17

TABLE OF CONTENTS

Chapter 2: Selenium Essentials: Setup and Browser Commands.....19

- Setting up Java on Your Machine 20
 - Step 1: Download the Java Development Kit (JDK) 20
 - Step 2: Install the JDK 21
 - Step 3: Set Environment Variables 22
 - Step 4: Verify the Installation 23
 - Step 5: Update When Necessary 24
- Installing Eclipse IDE 24
 - Step 1: Download Eclipse 24
 - Step 2: Run the Eclipse Installer 25
 - Step 3: Install Eclipse 25
 - Step 4: Complete the Installation 25
 - Step 5: Launch Eclipse 26
 - Step 6: Configure Eclipse (Optional) 26
 - Step 7: Create a Java Project to Test 26
- Selenium Installation 27
 - Step 1: Download Selenium WebDriver 27
 - Step 2: Create a New Java Project in Eclipse 29
 - Step 3: Add Selenium JARs to the Project 29
 - Step 4: Verify Installation 29
 - Step 5: Run the Test Script 30
- Browser Commands 31
- Opening a Selenium WebDriver Using Java Code 31
- Opening a Web Page Online or Offline 34
 - Online Web Pages 34
 - Offline Web Pages 35

Understanding close and quit Commands	35
close Command	36
quit Command	36
Setting up Browser Size.....	37
Maximizing the Browser Window	37
Minimizing the Browser Window	37
Setting a Specific Browser Window Size.....	38
Using Full-screen Mode.....	38
Setting the Browser Position with Selenium WebDriver	39
Setting the Size Using Coordinates.....	39
Getting the Browser Position	41
Getting the Window Size	41
Navigating Through Web Pages with Selenium WebDriver	42
Navigating Back.....	42
Navigating Forward	43
Refreshing the Page	44
Summary.....	44
Chapter 3: Mouse and Keyboard Actions.....	47
Action Chains	47
Mouse Actions.....	48
Click.....	48
Double Click.....	49
Context Click.....	50
Click and Hold.....	51
Perform.....	52
Pause.....	52
Release.....	53
Reset	53

TABLE OF CONTENTS

Mouse Movements.....	54
Move to Element.....	54
Move by Offset.....	55
Offset from Element	56
Offset from Viewport.....	58
Offset from Current Pointer Location.....	59
Drag and Drop on Element	60
Drag and Drop by Offset	65
Keyboard Actions	67
Keys.....	67
Key Down.....	68
Key Up	68
Send Keys.....	69
Scroll	70
Scroll to Element	70
Scroll by a Given Amount.....	71
Scroll from an Element by a Given Amount	72
Scroll from an Element with an Offset.....	74
Scroll from an Offset of Origin by Given Amount	75
Summary.....	76
Chapter 4: Web Elements.....	77
What Are Web Element Locators?	77
Why Are Web Element Locators Important in Web Automation?	78
Understanding the DOM.....	79
HTML and DOM Basics	80
Attributes	80
Relationships in the DOM	82

Locators 84

 ID Locator 84

 Name Locator 85

 Link Text Locator 85

 Partial Link Text Locator 86

 Tag Name Locator 87

 Class Name Locator 88

 XPath Locators 89

 CSS Selectors 96

Locating Multiple Web Elements 110

 Table for Locators to Locate Multiple Elements 112

 Common Challenges for Locating Web Elements 114

 Best Practices to Overcome Challenges 115

Summary 117

Chapter 5: Navigations 119

 Hyperlinks 120

 Hyperlink by ID 120

 Hyperlink by Text 121

 Hyperlink by Partial Link Text 121

 Hyperlink by XPath 121

 nth Hyperlink 122

 Return All Hyperlinks 122

 Testing Hyperlinks 122

 Check for a Valid Hyperlink 123

 Check for Broken Images 125

 Data Attributes Hyperlinks 127

 Summary 127

TABLE OF CONTENTS

- Chapter 6: Buttons..... 129**
- Standard HTML Button..... 130
 - Asserting Button Type..... 131
 - Submit Button..... 131
 - Asserting Button Type..... 132
 - Image Button..... 133
 - Asserting Button Type..... 134
- JavaScript Button..... 134
- Disabled Button..... 135
 - Asserting Button Type..... 136
- Toggle Button..... 137
- Radio Buttons..... 138
 - Locating and Selecting with Radio Buttons..... 139
 - By ID..... 140
 - Using Label..... 140
 - By Index Values..... 140
 - Myth of Deselecting Radio Buttons..... 141
 - Validating Your Choices with Assertions..... 141
 - Verifying the Selection State..... 141
- Checkboxes..... 142
 - Locating and Selecting Checkboxes..... 143
 - By ID..... 143
 - Using Label..... 144
 - By Name..... 144
 - Selecting a Checkbox by Visible Text..... 144
 - Selecting a Checkbox by Value..... 145
 - Select All Checkboxes at Once..... 145
 - Selecting and Deselecting by Index..... 146

