**实习基地申请模块设计文档**

1. 实习基地申请模块概述

实习基地申请模块的功能主要包括：用户对实习基地的申请，管理员对实习基地申请的审核和管理。

1. 流程图

（一）实习基地申请流程图

用户

点击基地申请

填写基地信息表

提交

（二）实习基地申请审核流程图

管理员

选择审核记录状态

续期

申请成功

同意申请

拒绝申请

申请失败

显示实习基地申请记录

显示实习基地续期记录

申请

续期申请成功

同意续期

续期申请失败

拒绝续期

（三）实习基地管理流程图

3.1实习基地批量导入流程图

管理员

点击下载导入模板

模板填充数据

点击导入

选择文件

导入成功

3.2基地删除流程图

管理员

点击基地管理

选择基地记录

点击删除按钮

删除成功

3.3基地增加流程图

管理员

点击基地管理

点击增加按钮

填写基地信息

确认提交

增加成功

3.3基地导出流程图

管理员

点击导出按钮

选择相关信息

确认

导出成功

1. 实习基地续期流程图

4.1实习基地过期（数据库事件）

完毕

开始

获取所有基地的结束时间

超过结束时间

是

否

基地的状态变为17（到期失效）

4.2实习基地续期申请

用户

点击到期失效的基地记录

更改基地的结束时间

确认

续期申请提交成功

**三．表的创建**

1.实习基地申请表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute name | Key Type | Null/Unique | Default | Data Type | Sample Data |
| id(序号) | P | Not null |  | int | 自增 |
| bid(基地编号) |  | Not null |  | varchar |  |
| name  （基地名称） |  | Not null |  | varchar | “长安基地” |
| land\_address  （通信地址） |  | Not null |  | varchar |  |
| type  （基地类型） |  | Not null |  | int | 1 |
| landarea  （土地面积） |  | Not null |  | varchar |  |
| constructionarea  （建筑面积） |  | Not null |  | varchar |  |
| undertake  （一次性可承担实习学生人数） |  | Not null |  | int |  |
| applydp  （申报部门） |  | Not null |  | int |  |
| phone  (联系人电话) |  | Not null |  | varchar |  |
| material\_path  (材料路径) |  | null |  | varhcar |  |
| userid  (用户编号) |  | Not null |  | varchar |  |
| username  (联系人姓名) |  | Not null |  | varchar |  |
| status  （状态） |  | Not null | 2 | int | 2 |
| resperson  (法定责任人) |  | Not null |  | varchar |  |
| applytime  （基地星级） |  | Not null |  | varchar |  |
| buildtime  (基地建立时间) |  | null |  | varchar |  |

注：此表中的基地类型一共有五种，分别对应于basetype表中的值

1. 基地类型表（basetype）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute name | Key Type | Null/Unique | Default | Data Type | Sample Data |
| id  （序号） | P | Not null |  | int | 自增 |
| name  （基地类型说明） |  | Not null |  | varchar |  |

1. 实习基地信息表（prabaseinfo）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute name | Key Type | Null/Unique | Default | Data Type | Sample Data |
| id  (基地编号) | P | Not null |  | varchar |  |
| name  （基地名称） |  | Not null |  | varchar | “长安基地” |
| land\_address  （通信地址） |  | Not null |  | varchar |  |
| type  （基地类型） |  | Not null |  | int | 1 |
| landarea  （土地面积） |  | Not null |  | varchar |  |
| constructionarea  （建筑面积） |  | Not null |  | varchar |  |
| undertake  （一次性可承担实习学生人数） |  | Not null |  | int |  |
| applydp  （申报部门） |  | Not null |  | int |  |
| phone  (联系人电话) |  | Not null |  | varchar |  |
| material\_path  (材料路径) |  | null |  | varhcar |  |
| userid  (用户编号) |  | Not null |  | varchar |  |
| username  (联系人姓名) |  | Not null |  | varchar |  |
| status  （状态） |  | Not null | 2 | int | 2 |
| resperson  (法定责任人) |  | Not null |  | varchar |  |
| applytime  （基地星级） |  | Not null |  | varchar |  |
| buildtime  (基地建立时间) |  | null |  | varchar |  |
| star  (星级) |  | null |  | int |  |
| validate  (续期月数) |  | null |  | int |  |

