Automation Analyst Assignment

Tools and Technologies Used:

1. **Selenium WebDriver:** Automates browser actions like clicking, typing, and validating elements.
2. **TestNG:** A testing framework for organizing and executing tests with features like annotations and assertions.
3. **Java:** The programming language for writing the test script and logic.
4. **Actions Class:** Handles advanced user interactions (e.g., drag-and-drop).
5. **JavaScriptExecutor:** Executes JavaScript for tasks like scrolling.
6. **SoftAssert:** Validates multiple conditions without stopping the test.

How to Run This Test

1. **Prerequisites:**
   - Install **JDK 17** and Eclipse IDE.
   - Add **Selenium** and **TestNG** dependencies (Maven/Manual).
2. **Setup Project:**
   - Create a Java project.
   - Save the test script in the src/test/java folder or File converted into TestNG
   - Ensure ChromeDriver is configured.
3. **Execution:**
   - **In IDE:** Right-click the class and select **Run as TestNG Test**.

Task Details & Source code

Automated Test Script for Validating Revenue Calculator Functionality Using Selenium and TestNG and each task details in comments

```
package com.Automation.TestNG;

import org.testng.annotations.Test;
import org.testng.asserts.SoftAssert;

import static org.testng.Assert.assertEquals;

import java.time.Duration;

import org.openqa.selenium.By;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Keys;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.interactions.Actions;

//Class Setup
public class Automation_Analyst_Assignment_1 {
        WebDriver driver;
        Actions actions;
        JavascriptExecutor js;

// Test Method
        @Test
        public void seleniumTest_Automation_Analyst_Assignment_1() {
//Locators
                By btn_Menu_Revenue_Calculator = By.xpath("//div[text()='Revenue Calculator']");
                By sliderCalculator = By.xpath("//input[@aria-orientation='horizontal']");
                By adjustSliderCalculator = By.xpath("//input[@aria-
orientation='horizontal']//parent::span");
                By adjustSlider = By.xpath("//input[@aria-
orientation='horizontal']//parent::span//parent::span");
                By textNumber = By.xpath("//input[@type='number']");
                By SliderIndex = By.xpath("//input[@data-index='0']");
//Checkbox Locators
                By chkCPT99091 = By.xpath("//p[text()='CPT-99091']//following-sibling::label//input");
                By chkCPT99453 = By.xpath("//p[text()='CPT-99453']//following-sibling::label//input");
                By chkCPT99454 = By.xpath("//p[text()='CPT-99454']//following-sibling::label//input");
                By chkCPT99474 = By.xpath("//p[text()='CPT-99474']//following-sibling::label//input");
//Revenue and Reimbursement Locators
                By TotalGrossRevenuePerYear = By.xpath("//p[text()='Total Gross Revenue Per Year:']/p");
                By OneTimeReimbursementforallPatientsAnnually = By
                        .xpath("//p[text()='One Time Reimbursement for all Patients
Annually:']/p");
                By TotalRecurringReimbursementforallPatientsPerMonth = By
                        .xpath("//p[text()='Total Recurring Reimbursement for all Patients Per
Month:']/p");

// Browser Setup
                driver = new ChromeDriver();
                actions = new Actions(driver);
                js = (JavascriptExecutor) driver;
                driver.manage().window().maximize();
                driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(30)); // Set implicit wait
                driver.get("https://www.fitpeo.com/");
                /*
                 * Navigate to the FitPeo Homepage: Open the web browser and navigate to FitPeo
                 * Homepage.
                 */

// Click on Revenue Calculator
                driver.findElement(btn_Menu_Revenue_Calculator).click();
                /*
                 * Navigate to the Revenue Calculator Page: From the homepage, navigate to the
                 * Revenue Calculator Page.
                 */

//Slider Adjustment
                WebElement elementSliderCalculator = driver.findElement(SliderIndex);
                actions.moveToElement(elementSliderCalculator).perform();
                actions.dragAndDropBy(elementSliderCalculator, 94,
0).keyDown(Keys.ARROW_LEFT).keyDown(Keys.ARROW_LEFT)
