**How to launch the Webdriver without using Exe file?**

In general, we need to download the webdriver exe file path and provide the exe file path i.e. system.setProperty(“webdriver.chrome.driver”,”Webdriver exe file path”) to launch the Browser web driver.

By using this bonigarcia WebDriverManager API we don’t need to provide the above, and we can directly lauch the specific browser .

If you are using the Maven project, use the below dependency in the POM.xml file or you can get the same from the github url <https://github.com/bonigarcia/webdrivermanager/>

<dependency>

<groupId>io.github.bonigarcia</groupId>

<artifactId>webdrivermanager</artifactId>

<version>3.1.1</version>

<scope>test</scope>

</dependency>

Now you can the write the code

**package** seleniumTests;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.chrome.ChromeDriver;

**import** org.openqa.selenium.edge.EdgeDriver;

**import** org.openqa.selenium.firefox.FirefoxDriver;

**import** org.openqa.selenium.ie.InternetExplorerDriver;

**import** org.openqa.selenium.opera.OperaDriver;

**import** org.openqa.selenium.phantomjs.PhantomJSDriver;

**import** io.github.bonigarcia.wdm.WebDriverManager;

**public** **class** LaunchingWebDriverWithoutExeFile {

**public** **static** WebDriver *driver*;

**public** **static** **void** main(String[] args) {

//to launch chrome driver

//WebDriverManager.chromedriver().setup();

//driver=new ChromeDriver();

//To launch Firefox driver

WebDriverManager.*firefoxdriver*().setup();

*driver*=**new** FirefoxDriver();

//To launch operadriver

//WebDriverManager.operadriver().setup();

//driver=new OperaDriver();

//To launch phantomjs driver

//WebDriverManager.phantomjs().setup();

//driver=new PhantomJSDriver();

//To launch edgedriver

//WebDriverManager.edgedriver().setup();

//driver=new EdgeDriver();

//To launch iedriver

//WebDriverManager.iedriver().setup();

//driver=new InternetExplorerDriver();

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// If you want to use specific version of the browser then you have to provide the below

// WebDriverManager.chromedriver().version("2.26").setup();

// If you want to use 32 bit version of Gecko driver

// WebDriverManager.firefoxdriver().arch32().setup();

*driver*.get("https://www.google.com/");

String title=*driver*.getTitle();

System.***out***.println("\*\*\*\*\*\*\*\*\* Title is \*\*\*\*\*\*\*\*\* " + title);

*driver*.quit();

}

}

If you are using TestNG , we can write the below

public class ChromeTest {

private WebDriver driver;

@BeforeClass

public static void setupClass() {

WebDriverManager.chromedriver().setup();

}

@Before

public void setupTest() {

driver = new ChromeDriver();

}

@After

public void teardown() {

if (driver != null) {

driver.quit();

}

}

@Test

public void test() {

// Your test code here

}

}