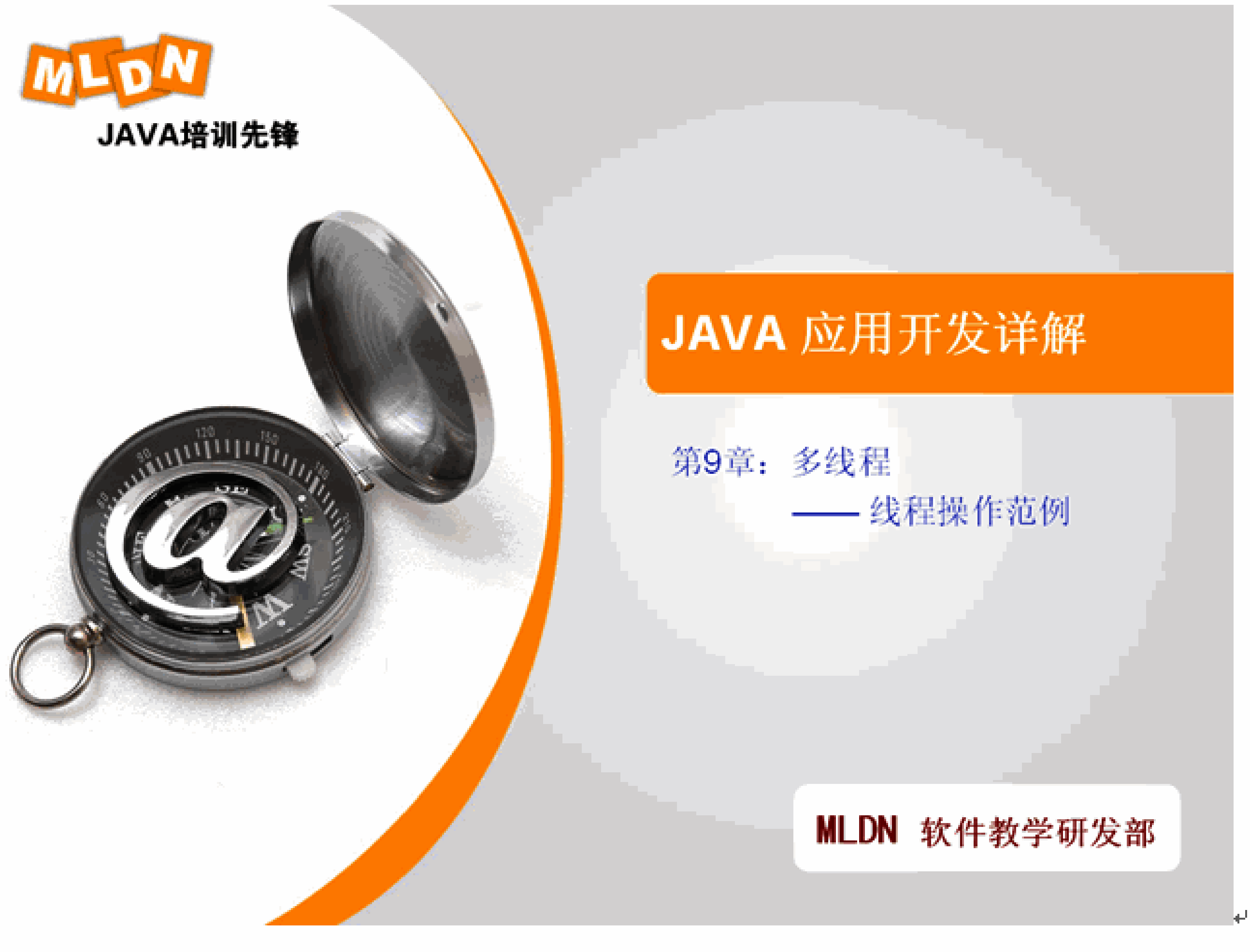
·1、课程名称：线程操作范例


MLDN JAVA培训先锋

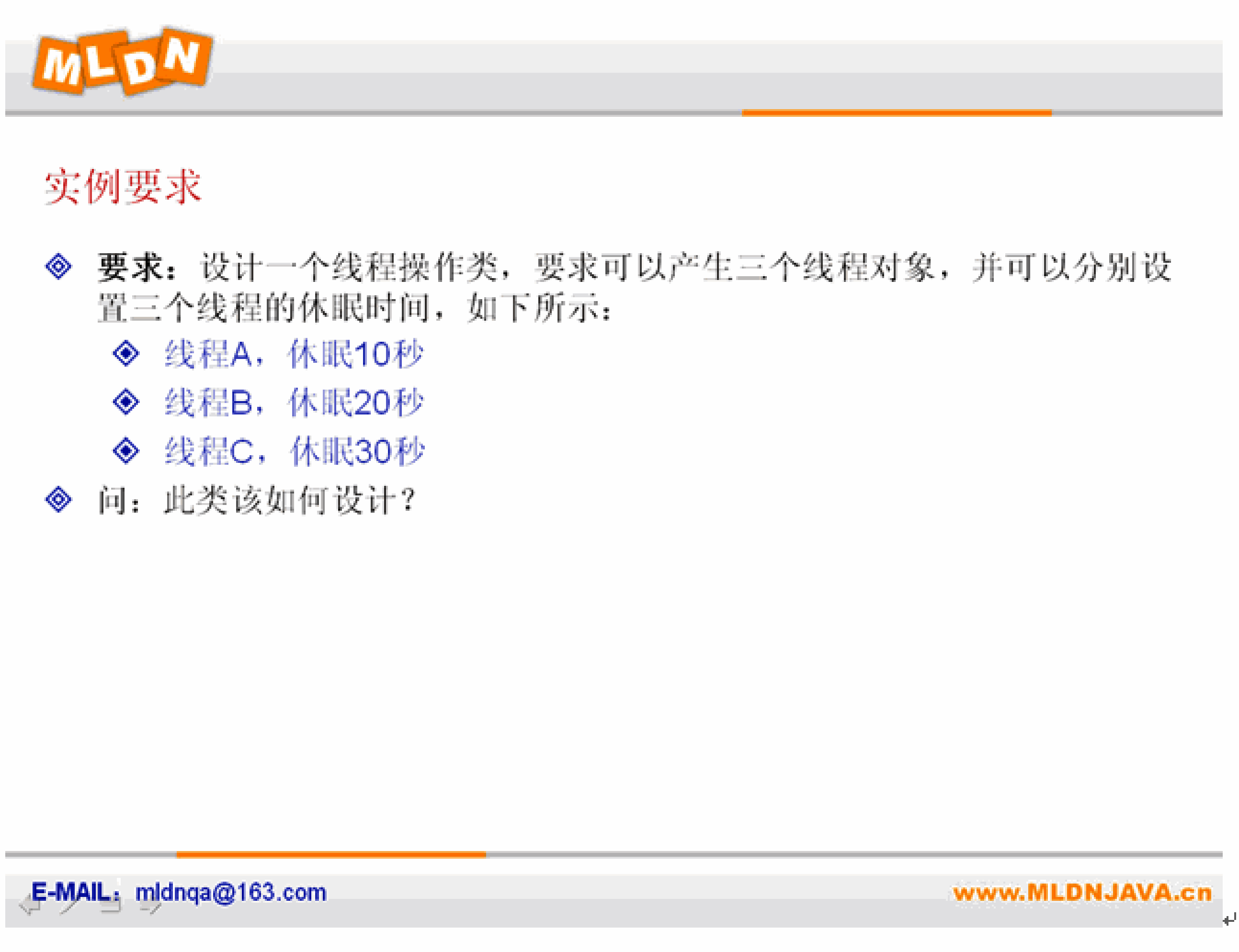
JAVA 应用开发详解

第9章：多线程

—— 线程操作范例

MLDN 软件教学研发部

·2、实例要求


实例要求

- **要求**：设计一个线程操作类，要求可以产生三个线程对象，并可以分别设置三个线程的休眠时间，如下所示：
  - 线程A，休眠10秒
  - 线程B，休眠20秒
  - 线程C，休眠30秒
- 问：此类该如何设计？

