Lesson 06 Demo 04

Implementing Lambda and Local Var in Java

Objective: To implement lambda expressions and the var keyword in Java 11

Tools required: Eclipse IDE

Prerequisites: None

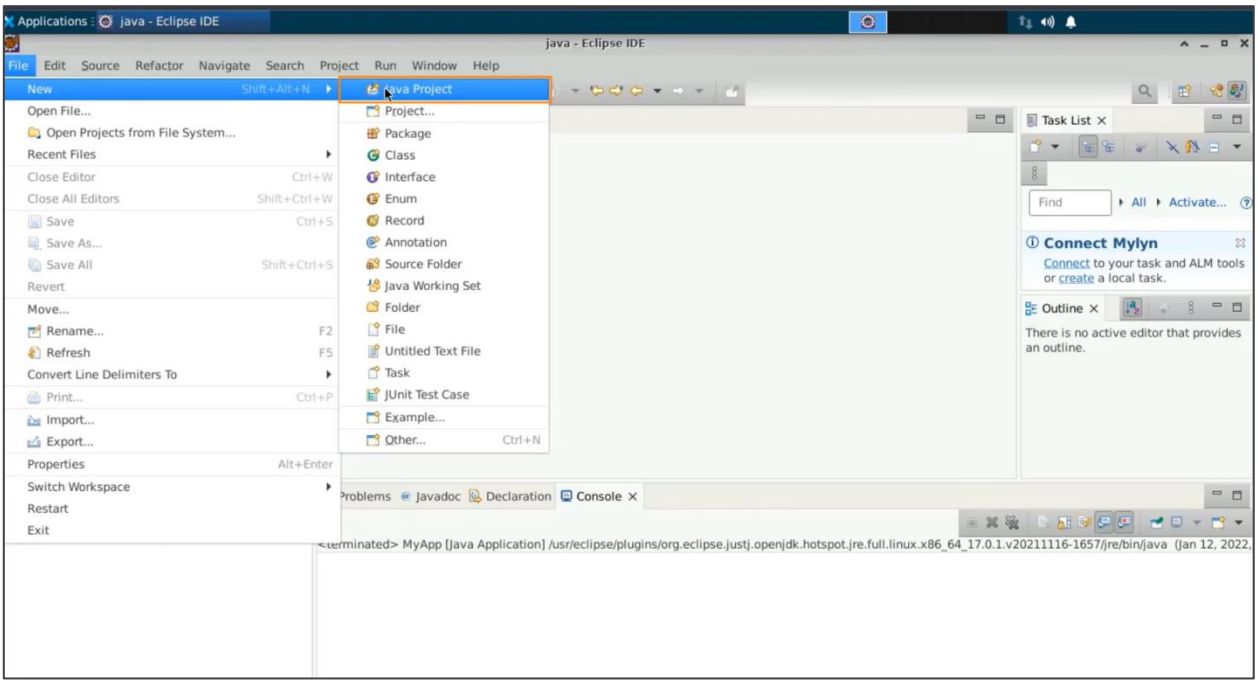
Steps to be followed:

1. Open the Eclipse IDE and create a new Java project
2. Create a list which goes as the list of type string and then the emails as arrays dot as a list and executing the code
3. Create the data as comma separated values and execute the code
4. Write the variable or the var keywords, inside the lambdas