1. 基地-专业表(basemajor)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute name | Key Type | Null/Unique | Default | Data Type | Sample Data |
| id  （序号） | P | Not null |  | int | 自增 |
| Pid  （实习基地编号） |  | Not null |  | varchar |  |
| maid(专业编号) |  | Not null |  | varchar |  |

5.学院-专业表(major)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute name | Key Type | Null/Unique | Default | Data Type | Sample Data |
| mid  （专业编号） | P | Not null |  | varchar |  |
| mname  （专业名称） |  | Not null |  | varchar |  |
| aid  （学院编号） |  | Not null |  | int | 参照学院信息表的主键 |

**四．代码设计**

本系统采用SSH框架实现，一共涉及四层分布：

controller层:控制层，主要实现页面的请求与响应,控制各个页面之间的跳转 ;

service层：业务逻辑层，主要实现代码内部逻辑功能；

dao层：数据库连接层，主要与数据库进行直接交互连接；

po层：实体层，每一个实体类对应数据库中的一张表。

下述为针对该模块所设计的action层，service层及对应dao层的相关接口：

Controller层: 1.实习基地申请控制：BaseApplyController

2.实习基地申请审核控制：BaseCheckController

3.我的实习基地申请控制：MybaseController

4.实习基地管理控制：BaseMaintenanceController

Service层： 1.实习基地申请逻辑：BaseApplyService

2.实习基地申请逻辑：BaseCheckService

3.我的实习基地申请控制：MybaseService

4.实习基地管理逻辑：BaseMaintenanceService

Dao层: 1.实习基地申请数据交互：BaseApplyDao

2.实习基地申请数据交互：BaseCheckDao

3.我的实习基地申请控制：MybaseDao

4.实习基地管理数据交互：BaseMaintenanceDao

**Controller层：**

//实习申请控制层

@Controller("BaseApplyController")

**public** **class** BaseApplyController {

//提交用户的实习基地申请

@RequestMapping("/getRequestBaseInfo.do")

**public** String getRequestBaseInfo(HttpServletRequest request,

HttpServletResponse response);

//获取学院和基地类型

@RequestMapping("/BaseApplyAllInfo.do")

**public** String BaseApplyInfo(HttpServletRequest request,

HttpServletResponse response, ModelMap map);

//根据部门类型获取部门

@RequestMapping("/getBaseSingleDept.do")

**public** String getBaseSingleDept(HttpServletRequest request,

HttpServletResponse response);

//根据学院编号获取专业

@RequestMapping("/getMajor.do")

**public** String getMajor(HttpServletRequest request,

HttpServletResponse response);

//检测用户输入的基地名称是否存在

@RequestMapping("/CheckName.do")

**public** String CheckName(HttpServletRequest request,

HttpServletResponse response);

}

//实习基地申请审核控制层

@Controller("BaseCheckController")

**public** **class** BaseCheckController {

//获取申请中的实习基地申请记录

@RequestMapping("/getBaseCheck.do")

**public** String getBaseCheck(HttpServletRequest request,

HttpServletResponse response);

//获取续期中的实习基地申请记录

@RequestMapping("/getaddCheck.do")

**public** String getaddCheck(HttpServletRequest request,

HttpServletResponse response);

// 获取刷选的部门

@RequestMapping("/getApplyDept.do")

**public** String getApplyDept(HttpServletRequest request,

HttpServletResponse response);

// 刷选功能（根据部门id刷选）

@RequestMapping("/getXUBaseCheck.do")

**public** String getXUBaseCheck(HttpServletRequest request,

HttpServletResponse response);

// 同意实习基地申请

@RequestMapping("/BasereAgreeApply.do")

**public** String agreeApply(HttpServletRequest request,

HttpServletResponse response);

// 拒绝实习基地申请

@RequestMapping("/BaserefuseApply.do")

**public** String refuseApply(HttpServletRequest request,

HttpServletResponse response);

//同意续期申请

@RequestMapping("/addAgreeApply.do")

**public** String addagreeApply(HttpServletRequest request,

HttpServletResponse response);

// 拒绝续期申请

@RequestMapping("/refuseAddApply.do")

**public** String refuseAddApply(HttpServletRequest request,

HttpServletResponse response);

