**Section-End Project - Lesson 3**

## **[Synchronization](https://personales.unican.es/corcuerp/java/Labs/LAB_23.htm" \l "Exercise_1)**

In this exercise, you are going to exercise how to do synchronization among threads.

1. **[Build and run a program in which threads are NOT synchronized](https://personales.unican.es/corcuerp/java/Labs/LAB_23.htm" \l "4.1)**
2. **[Build an run a program in which threads are synchronized through synchronized method](https://personales.unican.es/corcuerp/java/Labs/LAB_23.htm" \l "4.2)**
3. **[Build and run a program in which threads are synchronized through synchronized statement on a common object](https://personales.unican.es/corcuerp/java/Labs/LAB_23.htm" \l "4.3)**

### **(4.1) Build and run a program in which threads are NOT synchronized**

In this step, you are going to build an application that displays a result that is not desirable since threads are not synchronized. 

0. Start Spring Tools Suite IDE if you have not done so yet.  
  
1. Create a new Java project

* Select **File->New Java Project (Alt+Shift+N)**. The **New Java Project** dialog box appears.
* For the Project Name field, type in **UnsynchronizedExample**as project name.
* Choose ‘Use default JRE (11….). Make sure you have set your default JRE set to version 11 and the appropriate compliance settings as well. Deselect ‘Create module-info.java file’ if already selected. Click Next.Click Finish.
* Observe that **RunnableThreadTest1** project appears
* Right click on the project and select **File**->**New Class.**
* Enter in a package name of your choosing or use the default. Type in **UnsynchronizedExample**as the class name. The main method stub should also be created.
* The IDE generated **UnsynchronizedExample.java** is displayed in the source editor window of STS IDE.

2. Modify the IDE generated **UnsynchronizedExample.java**as shown in Code-4.11 below.

|  |
| --- |
| public class UnsynchronizedExample {         public static void main(String[] args) {         new PrintStringsThread("Hello ", "there.");         new PrintStringsThread("How are ", "you?");         new PrintStringsThread("Thank you ", "very much!");     }     } |

Code-4.11: UnsynchronizedExample.java  
  
3. Write PrintStringsThread.java as shown in Code-4.12 below.

|  |
| --- |
| public class PrintStringsThread implements Runnable {         Thread thread;     String str1, str2;         PrintStringsThread(String str1, String str2) {         this.str1 = str1;         this.str2 = str2;         thread = new Thread(this);         thread.start();     }         public void run() {         TwoStrings.print(str1, str2);     }     } |

Code-4.12: PrintStringsThread.java  
  
4. Write TwoStrings.java as shown in Code-4.13 below.  Study the code by paying special attention to the bold fonted parts.  Note that the print method is not synchronized.

|  |
| --- |
| public class TwoStrings {         **// This method is not synchronized**     **static void print(String str1, String str2)** {         System.out.print(str1);         try {             Thread.sleep(500);         } catch (InterruptedException ie) {         }         System.out.println(str2);     } } |

Code-4.13: TwoStrings.java  
  
5. Build and run the project

* Right click **UnsynchronizedExample**project and select **Run**.
* Observe the result in the **Output**window. (Figure-4.14 below)

|  |
| --- |
| Hello How are Thank you there. very much! you? |

Figure-4.14: Result of running UnsynchronizedExample application

### **(4.2) Build and run a program in which threads are synchronized through synchronized method**

In this step, you are going to build an application that displays a desired result because the threads are synchronized.

1. Create a new Java project

* Select **File->New Java Project (Alt+Shift+N)**. The **New Java Project** dialog box appears.
* For the Project Name field, type in **SynchronizedExample1**as project name.
* Choose ‘Use default JRE (11….). Make sure you have set your default JRE set to version 11 and the appropriate compliance settings as well. Deselect ‘Create module-info.java file’ if already selected. Click Next.Click Finish.
* Observe that **SynchronizedExample1**project appears
* Right click on the project and select **File**->**New Class.**
* Enter in a package name of your choosing or use the default. Type in **SynchronizedExample1** as the class name. The main method stub should also be created.
* The IDE generated **SynchronizedExample1 .java** is displayed in the source editor window of STS IDE.