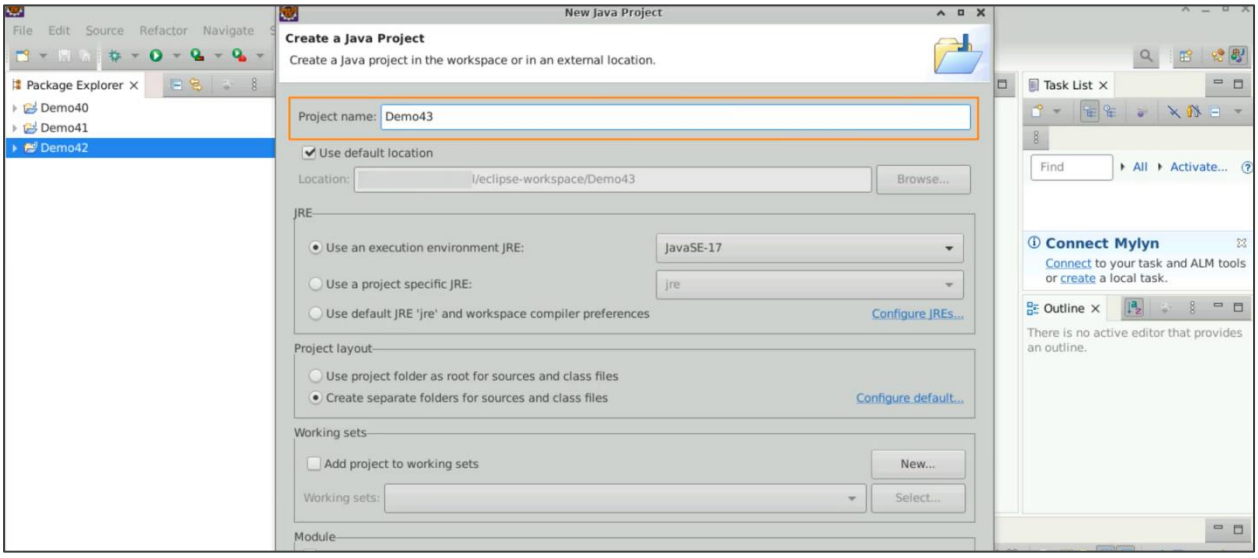
Step 1: Open the Eclipse IDE and create a new Java project

1.1 Open the **Eclipse IDE**

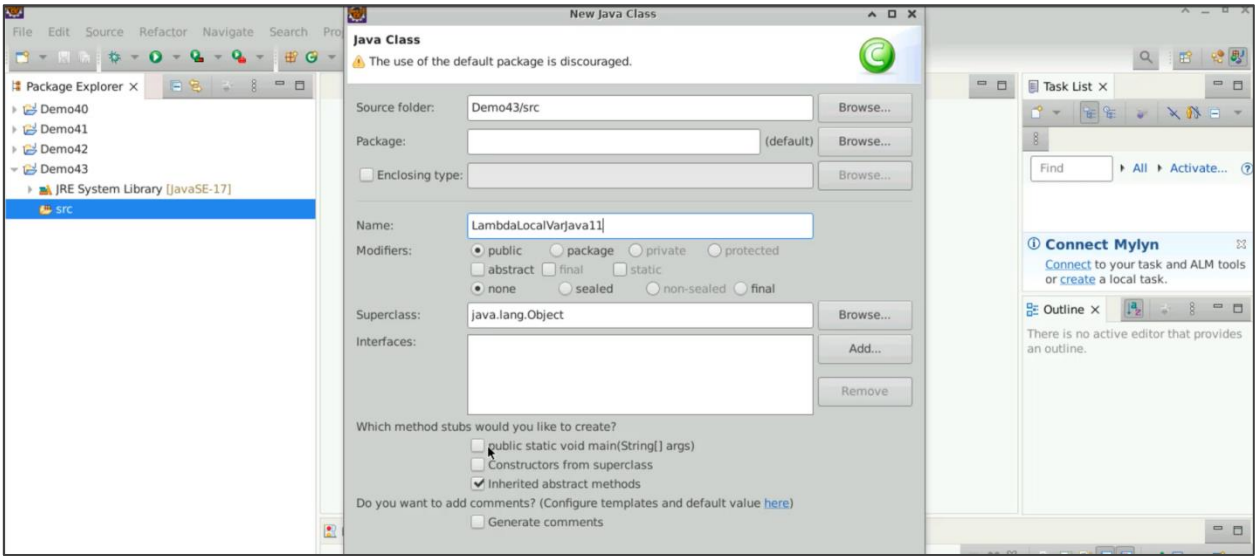
1.2 Select **File**, then **New,** and then **Java project**



