

Servlet的线程安全问题

1、程序出现线程安全问题的情况

满足下列条件：

--多线程并发

--有共享的数据

--共享的数据有修改的情况

2、在JVM中可能在线程安全的数据

--所有线程是共享一个堆的，所以：

- * 由于堆内存中存储了new出来的对象，该对象中有"实例变量"，所以"实例变量"的内存多线程共享。当"实例变量"被共同访问，且涉及到修改操作时存在线程安全问题

--所有线程共享一个方法区

- * 方法区中的静态变量被多线程共享，涉及到修改操作时存在线程安全问题
方法区的常量是修改不了的，不存在线程安全问题

--一个线程占用一个栈内存

- * 局部变量是存储在栈中的，所以局部变量内存空间不会被共享，没有线程安全问题

3、线程安全不仅仅存在于JVM中，也发生在数据库中，例如：多个线程共享一张表，如果这些线程都有修改操作就会存在线程安全问题

解决数据库中的线程安全问题：

- 第一种：在Java中使用synchronized关键字，线程排队执行，数据库中不会并发
- 第二种：行级锁（悲观锁）
- 第三种：事务隔离级别，使其串行化
- 第四种：乐观锁
- ...

4、Servlet中也有线程安全问题

--一般怎么解决线程安全问题？

- * 完成需求尽量用局部变量，不使用成员变量
- * 若必须使用成员变量，考虑将该对象变成多例对象，使其一个线程一个java对象，实例变量的内存也不会被共享
- * 若必须使用单例，用synchronized关键字让线程排队执行

--解决Servlet的线程安全问题

首先我们要知道Servlet是在单实例多线程的环境下运行的，所以只能用两个方式解决

- * 完成需求尽量用局部变量，不使用成员变量
- * 用synchronized关键字让线程排队执行