

说明

- 提交参数以 键值对的数据格式提交（ **application/x-www-form-urlencoded** ），浏览器将form数据封装到http body中。
- 认证方式（ **/hrm/websevice/hrm_webservice_config.jsp** 提供认证方式设置 ）：
 - IP方式：在/WEB-INF/prop/HrmWebserviceIP.properties配置客户端IP；多个值的以逗号分隔。
 - TOKEN认证：提交参数额外添加

名称	类型	必选	说明
token	string	是	json字符串，1个token只有5分钟的有效期（请求端时间与服务端时间原则是一致的），建议每次请求产生新的token。

■ 示例：

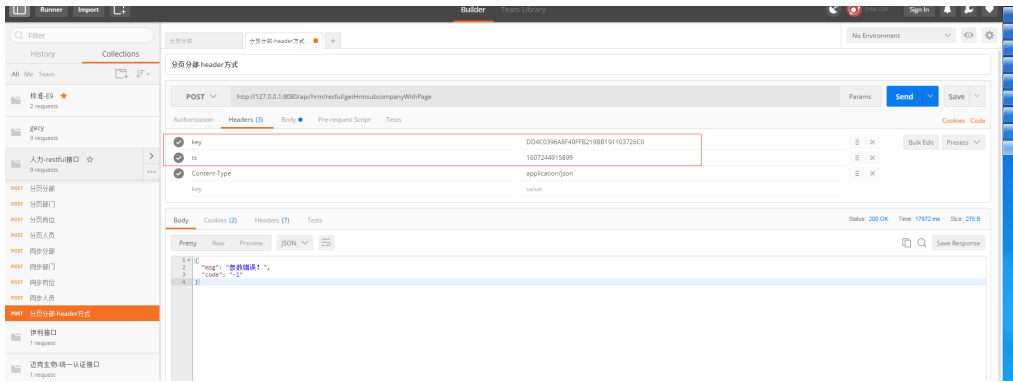
```
1 {
2     "key": "92B4BF9322C110E225683F048013B699", // {标识}+{时间戳毫秒数}, 然后进行 md5 / sm3 加密, 产生key ;
3     "ts": "1603600893041" // 时间戳毫秒数
4 }
```

■ key的生成方式，java示例：

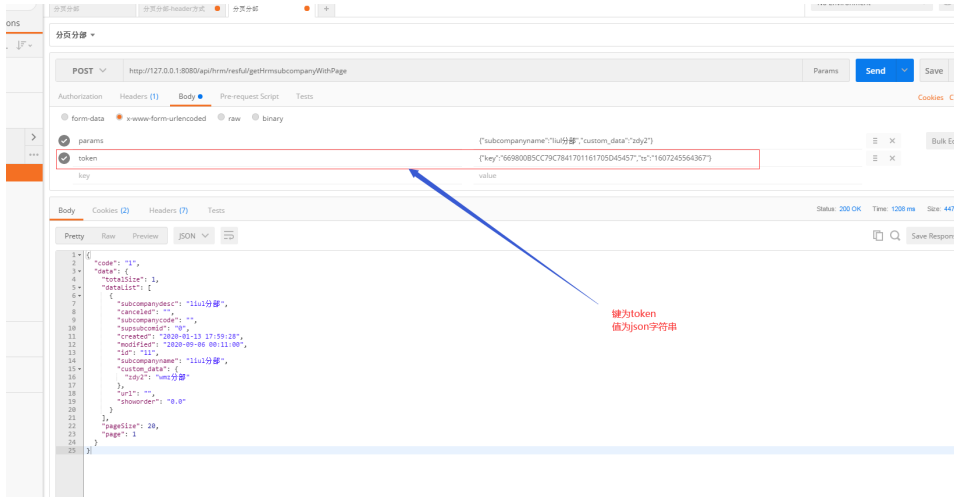
```
1 public static void main(String[] args) throws Exception{
2     String key = "5936562a-d47c-4a29-9b74-b310e6c971b7" ; //
标识 获取方式, 请访问 /hrm/websevice/hrm_webservice_config.jsp 进行
获取
3     long l = System.currentTimeMillis() ; // 时间戳毫秒数
4     String code = key.concat(Long.toString(l)) ;
5     String md5key = Util.getEncrypt(code).toUpperCase() ; //
md5加密 然后转大写
6     Map<String,String> map = new HashMap<>() ;
7     map.put("key",md5key) ;
8     map.put("ts",Long.toString(l)) ;
9     System.out.println(JSONObject.toJSONString(map)); //
token字符串
10 }
```