Deselecting a Checkbox by Visible Text..... 146

Deselecting a Checkbox by Value..... 147

Deselect All Checkboxes at Once 147

Validating a Checkbox with Assertions..... 148

Asserting Selection..... 148

Asserting Deselection..... 148

Asserting Element Type 149

SelectList..... 149

Locating and Interacting with SelectList 150

Locate and Select by Visible Text..... 151

Locate and Select by Value..... 151

Locate and Select by Index..... 151

Retrieving All Available Options 151

Get All Options 152

Deselecting Options in SelectList..... 152

Simulating Deselect by Selecting a Default Option 152

Validating SelectList Options and Selections 153

Assert the Selected Option 153

Assert Element Type 153

MultiSelectList..... 154

Selecting and Deselecting Options with MultiSelectList..... 155

 Selecting Multiple Options..... 156

 Deselecting Options..... 156

 Validating Multiple Selections and Deselections..... 156

 Asserting Multiple Selections:..... 157

 Asserting Deselections..... 157

TABLE OF CONTENTS

- Asserting Element Type for MultiSelectList 157
- Testing Considerations 158
- Summary..... 161
- Chapter 7: iframes and Textboxes 163**
- iframes 163
- Switching to an iframe..... 165
- Switch Using ID..... 166
- Switch Using Name..... 166
- Switch Using Index Value 166
- Hierarchy in Switching iframes..... 168
- Navigating an iframe in a Hierarchy..... 169
- Switch as an Element 171
- Frames with Waits 172
- Textboxes 173
- Single Line Textbox 174
- Multiline Textbox 175
 - Locating the Multiline..... 176
 - Inserting Values 176
 - Retrieving a Value from a Multiline Textbox..... 177
- Summary..... 177
- Chapter 8: Assertions 179**
- What Are Assertions? 180
- The Need for Assertions..... 180
 - Hard Assertions (Asserts) 181
 - Soft Assertions (Verify) 183
 - Hard vs. Soft..... 185

Assert Methods in Selenium..... 187

Handling Assertion Failures 191

 What Happens When an Assertion Fails? 191

 Handling Assertion Failures Gracefully..... 191

 Logging and Reporting Assertion Failures..... 192

Custom Assertions 192

Creating Custom Assertion Methods for Specific Application Requirements.... 193

 Example: Custom Assertion for Checking Data Validity 194

Extending Assertion Functionality..... 195

 Example: Extending Assertion Functionality in TestNG 195

Common Pitfalls and Mistakes in Selenium Assertions 197

 Common Mistakes When Using Assertions 197

 Avoiding False Positives and False Negatives..... 199

Best Practices for Using Assertions in Selenium 200

 When to Use Assertions..... 200

 Keeping Assertions Simple and Specific 201

 Using Meaningful Error Messages..... 201

 Organizing Assertions within Test Cases 202

Summary..... 202

Chapter 9: Exceptions.....205

 What Is an Exception in Selenium? 205

 Types of Exceptions 206

 Common Exceptions in Selenium..... 206

 Connection Exception 206

 Element Interaction Exceptions 207

TABLE OF CONTENTS

State-based Exceptions.....	208
Timeout and Delay Exceptions	209
Navigation Issues	210
Selector and Search Issues	210
JavaScript Execution Exceptions.....	211
Session Exceptions.....	212
Driver Configuration and Capability Exceptions	212
Input and Argument Exceptions.....	212
Alert and Pop-up Exceptions	213
Screenshot Exception.....	213
Movement and Action Exception	213
Browser Capability and Support Exception	214
Attribute and Property Exception.....	215
Cookie Handling Exception.....	215
Window Handling Exceptions	216
Element State Exceptions.....	216
Server and Response Exceptions	218
Other Exceptions	218
Handling Exceptions in Selenium.....	219
Why Exception Handling Is Essential in Selenium WebDriver.....	219
Handling Exceptions	220
Element Not Found Using try-catch.....	220
Timeout Exception Using try-catch-finally.....	221
Stale Element Exception Using try-catch-finally with throw	222
Handling Various Exceptions Using Multiple catch Blocks.....	223
Handling Custom Exceptions	224

