Java 多线程之 Join

作者：hades
原文链接：https://ld246.com/article/1516612755991
来源网站：链滴

在开发中，有些功能代码执行的过程需要依次完成的，这就涉及到了一个顺序的问题，例如有三个程序分别为 Thread1 ，Thread2，Thread3 这三个线程类怎么依次执行完毕呢。

Thread1的代码如下：

```
public class Thread1 implements Runnable {

@Override

public void run() {
try {
System.out.println("Thread1开始执行");
Thread.sleep(300);
System.out.println("Thread1执行完毕");
} catch (InterruptedException e) {
e.printStackTrace();
}
}
}
```

Thread2的代码如下：

```
public class Thread2 implements Runnable {

@Override

public void run() {
try {
System.out.println("Thread2开始执行");
Thread.sleep(300);
System.out.println("Thread2执行完毕");
} catch (InterruptedException e) {
e.printStackTrace();
}
}
}
```

Thread3的代码如下：

```
public class Thread3 implements Runnable {

@Override
```

```
public void run() {
try {
System.out.println("Thread3开始执行");
Thread.sleep(300);
System.out.println("Thread3执行完毕");
} catch (InterruptedException e) {
e.printStackTrace();
}
}
}
```

线程启动类代码如下

```
public class MainTest {
public static void main(String[] args) throws InterruptedException {
Thread thread1 = new Thread(new Thread1());
Thread thread2 = new Thread(new Thread2());
Thread thread3 = new Thread(new Thread3());
thread1.start();
thread2.start();
thread3.start();
}
}
```

运行main方法，第一次结果如下

```
Thread2开始执行
Thread3开始执行
Thread1开始执行
Thread1执行完毕
Thread2执行完毕
Thread3执行完毕

Process finished with exit code 0
```

第二次结果

```
Thread1开始执行
Thread3开始执行
Thread2开始执行
Thread1执行完毕
Thread3执行完毕
```

```
Thread2执行完毕

Process finished with exit code 0
```

并没有按照预想的结果来执行代码。
查看API java.lang.Thread 类有三个join()方法，大概意思是等待该线程终止。

在main方法中加入以下代码

```
public class MainTest {
public static void main(String[] args) throws InterruptedException {
Thread thread1 = new Thread(new Thread1());
Thread thread2 = new Thread(new Thread2());
Thread thread3 = new Thread(new Thread3());
thread1.start();
thread1.join();
thread2.start();
thread2.join();
thread3.start();
thread3.join();

}
```

}

然后再次运行

```
Thread1开始执行
Thread1执行完毕
Thread2开始执行
Thread2执行完毕
Thread3开始执行
Thread3执行完毕
```

结果是不是对了呢 。