}

//我的实习申请控制层

@Controller("MybaseController")

**public** **class** MybaseController {

// 获取用户的当前阶段的实习基地申请

@RequestMapping("/MybaseInfo.do")

**public** String MybaseInfo(HttpServletRequest request,

HttpServletResponse response);

// 获取用户的所有实习基地申请

@RequestMapping("/MybaseInfo2.do")

**public** String MybaseInfo2(HttpServletRequest request,

HttpServletResponse response);

// 实习基地申请撤回

@RequestMapping("/recall.do")

**public** String recall(HttpServletRequest request,

HttpServletResponse response);

//实习基地续期

@RequestMapping("/updateMyBaseDate.do")

**public** String updateBaseInfo(HttpServletRequest request,

HttpServletResponse response);

// 实习记录刷选(根据状态值刷选)

@RequestMapping("/getStatus.do")

**public** String getStatus(HttpServletRequest request,

HttpServletResponse response);

}

//实习基地管理控制层

@Controller("baseMaintenance")

**public** **class** BaseMaintenanceController {

//获取实习基地数据信息

@RequestMapping("/sendBaseinfo.do")

**public** String sendBaseinfo(HttpServletRequest request,

HttpServletResponse response);

//删除实习基地(批量)

@RequestMapping("/delBaseinfo.do")

**public** String delBaseinfo(HttpServletRequest request,

HttpServletResponse response);

//获取部门和基地类型等基本信息

@RequestMapping("/getManyinfo.do")

**public** String getManyinfo(HttpServletRequest request,

HttpServletResponse response);

//获取实习记录中所存在的部门

@RequestMapping("/getExistDept.do")

**public** String getExistDept(HttpServletRequest request,

HttpServletResponse response);

//根据刷选条件获取实习基地数据信息

@RequestMapping("/getshaiBaseinfo.do")

**public** String getshaiBaseinfo(HttpServletRequest request,

HttpServletResponse response);

//修改实习基地信息

@RequestMapping("/updateBaseInfo.do")

**public** String updateBaseInfo(HttpServletRequest request,

HttpServletResponse response);

//导出实习基地信息

@RequestMapping("/exportThisInfo.do")

**public** String exportThisInfo(HttpServletRequest request,

HttpServletResponse response);

//批量导入实习基地信息

@RequestMapping("/importBaseExcel.do")

**public** String importBaseInfo(HttpServletRequest request,

HttpServletResponse response);

//增加实习基地记录

@RequestMapping("/increaseBaseInfo.do")

**public** String increaseBaseInfo(HttpServletRequest request,

HttpServletResponse response);

}

**Service层：**

//实习申请逻辑层

**public** **interface** baseApplyService {

/\*

参数说明：type,字符串型，为部门类型

返回值： List<ApplyDept>，为ApplyDept对象的集合

函数功能： 根据部门类型获取部门

\*/

**public** List<ApplyDept> getDept(**int** type);

/\*

参数说明：无参数值

返回值： List<basetype>，为basetype对象的集合

函数功能：获取基地类型

\*/

**public** List<basetype> getBasetype();

/\*

参数说明：aid，整型，为学院编号

返回值： List<Major>，为Major对象的集合

函数功能：根据学院编号获取专业

\*/

**public** List<Major> getMajor(**int** aid);

/\*

参数说明：str1, 字符串型，为('基地编号','专业编号')的封装;str2,字符串型，为实习基地申请的信息;

info,字符串型，为要发送的消息

返回值： 无返回值

函数功能：增加实习基地申请

\*/

**public** **void** getRequestBaseInfo(String str1,String str2,String infostr);

/\*

参数说明：name,为实习基地的名称

返回值： int型，标志是否存在该基地名字

函数功能：判断是否存在该实习基地名称

\*/

**public** **int** CheckName(String name);

}

//实习基地申请审核逻辑层

**public** **interface** BaseCheckService {

/\*

参数说明：applydpid,整型，为申请部门编号; pageindex,为当前页数;size,为每页的条数; order,排序列;

orderDir,为排序的顺序; searchValue,为模糊查询的值.

