**前端页面API设计**

初始化页面

@Path(“/Server”)

@Path(“/InitPage”)

Path-->localhost:8080/Model/rest/Server/InitPage

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:{成功返回的json数据集},

failMessage:’error’

}

创建服务

@Path(“/Server”)

@Path(“/CreateServer”)

Path-->localhost:8080/Model/rest/Server/InitPage

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:{成功返回的json数据集},

failMessage:’error’

}

编辑服务

@Path(“/Server”)

@Path(“/EditServer/{ServerCode}”)

Path-->localhost:8080/Model/rest/Server/EditServer/{ServerCode}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:{成功返回的json数据集},

failMessage:’error’

}

更新服务

@Path(“/Server”)

@Path(“/UpdateServer”)

Path-->localhost:8080/Model/rest/Server/UpdateServer

前端--->Model服务端:

{

"F1":"服务编码",

"F2":"服务名称",

"F3":"模型服务说明",

"F4":"集群名称",

"F5":"IPV4",

"F6":"IPV6",

"F7":"端口",

"F8":"URL地址",

"F9":"注册码",

"F10":"创建时间",

"F11":"是否有效"

}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:’success message’,

failMessage:’error’

}

打开服务

@Path(“/Server”)

@Path(“/OpenServer/{ServerCode}”)

Path-->localhost:8080/Model/rest/Server/OpenServer/{ServerCode}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:{成功返回的json数据集},

failMessage:’error’

}

创建服务单元

编辑服务单元

@Path(“/Server”)

@Path(“/EditServerUnit/{ServerUnitCode}”)

Path-->localhost:8080/Model/rest/Server/EditServerUnit/{ServerUnitCode}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:{成功返回的json数据集},

failMessage:’error’

}

更新服务单元

@Path(“/Server”)

@Path(“/UpdateServerUnit”)

Path-->localhost:8080/Model/rest/Server/UpdateServerUnit

前端--->Model服务端:

{

"F1":"服务编码",

"F2":"服务单元编码",

"F3":"服务单元名称",

"F4":"服务单元说明",

"F5":"报警主题编码",

"F6":"报警主题名称",

"F7":"驱动jar包文件名",

"F8":"接口方法名”,

"F9":"创建时间",

"F10":"是否有效"

}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:’success message’,

failMessage:’error’

}

删除服务单元

创建实时对象

@Path(“/Server”)

@Path(“/CreateRTObject”)

Path-->localhost:8080/Model/rest/Server/CreateRTObject

前端--->Model服务端:

{

"ServerCode":"服务编码",

"ServerUnitCode":"服务单元编码"

}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:’success message’,

failMessage:’error’

}

编辑实时对象

@Path(“/Server”)

@Path(“/EditRtObject/{RtObjectCode}”)

Path-->localhost:8080/Model/rest/Server/EditRtObject/{RtObjectCode}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:{成功返回的json数据集},

failMessage:’error’

}

更新实时对象

@Path(“/Server”)

@Path(“/UpdateRtObject”)

Path-->localhost:8080/Model/rest/Server/UpdateRtObjec

前端--->Model服务端:

{

"F1":"服务编码",

"F2":"服务单元编码",

"F3":"实时对象编码",

"F4":"实时对象名称",

"F5":"实时对象说明",

"F6":"创建时间",

"F7":"是否有效"

}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:’success message’,

failMessage:’error’

}

删除实时对象

@Path(“/Server”)

@Path(“/DeleteRtObject/{RtObjectCode}”)

Path-->localhost:8080/Model/rest/Server/DeleteRtObject/{RtObjectCode}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:’success message’,

failMessage:’error’

}

创建实时对象属性目录

@Path(“/Server”)

@Path(“/CreateRTOAttrCatalog”)

Path-->localhost:8080/Model/rest/Server/CreateRTOAttrCatalog

前端--->Model服务端:

{"ServerCode":"100000000004",

"NodeCode":"1110000000000131",

"NodeName":"DI",

"CategoryCode":"1111000",

"ObjectStatus":"2",

"ParentNode":"111000000000013"

}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:’success message’,

failMessage:’error’

}

删除实时对象属性目录

@Path(“/Server”)

@Path(“/DeleteRTOAttrCatalog/{RTOAttrCatalogCode}”)

Path-->localhost:8080/Model/rest/Server/DeleteRTOAttrCatalog/{RTOAttrCatalogCode}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:’success message’,

failMessage:’error’

}

创建实时对象属性

@Path(“/Server”)

@Path(“/CreateRTOAttr”)

Path-->localhost:8080/Model/rest/Server/CreateRTOAttr

前端--->Model服务端:

{

"ServerCode":"100000000004",

"RTOCode":"111000000000013",

"ParentId":"1110000000000132",

"DotCategory":"2",

"AttrName":"DO",

"AttrDesc":"111000000000013下的属性点"

}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:’success message’,

failMessage:’error’

}

编辑实时对象属性

@Path(“/Server”)

@Path(“/EditRTOAttr/{RTOAttrCode}”)

Path-->localhost:8080/Model/rest/Server/EditRTOAttr/{RTOAttrCode}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:{成功返回的json数据集},

failMessage:’error’

}

修改实时对象属性

@Path(“/Server”)

@Path(“/UpdateRTOAttr”)

Path-->localhost:8080/Model/rest/Server/UpdateRTOAttr

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:’success message’,

failMessage:’error’

}

删除实时对象属性

@Path(“/Server”)

@Path(“/DeleteRTOAttr/{RTOAttrCode}”)

Path-->localhost:8080/Model/rest/Server/DeleteRTOAttr/{RTOAttrCode}

Model服务端--->前端:

{

isSuccess:1或0,

resultMessage:’success message’,

failMessage:’error’

}

创建数据对象

@Path(“/Server”)

@Path(“/CreateDataObject”)

Path-->localhost:8080/Model/rest/Server/CreateDataObject

编辑数据对象

@Path(“/Server”)

@Path(“/EditDataObject/{DataObjectCode}”)

Path-->localhost:8080/Model/rest/Server/EditDataObject/{DataObjectCode}

修改数据对象

@Path(“/Server”)

@Path(“/UpdateDataObject”)

Path-->localhost:8080/Model/rest/Server/UpdateDataObject

删除数据对象

@Path(“/Server”)

@Path(“/DeleteDataObject/{DataObjectCode}”)

Path-->localhost:8080/Model/rest/Server/DeleteDataObject/{DataObjectCode}

显示创建的子节点

@Path(“/Server”)

@Path(“/ChildrenNode/{parentCode}”)

Path-->localhost:8080/Model/rest/Server/ChildrenNode/{parentCode}