---------------------------------------------------

计划课程安排：

FUNDAMENTAL(7天)     该阶段已结束

OOP(11天)         该阶段已结束

JAVASE01(10天)        该阶段已结束

JAVASE02(12天)       该阶段已结束

WEBBASIC(11天)       该阶段已结束

JQUERY(2天)         该阶段已结束

DATABASE(5天)       该阶段已结束

JDBC(2天)         该阶段已结束

SERVLETJSP(13天) ←当前为该阶段第7天

SPRINGMYBATIS01(8天)

AJAX(2天)

~~SPRINGMYBATIS02(12天)~~ （该阶段已被取消）

~~LINUX(2天)~~ （该阶段已被取消）

~~NGINX(2天)~~ （该阶段已被取消）

~~REDIS(2天)~~ （该阶段已被取消）

~~T-STORE(5天)~~ （该阶段已被取消）

预计合计：100天

预计剩余：20天 （17天正课+3天自习）

注：上述信息随时可能发生更改，仅供参考。

---------------------------------------------------

**2019年6月4日**

**Java就业班-第80天**

**8:30实训室2-早自习**

今日晨讲主题——内部类和Lambda表达式

这位老哥的一手lambda表达式写的还是很溜的，学习到了。

**9:03实训室2-正课**

开始上课。

课程紧接昨天内容。

### 6)cookie的编码问题

a.什么是cookie的编码问题?

cookie只能存放合法的ascii字符，对于中文，需要将其转换成

合法的ascii字符的形式。

b.如何处理?

在添加cookie时，使用encode方法来编码

String URLEncoder.encode(String str,String charset);

在读取cookie时，使用decode方法来解码

String URLDecoder.decode(String str,String charset);

c.建议，在添加cookie时，不管是否是中文，都统一使用encode方法

来处理。

### 7)cookie的路径问题

a.什么是cookie的路径问题?

浏览器在向服务器发送请求时，会比较请求地址是否

符合cookie的路径要求，只有符合要求的cookie才会被

发送。

b.Cookie的默认路径

cookie都有一个默认路径，默认路径等于添加该cookie的

web组件的路径，比如/servlet06-lab2/biz01/addCookie.jsp

添加了一个cookie,则该cookie的默认路径就是"/servlet06-lab2/biz01"。

c.哪一些cookie会被发送?

请求地址要么等于cookie的路径，要么是其子路径。

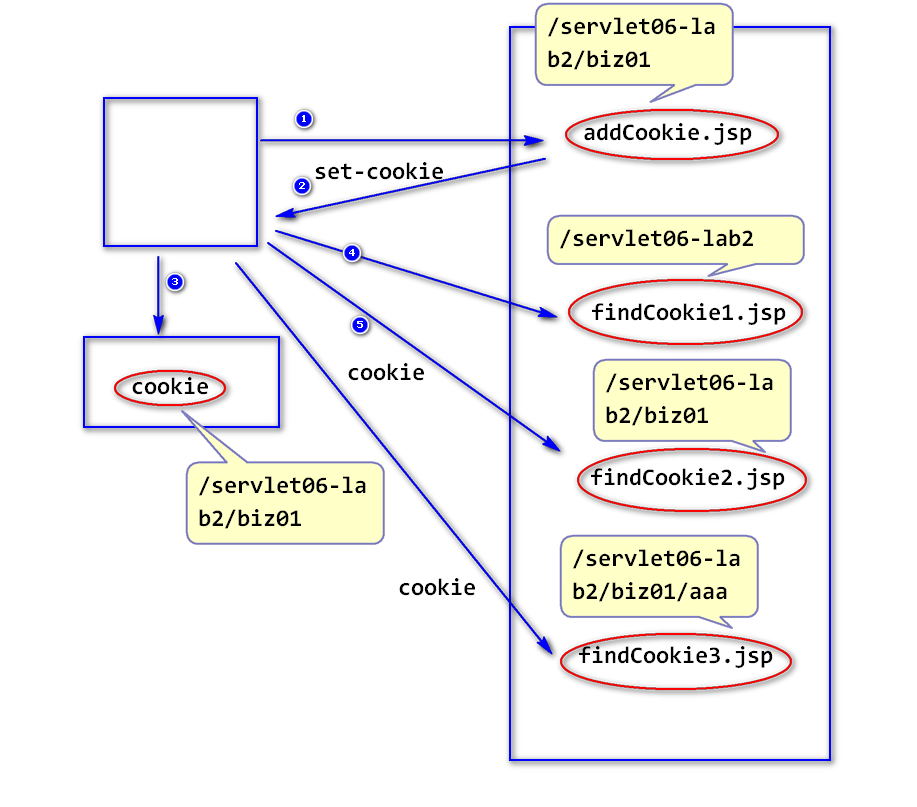
比如，cookie的路径是"/servlet06-lab2/biz01",则:

/servlet06-lab2/findCookie1.jsp no

/servlet06-lab2/biz01/findCookie2.jsp yes

/servlet06-lab2/biz01/aaa/findCookie3.jsp yes

![](path.png)



d.可以修改cookie的默认路径

cookie.setPath(String path);