■ 示例（下面2种设置token的方式任选一个）

- token设置方式1，使用header方式，放入 key, ts 2个参数



- token设置方式2,直接在body中添加参数



1.获取分部列表

请求URL

- /api/hrm/resful/getHrmsubcompanyWithPage

请求方式

- POST

参数

名称	类型	必选	说明
params	string	是	json格式数据

- 请求参数-json示例

名称	类型	必选	说明
subcompanyname	string		分部名称
subcompanycode	string		分部编号
modified	string		修改时间戳；（使用>=）
canceled	string		封存标志；默认查询非封存数据。1:封存。
custom_data	string		指定获取自定义字段数据字段列表，（多个逗号分隔），具体参考【组织权限中心-自定义设置-分部字段定义】；示例：zdy2
id	string		oa中分部id
pagesize	int		页大小，默认20
curpage	int		当前页，默认1

- 示例

```

1  {
2      "subcompanyname": "分部",
3      "subcompanycode": "code1",
4      "modified": "2020-11-08 00:00:00",
5      "created": "2020-11-08 00:00:00",
6      "canceled": "",
7      "custom_data": "field1,field2",
8      "id": "10",
9      "pagesize": 20,
10     "curpage": 1
11 }
```

返回示例

```

1  {
2      "code": "1", // 1 成功，其他为失败
3      "data": {
4          "totalSize": 1, // 分页-总数
5          "dataList": [ // 数据集
6              {
7                  "subcompanydesc": "liu1分部", // 分部全称
8                  "canceled": "", // 封存标志，1 封存，其他为未封存
9                  "subcompanycode": "", // 分部编码
10                 "supsubcomid": "0", // 上级分部id，0 或者 空为表示没有上级分部
11                 "created": "2020-01-13 17:59:28", // 创建时间戳
12                 "modified": "2020-09-06 00:11:00", // 修改时间戳
13                 "id": "11", // 分部id，唯一标识id
14                 "subcompanyname": "liu1分部", // 分部简称
15                 "url": "", // 废弃字段
16                 "custom_data": { // 自定义数据
```

```
17         "zdy2": "wmz分部" // 自定义数据，具体参考 【组织权限中心-自定义设置-分部字段定义】
18     },
19     "showorder": "0.0" // 排序
20 }
21 ],
22 "pageSize": 20, // 分页-单页条目数
23 "page": 1 // 分页-当前页数
24 }
25 }
```

2.获取部门列表

请求URL

- /api/hrm/resful/getHrmdepartmentWithPage

请求方式

- POST

参数

名称	类型	必选	说明
params	string	是	json格式数据

- json参数说明

名称	类型	必选	说明
created	string		创建时间戳；（使用>=）
modified	string		修改时间戳；（使用>=）
departmentcode	string		编号
departmentname	string		名称
subcompanyid1	string		分部id
canceled	string		封存标志；默认查询非封存数据。1:封存。
id	string		oa中部门id
pagesize	int		页大小，默认20
curpage	int		当前页，默认1
custom_data	string		值内容是指定获取OA自定义字段的列表（具体看【组织权限中心】-【自定义设置】-【部门字段定义】），示例: custom_data:'x1k,field2,field3,field4' 获取自定义信息中

- 示例

```
1 {
2   "departmentname": "部门",
3   "departmentcode": "code1",
4   "pagesize": 20,
5   "curpage": 1
6 }
```

返回示例

```
1 {
2   "code": "1", // 1 成功，其他为失败；
3   "data": {
4     "totalSize": 1, // 分页-总数
5     "dataList": [ // 数据集合
6       {
7         "canceled": "", // 封存标志，1 封存，其他为未封存
8         "supdepid": "0", // 上级部门id
9         "departmentmark": "yhm-主分部A-下级部门一", // 简称
10        "departmentname": "yhm-主分部A-下级部门一", // 全称
11        "created": "2019-08-28 10:44:12", // 创建时间戳
12        "departmentcode": "", // 编码
13        "modified": "2020-09-07 19:40:09", // 修改时间戳
14        "id": "2", // 部门id
15        "subcompanyid1": "5", // 分部id
16        "custom_data": { // 自定义数据
17          "x1k": "使用" //
18        },
19        "showorder": "0.0" // 自定义数据，具体参考 【组织权限中心-自定义设置-部门字段定义】
20      }
21    ],
22    "pageSize": 20, // 分页-单页条目数
23    "page": 1 // 分页-当前页数
24  }
25 }
```

