|  |
| --- |
| Hands-on Exercise Objective |
| After completing the hands-on exercises, you will be able to:   * Develop threads using Runnable interface. * How to create multiple threads and use synchronized methods. |

|  |
| --- |
| Problem Statement 1:Develop a simple java application which needs to store 100Lakh Integers in an ArrayList starting from 0…100lakhs.Create a main method which can have a loop ranging from 0 -100lakh. Calculate the response time required for the process. The response time can be calculated as the difference between the start time and end time of the process which can be calculated using the system time.  1. Develop a class ListLoader with a method loadList()   **Method Name :** loadList  **Method Arguments** : Integer ***startNumber***, Integer ***lastNumber***  **Method return type :** void  **Method Logic:** Should iterate using a looping starting from ***startNumber*** andending with ***lastNumber*** and load an array list (***L***) with the Integer numbers. The Array List (**L**) should be defined as a member variable in the class.  Develop a main method which will invoke the loadList() and pass the parameter ***startNumber*** as 0 and ***lastNumber*** as 100 lakh. Print the time taken for the method to run.  **Hint:** Use **System.currentTimeMillis()** before and after the method execution to print the response time. Problem Statement # 2: Now we will see how multi threading can improve the performance by reducing the response time. Modify the application as a multi threaded program which can load the array list in X/2 seconds.  **Hint for problem statement # 2:**   * Spawn two threads to store the values in the array list. * Thread 1 should load values from 0….50 lakh * Thread 2 should load values from 50 lakh….100 lakh * Use a Arraylist variable which can be shared by two threads to load the values.  1. Develop a class with name LoaderThread using the Runnable interface.    1. Should have two instance variables Integer ***startNumber*** and Integer ***lastNumber***.    2. Overload the constructor which accepts two integer values ***startNumber & lastNumber*** and sets the value in the instance variables specified in the above point.    3. Override the run method and invoke the method loadList of the ListLoader class passing the values stored in instance variables ***startNumber &*** ***lastNumber***. 2. Develop a class MainThread which should use the LoaderThread spawn two threads using the overloaded constructor by passing  ***startNumber & lastNumber***, each thread to print the range of numbers as mentioned in Hint section.   The Response time in this case should be calculated. This is the time between the first thread getting started and the time at which both the threads completes and joins with the main thread.  **Hint:** Use threads join methods to ensure that two spawned thread completes execution before the main thread gets completed. |