二、3月4日下午（13:00—17:30）：setTimeout的使用；JS处理事件；JS的复选框判断等；JS做表单验证、JS操作CSS：

1、定时器的使用：

1）setTimeout()、clearTimeout()：

2）setInterval()、clearInterval()：

3）思考：我想定时执行5次后，停止定时器，怎么办？

4）练习：以上次课程的作业（计算现在时间到某个节日的间隔）为基础，实现可以每秒刷新一次那个文本框的值；

5）练习：在页面中提供一个文本框（长度设置长一些）和一个按钮，当点击按钮时，把文本框中的文字以跑马灯方式滚动；说明：结合字符串的substring()方法，300毫秒修改一下文本框的值；

2、事件处理：

1）简单概念：事件、事件处理

2）键盘、鼠标及窗口的基本事件：掌握onclick、onkeypress、onload；

3）表单域的常用方法：focus()、blur()；

4）具体每个表单域有哪些事件：要记住常用的表单域及其事件的使用！

a、最常用的表单域的事件：onclick、onchange、onfocus、onblur、onmouseover、onmouseout；

b、常用的触发js的方式：按钮的onclick、超链接、表单域的onchange和onblur、表单的onsubmit；

c、补充：建议我们定义的js函数名：至少两个单词，如 doTest()、doSum()等等。

d、文本框、单选按钮、复选框、下拉列表框的常见JS的使用（编写代码演示）；

d1、文本框：value、de faultValue、onfucus；

d2、单选按钮取值；

d3、复选框判断是否选中；

d4、下拉列表框的选项个数、选项的text和value属性、动态增加选项；

5）练习：给出5个复选框带数字值、1个文本框和1个按钮；要求点击按钮后验证至少选中一个复选框，并计算所有选中复选框的值的和，写入到文本框中；

6）练习：给出6个复选框；其中有1个是主复选框，当选中主复选框时，将其它复选框都选中；取消选中主复选框时，其它复选框都取消选中；

3、纯JS实现div的显示和隐藏：

1）主要通过css来设置div的样式；

2）两个div配合，一个用于遮盖底层，另一个用于显示顶层内容；

4、<span>的innerText和innerHTML的使用；

5、表单验证（常用功能）：实际就是通过表单和表单域事件触发js验证函数的执行；

1）通常是两种方式：

a、对每个表单域单独验证（需要给每个表单域的onblur事件设置js函数），建议通过<span>显示错误信息；

b、在提交时统一验证（只要给表单的onsubmit事件设置js函数）；

c、可以通过字符的ASCII值确定字符是英文字母还是数字；

2）正则表达式：能够调用即可；

a、说明：是一个描述字符模式的对象；JS中的RegExp对象和String对象定义了使用正则表达式来执行强大的模式匹配和文本检索与替换函数的方法。

b、两种创建正则表达式的方式：

var pattern = /^\d{6}$/;

var pattern = new RegExp(“^\\d{6}$”); //注意字符串里的两个“\”

c、正则表达式中常见字符的含义（我也记不住）：

^ 匹配一个输入或一行的开头，/^a/匹配"an A"，而不匹配"An a"   
$ 匹配一个输入或一行的结尾，/a$/匹配"An a"，而不匹配"an A"   
\* 匹配前面元字符0次或多次，/ba\*/将匹配b,ba,baa,baaa   
+ 匹配前面元字符1次或多次，/ba+/将匹配ba,baa,baaa   
? 匹配前面元字符0次或1次，/ba?/将匹配b,ba   
(x) 提取匹配x的字符串   
x|y 匹配x或y   
{n} 精确匹配n次   
{n,} 匹配n次以上   
{n,m} 匹配n到m次   
[xyz] 字符集(character set)，匹配这个集合中的任一一个字符(或元字符)   
[^xyz] 不匹配这个集合中的任何一个字符

\ 做为转义

\d 匹配一个字数字符，/\d/ = /[0-9]/   
\D 匹配一个非字数字符，/\D/ = /[^0-9]/

\s 匹配一个空白字符，包括\n,\r,\f,\t,\v等   
\S 匹配一个非空白字符，等于/[^\n\f\r\t\v]/

\w 匹配一个可以组成单词的字符(含数字)，包括下划线，如[\w]匹配"$5.98"中的5，等于[a-zA-Z0-9]   
\W 匹配一个不可以组成单词的字符，如[\W]匹配"$5.98"中的$，等于[^a-zA-Z0-9]

d、演示邮编的正在表达式的使用；

3）练习：提供两个文本框，分别是邮箱地址和用户名；要求使用JS代码编写验证规则（不使用正则表达式），规则如下：

a、邮箱地址要有“@”和“.” ，“@”在“.”前面并且中间至少有一个字符；“@”前面至少有一个字符；“.”后面至少有一个字符；

b、用户名必须英文字符开头，只能包含字母和数字；

6、JS操作CSS：

1）通过style对象修改单个样式；

2）通过className属性修改复合样式；

7、作业：做一个注册页面，实现常见的验证（非空、基本格式验证等），参考页面效果如下：

