

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).

```
package javaapplication20;

public class JavaApplication20 {

    public static void main(String[] args) {
        Thread alphabetThread = new Thread() {
            @Override
            public void run() {
                try {
                    for (int i = 0; i < 26; i++) {
                        int delay = (int) (Math.random() * 900) + 100;
                        char ch = (char) ('A' + i);
                        System.out.print(ch + " ");
                        Thread.sleep(delay);
                    }
                } catch (InterruptedException e) {
                    System.out.println("Thread interrupted!");
                }
            }
        };
        alphabetThread.start();
        try {
            alphabetThread.join();
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        System.out.println("\nAll alphabets printed successfully!");
    }
}
```

Output - JavaApplication20 (run)

```
run:
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
All alphabets printed successfully!
BUILD SUCCESSFUL (total time: 13 seconds)
```

Github repository

Screenshot:

