Step 1: Set up your Java project

1. Install Java:

Ensure that you have Java installed on your machine. You can download and install it from the [official website](https://www.oracle.com/java/technologies/downloads/).

2. Install an Integrated Development Environment (IDE):

Choose an IDE for Java development. Popular choices include Eclipse, IntelliJ IDEA, or Visual Studio Code.

3. Create a new Java project:

Set up a new Java project in your chosen IDE.

Step 2: Add Selenium WebDriver to your project

1. Download Selenium WebDriver:

Download the Selenium WebDriver Java bindings from the [official Selenium website](https://www.selenium.dev/downloads/). Add the JAR files to your project.

2. Configure your project:

Set up the Selenium WebDriver in your project. You can use Maven or Gradle for dependency management. If you're using Maven, add the following dependency to your **pom.xml** file:

```
<dependencies>
  <dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-java</artifactId>
    <version>3.141.59</version> <!-- Use the latest version available -->
  </dependency>
</dependencies>
```

```
}
```

Step 3: Write your first Selenium script

Now, let's create a simple Selenium script to open a browser, navigate to a website, and perform basic interactions.

```
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class SeleniumTutorial {

  public static void main(String[] args) {
    // Set the path to the ChromeDriver executable (download from https://sites.google.com/chromium.org/driver/ )
    System.setProperty("webdriver.chrome.driver", "path/to/chromedriver.exe");

    // Create a new instance of the ChromeDriver
    WebDriver driver = new ChromeDriver();

    // Navigate to a website
    driver.get("https://www.example.com");

    // Perform some actions (e.g., print the title)
    System.out.println("Page Title: " + driver.getTitle());

    // Close the browser
    driver.quit();
  }
}
```

Step 4: Run your Selenium script

1. Make sure you have the ChromeDriver executable downloaded and its path correctly set in your script.
2. Run your Java program.

This script opens a Chrome browser, navigates to "https://www.example.com," prints the title, and then closes the browser.