3.获取岗位列表

请求URL

- /api/hrm/resful/getJobtitleInfoWithPage

请求方式

- POST

参数

名称	类型	必选	说明
params	string	是	json格式数据

- json参数说明

名称	类型	必选	说明
created	string		创建时间戳；（使用>=）
modified	string		修改时间戳；（使用>=）
id	string		岗位id
jobtitlename	string		岗位名称
curpage	int		当前页，默认1
pagesize	int		页大小，默认20

返回示例

```
1  {
2    "code": "1", // 1 成功 ， 其他为失败；
3    "data": {
4      "totalSize": 1, // 分页-总数
5      "dataList": [ // 数据集合
6        {
7          "jobdoc": "", // 相关文档id
8          "jobtitleremark": "", // 备注
9          "jobresponsibility": "", // 职责
10         "jobtitlename": "Default", // 全称
11         "jobcompetency": "", // 任职资格
12         "created": "2019-03-07 13:12:47", // 创建时间戳
13         "modified": "2019-03-07 13:12:47", // 修改时间戳
14         "id": "1", // 岗位id
15         "jobtitlemark": "Default", // 简称
16         "jobdepartmentid": "" // 部门id（废弃字段）
17       }
18     ],
19     "pageSize": 20, // 分页-单页条目数
20     "page": 1 // 分页-当前页数
21   }
22 }
```

4.获取人员信息列表

请求URL

- /api/hrm/resful/getHrmUserInfoWithPage

请求方式

- POST

参数

名称	类型	必选	说明
params	string	是	json格式数据

- json参数说明

名称	类型	必选	说明
created	string		创建时间戳；（使用>=）
modified	string		修改时间戳；（使用>=）
workcode	string		编号
subcompanyid1	string		分部id
departmentid	string		部门id
jobtitleid	string		岗位id
id	string		人员id
loginid	string		登录名
base_custom_data	string		指定获取OA自定义字段的列表（具体看【组织权限中心】-【自定义设置】-【人员卡片字段定义】基本信息），示例: base_custom_data:'field1,field2,field3,field4' 获取基本信息中
person_custom_data	string		同base_custom_data解释（个人信息）
work_custom_data	string		同work_custom_data解释（工作信息）
pagesize	int		页大小，默认20
curpage	int		当前页，默认1

返回示例

```
1 {
2   "code": "1", // 1 成功, 其他失败
3   "data": {
4     "totalSize": 1, // 分页-总数
5     "dataList": [ // 数据列表
6       {
7         "tempresidentnumber": "xxxx", // 暂住证号码
8         "person_custom_data": { // 个人自定义数据
9           "field6": "2020-09-10"
10        },
11        "createdate": "2019-08-28", // 创建日期
12        "language": "", // 系统语言
13        "subcompanyid1": "5", // 分部id
14        "subcompanyname": "yhm-主分部A", // 分部名称
15        "joblevel": "0", // 职级
16        "startdate": "", // 合同开始日期
17        "password": "lK1l5Yaeg5csNkZvr/uaYnioLAv02khMFnUS9/pv8Tg=",
        // 密码 密文
18        "subcompanycode": "", // 分部编码
19        "jobactivitydesc": "xxxx", // 职责描述
20        "bememberdate": "2010-08-08", // 入团日期
21        "modified": "2020-09-09 19:46:40", // 修改时间
22        "id": "3", // 人员id
23        "mobilecall": "123456", // 其他电话
24        "nativeplace": "湖北", // 籍贯
25        "certificatenum": "1", // 身份证
26        "height": "0", // 身高
27        "loginid": "yhm01", // 登录名
28        "created": "2019-08-28 10:44:12", // 创建日期(旧字段)
29        "degree": "xxx", // 学位
30        "bepartydate": "2010-08-10", // 入党日期
31        "weight": "110", // 体重
32        "telephone": "123", // 办公电话
33        "residentplace": "xxxx", // 户口
34        "lastname": "yhm01tt", // 人员名称
35        "healthinfo": "1", // 健康状况
36        "enddate": "", // 合同结束日期
37        "maritalstatus": "已婚", // 婚姻状况
38        "departmentname": "yhm-主分部A-下级部门一", // 部门名称
39        "folk": "汉", // 民族
40        "status": "1", // 状态: 0 试用 1 正式 2 临时 3 试用延期 4 解聘 5
        离职 6 退休 7 无效
41        "birthday": "2010-10-10", // 生日
42        "accounttype": "0", // 主次账号标志: 1 次账号, 其他 主账号
43        "jobcall": "2", // 职称
44        "managerid": "0", // 上级人员id
45        "assistantid": "0", // 助理人员id
46        "departmentcode": "", // 部门编码
47        "belongto": "", // 主账号id (当accounttype 为 1 有效)
48        "email": "123123123@123.com", // 邮箱
49        "secllevel": "10", // 安全级别
50        "policy": "团员", // 政治面貌
51        "jobtitle": "2", // 岗位id
52        "workcode": "yhm01tt", // 编号
```



```

53         "sex": "男", // 性别
54         "departmentid": "2", // 部门id
55         "homeaddress": "xxxx", // 家庭联系方式
56         "mobile": "123123123", // 移动电话
57         "lastmoddate": "2020-09-09", // 最后修改日期
58         "educationlevel": "3", // 学历
59         "islabouunion": "0", // 工会会员
60         "base_custom_data": { // 基本信息自定义数据
61             "field1": "test_yhm01",
62             "field3": "liu1部门1,liu1部门2",
63             "field2": "测试人员1"
64         },
65         "locationid": "2", // 办公地点
66         "regresidentplace": "武汉", // 户口
67         "dsporder": "3.0" // 排序
68     }
69 ],
70 "pageSize": 20, // 分页-单页条目数
71 "page": 1 // 分页-当前页数
72 }
73 }

