## 注册用户

* **URL：**

http:// 172.31.24.34:8031/enroll

* **请求方式：**Post
* **请求参数如下:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 元素标识 | 元素全名 | 元素名称 | 参数类型 | 请求 | 备注 |
|  | userName |  | String |  | 用户名 |
|  | peerWithOrg |  | String |  | 用户所属机构 |

* **HTTP头中的状态码200 OK**
* **响应请求参数如下:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 元素标识 | 元素全名 | 元素名称 | 响应 | 备注 |
|  | code | 响应码 | M |  |
|  | data |  | M |  |
|  | msg | 响应内容 | M |  |

* **Data包含一个或者多个JSON数据：**

|  |  |  |  |
| --- | --- | --- | --- |
| 元素标识 | 元素全名 | 参数类型 | 元素名称 |
|  |  |  |  |
|  |  |  |  |

**请求例子:**

**{**

**"userName": "111",**

**"peerWithOrg": "peerOrg1"**

**}**

**返回例子:**

**{**

**"code": "000000",**

**"data": "",**

**"msg": "请求处理成功"**

**}**

## 调用保存

* **URL：**

http:// 172.31.24.34:8031/api/invoke

* **请求方式：**Post
* **请求参数如下:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 元素标识 | 元素全名 | 元素名称 | 参数类型 | 请求 | 备注  (具体见例子) |
|  | args |  | String[] |  | 方法需要调用的参数(以数组的方式提交) |
|  | chainCodeName |  | String |  | **智能合约名称** |
|  | chainCodeVersion |  | String |  | 智能合约版本1.0 |
|  | channelName |  | String |  | **通道名称** |
|  | function |  | String |  | 智能合约里面需要调用的方法名 |
|  | peerWithOrgs |  | String[] |  | 智能合约需要背书的机构 |
|  | belongWithOrg |  | String |  | 用户所属机构 |
|  | userName |  |  |  | 用户名 |

* **响应请求参数如下:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 元素标识 | 元素全名 | 元素名称 | 响应 | 备注 |
|  | code | 响应码 | M |  |
|  | data | 内容 | M |  |
|  | msg | 响应内容 | M |  |
|  |  |  |  |  |

### 保存模型调用结果请求例子:

**{**

**"args": [**

**"RecordId1", //记录id 唯一**

**"BusinessNumber", //业务流水号**

**"BusinessTypeId", //业务类型**

**"EstimateModelId", //模型id**

**"EstimateModelVersion",//模型版本**

**"EstimateModelType",//模型类型**

**"hashdata", //值**

**"Remark",//备注**

**"EstimateDate"//日期**

**],**

**"chainCodeName": "EstimateChainCode",**

**"function": "estimate",**

**"chainCodeVersion": "1.0",**

**"userName": "111",**

**"belongWithOrg": "peerOrg1",**

**"channelName": "idealchannel",**

**"peerWithOrgs": [**

**"peerOrg1"**

**]**

**}返回例子:**

**{**

**"code": "000000",**

**"data": "Transaction invoked successfully",**

**"msg": "请求处理成功"**

**}**

### 向其他机构获取数据请求例子:

{

"args": [

"RecordId2", //记录号 唯一

"BusinessNumber", //业务流水号

"BusinessTypeId", //业务类型

"idNo", //身份证

"realName",//真实姓名

"mobile",//手机

"bankCard",//银行卡号

"source",//数据获取方

"VerifyDate",//日期

"modeName",//模型名称

"https://www.baidu.com/", //数据拉取方url

"id=111&name =222" //数据拉取时需要提交的参数

],

"chainCodeName": "VerifyChainCode",

"function": "verify",

"chainCodeVersion": "1.0",

"userName": "111",

"belongWithOrg": "peerOrg1",

"channelName": "idealchannel",

"peerWithOrgs": [

"peerOrg1"

]

}**返回例子:**

**{**

**"code": "000000",**

**"data": "Transaction invoked successfully",**

**"msg": "请求处理成功"**

**}**

## 查询不含分页

* **URL：**

http:// 172.31.24.34:8031/api/query

* **请求方式：**Post
* **请求参数如下:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 元素标识 | 元素全名 | 元素名称 | 参数类型 | 请求 | 备注 |
|  | args |  | String[] |  | 方法需要调用的参数(以数组的方式提交) |
|  | chainCodeName |  | String |  | **智能合约名称** |
|  | chainCodeVersion |  | String |  | 智能合约版本 |
|  | channelName |  | String |  | **通道名称** |
|  | function |  | String |  | 智能合约里面需要调用的方法名 |
|  | **peerWithOrg** |  | String |  | 用户所属机构 |
|  | **userName** |  | String |  | 用户名 |

* **响应请求参数如下:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 元素标识 | 元素全名 | 元素名称 | 响应 | 备注 |
|  | code | 响应码 | M |  |
|  | data | 内容 | M |  |
|  | msg | 响应内容 | M |  |

