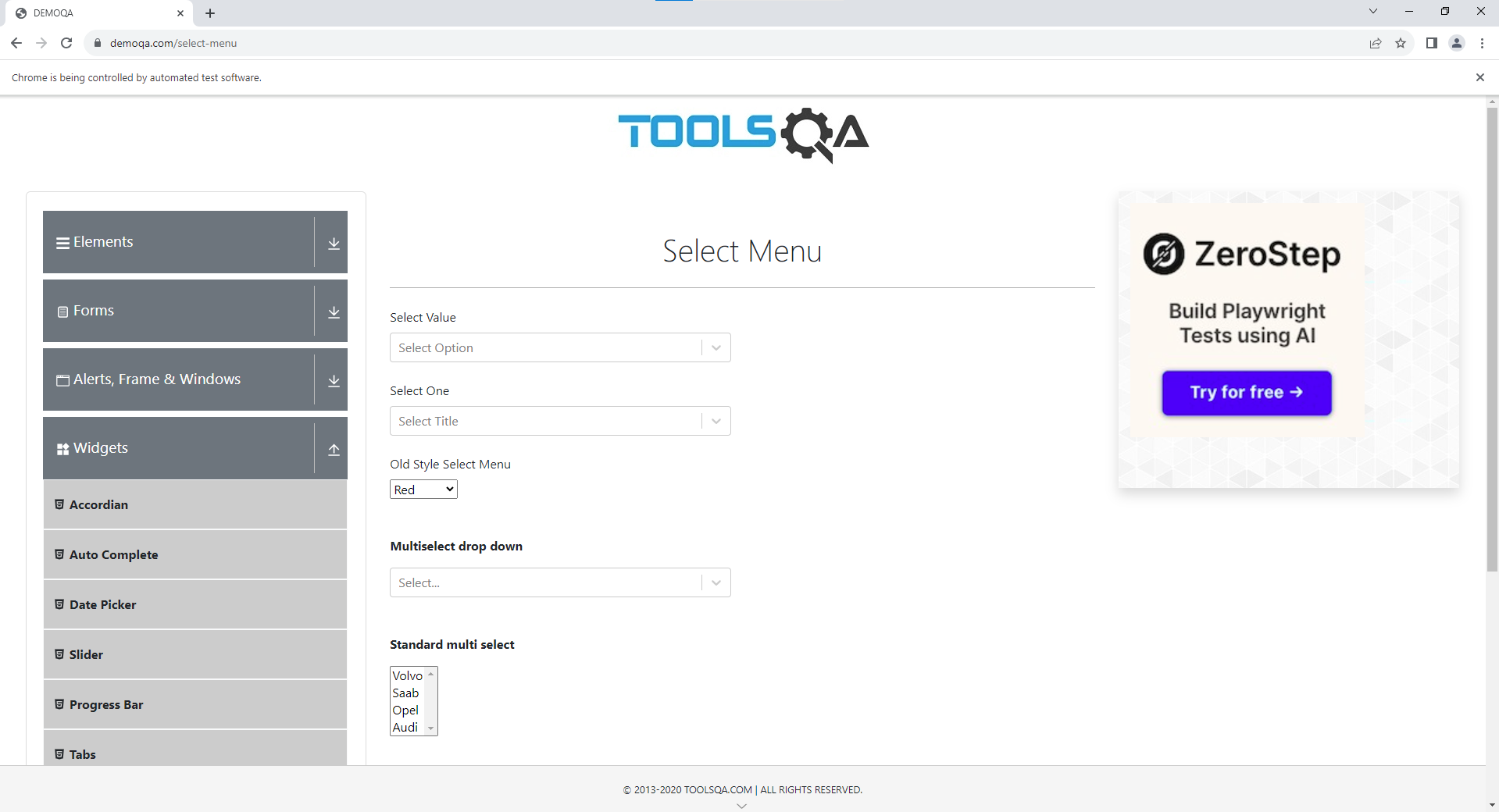
**: Demonstrate Handling Drop Down, List Boxes.**