```

5.同步分部

请求URL

- /api/hrm/resful/synSubcompany

请求方式

- POST

参数

名称	类型	必选	说明
data	String		同步数据体，json格式数据

- 示例

```

1  [
2      {
3          "@action": "add", // add 新增, edit 编辑 delete 删除（只会处理canceled字
          段）
4          "code": "test", // 分部编号
5          "shortname": "测试-分部简称", // 分部简称
6          "fullname": "测试-分部全称", // 分部全称
7          "parent_code": "上级分部编码", // 上级分部编码
8          "canceled": "0", // 封存标志 （@action 为 delete有效, 0 表示封存该分部, 1
          表示解封）
9          "order": "3", // 排序
10         "custom_data": { // 自定义数据

```

```
11 //字段确认请看【组织权限中心】-【自定义设置】-【分部字段定义】 数据库字段
12 名
13 "zdy2": "wmz分部",
14 "zdy3": "测试人员1,yhm05"
15 }
16 ]
```

返回示例

```
1 {
2   "result": {
3     "test": { //分部编号
4       "msg": "插入成功",
5       "code": "1"
6     }
7   },
8   "code": "1" // 1 成功
9 }
```

6.同步部门

请求URL

- /api/hrm/resful/synDepartment

请求方式

- POST

参数

名称	类型	必选	说明
data	String		同步数据体，json格式数据

- 示例

```
1 [
2   {
3     "@action": "add", // add 新增, edit 编辑 delete 删除（只会处理canceled字
4     "code": "test", // 编号
5     "shortname": "测试-部门简称", // 简称
6     "fullname": "测试-部门全称", // 全称
7     "org_code": "test", // 分部编号
8     "parent_code": "super_test", // 上级部门编号
9     "canceled": "0", // 封存标志，（@action 为 delete 有效，0 表示封存该分
10    "order": "3", // 排序
```

```
11         "custom_data":{ // 自定义数据
12             //字段确认请看【组织权限中心】-【自定义设置】-【部门字段定义】 数据库字段名
13             "xlk":"不用"
14         }
15     }
16 ]
```

返回示例

```
1  {
2      "result": {
3          "test": { //部门编号
4              "msg": "插入成功",
5              "code": "1"
6          }
7      },
8      "code": "1" // 1 成功
9  }
```

7.同步岗位

请求URL

- /api/hrm/resful/synJobtitle

请求方式

- POST

参数

名称	类型	必选	说明
data	String		同步数据体，json格式数据

- 示例

```
1  [
2      {
3          "@action":"add", // add 新增, edit 编辑 delete 删除
4          "jobtitlecode":"job_test", // 岗位编号
5          "jobtitlename":"测试-岗位简称", // 简称
6          "jobtitleremark":"测试-岗位全称", // 全称
7          "jobtitledept":"" // 废弃字段
8      }
9  ]
```

返回示例

```
1 {
2   "result": {
3     "job_test": { //岗位编号
4       "msg": "操作成功",
5       "code": "1"
6     }
7   },
8   "code": "1" // 1 成功
9 }
```