1.3 Name the project **Demo43,** uncheck **Create a module-info.java file**, and press **Finish**

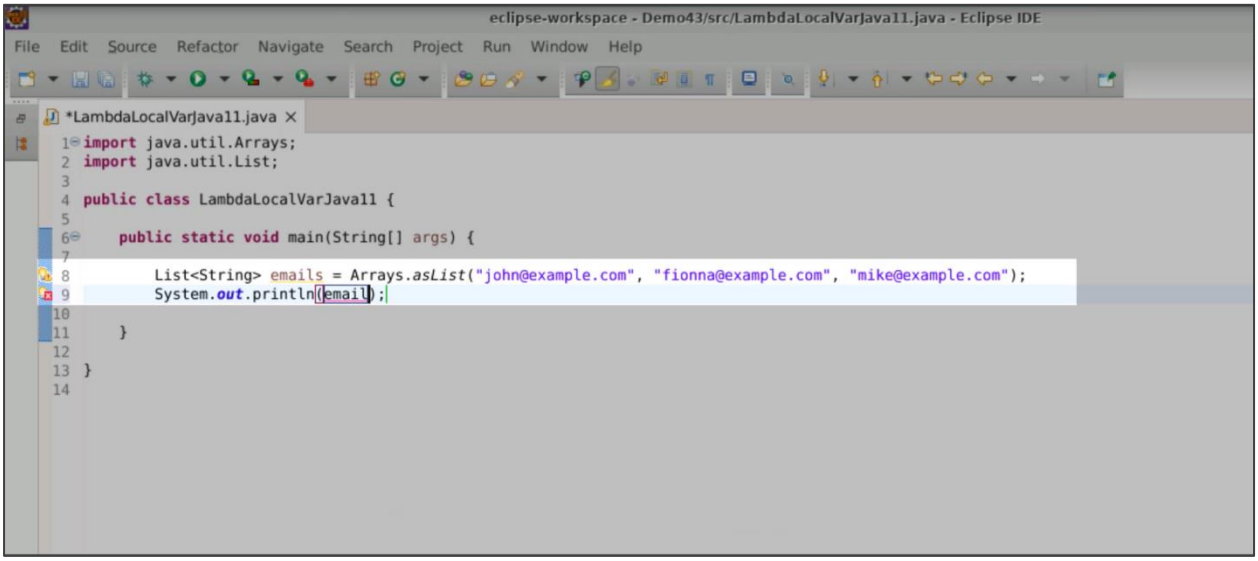


1.4 With **Demo43** selected in the **src** folder, right-click and create a new class. Name this class **LambdaLocalVarJava11**, then select the **main** method, and then select **Finish**

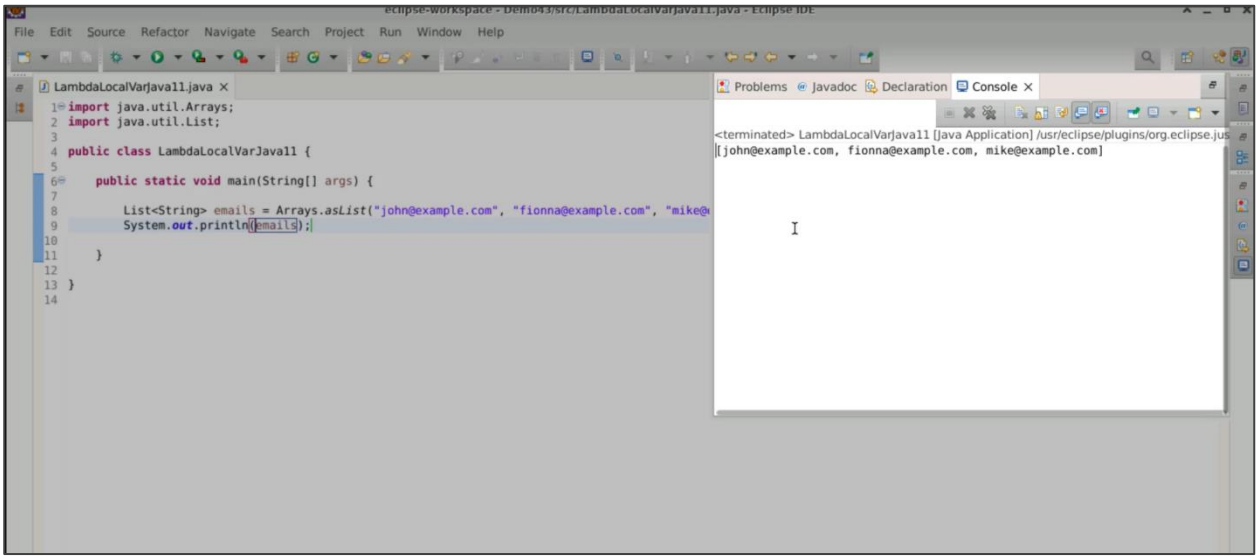


Step 2: Create a list of type String and add emails using Arrays.asList(), then execute the code