* **Data包含一个或者多个JSON数据：**

|  |  |  |  |
| --- | --- | --- | --- |
| 元素标识 | 元素全名 | 参数类型 | 元素名称 |

### 模型调用查询例子:

**根据流水号查询**

**{**

**"userName" : "111",**

**"args" : ["BusinessNumber"],**

**"channelName" : "idealchannel",**

**"chainCodeName" : "EstimateChainCode",**

**"chainCodeVersion": "1.0",**

**"function" : "queryByBN",**

**"peerWithOrg" :"peerOrg1"**

**}返回例子:**

**{**

**"code": "000000",**

**"data": "{\"certificateId\":\"certificateId1\",\"certificatePath\":\"certificatePath\",\"certificateType\":\"certificateType2\",\"code\":\"code\",\"codeStatus\":\"codeStatus\",\"createDate\":\"2018-05-02 06:44:56\",\"email\":\"email\",\"fileHash\":\"fileHash\",\"fileName\":\"fileName\",\"filePath\":\"filePath\",\"fileType\":\"fileType\",\"name\":\"name\",\"note\":\"note\",\"phone\":\"phone\",\"userId\":\"userId1\"}",**

**"msg": "请求处理成功"**

**}**

### 拉取数据调用查询例子:

**根据流水号查询**

**{**

**"userName" : "111",**

**"args" : ["BusinessNumber"],**

**"channelName" : "idealchannel",**

**"chainCodeName" : "VerifyChainCode",**

**"chainCodeVersion": "1.0",**

**"function" : " queryListByBusinessNumber**

**",**

**"peerWithOrg" :"peerOrg1"**

**}返回例子:**

**{**

**"code": "000000",**

**"data": "{\"certificateId\":\"certificateId1\",\"certificatePath\":\"certificatePath\",\"certificateType\":\"certificateType2\",\"code\":\"code\",\"codeStatus\":\"codeStatus\",\"createDate\":\"2018-05-02 06:44:56\",\"email\":\"email\",\"fileHash\":\"fileHash\",\"fileName\":\"fileName\",\"filePath\":\"filePath\",\"fileType\":\"fileType\",\"name\":\"name\",\"note\":\"note\",\"phone\":\"phone\",\"userId\":\"userId1\"}",**

**"msg": "请求处理成功"**

**}**

### 模型调用所有数据查询

**{**

**"userName" : "111",**

**"args" : [],**

**"channelName" : "idealchannel",**

**"chainCodeName" : "EstimateChainCode",**

**"chainCodeVersion": "1.0",**

**"function" : "queryAllList",**

**"peerWithOrg" :"peerOrg1"**

**}**

### 拉取数据所有数据查询

**{**

**"userName" : "111",**

**"args" : [],**

**"channelName" : "idealchannel",**

**"chainCodeName" : "VerifyChainCode",**

**"chainCodeVersion": "1.0",**

**"function" : "queryAllList",**

**"peerWithOrg" :"peerOrg1"**

**}**

## 查询含分页

* **URL：**

http:// 172.31.24.34:8031/api/query

* **请求方式：**Post
* **请求参数如下:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 元素标识 | 元素全名 | 元素名称 | 参数类型 | 请求 | 备注 |
|  | args |  | String[] |  | 方法需要调用的参数(以数组的方式提交) |
|  | chainCodeName |  | String |  | **智能合约名称** |
|  | chainCodeVersion |  | String |  | 智能合约版本 |
|  | channelName |  | String |  | **通道名称** |
|  | function |  | String |  | 智能合约里面需要调用的方法名 |
|  | **peerWithOrg** |  | String |  | 用户所属机构 |
|  | **userName** |  | String |  | 用户名 |
|  |  |  |  |  |  |

* **响应请求参数如下:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 元素标识 | 元素全名 | 元素名称 | 响应 | 备注 |
|  | code | 响应码 | M |  |
|  | data | 内容 | M |  |
|  | msg | 响应内容 | M |  |