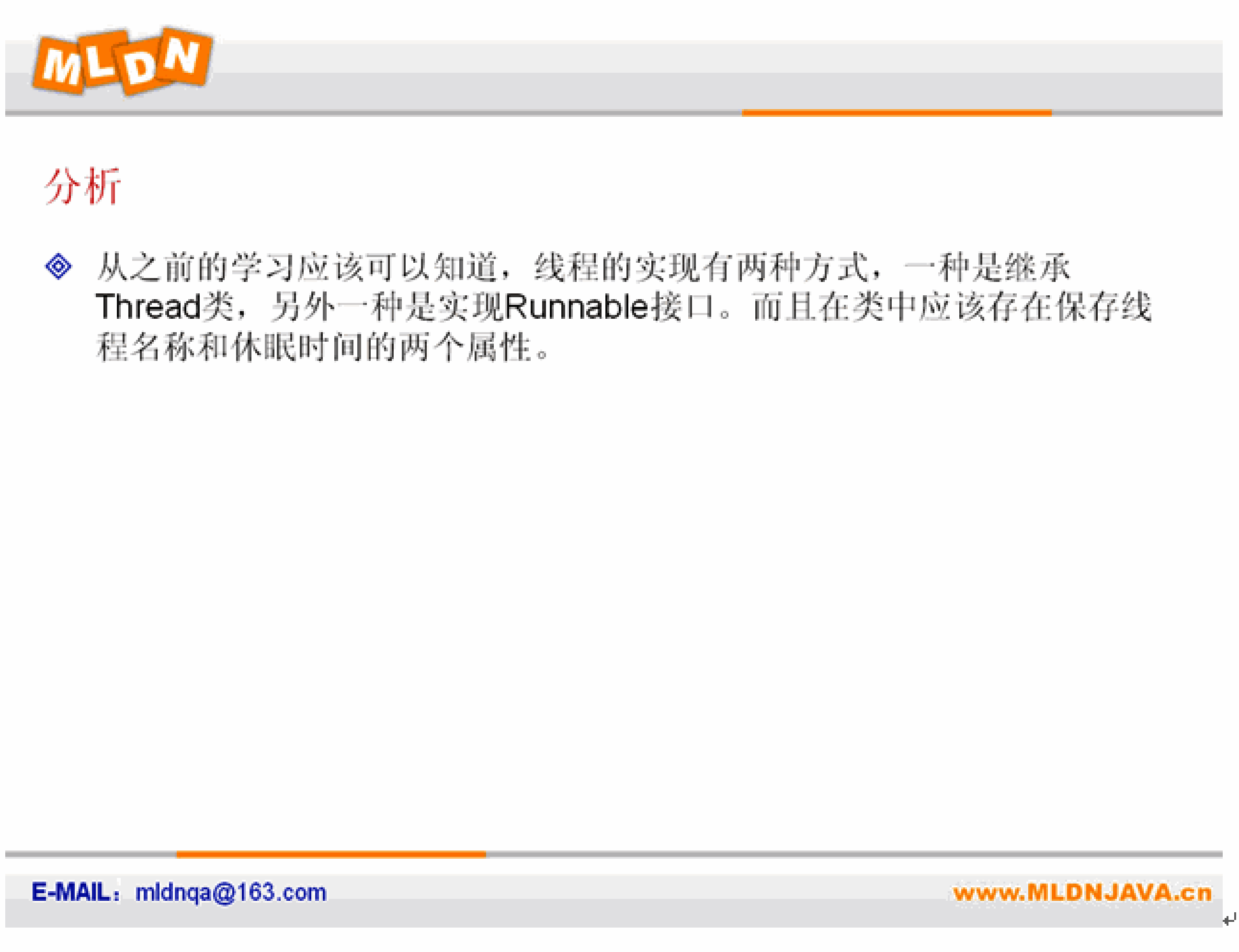
E-MAIL：mldnqa@163.com www.MLDNJAVA.cn

·3、本实例主要采用的知识

1、 Thread 类

2、 Runnable

·4、具体内容


分析

- 从之前的学习应该可以知道，线程的实现有两种方式，一种是继承Thread类，另外一种是实现Runnable接口。而且在类中应该存在保存线程名称和休眠时间的两个属性。

E-MAIL：mldnqa@163.com www.MLDNJAVA.cn

·4.1、使用 Thread 类

在 Thread 类中直接存在了 name 属性。

```
class MyThread extends Thread{
	private int time ;
	public MyThread(String name,int time){
		super(name) ;	// 设置线程名称
		this.time = time ;// 设置休眠时间
	}
	public void run(){
```

```
		try{
			Thread.sleep(this.time) ;	// 休眠指定的时间
		}catch(InterruptedException e){
			e.printStackTrace() ;
		}
		System.out.println(Thread.currentThread().getName() + "线程，休眠"
			+ this.time + "毫秒。") ;
	}
};
public class ExecDemo01{
	public static void main(String args[]){
		MyThread mt1 = new MyThread("线程 A",10000) ;// 定义线程对象，指定休眠时间
		MyThread mt2 = new MyThread("线程 B",20000) ;// 定义线程对象，指定休眠时间
		MyThread mt3 = new MyThread("线程 C",30000) ;// 定义线程对象，指定休眠时间
		mt1.start() ;	// 启动线程
		mt2.start() ;	// 启动线程
		mt3.start() ;	// 启动线程
	}
};
```