2.1 First, create a list of type String using **Arrays.asList()**. Then, add a few elements such as **john@example.com, fionna@example.com, and mike@example.com**. This will give us a list of emails. Finally, print the emails

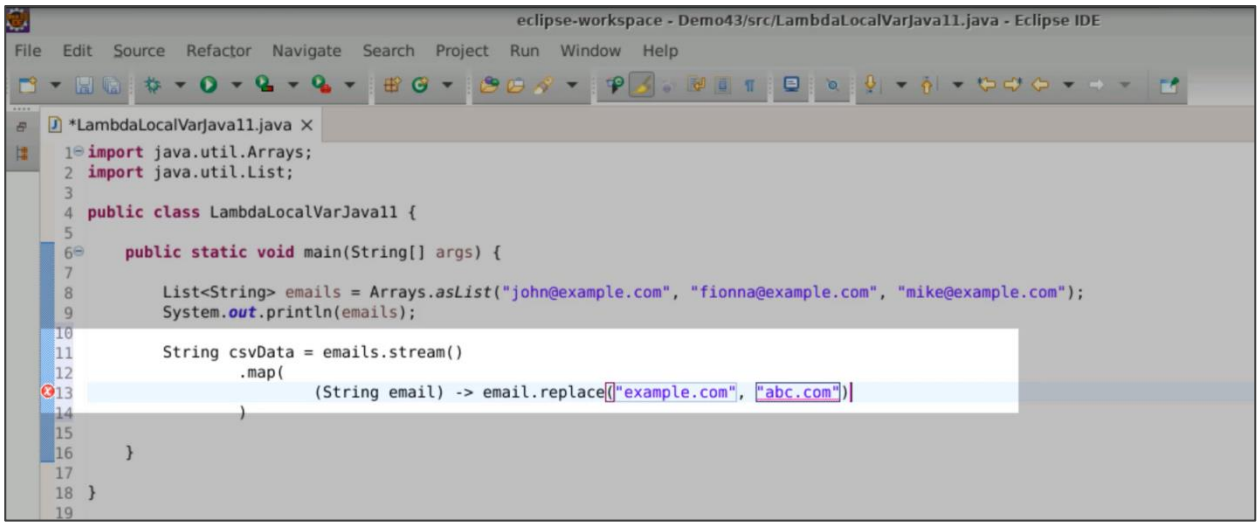