返回值： BaseCheckList,该对象包括实习基地申请数据记录+记录条数

函数功能： 获取实习基地申请数据

\*/

**public** BaseCheckList getBaseCheck(**int** applydpid, **int** pageindex, **int** size,

**int** order, String orderDir,String searchValue);

/\*

参数说明：无参数值

返回值： List<Map<String,String>>,为Map<String,String>对象的集合

函数功能：获取部门集合

\*/

**public** List<Map<String,String>> getDept();

/\*

参数说明：recordstr，字符串型，为实习基地申请记录编号的封装;infoStr,字符串型，为要发送的消息

返回值： 无返回值

函数功能：拒绝实习基地申请

\*/

**public** **void** refuseapply(String recordstr,String infostr);

/\*

参数说明：str,字符串型，为('实习基地编号','建立时间','结束时间')的封装;

infoStr,字符串型，为要发送的消息

recordstr，字符串型，为实习基地申请记录编号的封装

返回值： 无返回值

函数功能：同意实习基地申请

\*/

**public** **void** agreeApply(String str,String infoStr,String recordstr);

/\*

参数说明：applydpid,整型，为申请部门编号; pageindex,为当前页数;size,为每页的条数; order,排序列;

orderDir,为排序的顺序; searchValue,为模糊查询的值.

返回值： BaseCheckList,该对象包括续期申请数据记录+记录条数

函数功能： 获取续期申请数据

\*/

**public** BaseCheckList getaddCheck(**int** applydpid, **int** pageindex, **int** size,

**int** order, String orderDir,String searchValue);

/\*

参数说明：recordstr，字符串型，为实习基地申请记录编号的封装;infoStr,字符串型，为要发送的消息

返回值： 无返回值

函数功能：同意续期申请

\*/

**public** **void** addDateApply(String infostr, String recordstr);

/\*

参数说明：recordstr，字符串型，为实习基地申请记录编号的封装;infoStr,字符串型，为要发送的消息

返回值： 无返回值

函数功能：拒绝续期申请

\*/

**public** **void** refuseDateApply(String infostr, String recordstr);

}

//我的实习基地申请逻辑层

**public** **interface** MyBaseInfoService {

/\*

参数说明：year,整型，为当前年份;status,整型，为标志值;userid,字符串型，为用户编号;

pageindex,为当前页数;size,为每页的条数; columnName,排序列;

orderDir,为排序的顺序;

返回值： MyBaseList,该对象包括实习基地申请数据记录+记录条数

函数功能：获取该用户当年的特定状态的实习基地申请数据

\*/

**public** MyBaseList MybaseInfo(**int** pageindex,**int** size,String columnName,

String orderDir,**int** year,**int** status,String userid);

/\*

参数说明：year,整型，为当前年份;status,整型，为标志值;userid,字符串型，为用户编号;

pageindex,为当前页数;size,为每页的条数; order,排序列;

orderDir,为排序的顺序;

返回值： MyBaseList,该对象包括实习基地申请数据记录+记录条数

函数功能：获取该用户当年的所有的实习基地申请数据

\*/

**public** MyBaseList MybaseInfo2(**int** pageindex,**int** size,**int** order,

String orderDir,**int** year,**int** status,String userid);

/\*

参数说明：id,为实习申请记录的编号

infoStr,字符串型，为要发送的消息

返回值： 无返回值

函数功能：撤回实习基地申请

\*/

**public** **void** recall(String id,String infostr);

/\*

参数说明：id,为实习申请记录的编号

adddate，为续期的月数

返回值： 无返回值

函数功能：续期

\*/

**public** **void** updateDate(**int** id,String adddate);

}

//实习基地管理逻辑层

**public** **interface** MaintenanceService {

/\*

参数说明：pageindex,为当前页数;size,为每页的条数; order,排序列;

orderDir,为排序的顺序;

返回值： MaintenanceList,该对象包括实习基地数据记录+记录条数

函数功能：获取该用户实习基地数据

\*/

**public** MaintenanceList maintenance(**int** pageindex, **int** size,**int** order,String orderDir,String searchValue);