### 8)cookie的限制

a. cookie可以被用户禁止。

b. cookie不安全。

对于敏感数据，一定要加密处理。

c. cookie只能存放少量数据。

大约4k左右。

d. cookie的数量也有限制。

大约几百个。

e. cookie只能存放字符串。

# 2.Session(会话)

## (1)session是什么?

服务器为了保存用户状态而创建的一个特殊的对象，该对象一般称之为

session对象。

## (2)session工作原理

当浏览器第一次访问服务器时，服务器会创建一个session对象

(该对象有一个唯一的id,一般称之为sessionId),接下来服务器

会将sessionId以cookie的方式发送给浏览器。

当浏览器再次访问服务器时，会将sessionId发送过来，服务器

就可以利用sessionId找到对应的session对象。

## (3)如何获得session对象?

方式一:

HttpSession s = request.getSession(boolean flag);

注：

当flag为true时，先查看请求当中是否有sessionId,如果没有

sessionId,则创建一个session对象；如果有sessionId,则

依据该sessionId去查找对应的session对象，如果找到了，则

返回该session对象，如果没有找到，则创建一个新的session对象。

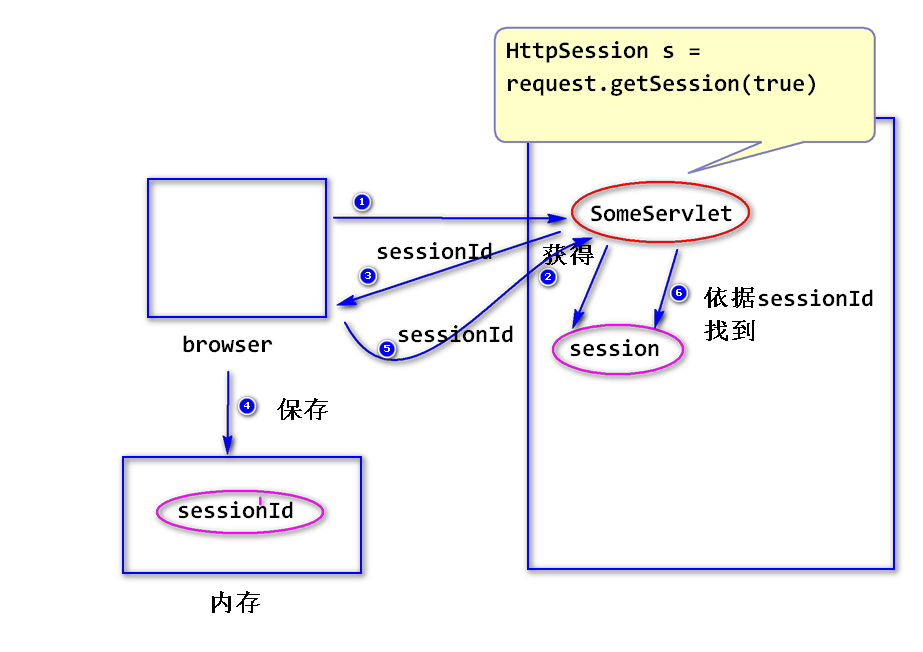
当flag为false，先查看请求当中是否有sessionId,如果没有，

返回null;如果有sessionId,则依据该sessionId去查找对应的

session对象，如果找到了，则返回该session对象，如果没有找到,

返回null。

![](session.png)



方式二:

HttpSession s = request.getSession();

等价于 request.getSesion(true)。

## (4)使用session进行状态管理

a.将数据绑订到session对象上:

session.setAttribute(String name,Object obj);

b.依据绑订名获得绑订值:

Object session.getAttribute(String name);

c.解除绑订:

session.removeAttribute(String name);

## (5)session超时

a.什么是session超时?

服务器会把空闲时间过长的session对象从内存当中删除。

为了节省内存空间。

注：

空闲时间一般是半个小时。

b.如何修改超时时间长度?

方式一: 修改web.xml

<session-config>

<session-timeout>30</session-timeout>

</session-config>

注:

每一个web应用都有一个web.xml，另外，容器也有一个

web.xml。

方式二: session.setMaxInactiveInterval(int seconds);

注:

用于设置两次请求之间最大的间隔时间长度，如果超过这个

时间长度，服务器会将session删除。

## (6)删除session

session.invalidate();

# session验证

step1.登录成功之后，在session对象上绑订一些数据，比如:

session.setAttribute("username","Tom");

step2.当用户访问需要保护的资源时(即只有登录成功之后才能访问的

资源，比如success.jsp),则进行session验证。

Object obj = session.getAttribute("username");

