

Let's create two separate classes, each demonstrating implicit and explicit waits in Selenium with WebDriverManager in Java.

Implicit Wait Example

```
```java
// ImplicitWaitExample.java
import io.github.bonigarcia.wdm.WebDriverManager;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;

import java.util.concurrent.TimeUnit;

public class ImplicitWaitExample {
 public static void main(String[] args) {
 // Setup WebDriver using WebDriverManager
 WebDriverManager.chromedriver().setup();
 WebDriver driver = new ChromeDriver();

 // Implicit Wait
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

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

 // Find an element using implicit wait
 WebElement element = driver.findElement(By.id("exampleElement"));

 // Perform actions on the element
 element.click();

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

In this example, the `implicitlyWait` method sets an implicit wait of 10 seconds. This means that Selenium will wait up to 10 seconds when attempting to find an element before throwing a `NoSuchElementException`.

Explicit Wait Example

```
```java
// ExplicitWaitExample.java
import io.github.bonigarcia.wdm.WebDriverManager;
```

```

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.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;

public class ExplicitWaitExample {
 public static void main(String[] args) {
 // Setup WebDriver using WebDriverManager
 WebDriverManager.chromedriver().setup();
 WebDriver driver = new ChromeDriver();

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

 // Explicit Wait
 WebDriverWait wait = new WebDriverWait(driver, 10);

 // Example of using ExpectedConditions for explicit wait
 WebElement element =
wait.until(ExpectedConditions.presenceOfElementLocated(By.id("exampleElement")));

 // Perform actions on the element
 element.click();

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

```

In this example, an explicit wait using `WebDriverWait` is set up with a timeout of 10 seconds. The `ExpectedConditions.presenceOfElementLocated` condition is used to wait until an element with the ID "exampleElement" is present on the page before proceeding.

These examples showcase both implicit and explicit waits in Selenium with WebDriverManager. You can run each class independently to observe the behavior of implicit and explicit waits. Adjust the timeouts and locators based on your specific requirements.