

```

package com.sxt.others;

/**
 * ThreadLocal:每个线程自身的存储本地或局部区域
 * get/set/initialValue
 * @author 江
 *
 */

public class ThreadLocalTest01 {
    //private static ThreadLocal<Integer> threadLocal=new ThreadLocal<>();

    //更改初始值
    // private static ThreadLocal<Integer> threadLocal=new ThreadLocal<Integer>()
    {
    //      protected Integer initialValue() {
    //          return 200;
    //      }
    // };

    private static ThreadLocal<Integer> threadLocal=ThreadLocal.withInitial(()->{
        return 20;
    });

    public static void main(String[] args) {
        //获取值
        System.err.println(Thread.currentThread().getName()+"--
>" +threadLocal.get());
        //设置值
        threadLocal.set(99);
        System.err.println(Thread.currentThread().getName()+"--
>" +threadLocal.get());

        new Thread(new MyRun()).start();
    }

    public static class MyRun implements Runnable{

```

```
@Override
public void run() {
    threadLocal.set((int)Math.random()*99);
    System.err.println(Thread.currentThread().getName()+"--
>" + threadLocal.get());
}

}

}
```