/\*

参数说明：str,为实习基地记录的编号

返回值： 无返回值

函数功能：删除实习基地信息

\*/

**public** **void** delInfo(String str);

/\*

参数说明：无参数

返回值： List<ApplyDept>，为ApplyDept对象的集合

函数功能：获取部门集合

\*/

**public** List<ApplyDept> getExistDept();

/\*

参数说明：basetype，为实习基地类型;dept,为部门;star,为星级;

pageindex,为当前页数;size,为每页的条数; order,排序列;

orderDir,为排序的顺序;

返回值： MaintenanceList,该对象包括实习基地数据记录+记录条数

函数功能：根据条件刷选获取实习基地数据

\*/

**public** MaintenanceList getshaiBaseInfo(**int** basetype,**int** dept,**int** star,**int** pageindex, **int** size,**int** order,String orderDir,String searchValue);

/\*

参数说明：baseid，为实习基地编号;star,为星级;adddate,为续期月数

返回值： 无返回值

函数功能：修改实习基地截止日期

\*/

**public** **void** updateBaseInfo(String baseid,**int** star,String adddate);

/\*

参数说明：basetype，为实习基地类型;dept,为部门;star,为星级;

返回值： List<ExportBase>，为ExportBase对象的集合

函数功能：获得导出的实习基地数据

\*/

**public** List<ExportBase> getExportBaseInfo(**int** basetype,**int** dept,**int** star);

/\*

参数说明：str1, 字符串型，为('基地编号','专业编号')的封装;str2,字符串型，为实习基地申请的信息;

返回值： 无返回值

函数功能：增加实习基地

\*/

**public** **void** increaseBaseInfo(String str1,String str2);

}

**Dao层：**

//实习基地申请的数据交互层

**public** **interface** baseApplyDao {

/\*

参数说明：type,字符串型，为部门类型

返回值： List<ApplyDept>，为ApplyDept对象的集合

函数功能： 根据部门类型获取部门

\*/

**public** List<ApplyDept> getDept(**int** type);

/\*

参数说明：无参数值

返回值： List<basetype>，为basetype对象的集合

函数功能：获取基地类型

\*/

**public** List<basetype> getBasetype();

/\*

参数说明：aid，整型，为学院编号

返回值： List<Major>，为Major对象的集合

函数功能：根据学院编号获取专业

\*/

**public** List<Major> getMajor(**int** aid);

/\*

参数说明：str1, 字符串型，为('基地编号','专业编号')的封装;str2,字符串型，为实习基地申请的信息;info,字符串型，为要发送的消息

返回值： 无返回值

函数功能：插入实习基地申请

\*/

**public** **void** getRequestBaseInfo(String str1,String str2);

/\*

参数说明：name,为实习基地的名称

返回值： int型，标志是否存在该基地名字

函数功能：判断是否存在该实习基地名称

\*/

**public** **int** CheckName(String name);

/\*

参数说明：sql,为消息的封装

返回值： 无返回值

函数功能：插入消息

\*/

**public** **void** insertMessage(String sql);

}

//实习基地审核逻辑层

**public** **interface** BaseCheckDao {

/\*

参数说明：applydpid,整型，为申请部门编号; pageindex,为当前页数;size,为每页的条数; order,排序列; orderDir,为排序的顺序; searchValue,为模糊查询的值.

返回值： BaseCheckList,该对象包括实习基地申请数据记录+记录条数

函数功能： 获取实习基地申请数据

\*/

**public** BaseCheckList getBaseCheck(**int** applydpid,**int** pageindex,**int** size, String columnName, String orderDir,String searchValue);

/\*

参数说明：无参数值

返回值： List<Map<String,String>>,为Map<String,String>对象的集合

函数功能：获取部门集合

\*/

**public** List<Map<String,String>> getDept(**int** tag);

/\*

参数说明：recordstr，字符串型，为实习基地申请记录编号的封装;infoStr,字符串型，为要发送的消息

返回值： 无返回值

函数功能：申请记录的状态值变为申请失败状态12

\*/

**public** **void** refuseapply(String recordstr);

/\*

参数说明：str,字符串型，为('实习基地编号','建立时间','结束时间')的封装;

infoStr,字符串型，为要发送的消息

recordstr，字符串型，为实习基地申请记录编号的封装

返回值： 无返回值

函数功能：申请记录的状态值变为申请成功状态6,

\*/

**public** **void** agreeApply(String str,String recordstr);

/\*

参数说明：sql,为消息的封装

返回值： 无返回值

函数功能：插入消息

\*/

**public** **void** insertMessage(String sql);

/\*

参数说明：applydpid,整型，为申请部门编号; pageindex,为当前页数;size,为每页的条数; order,排序列;

orderDir,为排序的顺序; searchValue,为模糊查询的值.

