# 前端分组函数

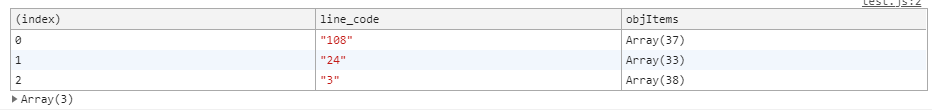
## 来源

参看文档：https://blog.csdn.net/lamyuqingcsdn/article/details/51123177

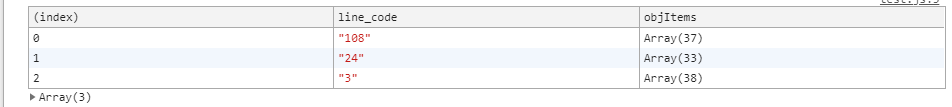
前段时间，写了一条sql语句，一对多关系，需求是一个1表的一条和多表的多条数据构成一个list，再构成多个list，由于没有对应实体接受，如果写成关联查询，就需要在java里面去循环转换；如果再low一点，就先查1表，再循环去查多表，多次请求数据库，有点吃饱了，老夫的原则，能少访问数据库，就尽量少访问，所以老夫就想到了在后台直接关联查询出所有数据，然后在前端进行分组

新版本使用objItems,取代data，避免data太多引起混淆

## 原始getObjArrGroupJs



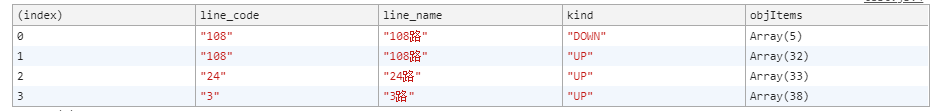
## （单）getObjArrGroup



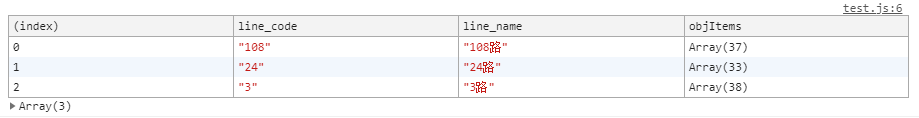
构成，多个数组对象，此数组对象属性（条件，对应的数据数据）

此方法有局限性，只能查询条件是一个，下面重头戏，多个查询条件

## （多）getObjArrGroup2



多个不同条件进行不同分组，当所有筛选条件(合)唯一时，完全兼容单条件分组



# 数组分段

## 来源

在弄高德地图时候，在一条线路画点的时候，由于途径点16个限制死了的，所以一条线路上的点超过16个必须分段渲染，下面那个去掉屏蔽的便适用于高德地图（取15 + 1）

## 示例

*//js拆分数组*splitArr: **function** (chartArr) {  
 **var** allData = []; *//用来装处理完的数组* **var** currData = []; *//子数组用来存分割完的数据  
 //循环需要处理的数组* **for**(**var** i = 0; i < chartArr.length; i++) {  
 *//将chartArr[i]添加到子数组* currData.push(chartArr[i]);  
 *//在这里求4的余数,如果i不等于0,且可以整除 或者考虑到不满4个或等于4个的情况就要加上 i等于当前数组长度-1的时候* **if**((i != 0 && (i + 1) % 15 == 0) || i == chartArr.length - 1) {  
 // **if**(i < chartArr.length - 1) {*//这里做个小变动，取后面的第一条加入数组  
 //* currData.push(chartArr[i+1]);  
 // }  
 *//把currData加到allData里* allData.push(currData);  
 *//在这里清空currData* currData = [];  
 }  
 };  
 **return** allData;  
 },

# 取差集

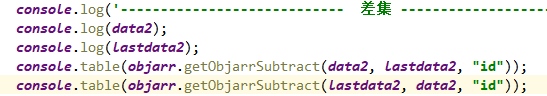
## 以id为标准

JS数组查重、数组对象求差集

https://blog.csdn.net/lyn1772671980/article/details/79664855

原理：类似java的双重for循环取差

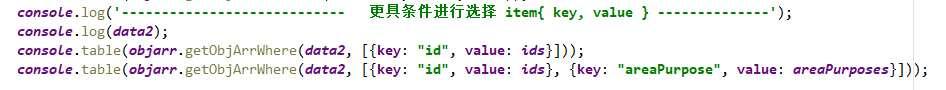
两个数组肯定是对象元素都一样，比较一个就行了



# Where条件过滤

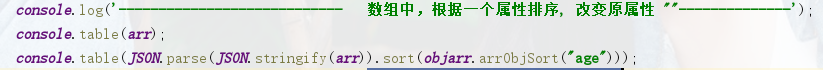
## 单，多条件过滤

支持单条件，多条件过滤



# 对象数组排序

## 改变原数组



# 数组中，根据几个属性组成新的数组



# Jpa 数据合并

## 多对一合并

