

排序接口返回乱序的字典数据

1. 问题：一般来讲，排序接口请求的结果都是一个数组的 json 数据，但是有的第三方服务器接口返回的并不是这种格式的数据，而是像下面所示，怎么处理？

```
/** 适用json格式:
{
  "fields": [
    "current_amount",
    "business_balance_scale"
  ],
  "603358.SS": [
    1200,
    0
  ],
  "603966.SS": [
    1200,
    0
  ]
}
*/
```

2. 解决方案：

- (1) 定义基类：

```
@interface TheMarketDataModel : NSObject //基类

@property (strong, nonatomic) NSArray *keys; /* json字典中的fields数组，如 ["current_amount","business_balance_scale"] */
@property (strong, nonatomic) NSArray *values; /* json字典剔除fields数组后的任意数组，如：[1200,0] */
@property (strong, nonatomic) NSString *stockCode; /* json字典剔除fields数组后的任意数组的key值，如股票代码 "603358.SS" 或板块代码"420100.XBHS"*/

- (id)initWithValues:(NSArray *)values Keys:(NSArray *)keys withStockCode:(NSString *)stockCode;
- (NSInteger)indexOffields:(NSString *)param;
- (void)setParam:(NSString *)paramName;

@end
```

```
#import "TheMarketDataModel.h"

@implementation TheMarketDataModel

- (id)initWithValues:(NSArray *)values Keys:(NSArray *)keys withStockCode:(NSString *)stockCode{
    if(self = [super init]){
        self.keys = keys;
        self.values = values;
        self.stockCode = stockCode;
    }
    return self;
}

- (NSInteger)indexOffields:(NSString *)param{
    NSString *index = [NSString stringWithFormat:@"%ld", (long)[self.keys indexForObject:param]];
    return [index length] > 4 ? -1:[index integerValue];
}

- (void)setParam:(NSString *)paramName{
    NSInteger index= [self indexOffields:paramName];
    if(index >= 0){
        NSString *str=[self.values objectAtIndex:index];
        if([str isKindOfClass:[NSNull class]] == NO)
            [self setValue:[NSString stringWithFormat:@"%@",str] forKey:paramName];
    }
}

@end
```

(2) 创建排序 model

```
@interface Sort_Detail_Data : TheMarketDataModel//排序详情Object

@property (strong, nonatomic) NSString *prod_name; /* 股票名称 */
@property (strong, nonatomic) NSString *last_px; /* 最新价 */
@property (assign, nonatomic) float px_change; /* 涨跌价格 */
@property (assign, nonatomic) float px_change_rate; /* 涨跌幅 */

@property (strong, nonatomic) NSString *selfstock_id; /* 自选股id */
@property (strong, nonatomic) NSString *market; /* 市场代码 */

@property (strong, nonatomic) NSString *min5_chgpct; /* 五分钟涨跌幅 */
@property (strong, nonatomic) NSString *rise_count; /* 上涨家数 */
@property (strong, nonatomic) NSString *fall_count; /* 下跌家数 */
@property (strong, nonatomic) NSString *member_count; /* 成员个数 */
@property (strong, nonatomic) NSString *preclose_px; /* 昨收价 */
@property (strong, nonatomic) NSString *high_px; /* 最高价 */
@property (strong, nonatomic) NSString *low_px; /* 最低价 */
@property (strong, nonatomic) NSString *business_amount; /* 成交数量 */
@property (strong, nonatomic) NSMutableArray <rise_first_grp *> *rise_first_grp; /* 领涨股 */
@property (strong, nonatomic) NSString *business_balance; /* 成交金额 */
@property (strong, nonatomic) NSString *open_px; /* 开盘价 */
@property (strong, nonatomic) NSString *data_timestamp; /* 最后更新时间 */
@property (strong, nonatomic) NSString *trade_mins; /* 交易分钟数 */

@property (assign, nonatomic) CGFloat eps; /* 每股收益 */
@property (assign, nonatomic) CGFloat bps; /* 每股净资产 */
@property (assign, nonatomic) CGFloat cash_flowps; /* 每股现金流 */
@property (assign, nonatomic) CGFloat net_profit; /* 净利润 */
@property (assign, nonatomic) CGFloat operating_revenue; /* 营业收入 */
@property (assign, nonatomic) CGFloat total_assets; /* 总资产 */

@property (assign, nonatomic) NSInteger extIdentifier; /* 额外标识符，保留属性 */

@end
```