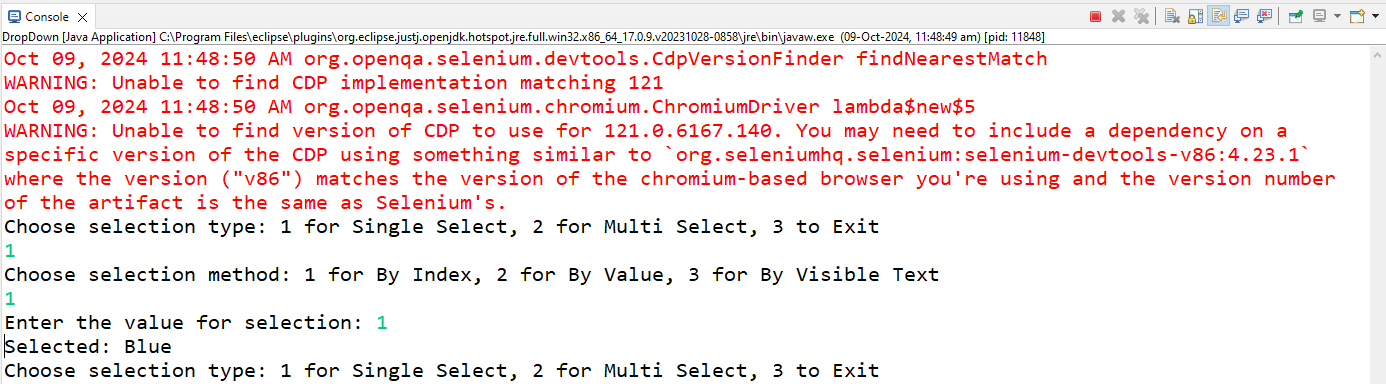
**Code:**

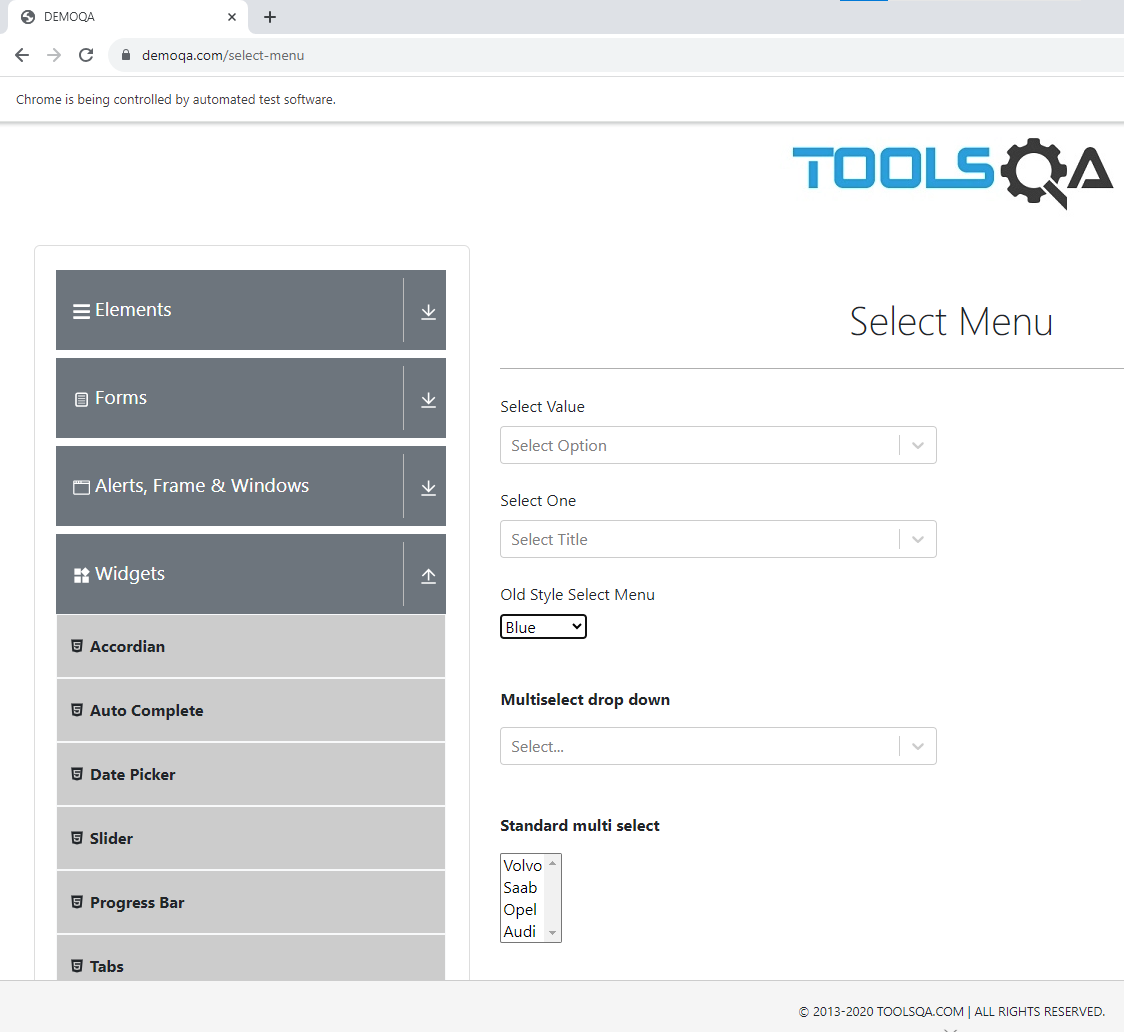
|  |
| --- |
| package demowork;  import java.util.Scanner;  import org.openqa.selenium.By;  import org.openqa.selenium.WebDriver;  import org.openqa.selenium.WebElement;  import org.openqa.selenium.chrome.ChromeDriver;  import org.openqa.selenium.support.ui.Select;  public class DropDown {  public static void main(String[] args) throws InterruptedException {  WebDriver driver = new ChromeDriver();  driver.get("https://demoqa.com/select-menu");  driver.manage().window().maximize();  Scanner scanner = new Scanner(System.*in*);  boolean exitProgram = false;  while (!exitProgram) {  System.*out*.println("Choose selection type: 1 for Single Select, 2 for Multi Select, 3 to Exit");  int selectionType = scanner.nextInt();  scanner.nextLine(); // Consume newline left over  switch (selectionType) {  case 1:  WebElement singleSelectElement = driver.findElement(By.*id*("oldSelectMenu"));  Select selObj1 = new Select(singleSelectElement);  System.*out*.println("Choose selection method: 1 for By Index, 2 for By Value, 3 for By Visible Text");  int singleSelectMethod = scanner.nextInt();  System.*out*.print("Enter the value for selection: ");  String singleSelectValue = scanner.next();  switch (singleSelectMethod) {  case 1:  selObj1.selectByIndex(Integer.*parseInt*(singleSelectValue));  System.*out*.println("Selected by Index: " + selObj1.getFirstSelectedOption().getText());  break;  case 2:  selObj1.selectByValue(singleSelectValue);  System.*out*.println("Selected by Value: " + selObj1.getFirstSelectedOption().getText());  break;  case 3:  selObj1.selectByVisibleText(singleSelectValue);  System.*out*.println("Selected by Visible Text: " + selObj1.getFirstSelectedOption().getText());  break;  default:  System.*out*.println("Invalid selection method.");  }  break;  case 2:  WebElement multiSelectElement = driver.findElement(By.*id*("cars"));  Select selObj2 = new Select(multiSelectElement);  System.*out*.println("Choose selection method: 1 for By Index, 2 for By Value, 3 for By Visible Text");  int multiSelectMethod = scanner.nextInt();  System.