## 单元测试报告

测试模块：响应事件监听模块/加入购物车，条件组合覆盖

测试工具：unittest

测试人员：吕莉

测试代码: test响应事件监听模块-测试用例&单元测试结果.py，添加至项目文件夹附件

#### 1程序结构：

<input onclick="add\_cart()" style="background-color: #20B2AA;color: black; font-weight: bolder" type="button" value="加入购物车" />

<script type="text/javascript">

function add\_cart(){

if(document.getElementById("type").value=="nothing"){

toastr.options.positionClass = 'toast-top-center';

toastr.error("请选择一个购买的规格吧亲");

}

else {

var amount\_index = document.getElementById("amount\_index").value

var value = document.getElementById("type").value

var color = document.getElementById("color").value

var amount\_str = document.getElementById("re\_username").value;

var amount = parseInt(amount\_str);

var id = "{{product.id}}";

$.ajax({

url: "/ajax\_add\_cars",

data: {"id": id, "amount": amount,"color":color,"value":value,"amount\_index":amount\_index},

dataType: 'text',

type: 'POST',

success: function (data) {

if (data == "添加购物车成功") {

toastr.options.positionClass = 'toast-top-center';

toastr.success(data);

} else {

toastr.options.positionClass = 'toast-top-center';

toastr.error(data);

}

}

});

}

}

</script>

def ajax\_add\_cars(request):

if 'user\_id' not in request.session.keys():

return HttpResponse(200)

if request.method == "POST":

value = request.POST.get("id")

pro = Products.objects(\_id=value)[0]

amount = request.POST.get("amount")

amount\_index = request.POST.get("amount\_index")

amount\_index\_delete = int(amount\_index)-1#要-1的amount下标.

Amount\_delete = int(amount)

try:

pro.amount[amount\_index\_delete] = pro.amount[amount\_index\_delete]-Amount\_delete

pro.save()

except:

return HttpResponse("库存量不足，添加失败")

amount = request.POST.get("amount")

amount\_ca = int(amount)

total\_money = amount\_ca \* pro.sale\_price

#print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"+product.name )

single = str(int(time.time() \* 1000000)) + request.session['user\_id']

cart = Carts(single=single,

product\_id=pro.id,

product\_name=pro.name,

amount\_index=amount\_index\_delete,

size=request.POST.get("value"),

price=pro.sale\_price,

color=request.POST.get("color"),

amount=amount,

total\_money=total\_money,

src=pro.src)

try:

customer = Customer.objects(\_id=request.session['user\_id'])[0]

customer.carts.append(cart)

customer.save()

except:

return HttpResponse("except\_line\_220")

return HttpResponse(200)

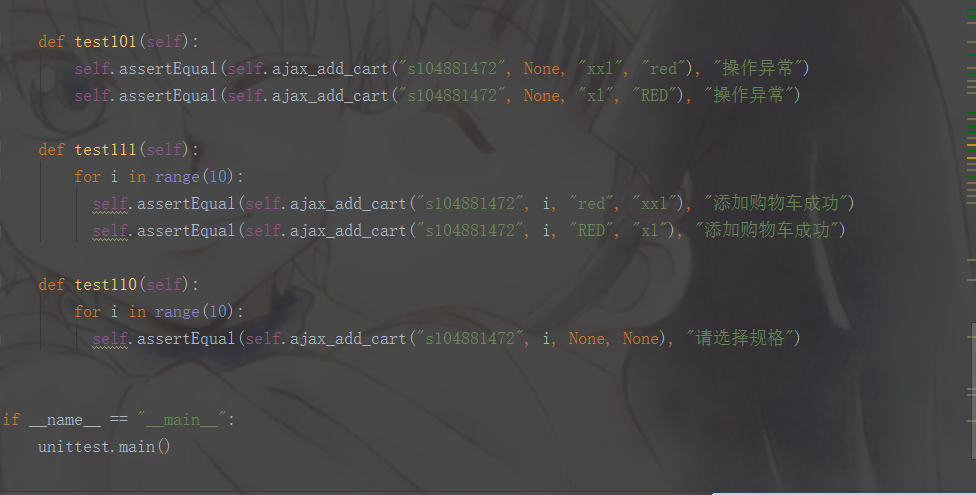
#### 代码附图











#### 代码描述

测试样例代码:

