2017-12-3 11:49:17

现在开始js任务3.

由于时间关系，我已经在手机端截图了。看来这个页面的逻辑也很简单吧，就是点击查看身份的时候显示玩家身份信息，点击擦掉身份的时候，隐藏信息并且在下一次点击查看身份的时候显示下一个玩家的身份。当全部身份都显示并隐藏之后，开始游戏。

做完这个任务之后，收获什么： 进一步学习页面间数据传递； 如何根据需要显示和隐藏视图； JS对DOM样式和内容进行更复杂的操作

* 需要的技能应该是：
* 根据需要显示和隐藏视图，这应该可以通过控制display:none等等来控制
* 用变量保存查看状态和当前查到几号身份
* 浏览器的本地存储
* 身份传递方式有 1、url方式传递 2、本页面不跳转，根据需要隐藏和显示文本内容和按钮
* 法官页面，可以查看所有的身份

浏览器的本地存储：

1. cookie
2. DHTML 行为 其中的一个行为:用户数据 userData 一个页面保存64kb的数据
3. Flash-JavaScript桥
4. Google的Gears浏览器插件 无限大小的数据
5. Web Storage HTML5储存 独立的标准，被叫做本地储存Local Storage DOM储存DOM Storage

看来这个文章主要讲的是Web Storage

关键点：

* web页面以键值对的形式存在
* 客户端web浏览器将数据存储在本地
* 离开web服务器，数据依然保存
* 不会被发送到远程web服务器(除非手动发送)
* 几乎所有的主流浏览器都支持

有点长，慢慢看完再继续。

22:58:04

经过一天的低效率编写……

结果是这样的。

* 在总人数上面输入数量，几乎自动计算出杀手数和水民数
* 按开始游戏，数量传递给新的页面
* 根据随机出来的数组，分配角色以及显示相应的文本

是这样做的。

首先，使用Storage中的sessionStorage保存玩家数量，然后根据玩家数量，生成一个随机数组，根据这个数组来分配角色。

接下来，显示和隐藏方块，当然是使用display的变更来做。当然，我这样的菜鸟，写出来的逻辑是很渣的，反正关键点在于“？”号玩家这个数字。接下来需要编写法官页面……

大概这样吧……

还有2个bug：

1. 游戏角色数组，不能大于10个角色，应该是哪里搞错了
2. 暂时不会使用JavaScript进行页面跳转。

2017-12-4 22:29:07

继续任务3.

关于页面跳转，可以使用

<script language="JavaScript">   
self.location='jb51.htm';   
</script>

指向需要跳转的页面即可

然后法官查看页面，使用DOM比较多，例如这三个标签的切换，分别写入js，js修改display即可。

接下来是这些数据都可以从sessionStorage里面获取。

最麻烦的是自动生成这个列表，要使用creatElement()和appendChild()等等操作DOM节点的方法。

差不多就这样了。