*out*.print("Enter the values for selection (comma separated): ");  String multiSelectValues = scanner.next();  for (String value : multiSelectValues.split(",")) {  switch (multiSelectMethod) {  case 1:  selObj2.selectByIndex(Integer.*parseInt*(value));  System.*out*.println("Selected by Index: " + selObj2.getAllSelectedOptions().get(selObj2.getAllSelectedOptions().size() - 1).getText());  break;  case 2:  selObj2.selectByValue(value);  System.*out*.println("Selected by Index: " + selObj2.getAllSelectedOptions().get(selObj2.getAllSelectedOptions().size() - 1).getText());  break;  case 3:  selObj2.selectByVisibleText(value);  System.*out*.println("Selected by Index: " + selObj2.getAllSelectedOptions().get(selObj2.getAllSelectedOptions().size() - 1).getText());  break;  default:  System.*out*.println("Invalid selection method.");  }  }  System.*out*.println("Choose deselection method: 1 for By Index, 2 for By Value, 3 for By Visible Text");  int multiSelectDeselectMethod = scanner.nextInt();    scanner.nextLine(); // Consume newline left over  System.*out*.print("Enter the values for deselection (comma separated): ");  String multiDeselectValues = scanner.nextLine();  for (String value : multiDeselectValues.split(",")) {  switch (multiSelectDeselectMethod) {  case 1:  selObj2.deselectByIndex(Integer.*parseInt*(value));  System.*out*.println("Delesected by Index: " + value);  break;  case 2:  selObj2.deselectByValue(value);  System.*out*.println("Delesected by Value: " + value);  break;  case 3:  selObj2.deselectByVisibleText(value);  System.*out*.println("Delesected by Visible Text: " + value);  break;  default:  System.*out*.println("Invalid deselection method.");  }  System.*out*.println("Deselected: " + value.trim());  }  // ... code for deselection based on the chosen method  System.*out*.println("Do you want to deselect all? (yes/no): ");  String deselectAll = scanner.next();  if (deselectAll.equalsIgnoreCase("yes")) {  selObj2.deselectAll();  System.*out*.println("All options deselected.");  }  break;  case 3:  exitProgram = true;  break;  default:  System.*out*.println("Invalid selection type. Please select 1, 2, or 3.");  break;  }  }  // Clean up and close resources  driver.quit();  scanner.close();  }  } |