Best Practices to Handle Exceptions.....	225
Summary.....	227
Chapter 10: Wait Strategies in Selenium Test Automation	229
Need for Waits.....	230
Dynamic Content Loading.....	230
Network Latency and Performance Variations	230
Synchronization.....	230
Reducing Flakiness	230
Uncertain User Input.....	231
Wait Types	231
Implicit Waits	231
Explicit Waits.....	233
Commonly Used ExpectedConditions in the Java Class.....	235
Fluent Waits	248
Key Features of a Fluent Wait	251
Selecting the Right Wait.....	251
Implicit Wait	251
Explicit Wait.....	252
Fluent Wait	253
Comparative Analysis of Implicit, Explicit, and Fluent Waits	255
Best Practices to Use Waits in Selenium Test Automation	256
Summary.....	259
Chapter 11: Page Object Model (POM).....	261
The Conventional Approach	262
What Is POM?	263

TABLE OF CONTENTS

Decoding DOM	263
Java Files in POM	266
Complete Analysis and Description of Creating a POM	266
Differences between Traditional and POM.....	268
POM Best Practices	269
Factory Page	272
Setting Up Page Factory	272
Summarizing the Setup	276
Steps to Implement Page Factory	277
When to Use Page Factory	277
Differences Between POM and Page Factory	278
Best Practices for Implementing Page Factory	279
Limitations of POM and Page Factory	280
Limitations of POM	280
Limitations of Page Factory	281
When and Which One to Use: POM vs. Page Factory	281
When to Use POM	282
When to Use Page Factory	282
Making the Decision	283
Summary.....	283
Chapter 12: TestNG	285
Understanding Frameworks in Depth	286
JUnit Overview	286
Exploring the Features of JUnit.....	286
Transitioning to TestNG: Elevating Beyond JUnit.....	287
TestNG Features	287

Comparative Analysis: JUnit vs. TestNG 288

TestNG in Selenium WebDriver: A Synergistic Combination 290

Setting Up TestNG: A Step-by-Step Guide 291

 Step 1: Installing TestNG in IDEs 291

 Step 2: Creating a New Java Project 292

 Step 3: Incorporating TestNG into the Project 292

 Step 4: Confirming Your TestNG Setup 294

TestNG Annotations and Attributes 294

@Test: The Core of TestNG 295

@Test with Other Annotations 298

 @BeforeSuite and @AfterSuite 298

 @BeforeTest/@AfterTest 299

 @BeforeClass/@AfterClass 299

 @BeforeMethod/@AfterMethod 300

TestNG Test Case with Selenium 301

 HTML Snippet Example 301

Writing TestNG Test Case 301

TestNG Assertions 304

Parameterized Testing in TestNG 307

 Implementing Parameterized Testing in TestNG 307

Advanced TestNG Configuration and Parallel Execution 309

 Utilizing testng.xml for Test Execution 310

 Enabling Parallel Execution in testng.xml 310

Implementing Parallel Execution: A Practical Example 311

Best Practices for Using TestNG with Selenium 312

Summary 315

Index 317

About the Author



Sujay Raghavendra is a distinguished expert in machine learning and software testing, with a strong background in Python programming. With a deep passion for both disciplines, Sujay Raghavendra has dedicated his career to exploring the intersection of these fields and delivering innovative solutions.

Recognizing the critical role of testing in software development, he expanded his expertise to include automated testing using Selenium with Python. With his books *Python Testing with Selenium: Learn to Implement Different Testing Techniques Using the Selenium WebDriver* and *Beginner's Guide to Streamlit with Python: Build Web-Based Data and Machine Learning Applications*, Sujay shares his extensive knowledge and practical insights into leveraging Selenium's capabilities with Python for efficient and reliable testing. The book provides a comprehensive guide for professionals looking to master automated testing techniques, harnessing the power of Python and Selenium to ensure the quality and functionality of web applications.

Sujay's expertise in machine learning and software testing allows him to bring a unique perspective to automated testing. He explores ways to integrate machine learning concepts into testing practices, enabling readers to discover innovative approaches to test analysis, anomaly detection, and intelligent test execution.

ABOUT THE AUTHOR

With a solid foundation in programming, Sujay leverages his expertise to provide readers with practical examples, best practices, and coding techniques specific to Python and Selenium. His clear explanations and step-by-step guidance make it easy for readers to implement effective testing strategies and optimize their automation workflows.

Beyond writing, Sujay actively contributes to the testing community through workshops, speaking engagements, and mentorship programs. He strives to empower professionals with the knowledge and skills needed to excel in automated testing, ensuring the delivery of high-quality software products. Through his work, Sujay continues bridging the gap between machine learning, software testing, and programming, inspiring others to embrace the synergy of these disciplines and drive advancements in automated testing.