* **Data包含一个或者多个JSON数据：**

|  |  |  |  |
| --- | --- | --- | --- |
| 元素标识 | 元素全名 | 参数类型 | 元素名称 |

### 模型调用查询例子:

**根据流水号查询**

**{**

**"userName" : "111",**

**"args" : ["BusinessNumber","2",""],**

**//最后2个参数 一个页面显示条数,一个是bookmark 第一次请求时bookmark可以为空 bookmarks为当前页面的标记 请求下一页是 需带上上一次返回的bookmark**

**"channelName" : "idealchannel",**

**"chainCodeName" : "EstimateChainCode",**

**"chainCodeVersion": "1.0",**

**"function" : "** **queryPageByBn",**

**"peerWithOrg" :"peerOrg1"**

**}返回例子:**

**{**

**"code": "000000",**

**"data": "{\"records\":[{\"businessNumber\":\"BusinessNumber\",\"businessTypeId\":\"BusinessTypeId\",\"estimateDate\":\"EstimateDate\",\"estimateModelId\":\"EstimateModelId\",\"estimateModelType\":\"EstimateModelType\",\"estimateModelVersion\":\"EstimateModelVersion\",\"hashData\":\"63a3ab06748cf7da223f62dec1346d4a\",\"recordId\":\"RecordId2\",\"remark\":\"Remark\",\"txId\":\"180608f7e531a5006ece1c137a133bd198e93c5e6adf17ddd84429b944de0ee6\"},{\"businessNumber\":\"BusinessNumber\",\"businessTypeId\":\"BusinessTypeId\",\"estimateDate\":\"EstimateDate\",\"estimateModelId\":\"EstimateModelId\",\"estimateModelType\":\"EstimateModelType\",\"estimateModelVersion\":\"EstimateModelVersion\",\"hashData\":\"8058e0e1622877f1c275da54f11e2d17\",\"recordId\":\"RecordId1\",\"remark\":\"Remark\",\"txId\":\"60cabbc4f9dadd33b37a80bd3376f63fe1771b194234b6a74b2f52bab64053bc\"}],\"RecordsCount\":\"2\", \"Bookmark\":\"g1AAAABYeJzLYWBgYMpgSmHgKy5JLCrJTq2MT8lPzkzJBYqLMLgWl2TmJpakMgSlJucXpXimGDKA1HLA1OJUlQUAVP8Zzw\"}",**

**"msg": "请求处理成功"**

**}**

**RecordsCount 为当前记录返回条数**

**Bookmark 为当前页的标志位 作为查询下一页是需要传递的参数**

### 拉取数据查询例子:

**根据流水号查询**

**{**

**"userName" : "111",**

**"args" : ["BusinessNumber","2",""],**

**//最后2个参数 一个页面显示条数,一个是bookmark 第一次请求时bookmark可以为空 bookmarks为当前页面的标记 请求下一页是 需带上上一次返回的bookmark**

**"channelName" : "idealchannel",**

**"chainCodeName" : "VerifyChainCode",**

**"chainCodeVersion": "1.0",**

**"function" : " queryListForPage**

**",**

**"peerWithOrg" :"peerOrg1"**

**}返回例子:**

**{**

**"code": "000000",**

**"data": "{\"records\":[{\"businessNumber\":\"BusinessNumber\",\"businessTypeId\":\"BusinessTypeId\",\"estimateDate\":\"EstimateDate\",\"estimateModelId\":\"EstimateModelId\",\"estimateModelType\":\"EstimateModelType\",\"estimateModelVersion\":\"EstimateModelVersion\",\"hashData\":\"63a3ab06748cf7da223f62dec1346d4a\",\"recordId\":\"RecordId2\",\"remark\":\"Remark\",\"txId\":\"180608f7e531a5006ece1c137a133bd198e93c5e6adf17ddd84429b944de0ee6\"},{\"businessNumber\":\"BusinessNumber\",\"businessTypeId\":\"BusinessTypeId\",\"estimateDate\":\"EstimateDate\",\"estimateModelId\":\"EstimateModelId\",\"estimateModelType\":\"EstimateModelType\",\"estimateModelVersion\":\"EstimateModelVersion\",\"hashData\":\"8058e0e1622877f1c275da54f11e2d17\",\"recordId\":\"RecordId1\",\"remark\":\"Remark\",\"txId\":\"60cabbc4f9dadd33b37a80bd3376f63fe1771b194234b6a74b2f52bab64053bc\"}],\"RecordsCount\":\"2\", \"Bookmark\":\"g1AAAABYeJzLYWBgYMpgSmHgKy5JLCrJTq2MT8lPzkzJBYqLMLgWl2TmJpakMgSlJucXpXimGDKA1HLA1OJUlQUAVP8Zzw\"}",**

**"msg": "请求处理成功"**

**}**

### 模型调用所有数据分页查询

**{**

**"userName" : "111",**

**"args" : [“2”,””],**

**"channelName" : "idealchannel",**

**"chainCodeName" : "EstimateChainCode",**

**"chainCodeVersion": "1.0",**

**"function" : "queryAllList",**

**"peerWithOrg" :"peerOrg1"**

**}**

### 拉取数据所有数据分页查询

**{**

**"userName" : "111",**

**"args" : [“2”,””],**

**"channelName" : "idealchannel",**

**"chainCodeName" : "VerifyChainCode",**

**"chainCodeVersion": "1.0",**

**"function" : "queryAllList",**

**"peerWithOrg" :"peerOrg1"**

**}**

## 数据库信息

**172.31.24.34:**3306 root/123456

数据库 fabric 表fabric\_ca\_user 保存用户登录的信息