截取后端views里的代码。由于httprespond无法体现与模拟，本身httprespind返回值在web开发中也是一个表示错误异常的资源，因此在测试中所有的httprespond改变为易读的字符串，并与预期结果进行比对。

传入参数说明:

同时由于ajax传入数据的测试无法进入python测试的函数入口，因此测试数据的输入直接模拟ajax传入的数据[size,color,user\_id,amount]

测试函数:

Def test xxx([ajax]).x=0|1,表示8种不同的输入情况，对于有amount和type的情况采用多组数据测试。当然由于前端input的限定使得传入的amount不可能小于0，也不可能大于99，amount和type同理

#### 测试结果分析

测试1：



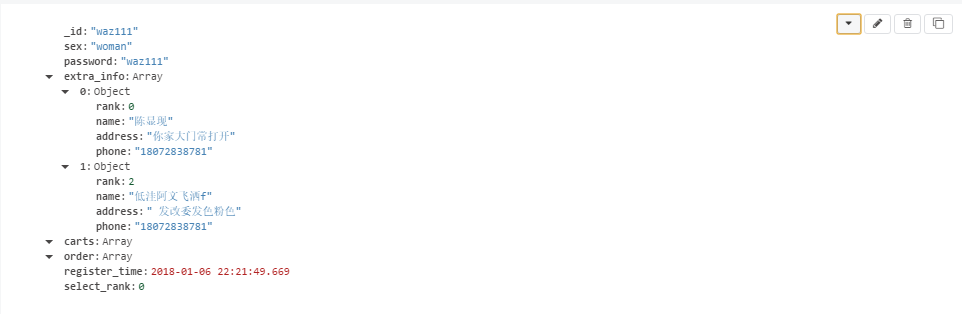
在使用None下的ajax传入的amount值被成功添加了购物车且是一个为0的商品

分析：前端在传来amount的时候已经确定不是一个NaN值（对应python None值）所以需要模拟前端处理

测试2:



使用的测试数据





|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 用例名称 | | 加入购物车 | | | |
| 测试类型 | | 单元测试 | | | |
| 测试目的 | | 测试出(指定的)一件商品在不同的条件下是否能成功被加入购物车 | | | |
| 测试方法 | | 在业务前网页层面，和业务代码内后端层面进行条件组合覆盖测试 | | | |
| 测试用例设计原则 | | 条件组合覆盖 | | | |
| 正常测试情况 | | | | | |
| 编号 | 测试数据 | 数据描述 | 预期结果 | 实际测试结果 | |
| 1 | userid:null | seeion.user为空 | 错误提示：请先登录 | 与预期结果相同 | |
| [color,size]:null | 不选择规格 |
| count: | 将购买数量改成了空 |
| 2 | userid:null | seeion.user为空 | 错误提示：请先登录 | 与预期结果相同 | |
| [color,size]:null | 不选择规格 |
| count:1 | 有正确的购买数量 |
| 3 | userid:null | seeion.user为空 | 错误提示：请先登录 | 与预期结果相同 | |
| [color,size]:red.xxl | 选择数据库中存在的规格 |
| count:1 | 有正确的购买数量 |
| 4 | userid:null | seeion.user为空 | 错误提示：请先登录 | 与预期结果相同 | |
| [color,size]:red.xxl | 选择数据库中存在的规格 |
| count: | 将购买数量改成了空 |
| 5 | userid：s104881472 | seeion.user为s104881472 | 错误提示：请选择规格 | 与预期结果相同 | |
| [color,size]:null | 不选择规格 |
| count: | 将购买数量改成了空 |
| 6 | userid：s104881472 | seeion.user为s104881472 | 错误提示：请选择规格 | 与预期结果相同 | |
| [color,size]:null | 不选择规格 |
| count:1 | 有正确的购买数量 |
| 7 | userid：s104881472 | seeion.user为s104881472 | 添加购物车成功 | 与预期结果相同 | |
| [color,size]:red.xxl | 选择数据库中存在的规格 |
| count:1 | 有正确的购买数量 |
| 8 | userid：s104881472 | seeion.user为s104881472 | 错误提示：操作错误异常 | 与预期结果相同 | |
| [color,size]:red.xxl | 选择数据库中存在的规格 |
| count: | 将购买数量改成了空 |
| 测试结论 | |  |  |  |  |
| 状态 | | 正常 |  |  |  |