正课:

1. Date

2. Error

3. \*\*\*\*Function

\*\*\*\*\*闭包

1. Date

什么是: 保存一个日期或时间，提供操作日期和时间的API

何时: 只要存储或操作日期和时间

如何: （PHP中:日期与时间相关扩展->日期和时间->Date/Time函数）

创建: 4种:

1. 获得系统当前日期和时间:

var now=new Date();

强调: 只能获得客户端本地时间，无法获得服务器时间

2. 自定义日期和时间:

var now=new Date("yyyy/MM/dd hh:mm:ss");

问题1: 不同编程语言中存储日期的格式不一致!

问题2: 字符串不便于计算和大小比较

问题3: 文字会受时区影响

解决: 不用字，也不用数组/对象来表示时间

用毫秒数

Date对象的本质: 其中保存了1970年1月1日至今的毫秒数

为什么: 1. 便于计算和比较大小

2. 数字在所有编程语言中都统一支持

3. 数字不受时区影响: 因为数字是一个时间段

3. 将ms转化为客户端日期对象:

var date=new Date(ms);

\*\*\*总结: 今后，在网络间传输时间或在服务器存储时间，都用整数（毫秒数/秒数）

4. 复制一个日期对象:

为什么复制: 日期计算默认都直接修改原日期对象

何时复制: 如果希望同时保留计算前后的开始时间和结束时间时，就要先复制开始时间，再用副本计算结束时间

如何: var date2=new Date(date1);

API: 3句话

1. 单位: FullYear Month Date Day ——都没有s结尾

Hours Minutes Seconds Milliseconds ——都有s结尾

2. 每个单位都有一对儿getXXX()/setXXX()方法

其中:

getXXX()负责获取单位的数值

比如: now.getFullYear() 2017

now.getHours() 11

setXXX(n) 负责修改指定单位的数值

比如: now.setFullYear(2018)

now.setDate(30)

福利: setXXX(n)

其中:n可以是任意值!

setXXX(n)可自动调整时间进制!

特例: day没有setDay()! 因为day是推算出来的

3. 取值范围:

Month: 0~11

强调: js中的月份比现实中小1

读取或修改时，可能需要+1或-1修正

Date: 1~31

Day: 0~6 日~六

Hours: 0~23

Minutes/Seconds: 0~59

日期计算: 2种情况:

1. 两日期对象可直接相减: 结果是毫秒差!

2. 对任意单位做加减:

3步:

1. 先取值: var d=now.getDate();

2. 做加减: d+=30;

3. 放回去: now.setDate(d); //setXXX()可自动调整时间进制

问题: setXXX() 直接修改原日期对象

解决: 先复制副本，再用副本计算

简写: now.setDate(now.getDate()+30);