

1. 补充方法说明，dao 层、服务层、控制器层、帮助类、前端 js、css 等

```
    }  
    /**  
     * 冻结库存减少，库存不增加，多个  
     * @param specCombIds amounts  
     * @return boolean  
     * @author 江伟  
     */  
    public synchronized boolean frozenDec(String[] specCombIds, int[] amounts) {  
        //两个数组不等  
        if (specCombIds.length != amounts.length) {  
            System.out.print("两个数组长度不等");  
            return false;  
        } else {  
            int i = 0;  
            List<LgItemSpecComb> lgItemSpecCombList = new ArrayList<>();  
            List<LgItemInfoChild> lgItemInfoChildList = new ArrayList<>();  
        }  
    }  
}
```

```
home / themes / basic / pay / bankcardpay.js  
fresh.properties x mvbatis-confio.xml x AddType.java x ApiControllerAdvice.java x Global.java  
/**  
 * Created by acer on 2016/11/22.  
 */  
var count=1;  
$(document).ready(function() {  
    //是否激活白条 1表示有 0表示无 2表示不足  
    var isnote = '';  
  
    //是否绑定银行卡 1表示有 0表示没有  
    var isbank = '';  
  
    //加载个人信息，判断是否绑定银行卡或者激活白条  
    $.ajax({  
        type:"POST",  
        async:false,  
        url:test+"/i/user/lgUser/queryInfo",  
        data:{  
            id:userId  
        },  
        success:function(data) {  
            console.log('我成功了');  
            //console.log(data);  
            isnote = data.lgUser.isActivate;  
            $('#limit_note').text((data.lgUser.surplus).toFixed(2)+'元');  
            if(data.lgUser.isActivate=='1'){  
                if(data.lgUser.surplus<$("#amountMoney").text()){
```

作者名称需要统一

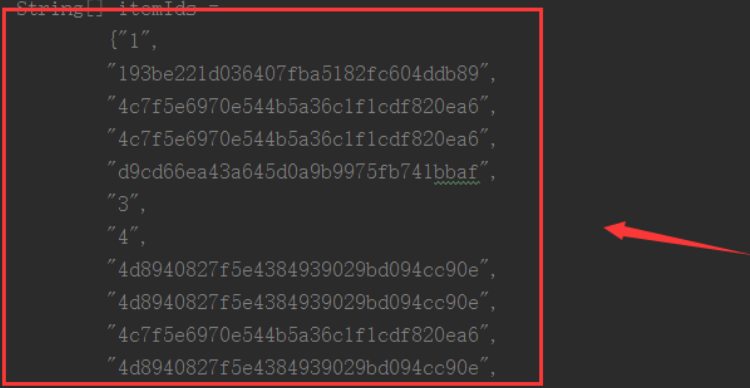
## 2. 方法主体，重点部分补充注释

```
byte[] keyBytes = new byte[0];
try {
    keyBytes = (new BASE64Decoder()).decodeBuffer(publicKey);
    // 构造X509EncodedKeySpec对象
    X509EncodedKeySpec keySpec = new X509EncodedKeySpec(keyBytes);
    // KEY_ALGORITHM 指定的加密算法
    KeyFactory keyFactory = KeyFactory.getInstance("RSA");
    // 取公钥对象
    PublicKey pubKey = keyFactory.generatePublic(keySpec);
    Signature signature = Signature.getInstance("MD5withRSA");
    signature.initVerify(pubKey);
    signature.update(signDataStr.getBytes("gbk"));
    // 验证签名是否正常
    boolean ret = signature.verify((new BASE64Decoder()).decodeBuffer(RSA));
    System.out.println("RSA验签结果: " + ret);
    if (!ret) {
```

