日期；---公共类dateUtil.js先将就用，以后再优化更好；

资金；

Js字符串字节长度；

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

Js日期：

<http://www.w3school.com.cn/js/js_obj_date.asp>

Js-date对象，

<http://www.w3school.com.cn/jsref/jsref_obj_date.asp>

定义日期，

var myDate=new Date()

设置一个日期，

var myDate=new Date();

myDate.setFullYear(2008,7,9);

alert(myDate);

//注意：表示月份的参数介于 0 到 11 之间。也就是说，如果希望把月设置为 8 月，则参数应该是 7。

//输出日期为2008-8-9的标准格式时间；

//日期+5; | 日期，月份，年的计算，都是这样；不使用下面麻烦的；

var myDate=new Date();

myDate.getDate()---获得日期值

myDate.setDate(myDate.getDate()+5);

~

getMonth()---月，getFullYear()---年；

getTime()---毫秒；

日期转字符串，---标准格式化的要自己写；

[toString()](http://www.w3school.com.cn/jsref/jsref_toString_date.asp);

toUTCString();

字符串转日期，

//应用，星期，

<script type="text/javascript">

var d=new Date()

var weekday=new Array(7)

weekday[0]="星期日"

weekday[1]="星期一"

weekday[2]="星期二"

weekday[3]="星期三"

weekday[4]="星期四"

weekday[5]="星期五"

weekday[6]="星期六"

document.write("今天是" + weekday[d.getDay()])

</script>

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

js数值： //数值比较或运算前，先转为number，再进行；

创建对象，

var myNum=Number(value);

把数字转换为字符串，结果的小数点后有指定位数的数字。

toFixed(len); ---四舍五入；

toFixed(0)，取整；

number的相加：---否则可能出错，js本身原理问题导致，必须如此处理下；

var mostAccount = Number(2.2);

var sumAccount = Number(9.7);

alert(Number((mostAccount +sumAccount).toFixed(2)) );

需要了引入math对象的相关方法；

Math.round(0.60)---四舍五入；

Math.random()

如何使用 random() 来返回 0 到 1 之间的随机数。

Math.max(5,7)---返回两个数中较大的一个；

Math.min(5,7)

Js中的数字相加；---先类型转换；

var sum = parseInt(num1) + parseInt(num2) + parseInt(num3) + parseInt(num4) + parseInt(num5);

/////////////////////////////////////////////////////////////////////////////////

dateUtil.js：---页面加载js文件，直接使用其中的方法；

页面，oa---contract---examinationstartenddetail.xhtml

<h:inputText styleClass=*"Wdate"* id=*"newstartdate"* onclick="WdatePicker()" value=*"#{contractApplyAction.financialExamination.signatureDate}"*

onchange=*"return changeredemptionDate(#{contractApplyAction.financialExamination.isMonthly});"*>

<f:convertDateTime pattern=*"yyyy-MM-dd"*></f:convertDateTime>

**function** changeredemptionDate(ismonthly){

//获得日期控件的值，| jsf控件,form名:

**var** thisstart=$('[id="examinationstartenddetailForm:newstartdate"]').val();

//天数+1，

**var** calculationDate=dateToStr(addDate(strToDate(thisstart),1));

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

dateUtil.js

//日期转字符串：

**function** dateToStr(mydate){

**var** day=mydate.getDate();

**if**(day<10){

**if**(mydate.getMonth()<9){

**return** mydate.getFullYear()+"-0"+(mydate.getMonth()+1)+"-0"+day;

}**else**{

**return** mydate.getFullYear()+"-"+(mydate.getMonth()+1)+"-0"+day;

}

}**else**{

**if**(mydate.getMonth()<9){

**return** mydate.getFullYear()+"-0"+(mydate.getMonth()+1)+"-"+day;

}**else**{

**return** mydate.getFullYear()+"-"+(mydate.getMonth()+1)+"-"+day;

}

}

};

//字符串转日期(格式：“2012-08-12 23:13:15”)：

**function** strToDate(str){

str = str.replace(/-/g,"/");

