Class NewThread implements Runnable {
    Thread t; /* create a variable 't' of type 'thread'*/
    NewThread()
    {
        t = new Thread(this, "Demo Thread"); /*thread creation using a constructor with arguments as an instance of a class that implements the Runnable interface and name for thread*/
        System.out.println("Child thread: " + t); /*printing child thread name*/
        t.start(); /* Start the thread, will call run() method.*/
    }
    public void run()
    {
        try{
            /*Child thread (and hence this function) will be called twice for every execution of main thread's for loop and hence this for loop will iterate twice before the main thread wakes up.*/
            for(int i = 5; i > 0; i--)
            {
                System.out.println("Child Thread says : " + i);
                Thread.sleep(500);
            }
        } catch (InterruptedException e)
        {
            System.out.println("Child interrupted.");
        }
        System.out.println("Exiting child thread.");
    }
}

class ThreadDemo {
    public static void main(String args[]) {
        new NewThread(); /* create a new thread*/
        try {
            for(int i = 5; i > 0; i--)
            {
                System.out.println("Main Thread says : " + i);
                Thread.sleep(1000);
            }
        } catch (InterruptedException e) {
            System.out.println("Main thread interrupted.");
        }
        System.out.println("Main thread exiting.");
    }
}

Output:

Child thread: Thread[Demo Thread,5,main]
Main Thread: 5
Child Thread: 5
Child Thread: 4
Main Thread: 4
Child Thread: 3
Child Thread: 2
Main Thread: 3
Child Thread: 1
Exiting child thread.
Main Thread: 2
Main Thread: 1
Main thread exiting.