vip过期提醒：

用户登录后，判断vip，临近过期的提醒，每天提醒一次；

~

实现方案，

使用缓存保存提醒过的状态，实现每天只提醒一次；---不受关机、用户退出的影响；

前提：

spring的相关配置，

Ehcache.xml

<!-- vip提醒cache-->

<cache name=*"vipCache"*

maxElementsInMemory=*"10000"*

eternal=*"false"*

timeToIdleSeconds=*"120" //0永不过期*

timeToLiveSeconds=*"120" //0永不过期*

overflowToDisk=*"true"*

diskPersistent=*"false"*

diskExpiryThreadIntervalSeconds=*"60"*/>

Controller，

@Autowired

**private** CacheManager cacheManager; //spring的cache配置；

//查询获取缓存信息，提醒状态

@RequestMapping(value = "getRemindStatus")

@ResponseBody

**public** MessageBox getRemindStatus() **throws** Exception {

ShiroUser shiroUser = currentUser();

String key = shiroUser.getUserId().toString();

Cache cache = cacheManager.getCache(***vipCache***);

//查找缓存，

ValueWrapper valueWrapper = cache.get(key);

**if** (valueWrapper != **null**) {

vipRemindVo2 = (VipRemindVo) valueWrapper.get();

**int** remindNum2 = vipRemindVo2.getRemindNum();

Long remindTime2 = vipRemindVo2.getRemindTime();

// 每天提醒一次的情形，

// 获得时间：同一天，返回1； | 不是同一天，重置cache；

**if** (remindNum2 == 1) {

**if** (org.apache.commons.lang3.time.DateUtils.*isSameDay*(**new** Date(remindTime2), **new** Date())) {

**return** **new** MessageBox("1", "");

} **else** {

msg = setRemindStatus();

**return** **new** MessageBox("0", msg);

}

}

// 只提醒一次的情形， | 存在,判断日期间隔是否还大于7，大于不再提醒；小于等于，重新设置缓存等；

**if** (remindNum2 == 2) {

**if** (DateUtils.*dayDiffByDay*(**new** Date(remindTime2), **new** Date()) > 7) {

**return** **new** MessageBox("1", "");

} **else** {

msg = setRemindStatus();

**return** **new** MessageBox("0", msg);

}

}

} **else** { // 无缓存的，建立缓存，返回相应的提示；

msg = setRemindStatus();

**return** **new** MessageBox("0", msg);

}

**return** **null**;

//设置缓存信息，提醒状态

**public** **void** setRemindStatus() **throws** Exception {

ShiroUser shiroUser = currentUser();

String key = shiroUser.getUserId().toString();

Cache cache = cacheManager.getCache(***vipCache***);

//存在，清空；

ValueWrapper valueWrapper = cache.get(key);

**if** (valueWrapper != **null**) {

cache.evict(key);

}

//保存数据，key-value形式；

//cache.put(key, System.*currentTimeMillis*());

VipRemindVo vipRemindVo = **new** VipRemindVo();

vipRemindVo.setRemindNum(1); // 区别于下面的内容；

vipRemindVo.setRemindTime(System.*currentTimeMillis*());

cache.put(key, vipRemindVo);

}

页面， //查询cache中的值，控制显隐；

//vip提醒时间满足的---先查询缓存，有缓存的返回结果，无缓存的在后台设置缓存---然后，前台无缓存不提醒，有缓存提醒；

$.ajax({

url : "${path}/myaccount/getRemindStatus.do",

data : {},

type : 'get',

success : **function**(data){

**if**(data.code=="-1"){

//alert("未登录，不做处理");

}

**else**{

**if**(data.code=="1"){

//alert("//有缓存，不提醒；");

}

**else**{

**if**(data.message =="3" ){

**if**(layer.confirm("温馨提示：您的vip临近过期。为享有本息保障，请提前做好VIP续费准备！",**function**(){

location.href="${path}/account/approve/vip/vipforinsert.do";

}));

}

**if**(data.message =="1" ){

**if**(layer.confirm("温馨提示：您的vip即将过期。为享有本息保障，请提前做好VIP续费准备！",**function**(){

location.href="${path}/account/approve/vip/vipforinsert.do";

}));

}

**if**(data.message =="2" ){

**if**(layer.confirm("温馨提示：您的vip已过期。为享有本息保障，请及时做好VIP续费准备！",**function**(){

location.href="${path}/account/approve/vip/vipforinsert.do";

}));

}

}

}

}

});

两个字段保存，使用key-value；

更多内容的，使用key-实体类；---声明一个实体类；

**public** **class** VipRemindVo **implements** Serializable{

**private** **static** **final** **long** ***serialVersionUID*** = -145170582010367476L;

**private** Integer remindNum;//提醒间隔类型 | 1，每天1次；2，只一次；

**private** Long remindTime; //首次提醒日期