**return** **new** Date(str);

}

//求日期所在月的最后一天日期

**function** maxDate(newdate){

**var** a=**new** Date(newdate.getFullYear(),Number(newdate.getMonth()+1),1);

**return** **new** Date(a.getTime()-24\*3600\*1000);

}

//两个日期比较大小：

**function** firstGreatSecond(date1,date2){

**var** a=date1.getTime();

**var** b=date2.getTime();

**if**(a>=b)**return** **true**;

**return** **false**;

}

//天数相加减

**~~function~~** ~~addDate(firstDate,i){~~

**~~return~~****~~new~~** ~~Date(firstDate.getTime()+i\*24\*3600\*1000);~~

~~}~~

//月份相加减

**~~function~~** ~~addMonths(firstDate,i){~~

**~~var~~** ~~tempdate=~~**~~new~~** ~~Date(firstDate.getFullYear()+i/12,(firstDate.getMonth()+i%12),1);~~

~~tempdate=maxDate(tempdate);~~

**~~var~~** ~~realdate=~~**~~new~~** ~~Date(firstDate.getFullYear()+i/12,(firstDate.getMonth()+i%12),firstDate.getDate());~~

**~~if~~**~~(firstGreatSecond(tempdate,realdate))~~**~~return~~** ~~realdate;~~

**~~return~~** ~~tempdate;~~

~~};~~

//两个日期的天数差：

**function** daysNum(date1,date2){

**return** (date1.getTime()-date2.getTime())/(24\*3600\*1000);

}

//两个日期的月份差：

**function** monthsNum(date1,date2){

**return** (date1.getFullYear()-date2.getFullYear())\*12+date1.getMonth()-date2.getMonth();

}

/////////////////////////////////////////////

数值计算：对应于资金；

/////////////////////////////////////////////

Js，页面倒计时：

<div class=*"tbzdc"* id=*"borrow\_remainingTime"*>剩余时间：</div>

<script type=*"text/javascript"*>

/\*\*秒数\*/

**var** SysSecond;

/\*\*循环计算倒计时\*/

**var** InterValObj;

$(**function**(){

**var** borrowStatus = '${borrow.status}';

**var** nowTime = $("#nowTime").val();

/\*\*预发标,发标剩余时间\*/

**if**(borrowStatus=="2"){

//有效时间，天数；默认3；

**var** validTime = '${borrow.validTime}';

**var** borrow\_addtime\_hide = $("#borrow\_addtime\_hide").val();

//这里获取倒计时的起始时间

SysSecond = parseFloat(**new** Date(borrow\_addtime\_hide).getTime()/1000+3600\*24\*parseFloat(validTime))-parseFloat(**new** Date(nowTime).getTime()/1000);

//剩余时间

InterValObj = window.setInterval(SetRemainTime, 1000); //间隔函数，1秒执行

}

});

/\*\*

\* 倒计时

\*/

**function** SetRemainTime() {

**if** (SysSecond > 0) {

SysSecond = SysSecond - 1;

**var** second = Math.floor(SysSecond % 60); // 计算秒

**var** minute = Math.floor((SysSecond / 60) % 60); //计算分

**var** hour = Math.floor((SysSecond / 3600) % 24); //计算小时

**var** day = Math.floor((SysSecond / 3600) / 24); //计算天

$("#borrow\_remainingTime").html('剩余时间：<span class="red">'+day+'</span>天<span class="red">'+hour+'</span>时<span class="red">'+minute+'</span>分<span class="red">'+second+'</span>秒');

} **else** {//剩余时间小于或等于0的时候，就停止间隔函数

window.clearInterval(InterValObj);

//这里可以添加倒计时时间为0后需要执行的事件

$("#borrow\_remainingTime").html("剩余时间：0天0时0分0秒");

}

}

Js字符串字节长度：

//中文+2；

function getStrBytes(str){

var byteLen=0,len=str.length;

for(var i=0; i<len; i++){

if(str.charCodeAt(i)>255){

byteLen += 2;

}

else{

byteLen++;

}

}

return byteLen;

}

~

function getBytes(str){

str = str.replace(/[^\x00-\xff]/g, '\*\*');//将非ascii码转换为2个ascii码

return str.length;

}