**Output:**

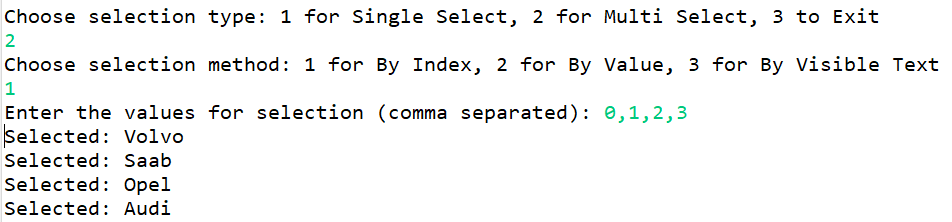


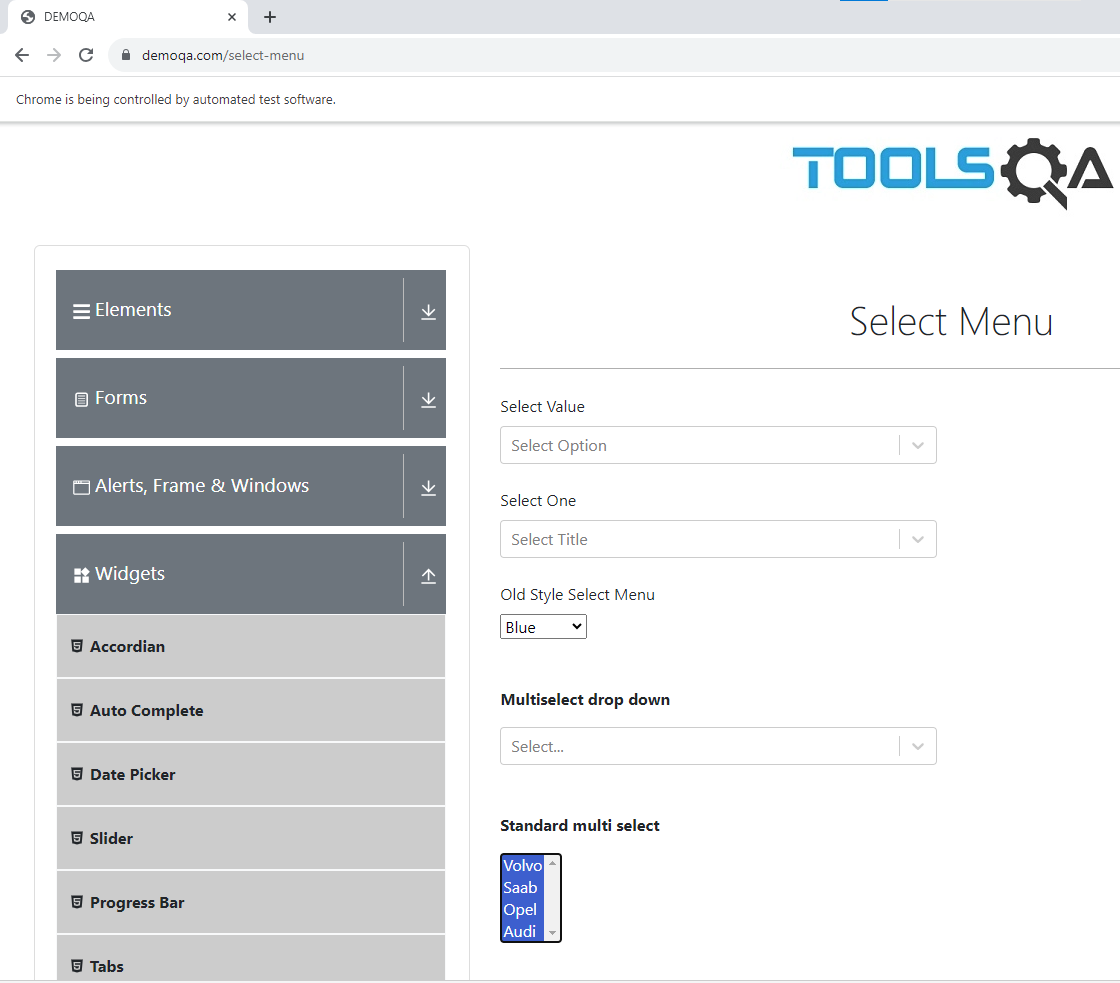
1. **SingleSelect Method:-**



