

实验 6 session 对象的使用

一、实验目的

本实验的目的是让学生了解 session 对象及其常用方法,掌握如何通过 session 对象以属性的方式存、取用户相关的信息。

二、实验要求

要求学生使用 session 对象完成一个小游戏,掌握 session 对象的工作机制及操作方法。

三、相关知识

(1) Session 对象:是用来分别保存每一个用户信息的对象,以便于跟踪用户的操作状态。Session 的信息保存在服务端,Session 的 ID 保存在客户机的 Cookie 中。当用户登录站点时,系统就会为其建立一个 Session 对象,以便于区别其他用户。这个 Session 对象记录该用户的个人信息,而当该用户退出网站时,该 Session 对象就会随之消失。

(2) 主要方法:

- getId()

此方法返回惟一的标识,这些标识为每个 Session 而产生。

- setAttribute(String name , java.lang.Object value)

设置指定名字 name 的属性值 value,并将之存储在 Session 对象中。

- getAttribute(String name)

获取指定名字 name 相联系的信息。

- removeAttribute(String name)

删除与指定名字 name 的相联系的信息。

- getAttributeNames()

返回 Session 对象中存储的每一个属性对象,其结果为一个枚举对象。

- getCreationTime()

返回 Session 被创建的时间。最小单位为千分之一秒。

- getLastAccessedTime()

返回当前 Session 对象最后被客户发送的时间,最小单位为千分之一秒。

- setMaxInactiveInterval()

设置会话超时时间间隔。

- getMaxInactiveInterval()

返回总时间(秒),负值表示 Session 永远不会超时。该时间为该 Session 对象的生存时间。

- invalidate()

注销会话，使会话失效。

(3) Java 中随机数生成

- Math 库里的静态方法 random()

该方法的作用是产生 0 到 1 之间（包括 0，但不包括 1）的一个 double 值。

```
double rand = Math.random();
```

- 通过 Random 类的对象

```
Random rand = new Random();
```

```
int num=rand.nextInt(100)+1);
```

(4) 当前系统时间的获取：

```
java.util.Date now=new java.util.Date();
```

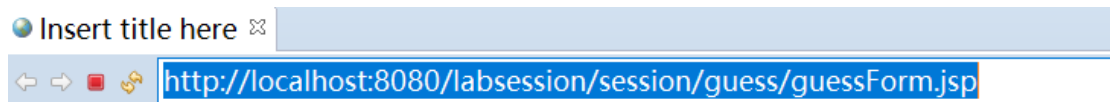
```
long time=now.getTime(); //返回的是以毫秒为单位的长整型值，表示的是距离  
1970 年 1 月 1 日 00: 00: 00 的毫秒数。
```

四、实验内容与步骤

利用 session 完成一个猜数字的游戏：

(1) 当客户端访问 guessForm.jsp 时，页面产生一个 1 ~ 100 的随机数，并将其存入 session 对象，然后让客户猜这个数字，页面如图 1 所示。

(2) 客户输入数后提交给 result.jsp，该页面将客户猜的数和产生的随机数进行比较，若猜对了，跳转到 success.jsp（图 4），显示客户猜数所用的时间等信息。若猜的数字大于随机数，连接到 large.jsp（图 2）；否则连接到 small.jsp（图 3）。这两个页面都要求客户重新猜数。



随机分给了你一个1到100之间的数字，请猜猜这个数

输入你猜的数

图 1 guessForm.jsp 页面

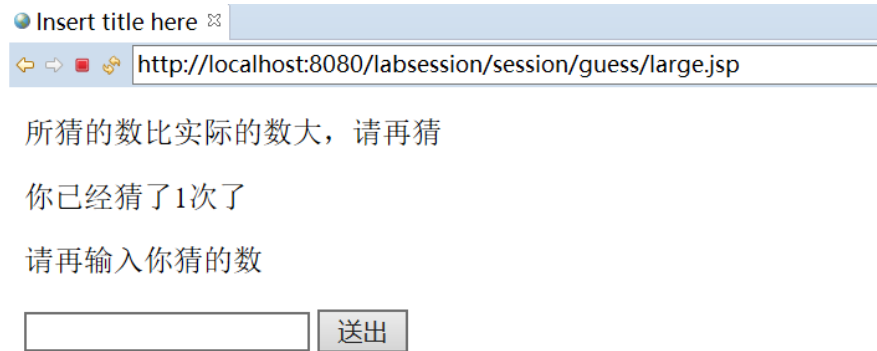


图 2 large.jsp 文件的运行效果图

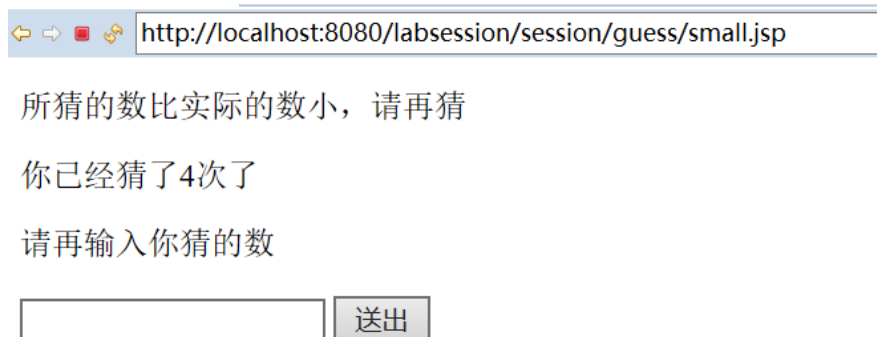


图 3 small.jsp 文件的运行效果图

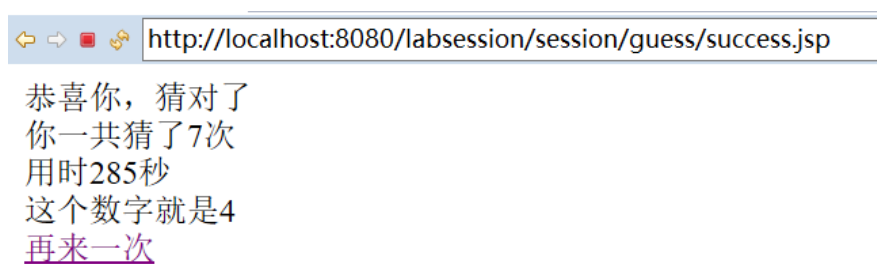


图 4 success.jsp 文件的运行效果图

五、实验作业的提交要求

- 1) 将实验程序的运行截图放到实验报告中。

2) 将源代码文件（或工程目录）打包，与实验报告文件一起提交到 MOOC 系统中，
文件名的命名规则：学号_姓名。