2.2 When you run the code, it will show the basic list of emails

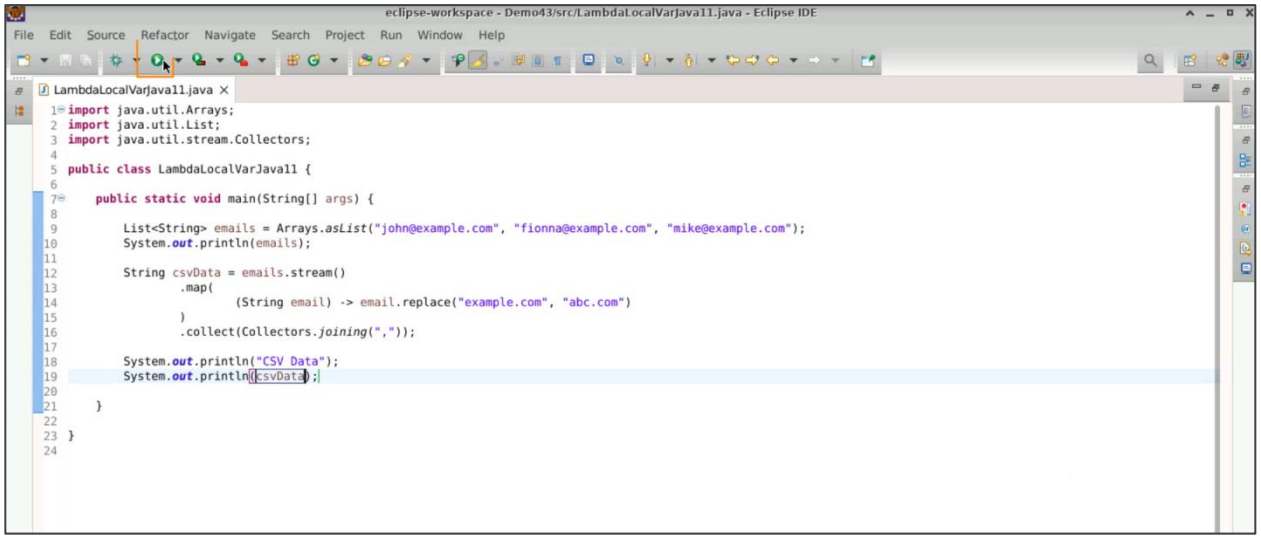


Step 3: Create the data as comma-separated values and execute the code

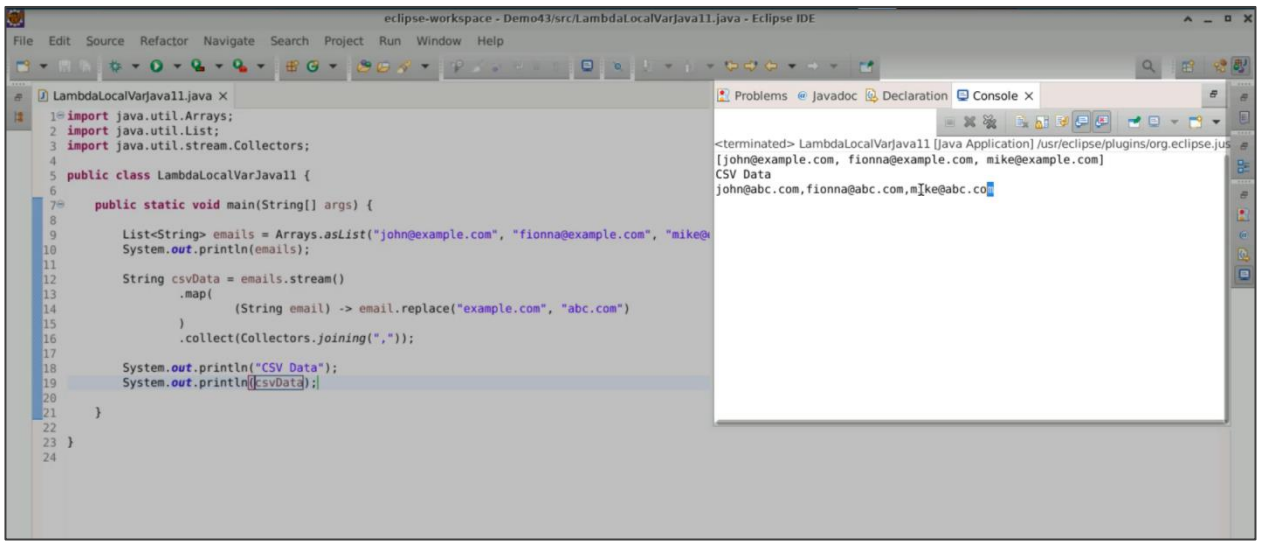
3.1 First, create the data as comma-separated values. Write it **String csvData = emails.stream()**