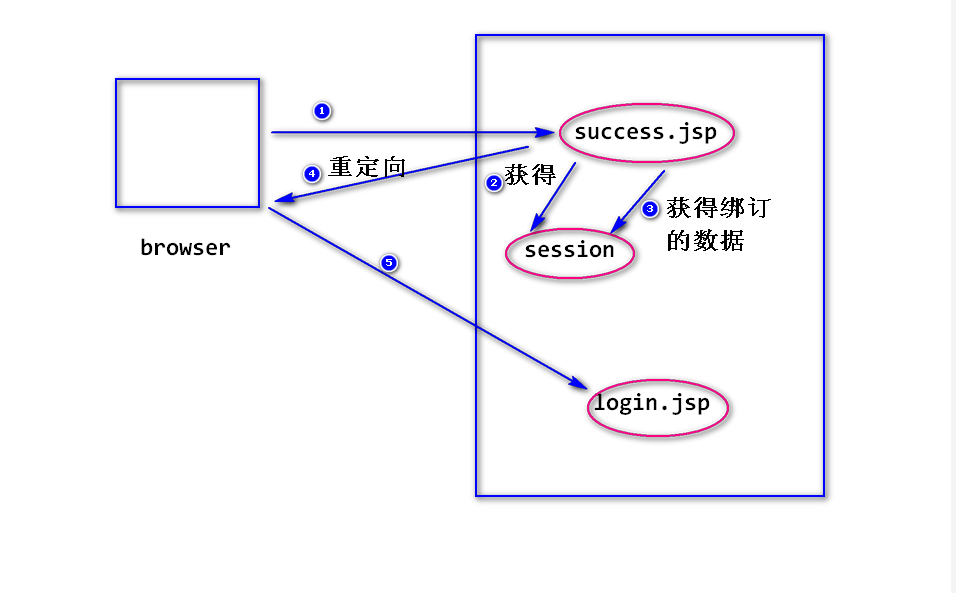
if(obj == null){

response.sendRedirect("login.jsp");

return;

}

![](session\_check.png)



# 练习：

写一个Servlet,输出用户上一次访问的时间。

如果是第一次访问，则输出"你是第一次访问"。

代码实现：

package web;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.Date;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

/\*

\* 练习：写一个Servlet,输出用户上一次访问的时间。

\* 如果是第一次访问，则输出第一次访问提示信息。

\*/

public class OtherServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

protected void service(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

PrintWriter out = response.getWriter();

request.setCharacterEncoding("utf-8");

response.setContentType("text/html;charset=utf-8");

HttpSession session = request.getSession();

Date date = (Date)session.getAttribute("date");

if(date == null) {

out.println("本次是第一次访问");

}else {

out.println("上次访问时间：" + date);

}

//将当前访问的时间保存下来

session.setAttribute("date", new Date());

}

}

**17:35实训室2-下课时间**

以上为今日课程内容。

**19:20实训室2-晚自习**

教室太热回宿舍。

**20:30宿舍-休息**

今天的内容不是很多，主要是讲解概念性的内容，虽然代码也有写，但是我想了一下觉得这种代码也没有什么贴的必要了，反正我都会上传到github上。

因为课程讲的比较慢，利用闲暇时间把IDEA破解版安装好了，虽然能够很直观的感觉到IDEA的高大上吧……我完全不会用啊！好多需要配置的地方，是真的麻烦啊，而且很多地方与我在eclipse上养成的编码习惯有冲突，如果说以后工作基本上都是用IDEA的话，似乎我得专门抽出时间来研究一下IDEA的使用教程了……

上午11点左右，项目经理找上我单独谈话，谈话时间也就几分钟，主要是为了提醒一下我上次摸底考试的基础选择题错了很多，觉得我的基础学的很差，这样的话没办法应付月底的面试题。同时还提到其他学员很多都是一题都没错（选择题），我和他们之间的差距很大……

我表示，考试的时候也没有监考，大多数人都是直接抄的答案，而我是自己想自己写的答案，比不过其他学员很正常。随后我装了一下B，表示如果面试题都是靠基础知识的话，我反倒是完全不担心了，因为基础知识的复习只需要在面试题考试的前几天背一背就没问题了。

项目经理苦笑了一阵……

随后提到这个月底的前3天左右就会有企业来招人，这个事情基本不会变，所以从现在开始就得加紧复习了。另外，晚自习项目经理通常都会进行一些面试题的讲解，所以提到要我晚自习不要再旷课了。（那恐怕是不太可能，只要室温超过35度，我是绝对不会上晚自习的。）

虽然说B是装足了，但我的基础水平自己也非常的清楚，所以对于复习我也有自己的安排，就是不知道实际效果如何了。

话说这两天都没看到过班主任的人啊，群里消息也不发了，看样子是不用再刷存在感了啊。还是说自从教室门上贴了一张“脱产班”的标签之后工作就交接了？

不管怎么样吧，都快结束了。

今日感想总结：课程进度开始越来越慢了，看来我得放弃机构的课程进度，自己看视频自学了。（我在网上找到了机构的全套授课视频……学费打水漂了，Shit!）

今日评分：-10/10分

今天就到这里，明天继续投稿。

感谢支持！

================================================================

**Acer程序员群：790482850**

**up主的java学习日记word文件及全部代码（随日记同步更新）：**

Adress：github.com/wt62635