8.同步人员

请求URL

- /api/hrm/resful/synHrmresource

请求方式

- POST

参数

名称	类型	必选	说明
data	String		同步数据体，json格式数据

- 示例

```
1 [
2   {
3     "workcode" : "test_code", // 编号
4     "subcompany" : "东展集团-pyy>东北分公司-pyy", // 分部：2种方式： 1)
        分部全路径（副作用：未找到，会新增），2) 以 {JSON}{"subcompanycode":"fw"} 为值的
        字符串，指定分部编码。（不会新增）
5     "department" : "东北-客服中心-pyy", // 部门：2种方式：1) 部门全路径
        （副作用：未找到，会新增），2) 以 {JSON}{"departmentcode":"fw"} 为值的字符串，
        指定部门编码（不会新增）
6     "lastname" : "测试人员1", // 人员名称
7     "loginid" : "test_loginid", // 登录名称
8     "password" : "123", // 明文密码，会转密文
9     "secllevel" : "3", // 安全级别
10    "sex" : "女", // 枚举值：默认 男 女 | 男
11    "jobtitle":"测试岗位", // 岗位名称：2种方式：1) 名称 （会新建） 2) 以
        {JSON}{"jobtitlecode":"fw"} 为值的字符串，指定岗位编码（不会新增）
12    "jobactivityid":"秘书", // 职务 岗位使用json格式不需要传递
13    "jobgroupid":"秘书类", // 职务类型 岗位使用json格式不需要传递
14    "jobcall":"2", // 职称 数据关联表: HrmJobCall
15    "joblevel":"0", // 职级 仅支持整数字
16    "jobactivitydesc":"职责描述_xxt", // 职责描述
17    "managerid":"test", // 直接上级 仅支持编号
```

```

18     "assistantid":"test", // 助理 仅支持编号
19     "status":"正式", // 状态 试用 | 正式 | 临时 | 试用延期 | 解聘 | 离
    职 | 退休 | 无效
20     "locationid":"上海", // 办公地点
21     "workroom":"办公地点", // 办公室
22     "telephone":"123-23334", // 办公室电话
23     "mobile":"123459877112", // 移动电话
24     "mobilecall":"123456", // 其他电话
25     "fax":"fx", // 传真
26     "email":"test@qq.com", // 邮箱
27     "systemlanguage":"简体中文", // 系统语言 简体中文 | 繁體中文 |
    English
28     "birthday":"2020-01-01", // 生日 格式: yyyy-MM-dd
29     "folk":"", // 民族
30     "nativeplace":"", // 籍贯
31     "regresidentplace":"", // 户口
32     "certificatenum":"", // 身份证号码 如果有值, 必须唯一
33     "maritalstatus":"已婚", // 婚姻状况 已婚 | 离异 | 未婚
34     "policy":"", // 政治面貌
35     "bememberdate":"2020-01-01", // 入团日期 格式: yyyy-MM-dd
36     "bepartydate":"1999-01-01", // 入党日期 格式: yyyy-MM-dd
37     "islabouunion":"", // 工会会员
38     "educationlevel":"", // 学历 数据关联表: HrmEducationLevel
39     "degree":"", // 学位
40     "healthinfo":"", // 健康状况 良好 | 一般 | 较差 | 优秀
41     "height":"100", // 身高(cm) 仅支持整数字
42     "weight":"130", // 体重(kg) 仅支持整数字
43     "residentplace":"", // 现居住地
44     "homeaddress":"", // 家庭联系方式
45     "tempresidentnumber":"", // 暂住证号码
46     "belongto":"", //主账号: accounttype为次账号 才有效。 仅支持编号
47     "dsporder":"", // 排序 数字
48     "accumfundaccount":"",
49     "accounttype":"", // 主次账号标志 主账号 | 次账号
50     "accountname":"", // 工资账号户名
51     "bankid1":"", // 工资银行
52     "accountid1":"", // 工资账号
53     "companystartdate":"", // 入职日期
54     "workstartdate":"", // 参加工作日期
55     "base_custom_data":{ // 基本信息自定义数据
56         //字段确认请看【组织权限中心】-【自定义设置】-【人员卡片字段定义】-
        【基本信息】 数据库字段名
57         "field1":"test"
58     },
59     "person_custom_data":{ // 个人信息自定义数据
60         // 字段确认请看【组织权限中心】-【自定义设置】-【人员卡片字段定义】-
        【个人信息】 数据库字段名
61         "field1":"张三,李四",
62         "field2":"在职"
63     },
64     "work_custom_data":{ // 工作信息自定义数据
65         // 字段确认请看【组织权限中心】-【自定义设置】-【人员卡片字段定义】-
        【工作信息】 数据库字段名
66         "field2" : "120"
67     }
68 }
69 ]

```

返回示例

```
1 {  
2   "result": {  
3     "workCode": { //编号  
4       "msg": "操作成功",  
5       "code": "1"  
6     }  
7   },  
8   "code": "1" // 1 成功  
9 }
```