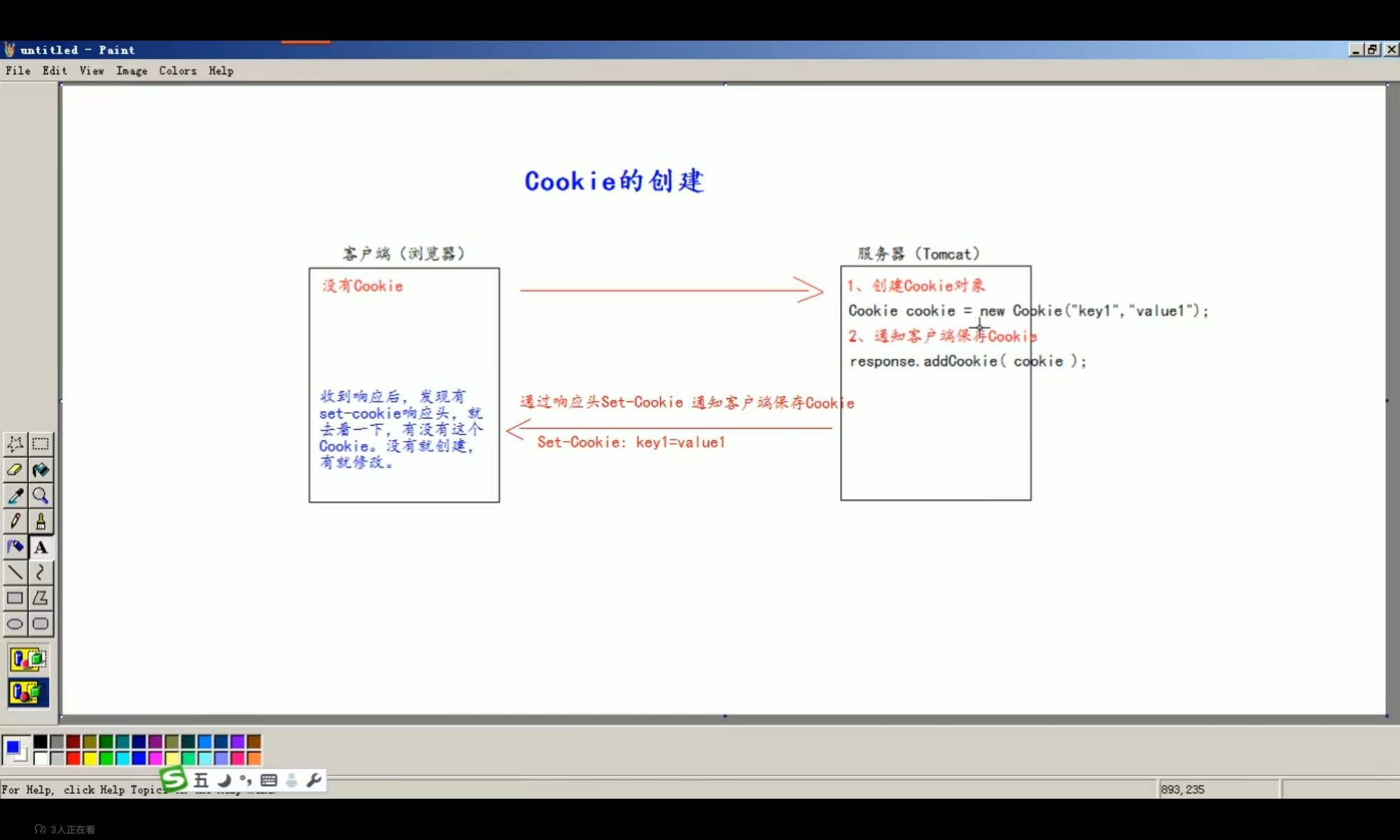
1. Cookie
2. 什么是Cookie?
3. Cookie翻译过来是饼干的意思
4. Cookie是服务器通知客户端保存键值对的一种技术
5. 客户端有了Cookie后，每次请求都发送给服务器
6. 每个Cookie的大小不能超过4kb
7. 如何创建Cookie



1. 如何获取Cookie
2. Cookie值的修改

方案一：

1. 先创建一个要修改的同名cookie对象
2. 在构造器，同时赋予新的Cookie值
3. 调用response.addCookie(Cookie):

Cookie cookie=new Cookie(“key1”,”newValue1”)

Resp.add(cookie)；

方案二：

1. 先查找到需要修改的Cookie对象
2. 调用setValue()方法赋予新的Cookie值
3. 调用response.addCookie()通知客户端保存修改
4. Cookie cookie= Cookieutils.*findCookie*(**"key1"**,cookies);

If(cookie!=null){

Cookie.setValue(“newValue2”);

Resp.addCookie(cookie);

}

1. Cookie生命控制

Cookie的生命控制是指如何管理cookie什么时候被销毁（删除）

setMaxAge()

