**Section-End Project - Lesson 3**

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

A thread is a thread of execution in a program. The Java Virtual Machine allows an application to have multiple threads of execution running concurrently.  This hands-on lab takes you through the basics of using Java threading.

In this exercise, you are going to create and start a thread by writing a class that implements Runnable interface.

1. **[Create and start a thread by implementing Runnable interface - start() method is not in the constructor](https://personales.unican.es/corcuerp/java/Labs/LAB_23.htm" \l "2.1)**
2. **[Create and start a thread by implementing Runnable interface - start() method is in the constructor](https://personales.unican.es/corcuerp/java/Labs/LAB_23.htm" \l "2.2)**

### **(2.1) Create and start a thread by implementing Runnable interface - start() method is not in the constructor**

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 **RunnableThreadTest1**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 **RunnableThreadTest1**as the class name. The main method stub should also be created.
* The IDE generated **RunnableThreadTest1 .java** is displayed in the source editor window of STS IDE.

2. Modify the IDE generated **RunnableThreadTest1.java**as shown in Code-2.11 below.  Study the code by paying special attention to the bold fonted parts.  Note that the **start()** method needs to be invoked explicitly after an object instance of the **PrintNameRunnable**class is created.

|  |
| --- |
| public class RunnableThreadTest1 {         public static void main(String args[]) {                 PrintNameRunnable pnt1 = new PrintNameRunnable("A");         Thread t1 = new Thread(pnt1);        **t1.start();**                 PrintNameRunnable pnt2 = new PrintNameRunnable("B");         Thread t2 = new Thread(pnt2);         **t2.start();**                 PrintNameRunnable pnt3 = new PrintNameRunnable("C");         Thread t3 = new Thread(pnt3);         **t3.start();**             } } |

Code-2.11: RunnableThreadTest1.java  
  
3. Write **PrintNameRunnable.java**as shown in Code-2.12 below.

|  |
| --- |
| // The class implements Runnable interface class PrintNameRunnable implements Runnable {         String name;         PrintNameRunnable(String name) {         this.name = name;     }         // Implementation of the run() defined in the     // Runnable interface.     public void run() {         for (int i = 0; i < 10; i++) {             System.out.print(name);         }     } } |

Code-2.12: PrintNameRunnable.java  
  
4. Build and run the project

* Right click **RunnableThreadTest1**project and select **Run**.
* Observe the result in the **Output**window. (Figure-2.13 below)

|  |
| --- |
| ACBACBACBACBACABCABCABCABCABCB |

Figure-2.13: Result of running RunnableThreadTest1 application  
  
  
5. For your own exercise, do the following. Build and run the application.

* Create another class called **MyOwnRunnableClass**that implements Runnable interface
* **MyOwnRunnableClass**displays values 1 to 10 inside its run() method
* Modify **RunnableThreadTest1.java** to start 2 thread instances of **MyOwnRunnableClass**.

### **(2.2) Create and start a thread by implementing Runnable interface - start() method is in the constructor**

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 **RunnableThreadTest2**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 **RunnableThreadTest2**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 **RunnableThreadTest2**as the class name. The main method stub should also be created.
* The IDE generated **RunnableThreadTest2 .java** is displayed in the source editor window of STS IDE.

2. Modify the IDE generated **RunnableThreadTest2.java**as shown in Code-2.21 below.  Study the code by paying special attention to the bold fonted parts.

|  |
| --- |
| public class RunnableThreadTest2 {         public static void main(String args[]) {          **// Since the constructor of the PrintNameRunnable         // object creates a Thread object and starts it,         // there is no need to do it here.         new PrintNameRunnable("A");**                 new PrintNameRunnable("B");         new PrintNameRunnable("C");     } } |

Code-2.21: RunnableThreadTest2.java  
  
3. Write PrintNameRunnable.java as shown in Code-2.22 below.  Study the code by paying special attention to the bold fonted parts.  Note that the start() method is in the constructor of the PrintNameRunnable class.

|  |
| --- |
| // The class implements Runnable interface class PrintNameRunnable implements Runnable {         Thread thread;         PrintNameRunnable(String name) { **thread = new Thread(this, name);         thread.start();**     }         // Implementation of the run() defined in the     // Runnable interface.     public void run() {         String name = thread.getName();         for (int i = 0; i < 10; i++) {             System.out.print(name);         }     } } |

Code-2.22: PrintNameRunnable.java  
  
4. Build and run the project

* Right click **RunnableThreadTest2**project and select **Run**.
* Observe the result in the **Output**window. (Figure-1.23 below)

|  |
| --- |
| ABCABCABCABCABCABCABCBACBACBAC |

Figure-2.23: Result of running RunnableThreadTest2 application

### **Summary**

In this exercise,  you have learned how to create a class that implements Runnable interface and starts a thread.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*