2. Modify the IDE generated **SynchronizedExample1.java**as shown in Code-4.21 below.

|  |
| --- |
| public class SynchronizedExample1 {         public static void main(String[] args) {         new PrintStringsThread("Hello ", "there.");         new PrintStringsThread("How are ", "you?");         new PrintStringsThread("Thank you ", "very much!");     }     } |

Code-4.21: SynchronizedExample1.java  
  
3. Write PrintStringsThread.java as shown in Code-4.22 below.

|  |
| --- |
| public class PrintStringsThread implements Runnable {         Thread thread;     String str1, str2;         PrintStringsThread(String str1, String str2) {         this.str1 = str1;         this.str2 = str2;         thread = new Thread(this);         thread.start();     }         public void run() {         TwoStrings.print(str1, str2);     }     } |

Code-4.22: PrintStringsThread.java  
  
4. Write TwoStrings.java as shown in Code-4.23 below.  Study the code by paying special attention to the bold fonted parts.

|  |
| --- |
| public class TwoStrings {        **// This method is now synchronized     synchronized** static void print(String str1, String str2) {         System.out.print(str1);         try {             Thread.sleep(500);         } catch (InterruptedException ie) {         }         System.out.println(str2);     } } |

Code-4.23: TwoStrings.java  
  
5. Build and run the project

* Right click **SynchronizedExample1**project and select **Run**.
* Observe the result in the **Output**window. (Figure-4.24 below)

|  |
| --- |
| How are you? Thank you very much! Hello there. |

Figure-4.24: Result of running SynchronizedExample1 application

### **(4.3) Build and run a program in which threads are synchronized through synchronized statement on common object**

In this step, you are going to build another application that displays a desired result because the threads are synchronized.

1. Create a new Java project

* Select **File->New Java Project (Alt+Shift+N)**. The **New Java Project** dialog box appears.
* For the Project Name field, type in **SynchronizedExample2**as project name.
* Choose ‘Use default JRE (11….). Make sure you have set your default JRE set to version 11 and the appropriate compliance settings as well. Deselect ‘Create module-info.java file’ if already selected. Click Next.Click Finish.
* Observe that **SynchronizedExample2**project appears
* Right click on the project and select **File**->**New Class.**
* Enter in a package name of your choosing or use the default. Type in **SynchronizedExample2**as the class name. The main method stub should also be created.
* The IDE generated **SynchronizedExample2 .java** is displayed in the source editor window of STS IDE.

2. Modify the IDE generated **SynchronizedExample2.java**as shown in Code-4.31 below.

|  |
| --- |
| public class SynchronizedExample2 {         public static void main(String[] args) {                 TwoStrings ts = new TwoStrings();                 new PrintStringsThread("Hello ", "there.", ts);         new PrintStringsThread("How are ", "you?", ts);         new PrintStringsThread("Thank you ", "very much!", ts);     }     } |

Code-4.31: SynchronizedExample2.java  
  
3. Write PrintStringsThread.java as shown in Code-4.32 below.  Study the code by paying special attention to the bold fonted parts.

|  |
| --- |
| public class PrintStringsThread implements Runnable {         Thread thread;     String str1, str2;     **TwoStrings ts;**         PrintStringsThread(String str1, String str2,                        TwoStrings ts) {         this.str1 = str1;         this.str2 = str2;         this.ts = ts;         thread = new Thread(this);         thread.start();     }         public void run() {   **// Synchronize over TwoString object         synchronized (ts) {             ts.print(str1, str2);         }**     } } |

Code-4.32: PrintStringsThread.java  
  
4. Write TwoStrings.java as shown in Code-4.33 below.

|  |
| --- |
| public class TwoStrings {         static void print(String str1, String str2) {         System.out.print(str1);         try {             Thread.sleep(500);         } catch (InterruptedException ie) {         }         System.out.println(str2);     } } |

Code-4.33: TwoStrings.java  
  
5. Build and run the project

* Right click **SynchronizedExample2**project and select **Run**.
* Observe the result in the **Output**window. (Figure-4.34 below)

|  |
| --- |
| Hello there.  Thank you very much!  How are you? |

Figure-4.34: Result of running SynchronizedExample2 application

### **Summary**

In this exercise, you have learned how to use synchronization.