正数，表示在指定的秒数后过期

负数，表示浏览器一关，cookie就会被删除

零，表示马上删掉

1. Cookie有效路径Path的设置

Cookie的path属性可以有效的过滤哪些cookie可以发送给服务器，哪些不可以

Path属性是通过请求的地址来进行有效的过滤。

cookieA path=/工程路径

cookieB path=/工程路径/abc

请求地址如下：

<http://ip:port/工程路径/a.html>

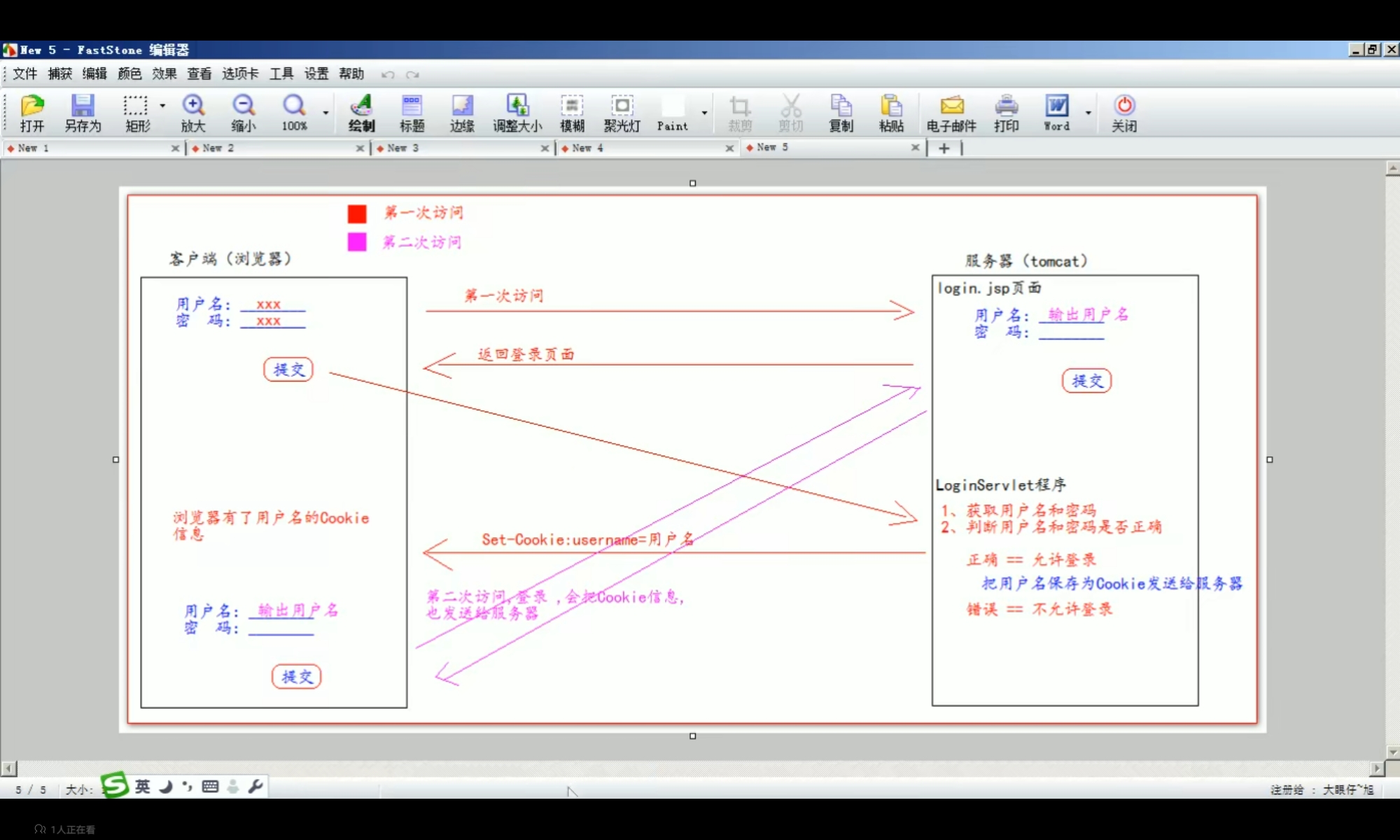
CookieA发送

CookieB不发送

<http://ip:port/工程路径/abc/a.html>

CookieA发送

CookieB发送

1. 用户免密登录
2. session会话

i)什么是session会话？

1.session就是一个接口（HttpSession）

2.Session就是会话。它用来维护一个客户端和服务器之间关联的一种技术

3.每个客户端都有自己的一个session会话

4.session会话中，我们经常用来保存用户登录之后的信息

j)如何创建session和获取id号（是否为新）

request.getSession()

