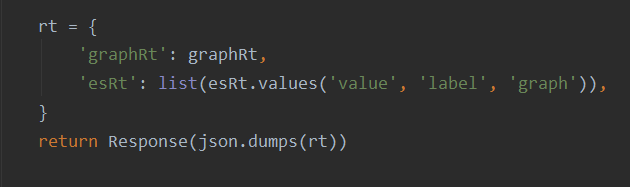
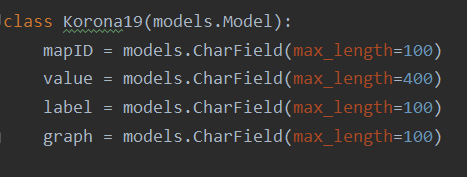
**事先声明**

* Value：实体的值
* Label：实体的label
* graph：实体对应的知识图谱，有8个
* 请求端口：<http://127.0.0.1:8000/se/search/>
* 总返回可查看：总返回实例(未json化).txt

**后端返回**



一个是graph另一个是es的，es的返回是一个querySet提取value、label和graph后转list的结果，model如下图



**ESRt: es返回数据**

**Case1**：-1 表示不用显示数据

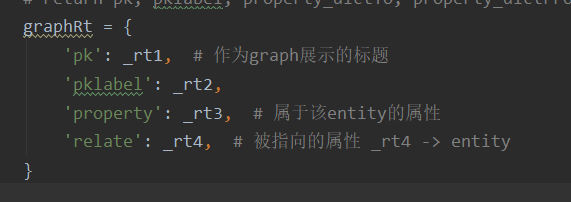
**Case2**: list

List里边是多个dict，dict中有value label和graph三个key

**GraphRt 图数据返回前端数据**

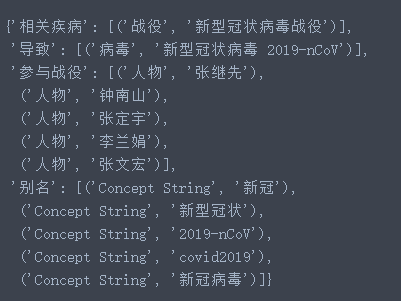
**Case1**：-1 表示不用显示图数据库数据

**Case2**: Dict



1. **Pk**是entity的名字（value），str
2. **Pklabel**表示其对应的多个label，list
3. **Property**是属于该entity的属性，dict 里边内容也有结构
4. **Relate**是与entity相关的属性，dict 里边内容也有结构

**Property、relate内部：**key是关系（一个关系可连接多个节点），里边是list，list里边是list(下图中是set，错误)，list.0是该节点的label，而list.1是该节点的value



**前端请求代码：**

**输入：q**

fetch('http://127.0.0.1:8000/se/search/', {

        method: 'POST',

        headers: {

            //'Content-Type': 'application/x-www-form-urlencoded',

        },

        body: JSON.stringify({

            "sequence": q,

        })

    }).then(res => res.text()).then(res => { strToJson(res).graphRt !== -1? setRes({pk:strToJson(res).graphRt.pk, pklabels:strToJson(res).graphRt.pklabels , property\_dictTo:strToJson(res).graphRt.property\_dictTo ,property\_dictFrom:strToJson(res).graphRt.property\_dictFrom, request: true }):setRes({request:false})

        ;strToJson(res).esRt !== -1?setRes1({esRt:strToJson(res).esRt.hits, request: true }):setRes1({request:false})})

        .catch((error) => {

            // alert("结果返回失败！");

        });