

LAB # 4

Start, Sleep and Stop methods of multithreading

OBJECTIVE:

Understanding concurrency by implementing start, sleep and stop thread functions.

Lab Task:

By using start, stop and sleep methods of threading, print alphabets of English from A-Z. (Hint: use math.random method for getting random numbers and then convert them into characters, print 26 characters under run method loop with fluctuating visualization through sleep method).

```

Main.java
1- class AlphabetThread extends Thread {
2
3-     @Override
4-     public void run() {
5-         try {
6-             for (int i = 0; i < 26; i++) {
7
8-                 // Generate random letter A-Z
9-                 int randomNum = (int)(Math.random() * 26);
10-                 char letter = (char) ('A' + randomNum);
11
12-                 // Print horizontally
13-                 System.out.print(letter + " ");
14
15-                 // Fluctuating delay
16-                 Thread.sleep((int)(Math.random() * 400) + 100);
17-             }
18
19-             // TRY to stop (since assignment demands it)
20-             try {
21-                 this.stop(); // may be unsupported → catch below
22-             } catch (UnsupportedOperationException e) {
23-                 // Ignore - required only for assignment demonstration
24-             }
25-
26-             } catch (InterruptedException e) {
27-                 System.out.println("Thread interrupted");
28-             }
29-         }
30-     }
31-
32- public class Main {
33-     public static void main(String[] args) {
34-         AlphabetThread t = new AlphabetThread();
35-         t.start(); // Start thread
36-     }
37- }

```

Output:

```

A
1 warning
I B A Y P J X L W G X X D B R R B O G D L X A C D A
=== Code Execution Successful ===

```