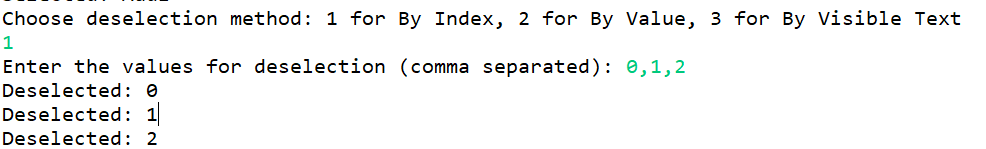
****

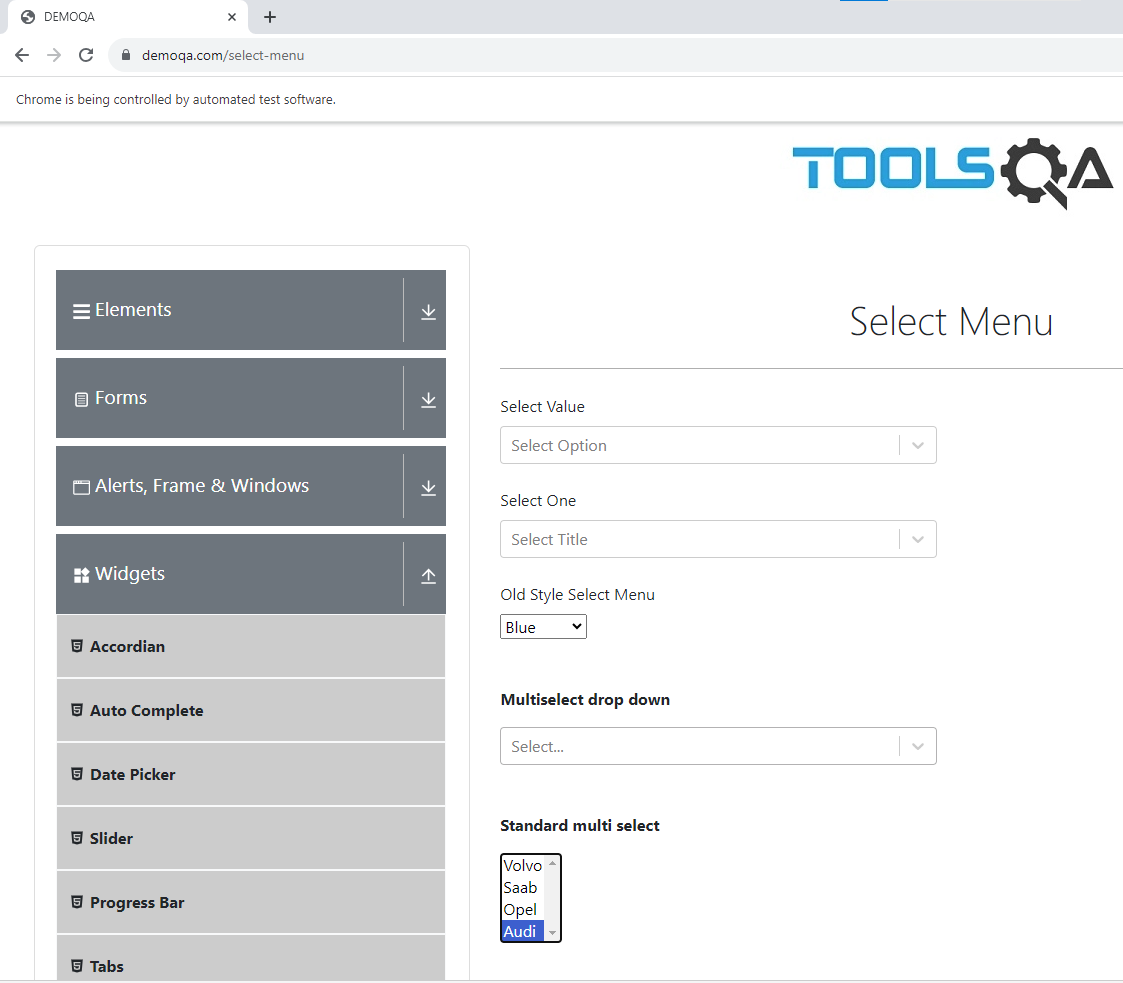
1. **MultiSelect Method:-**

****

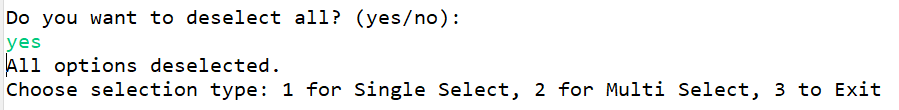
