## 一、java函数回调

String xyrmsgStr = (String) request.getParam("xyrmsg"); // 嫌疑人数据

List<ZhzjSuspectSource> zhzjSuspectSourceList = JSON.*parseArray*(xyrmsgStr, ZhzjSuspectSource.**class**);

// 修改source表嫌疑人信息

**if** (zhzjSuspectSourceList.size() > 0) {

zhzjSuspectSourceService.updateZhzjSuspectSourceByInfotype(zhzjSuspectSourceList);

String id = zhzjSuspectSourceList.get(0).getId(); // 嫌疑人ID

// 获取info表嫌疑人信息

ZhzjSuspectInfo zhzjSuspectInfo = zhzjSuspectInfoService.getZhzjSuspectInfoById(id);

Map<String, ZhzjSuspectSource> zhzjSuspectSourcemap = **new** HashMap<String, ZhzjSuspectSource>();

**for** (ZhzjSuspectSource zhzjSuspectSource : zhzjSuspectSourceList) {

zhzjSuspectSourcemap.put(zhzjSuspectSource.getInfoSource(), zhzjSuspectSource);

}

// 需要修改的信息

String[] models = { "Name", "Gender", "BirthDay", "CertificatesType", "CertificatesType",

"CertificatesCard", "BirthPlace", "Nationalilty", "Occupation", "Education", "CensusRegisterPlace",

"Domicile" };

**for** (String model : models) {

// 获取value来源方法

String getmethodNameinfo = "get" + model + "Source";

String getmethodName = "get" + model;

// 赋值的方法

String setmethodName = "set" + model;

// 如果是居住地 则更新info的居住地详细里

**if** ("Domicile".equals(model)) {

setmethodName = "setDomicileDetail";

}

Class clazzinfo = ZhzjSuspectInfo.**class**;

Class clazzsource = ZhzjSuspectSource.**class**;

Method methodgetsource = **null**;

Method methodget = **null**;

Method methodset = **null**;

**try** {

// 取值方法

methodgetsource = clazzinfo.getMethod(getmethodNameinfo);

methodget = clazzsource.getMethod(getmethodName);

// 赋值方法

methodset = clazzinfo.getMethod(setmethodName, String.**class**);

Object resultgetsource = **null**;

Object resultget = **null**;

// 获取来源

resultgetsource = methodgetsource.invoke(zhzjSuspectInfo);

// 如果得到的来源为空 跳出本次循环

**if** (**null** == resultgetsource) {

**continue**;

}

// 判断来源

**if** (zhzjSuspectSourcemap.keySet().contains(resultgetsource)) {

// 获取来源下的数据

ZhzjSuspectSource source = zhzjSuspectSourcemap.get(resultgetsource);

// 得到来源对应的数据

resultget = methodget.invoke(source);

// 如果得到的数据为空 跳出本次循环

**if** (**null** == resultget) {

**continue**;

}

// 给info赋值

methodset.invoke(zhzjSuspectInfo, resultget.toString());

}

} **catch** (NoSuchMethodException e) {

*logger*.error("获取方法异常异常:class=" + clazzinfo + clazzsource + ",method:" + getmethodNameinfo

+ setmethodName + getmethodName);

} **catch** (InvocationTargetException e) {

*logger*.error("取值方法反射异常,class:" + methodgetsource + methodget + methodset, e);

} **catch** (IllegalAccessException e) {

*logger*.error("实例化处理类异常,class:" + clazzinfo + clazzsource, e);

}

}

// 修改info表嫌疑人信息

zhzjSuspectInfoService.updateZhzjSuspectInfo(zhzjSuspectInfo);

}

**return** Response.*SUCCESS*();

}