第一次调用是：创建session会话

之后调用都是：获取前面创建好的session会话对象。

IsNew();判断到底是不是刚创建出来的（新的）

true表示刚创建

false表示获取之前创建的

每一个会话都有一个身份id，而且这个id是唯一的

getId（）得到Session的会话id值

k)session域数据的存取

l)session生命周期控制

**public void** setMaxinactiveInterval(**int** interval) 设置Session的超时时间（以秒为单位）超过指定的时长，Session就会被销毁

值为正数的时候，设定Session的超时时长

值为负数的时候表示永不超时（极少使用）

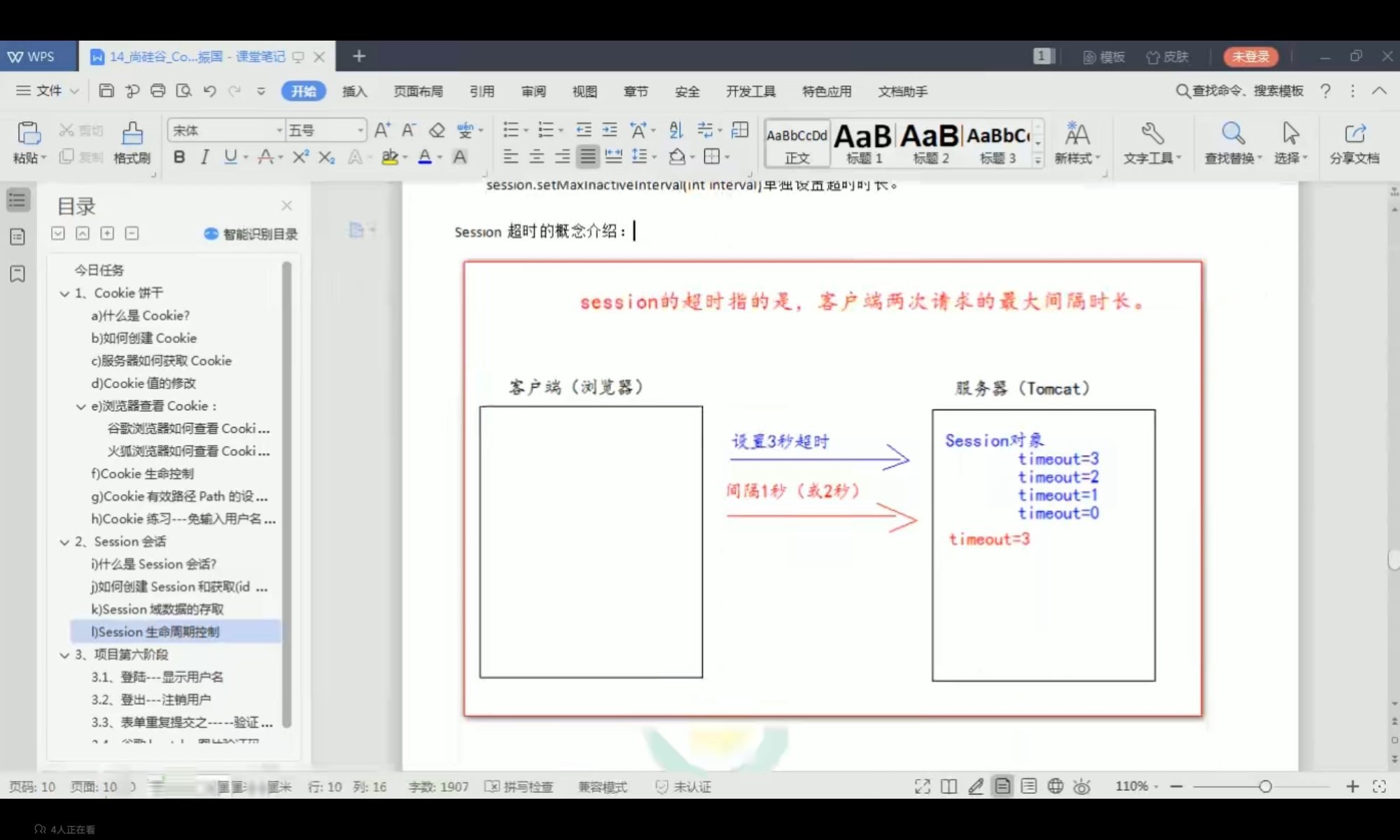
**public void** getMaxinactiveInterval() 获取Session的超时时间

public void invalidate()让当前Session会话马上超时无效

<**session-config**>  
 <**session-timeout**>20</**session-timeout**>  
</**session-config**>

session的默认超时时长为30分钟，如果想要把超时时长换成其他时间，你可以在你自己的web.xml配置文件中做以上相同的配置，就可以修改你wewb工程所以session的默认超时时长

如果你想只修改个别Session的超时时长，就可以使用上面的API，**public void** setMaxinactiveInterval(**int** interval)来进行单独的设置



m)浏览器和服务器session之间关系的技术内幕

session技术，底层技术就是基于cookie实现的