****

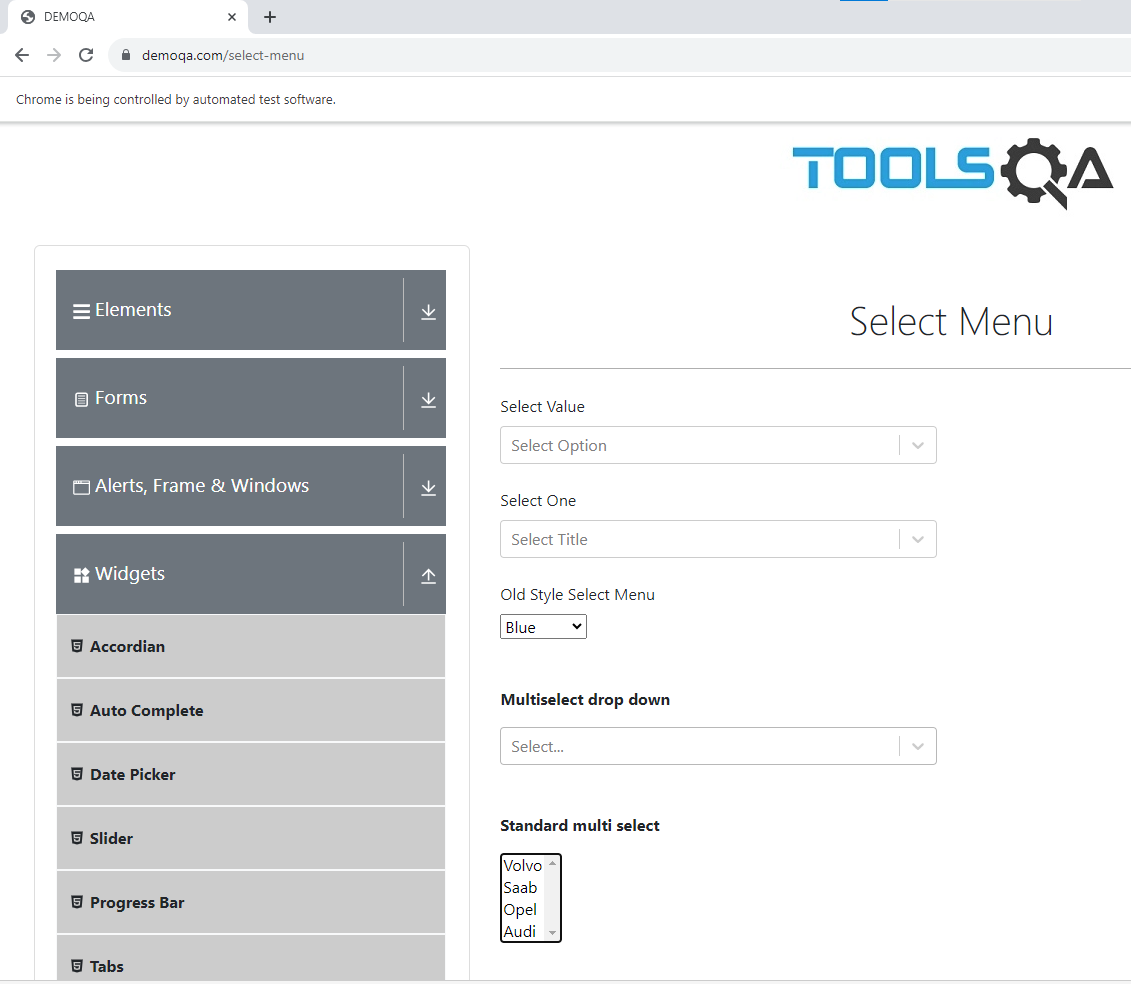
1. **MultiDeselect Method:-**

****

****

1. **Deselect All:-**

****

****