                                .keyDown(Keys.ARROW_LEFT).perform();
                /*
                 * Scroll Down to the Slider section: Scroll down the page until the revenue
                 * calculator slider is visible.
                 */

//Validate Slider Value
                WebElement elementTextNumber = driver.findElement(textNumber);
                String strNumber = elementTextNumber.getAttribute("value");
                assertEquals(strNumber, "820");
                /*
                 * Adjust the Slider: Adjust the slider to set its value to 820. we've
                 * highlighted the slider in red color, Once the slider is moved the bottom text
                 * field value should be updated to 820
                 */

//Update Slider Value
                WebElement elementUpdateNumber = driver.findElement(textNumber);
                js.executeScript("arguments[0].scrollIntoView({behavior: 'smooth', block: 'center'});",
elementUpdateNumber);
                elementUpdateNumber.click();
                actions.doubleClick(elementUpdateNumber).click().perform();
                elementUpdateNumber.sendKeys("560");

                /*
                 * Update the Text Field: Click on the text field associated with the slider.
                 * Enter the value 560 in the text field. Now the slider also should change
                 * accordingly
                 */
                String strUpdateNumber = elementUpdateNumber.getAttribute("value");
                assertEquals(strUpdateNumber, "560");
                /*
                 * Validate Slider Value: Ensure that when the value 560 is entered in the text
                 * field, the slider's position is updated to reflect the value 560.
                 */

//Select CPT Codes
                driver.findElement(chkCPT99091).click();
                driver.findElement(chkCPT99453).click();
                driver.findElement(chkCPT99454).click();
```

```
        driver.findElement(chkCPT99474).click();

        /*
         * Select CPT Codes: Scroll down further and select the checkboxes for
         * CPT-99091, CPT-99453, CPT-99454, and CPT-99474.
         */

//Extract and Calculate Revenue
        String strTotalGrossRevenuePerYear = 
driver.findElement(TotalGrossRevenuePerYear).getText();
        String replacestrTotalGrossRevenuePerYear = strTotalGrossRevenuePerYear.replace("$", 
"");
        double doubleTotalGrossRevenuePerYear = 
Double.parseDouble(replacestrTotalGrossRevenuePerYear);

        String strOneTimeReimbursementforallPatientsAnnually = driver
                .findElement(OneTimeReimbursementforallPatientsAnnually).getText();
        String replacestrOneTimeReimbursementforallPatientsAnnually = 
strOneTimeReimbursementforallPatientsAnnually
                .replace("$", "");
        double doubleOneTimeReimbursementforallPatientsAnnually = Double
                .parseDouble(replacestrOneTimeReimbursementforallPatientsAnnually);

        String strTotalRecurringReimbursementforallPatientsPerMonth = driver

    .findElement(TotalRecurringReimbursementforallPatientsPerMonth).getText();
        String replacestrTotalRecurringReimbursementforallPatientsPerMonth = 
strTotalRecurringReimbursementforallPatientsPerMonth
                .replace("$", "");
        double doubleTotalRecurringReimbursementforallPatientsPerMonth = Double

    .parseDouble(replacestrTotalRecurringReimbursementforallPatientsPerMonth);
//Validation
        double TotalRecurringReimbursement = (doubleTotalGrossRevenuePerYear
                - doubleOneTimeReimbursementforallPatientsAnnually) / 12;
        assertEquals(doubleTotalRecurringReimbursementforallPatientsPerMonth, 
TotalRecurringReimbursement);
        /* Validate Total Recurring Reimbursement: */
//Soft Assertion
        SoftAssert softAssert = new SoftAssert();
        softAssert.assertEquals(strTotalRecurringReimbursementforallPatientsPerMonth, 
"$110700",
                "Total Recurring Reimbursement for all Patients Per Month: shows the 
value $110700.");
        softAssert.assertAll();
        /*
         * Verify that the header displaying Total Recurring Reimbursement for all
         * Patients Per Month: shows the value $110700.
         */

//Close Browser
        driver.close();
        driver.quit();

    }

}
```