·4.2、使用 Runnable

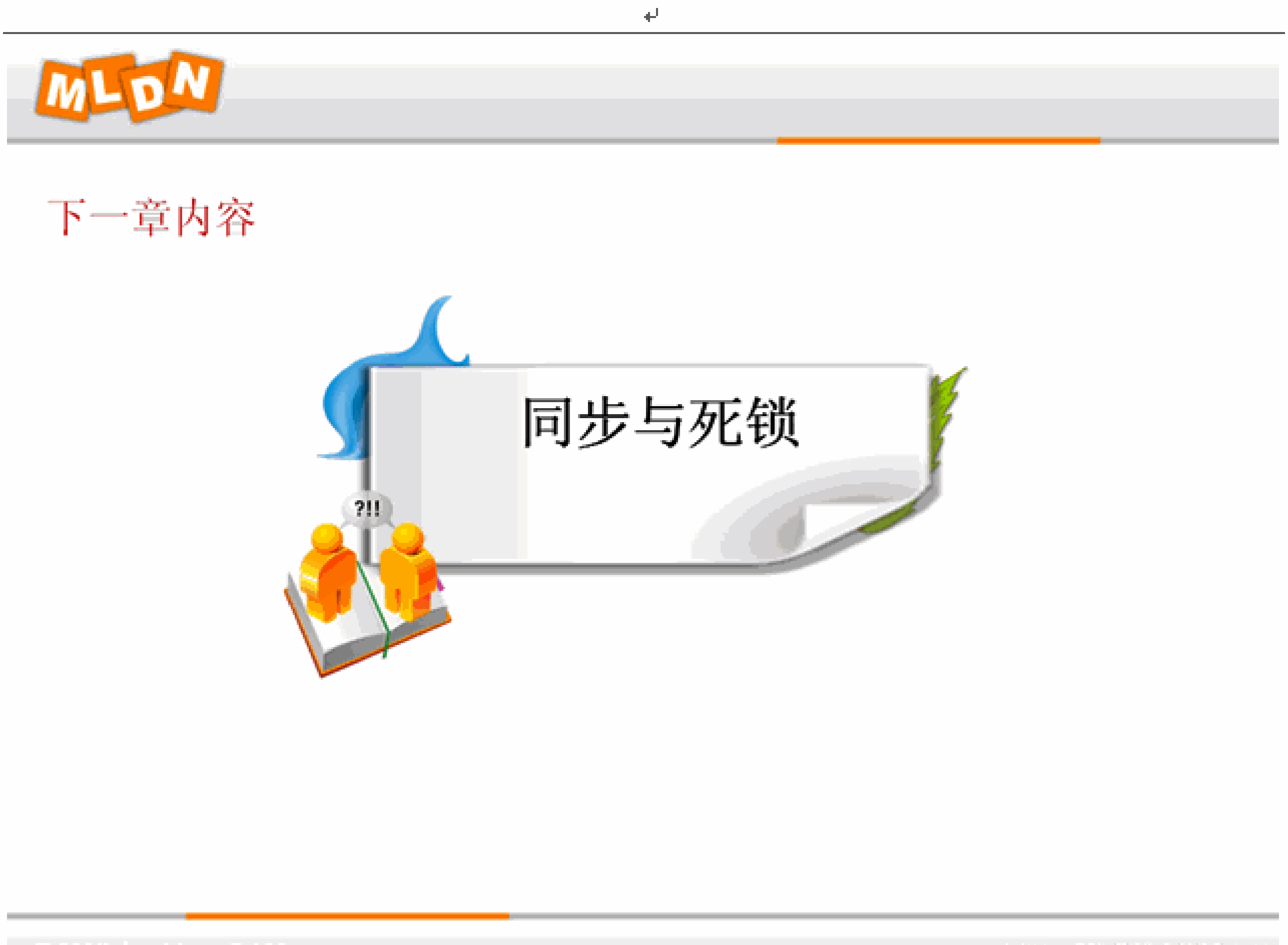
如果使用 Runnable 接口，则类中是没有线程名称存在的，所以应该单独建立一个 name 属性，以保存线程的名称。

```
class MyThread implements Runnable{
	private String name ;
	private int time ;
	public MyThread(String name,int time){
		this.name = name ;	// 设置线程名称
		this.time = time ;// 设置休眠时间
	}
	public void run(){
		try{
			Thread.sleep(this.time) ;	// 休眠指定的时间
		}catch(InterruptedException e){
			e.printStackTrace() ;
		}
```

```
		System.out.println(this.name + "线程，休眠"
			+ this.time + "毫秒。") ;
	}
};
public class ExecDemo02{
	public static void main(String args[]){
		MyThread mt1 = new MyThread("线程 A",10000) ;// 定义线程对象，指定休眠时间
		MyThread mt2 = new MyThread("线程 B",20000) ;// 定义线程对象，指定休眠时间
		MyThread mt3 = new MyThread("线程 C",30000) ;// 定义线程对象，指定休眠时间
		new Thread(mt1).start() ;	// 启动线程
		new Thread(mt2).start() ;	// 启动线程
		new Thread(mt3).start() ;	// 启动线程
	}
};
```

·5、总结

线程的实现方式及区别。线程的休眠操作。


下一章内容

同步与死锁

E-MAIL：mldnqa@163.com www.MLDNJAVA.cn