About the Technical Reviewer



Dolkun Tursun Tarim is a senior quality assurance manager, principal software developer, and the founder of Selenium Master LLC. Tarim has more than 12 years of progressive experience managing quality assurance test automation teams and projects and developing robust and scalable UI, API, and database automation frameworks with Java/.NET/Python Selenium WebDriver, RestAssured, and database libraries. He has technical expertise in Cypress and Playwright automation framework implementations in the CI/CD Pipeline. He mastered technical expertise in software accessibility, contract, and performance testing. He has worked on various automated functional and performance testing projects utilizing commercial and open source test automation applications and framework APIs. In his free time, he teaches full-stack software development engineer in test (SDET) courses online and assists with testing automation projects.

Acknowledgments

I extend my heartfelt gratitude to my dear mother, Indumati Raghavendra, and elder brother, Sumedh, for their steadfast love, guidance, and support. Their unwavering belief in me has been a source of strength and inspiration throughout my journey. My elder brother's wisdom, mentorship, and encouragement have significantly shaped my path and aspirations. Together with my mother's endless love and nurturing care, they have been my pillars of strength, guiding me through life's ups and downs. I am truly blessed to have such remarkable individuals in my life, and I am forever grateful for their unwavering presence and support. Thank you, Mom and elder brother, for everything you have done for me.

Introduction

Welcome to *Java Testing with Selenium*, a comprehensive guide designed to help you master automated testing of web applications using Java and Selenium WebDriver. This book is a journey through the intricacies of Selenium WebDriver, exploring its capabilities for web testing and leveraging the power of Java to create robust and efficient automation scripts.

Who Is This Book For?

This book is ideal for software developers, quality assurance professionals, and anyone interested in learning about automated testing with Selenium WebDriver using Java. Whether you are a beginner looking to get started with automated testing or an experienced tester seeking to enhance your skills, this book offers valuable insights and practical techniques to excel in web application testing.

Structure of the Book

Java Testing with Selenium is structured into twelve chapters, each focusing on different aspects of automated testing with Selenium WebDriver and Java. Here's a brief overview of what you can expect from each chapter:

- Chapter 1 introduces Selenium, its various tools and versions, and the architecture of Selenium WebDriver. You learn the advantages of using Selenium for web

INTRODUCTION

application testing and learn how to integrate Selenium with Python for automation.

- Chapter 2 dives into the essentials of Selenium testing by learning how to install Java and Selenium, set up browser drivers, execute basic browser commands, and run a Python test case using Selenium.
- Chapter 3 explores the capabilities of Selenium WebDriver for performing mouse and keyboard actions. Learn about action chains, mouse actions like click and drag, keyboard actions including sending keys, and more.
- Chapter 4 explains the fundamental concepts of web elements and locators in Selenium WebDriver. Master various types of web locators and techniques for locating multiple web elements.
- Chapter 5 teaches techniques for testing hyperlinks on web pages, including locating hyperlinks by different attributes, checking for valid hyperlinks, and handling broken images within hyperlinks.
- Chapter 6 shows how to interact with different types of buttons in Selenium, including image, radio, checkboxes, select lists, and multiple select lists.
- Chapter 7 explores the concepts of frames and textboxes in Selenium WebDriver. Learn techniques for switching to iframes and interacting with single-line and multiline textboxes.
- Chapter 8 explains the importance of assertions in test automation and how to implement them effectively in Selenium using hard and soft assertions.

- Chapter 9 describes handling exceptions in Selenium WebDriver, including common exceptions and strategies for effective exception handling.
- Chapter 10 delves into the concept of waits in Selenium WebDriver, including implicit and explicit waits, commonly used ExpectedConditions, and fluent waits.
- Chapter 11 explores the page object model (POM) and Page Factory patterns in Selenium, their advantages, implementation, and differences.
- Chapter 12 introduces TestNG, a powerful testing framework for Java, and explains how to integrate it with Selenium for creating robust and scalable automation test suites.

Java Testing with Selenium equips you with the knowledge and skills necessary to excel in the automated testing of web applications using Java and Selenium WebDriver. Each chapter provides a comprehensive discussion of topics, practical examples, and hands-on exercises to reinforce learning. Whether you're a novice or an experienced tester, this book is your ultimate guide to mastering Selenium WebDriver testing with Java. Let's embark on this journey together and unlock the full potential of automated testing!