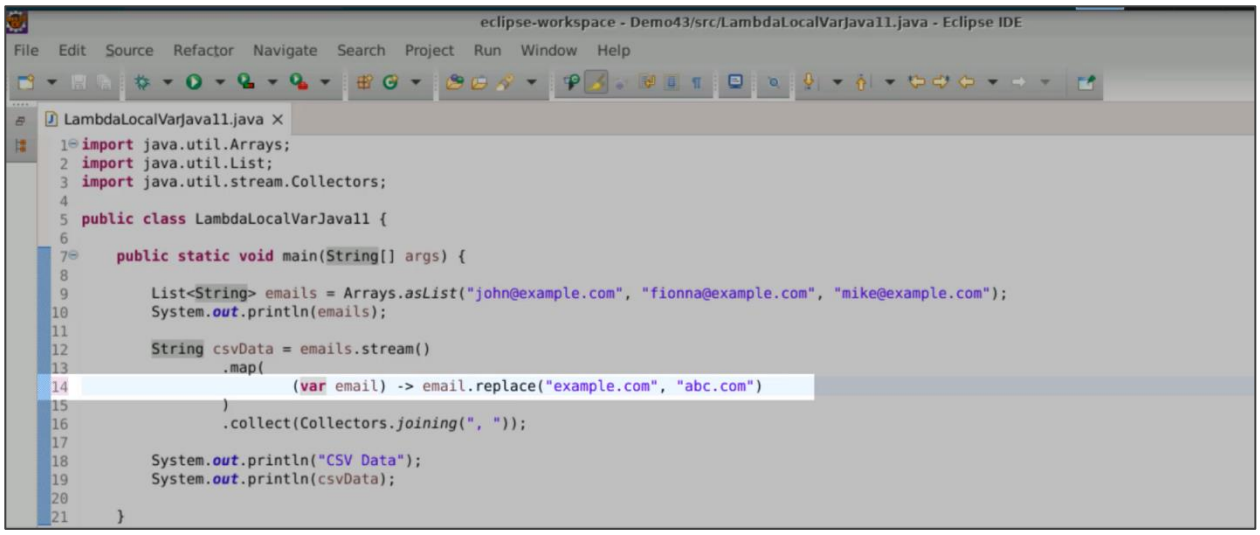
3.2 Use **.collect(Collectors.joining(","))** and add a semicolon at the end. Then, print the CSV data, which is a new joining created from a domain name



3.3 Run the code, and you will notice that the emails are separated by commas. You can even use a comma and a space to introduce a space in between

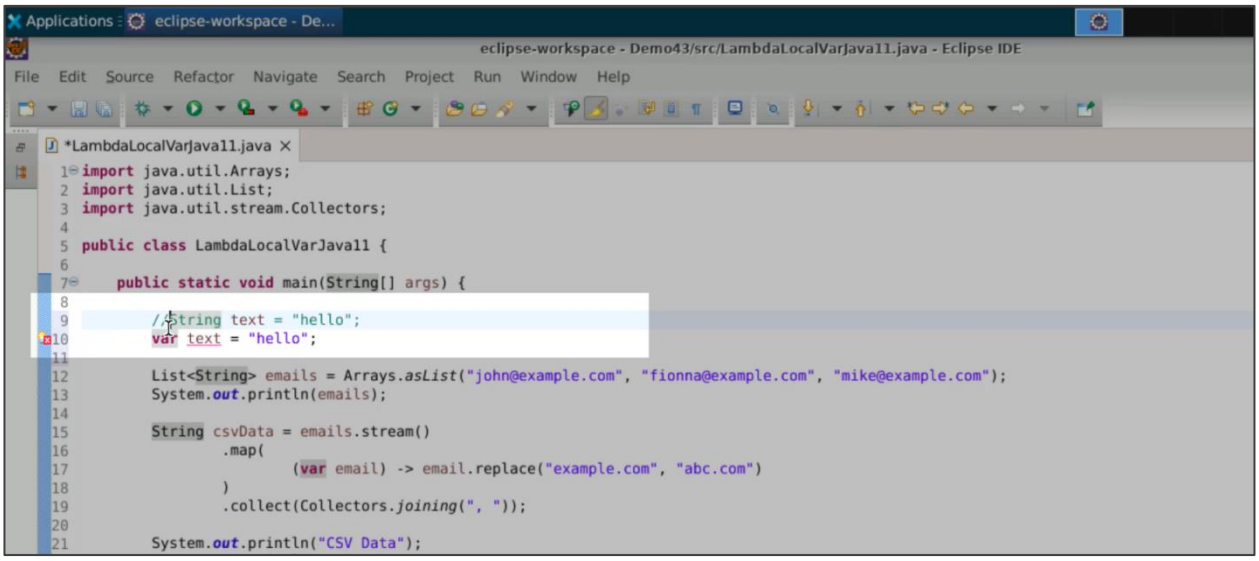


3.4 Instead of using **String** here, you can just write **var email**



Step 4: Write the variable or the var keywords, inside the lambdas

4.1 Use the variable **String text** as **hello**. You can even write **var text = "hello"**. The same variable or the **var** keyword can be used inside your lambdas as well. This is one of the new features in Java 11



By following these steps, you have successfully implemented lambda expressions and the var keyword in Java 11.