## 3. 清除无用的代码

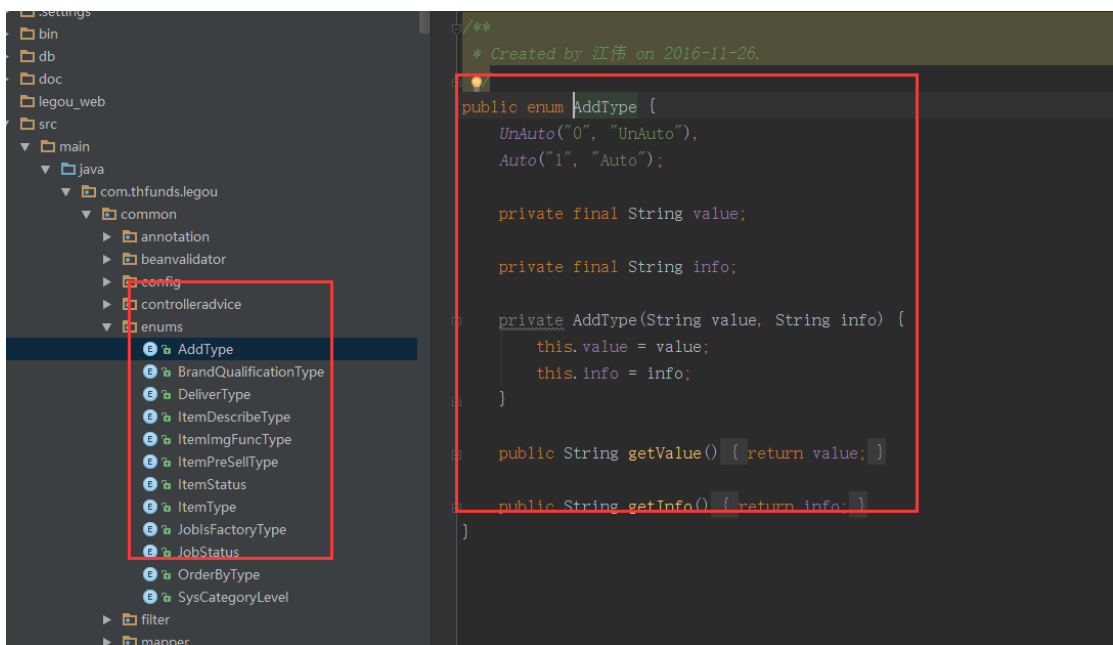
```
Map<String, Object> map = ApiResultUtils.getBaseMapSuccess();
Integer floorNum = 4; //楼层数量
Integer eachFloorItemNum = 8; //每层商品数量

List floorItemList = new ArrayList();
LgItemInfoChild lgItemInfoChild = new LgItemInfoChild();
lgItemInfoChild.setStatus(ItemStatus.OnShelves.getValue());
List<LgItemInfoChild> lgItemInfoChildList = lgItemInfoChildService.findList(lgItemInfoChild);
String[] itemIds = {
    "1",
    "193be221d036407fba5182fc604ddb89",
    "4c7f5e6970e544b5a36c1f1cdf820ea6",
    "4c7f5e6970e544b5a36c1f1cdf820ea6",
    "d9cd66ea43a645d0a9b9975fb741bbaf",
    "3",
    "4",
    "4d8940827f5e4384939029bd094cc90e",
    "4d8940827f5e4384939029bd094cc90e",
    "4c7f5e6970e544b5a36c1f1cdf820ea6",
    "4d8940827f5e4384939029bd094cc90e",
    "4c7f5e6970e544b5a36c1f1cdf820ea6",
    "4c7f5e6970e544b5a36c1f1cdf820ea6"
};
```



## 4. 类似参数，需注明其含义，采用枚举类型

```
public void move(LgAppInfo lgAppInfo, String type) {
    switch (type) {
        case "0":
            dao.moveTop1(lgAppInfo);
            dao.moveTop2(lgAppInfo);
            break;
        case "1":
            dao.moveBef1(lgAppInfo);
            dao.moveBef2(lgAppInfo);
            break;
        case "2":
            Integer count = dao.count(lgAppInfo);
            dao.moveDown1(lgAppInfo);
            dao.moveDown2(count, lgAppInfo.getId());
            break;
        case "3":
            Integer count1 = dao.count(lgAppInfo);
            dao.moveEnd1(lgAppInfo);
    }
}
```



```
/**
 * Created by 江伟 on 2016-11-26.
 */
public enum AddType {
    UnAuto("0", "UnAuto"),
    Auto("1", "Auto");

    private final String value;

    private final String info;

    private AddType(String value, String info) {
        this.value = value;
        this.info = info;
    }

    public String getValue() { return value; }

    public String getInfo() { return info; }
}
```

5. 全局变量统一整理到该类中

```
doc
legou_web
src
├── main
└── java
    ├── com.thfunds.legou
    │   ├── common
    │   │   ├── annotation
    │   │   ├── beanvalidator
    │   │   └── config
    │   └── Global
    └── controlleradvice
```

```
/**
 * 全局配置类
 * @author ThinkGem
 * @version 2014-06-25
 */
public class Global {

    /**
     * 当前对象实例
     */
}
```