## 1．Json

1. /\*\*
2. \* 复杂json格式字符串与JavaBean\_obj之间的转换
3. \*/

Teacher teacher = JSON.parseObject(COMPLEX\_JSON\_STR, new TypeReference<Teacher>() {});

1. /\*\*
2. \* json字符串-数组类型与JavaBean\_List之间的转换
3. \*/

ArrayList<Student> students = JSON.parseObject(JSON\_ARRAY\_STR, new TypeReference<ArrayList<Student>>() {});

## 2.Stream

### 2.1 List<Map> sort

passList=passList.stream().sorted((m1,m2)->{  
 Object s1 = m1.get(DynamicFieldEnum.*SORT*.getCode());  
 Object s2 = m2.get(DynamicFieldEnum.*SORT*.getCode());  
 if (s1==null || s2==null) return 0;  
 return ((Integer)s1).compareTo((Integer)s2);  
 }).collect(Collectors.*toList*());

Map<String, List<AwardsResult>> ballotId2ResultMap = awardsResults  
 .stream()  
 .collect(Collectors.*groupingBy*(AwardsResult::getSlimId));

### 2.2 stream对List的一些求和

List<Map<String,Object>> salesList = this.baseMapper.sales(params);

int totalAaleCount = salesList.stream().mapToInt(m -> Integer.valueOf(String.valueOf(m.get("saleCount")))).reduce(0,Integer::sum);

//BigDecimal 求和

BigDecimal totalPrice = salesStandardList.stream().map(m -> new BigDecimal(m.get("totalAmount").toString())).reduce(BigDecimal.ZERO,BigDecimal::add);