CHAPTER 1

Introduction to Selenium: Java Automation

Introduction

This exploration of the dynamic landscape of web application testing delves into the indispensable role of Selenium. This open source automation tool has fundamentally transformed the approach to ensuring software quality and reliability. The journey begins with acknowledging the critical need for application testing, a practice integral to identifying and addressing potential issues before they affect the end-user experience. A comprehensive overview introduces Selenium, highlighting its development, history, and the key reasons behind its widespread adoption in the web testing domain.

As you navigate through the intricacies of Selenium, the focus extends to its robust architecture—a foundation that enables the seamless integration and execution of complex testing scenarios. This architectural insight provides a backdrop for understanding how Selenium distinguishes itself from other tools in the market, offering unparalleled flexibility and compatibility across various platforms and browsers.

Central to the narrative is the affinity between Selenium and Java, the preferred programming language that amplifies Selenium’s capabilities. Java’s object-oriented nature, wide adoption, and platform independence make it an ideal companion for Selenium, enhancing your ability to develop sophisticated and scalable test scripts.

This introductory chapter sets the stage for a deeper dive into the functionalities, benefits, and strategic advantages of using Selenium in conjunction with Java. The aim is to illuminate the synergies between these powerful tools, showcasing why they remain at the forefront of web application testing practices.

The Need for Testing Application

The need for testing applications emerged as software development evolved and became an integral part of modern businesses. The recognition of this need can be traced back to the following key factors.

- **Quality assurance:** Testing helps identify defects and bugs in the application, allowing developers to fix them before the software is released to users. It improves the overall quality of the software.
- **Functionality verification:** Testing verifies that the application functions as intended, ensuring it meets user expectations and business requirements.
- **Regression testing:** As software evolves with new features and updates, testing ensures that existing functionality remains intact and unaffected by changes.

- **User satisfaction:** Quality assurance through testing leads to a better user experience, which is crucial for user satisfaction and retention.
- **Cost reduction:** Early detection and correction of defects reduce the cost of fixing issues after the software is in production.

What Is Selenium?

Selenium is a powerful and widely used open source framework for automating web browsers. It plays a crucial role in testing web applications by simulating user interactions with web elements. The core purpose of Selenium is to automate web application testing across various browsers and platforms, ensuring that web applications function correctly and efficiently.

Development and History of Selenium

This topic traces the remarkable development journey of Selenium, a tool that revolutionized the domain of web application testing. From its inception as a simple browser automation tool to its current status as a comprehensive test automation suite, Selenium's evolution mirrors the advancements and challenges in web technology over the past two decades.

The Genesis and Early Development (2004–2006)

The Birth of Selenium

The story of Selenium begins in 2004 with Jason Huggins, a software engineer at ThoughtWorks, who developed Selenium as an internal tool to address the need for automated testing of web applications. The initial release, Selenium Core, was a groundbreaking JavaScript-based testing system.

The name *Selenium* was chosen as a joke to a competitor named Mercury because selenium is a known antidote for mercury poisoning.

Selenium Remote Control (RC)

In 2005, Paul Hammant, another ThoughtWorks engineer, introduced Selenium RC to overcome the same-origin policy limitations inherent in Selenium Core. This development marked a significant step forward, allowing users to write test scripts in various programming languages.

Expanding Horizons (2006–2011)

Selenium IDE

In 2006, Shinya Kasatani of Japan contributed Selenium IDE, a Firefox extension, to the Selenium suite. It offered an easy-to-use interface for recording and playing back tests, making test automation more accessible to beginners.

The Introduction of WebDriver

In 2008, Simon Stewart developed WebDriver, a tool designed to address the limitations of Selenium RC. WebDriver's direct interaction with web browsers and its cohesive API marked a substantial improvement over its predecessors.

Selenium 2.0: A Major Milestone

The release of Selenium 2.0 in 2011 was a landmark event in Selenium's history. This version unified Selenium RC and WebDriver, offering a robust and streamlined framework for web application testing.

Maturing and Expanding (2011–2018)

The Advent of Selenium 3.0

In 2016, Selenium 3.0 represented a major leap forward, deprecating the original Selenium Core and replacing it with WebDriver. This version focused on modern web standards and enhanced browser support.

Growth of the Selenium Ecosystem

During this period, the Selenium community saw substantial growth. The tool's integration with other testing frameworks and CI systems underscored its adaptability and wide-ranging applicability.

The Modern Era of Selenium (2018–Present)

Selenium 4.0: The Future Realized

Announced in 2018 and released in 2021, Selenium 4.0 brought many new features and improvements. Adopting the W3C WebDriver standard, enhancements to Selenium Grid, new functionalities like relative locators, and improved window management exemplified the ongoing innovation in Selenium.