返回值： BaseCheckList,该对象包括续期申请数据记录+记录条数

函数功能： 获取续期申请数据

\*/

**public** BaseCheckList getaddCheck(**int** applydpid, **int** pageindex,**int** size,

String columnName, String orderDir,String searchValue);

/\*

参数说明：recordstr，字符串型，为实习基地申请记录编号的封装;infoStr,字符串型，为要发送的消息

返回值： 无返回值

函数功能：同意续期申请，申请记录的状态值变为申请成功状态6,

\*/

**public** **void** addDateApply(String recordstr);

/\*

参数说明：recordstr，字符串型，为实习基地申请记录编号的封装;infoStr,字符串型，为要发送的消息

返回值： 无返回值

函数功能：拒绝续期申请，申请记录的状态值变为到期已失效状态17

\*/

**public** **void** refuseDateApply(String recordstr);

}

//我的实习基地申请的数据交互层

**public** **interface** MyBaseInfoDao {

/\*

参数说明：year,整型，为当前年份;status,整型，为标志值;userid,字符串型，为用户编号;

pageindex,为当前页数;size,为每页的条数; columnName,排序列;

orderDir,为排序的顺序;

返回值： MyBaseList,该对象包括实习基地申请数据记录+记录条数

函数功能：获取该用户当年的特定状态的实习基地申请数据

\*/

**public** MyBaseList MybaseInfo(**int** pageindex, **int** size, String columnName,

String orderDir, **int** year, **int** status, String userid);

/\*

参数说明：sql,为消息的封装

返回值： 无返回值

函数功能：插入消息

\*/

**public** **void** insertMessage(String sql);

/\*

参数说明：id,为实习申请记录的编号

adddate，为续期的月数

返回值： 无返回值

函数功能：续期

\*/

**public** **void** updateDate(**int** id, String adddate);

}

//实习基地管理的数据交互层

**public** **interface** MaintenanceDao {

/\*

参数说明：pageindex,为当前页数;size,为每页的条数; order,排序列;

orderDir,为排序的顺序;

返回值： MaintenanceList,该对象包括实习基地数据记录+记录条数

函数功能：获取该用户实习基地数据

\*/

**public** MaintenanceList maintenance(**int** pageindex, **int** size,String order,String orderDir,String searchValue);

/\*

参数说明：str,为实习基地记录的编号

返回值： 无返回值

函数功能：删除实习基地信息

\*/

**public** **void** delInfo(String str);

/\*

参数说明：无参数

返回值： List<ApplyDept>，为ApplyDept对象的集合

函数功能：获取部门集合

\*/

**public** List<ApplyDept> getExistDept();

/\*

参数说明：basetype，为实习基地类型;dept,为部门;star,为星级;

pageindex,为当前页数;size,为每页的条数; order,排序列;

orderDir,为排序的顺序;

返回值： MaintenanceList,该对象包括实习基地数据记录+记录条数

函数功能：根据条件刷选获取实习基地数据

\*/

**public** MaintenanceList getshaiBaseInfo(**int** basetype,**int** dept,**int** star,**int** pageindex, **int** size,String order,String orderDir,String searchValue);

/\*

参数说明：baseid，为实习基地编号;star,为星级;adddate,为续期月数

返回值： 无返回值

函数功能：修改实习基地截止日期

\*/

**public** **void** updateBaseInfo(String baseid, **int** star, String adddate);

/\*

参数说明：basetype，为实习基地类型;dept,为部门;star,为星级;

返回值： List<ExportBase>，为ExportBase对象的集合

函数功能：获得导出的实习基地数据

\*/

**public** List<ExportBase> getInfo(**int** basetype, **int** dept, **int** star);

/\*

参数说明：str1, 字符串型，为('基地编号','专业编号')的封装;str2,字符串型，为实习基地申请的信息;

返回值： 无返回值

函数功能：增加实习基地

\*/

**public** **void** increaseBaseInfo(String str1, String str2);

}