属性

1.属性是什么？Attribute

属性就是一个对象，设置（绑定）到另外三个servlet api对象中：ServletContext、HttpServletRequest（或ServletRequest） 、HttpSession

可以简单当做一个键值对，键是String，值是Object

实际我们只关心属性的**作用域**和**生命周期**，不关注具体实现

2.属性不是参数！

注意和上下文/应用初始化参数、请求参数、servlet初始化参数区别开

属性的设置方法是setAttribute(String name, Obejct value)

而参数除了请求参数，都在DD中配置，请求参数来自Web页面的提交

属性的返回类型是Obejct，参数的返回类型是String

属性获取getAttribute(String name),参数获取 getInitParameter(String name)

3.三个作用域：上下文、请求、会话

1）Context上下文：应用中每一部分都能访问，通常是数据库连接、并发用户的计数和email信息等具有全局意义的属性对象

2）HttpSession会话：能访问特定HttpSession的部分才能访问

3）Request请求：能访问特定ServletRequest的部分才能访问

4.属性API

3个属性作用域分别由ServletContext、ServletRequest和HttpSession接口处理，每个接口中有关于属性的API方法完全相同

Object getAttribute(String name)

void setAttribute(String name, Object value)

void removeAttribute(String name)

Enumeration getAttributeNames()

5.上下文作用域不是线程安全的

应用中的多个部分都能访问上下文属性，这意味着可能有多个servlet并发访问某个属性，这会产生线程安全问题，更何况每个servlet还会有多个线程，每个请求在一个单独的线程栈中处理。

6.上下文属性线程安全的尝试

1）public **synchronized** void doGet

分析：这样做只是解决了同一个servlet中不同线程的线程安全问题，但是解决不了不同servlet的线程安全问题

2）因此，不是需要对servlet加锁，而是要对上下文加锁，比如把上下文（Web应用只有一个）作为同步对象来使用，保护业务代码 synchronized(getServletContext()) {…}

7.会话属性是线程安全的吗？

1）会话是一个对象，用于维护与**一个客户**的会话状态

2）但是当一个用户有多个请求时，会话中将会有多个线程并发访问会话属性

3）同样的，要把session对象作为同步对象 synchronized(session){…}

8.只有请求属性和局部变量是线程安全的