@implementation Sort_Detail_Data

```
- (id)initWithValues:(NSArray *)values Keys:(NSArray *)keys withStockCode:(NSString *)stockCode {
    if(self = [super initWithValues:values Keys:keys withStockCode:stockCode]){
        NSInteger index;
        [self setParam:@"prod_name"];
        [self setParam:@"last_px"];

        index = [self indexOffFields:@"px_change"];
        if(index >= 0){
            double str=[[self.values objectAtIndex:index] doubleValue];
            self.px_change = str;
        }

        index=[self indexOffFields:@"px_change_rate"];
        if(index >= 0){
            double str=[[self.values objectAtIndex:index] doubleValue];
            self.px_change_rate = str;
        }

        [self setParam:@"min5_chgpct"];
        [self setParam:@"rise_count"];
        [self setParam:@"fall_count"];
        [self setParam:@"member_count"];
        [self setParam:@"preclose_px"];
        [self setParam:@"high_px"];
        [self setParam:@"low_px"];
        [self setParam:@"business_amount"];
        [self setParam:@"business_balance"];
        [self setParam:@"open_px"];
        [self setParam:@"data_timestamp"];
        [self setParam:@"trade_mins"];
```

(3) Json 数据处理

```
[guPiaoService requestSortDataWithUrl:url andMethod:@"GET" success:^(id responseObject) {
    NSDictionary *dic=[NSJSONSerialization JSONObjectWithData:responseObject options:NSJSONReadingMutableContainers error:nil];
    NSMutableDictionary *sort = [[dic objectForKey:@"data"][@"sort"] mutableCopy];

    NSArray *keys = [sort objectForKey:@"fields"];
    [sort removeObjectForKey:@"fields"];

    NSMutableArray *responseArr = [NSMutableArray array];
    //json正则匹配出allKey
    NSArray *allKeys = [TextUtility getAllKeysFromjsonData:responseObject];
    [allKeys enumerateObjectsUsingBlock:^(NSString *stockCode, NSUInteger idx, BOOL * _Nonnull stop) {
        NSArray *values = sort[stockCode];
        Sort_Detail_Data *model = [[Sort_Detail_Data alloc] initWithValues:values Keys:keys withStockCode:stockCode];
        [responseArr addObject:model];

        [weakSelf.tempDic setValue:model forKey:model.stockCode];
        [weakSelf.tempStockCodes addObject:model.stockCode];
    }];
    responseArr = [[self sortArray:responseArr sortKey:@"cps" ascending:NO] mutableCopy];

    if([responseArr count]){
        [weakSelf.data replaceObjectAtIndex:0 withObject:responseArr];
        completion(NO, nil);//数据未请求结束，先返回部分数据

        //<<<<<继续请求其他数据
        NSInteger count = [weakSelf.tempStockCodes count];
        if(count < 20){
            [self loadOtherRequestWithStocks:weakSelf.tempStockCodes withResponseBlock:^(BOOL isLastData, NSError *error) {
```

为什么这里不是遍历sort的allKey

图中问题，原因是遍历字典的 key 会导致最终的 model 数组乱序，大家注意，字典是无序的！那么怎么获取顺序的 key 呢？可以使用正则匹配～

```
+ (NSArray *)getAllKeysFromjsonData:(NSData *)jsonData {
    NSString *jsonStr = [[NSString alloc] initWithData:jsonData encoding:NSUTF8StringEncoding];
    NSRegularExpression *expression = [NSRegularExpression regularExpressionWithPattern:@"(?:<=\\").{3,8}\\.[A-Z]{2,4}(?:=\\")" options:kNilOptions error:nil];
    NSMutableArray *keys = [NSMutableArray array];
    [[expression matchesInString:jsonStr options:NSMatchingReportCompletion range:NSMakeRange(0, jsonStr.length)] enumerateObjectsUsingBlock:^(NSTextCheckingResult * _Nonnull obj, NSUInteger idx
        , BOOL * _Nonnull stop) {
        [keys addObject:[jsonStr substringWithRange:obj.range]];
    }];
    return keys;
}
```