



最佳实践

文档版本: 20220425



# 法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。 如果您阅读或使用本文档,您的阅读或使用行为将被视为对本声明全部内容的认可。

- 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档,且仅能用 于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息,您应当严格 遵守保密义务;未经阿里云事先书面同意,您不得向任何第三方披露本手册内容或 提供给任何第三方使用。
- 未经阿里云事先书面许可,任何单位、公司或个人不得擅自摘抄、翻译、复制本文 档内容的部分或全部,不得以任何方式或途径进行传播和宣传。
- 由于产品版本升级、调整或其他原因,本文档内容有可能变更。阿里云保留在没有 任何通知或者提示下对本文档的内容进行修改的权利,并在阿里云授权通道中不时 发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠 道下载、获取最新版的用户文档。
- 4. 本文档仅作为用户使用阿里云产品及服务的参考性指引,阿里云以产品及服务的"现状"、"有缺陷"和"当前功能"的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引,但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的,阿里云不承担任何法律责任。在任何情况下,阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害,包括用户使用或信赖本文档而遭受的利润损失,承担责任(即使阿里云已被告知该等损失的可能性)。
- 5. 阿里云网站上所有内容,包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计,均由阿里云和/或其关联公司依法拥有其知识产权,包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意,任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外,未经阿里云事先书面同意,任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称(包括但不限于单独为或以组合形式包含"阿里云"、"Aliyun"、"万网"等阿里云和/或其关联公司品牌,上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司)。
- 6. 如若发现本文档存在任何错误,请与阿里云取得直接联系。

# 通用约定

格式	说明	样例
⚠ 危险	该类警示信息将导致系统重大变更甚至故 障,或者导致人身伤害等结果。	⚠ 危险 重置操作将丢失用户配置数据。
警告	该类警示信息可能会导致系统重大变更甚 至故障,或者导致人身伤害等结果。	警告 重启操作将导致业务中断,恢复业务 时间约十分钟。
〔) 注意	用于警示信息、补充说明等,是用户必须 了解的内容。	▶ 注意 权重设置为0,该服务器不会再接受新 请求。
⑦ 说明	用于补充说明、最佳实践、窍门等,不是 用户必须了解的内容。	⑦ 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置> 网络> 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在 <b>结果确认</b> 页面,单击 <b>确定</b> 。
Courier字体	命令或代码。	执行    cd /d C:/window    命令,进入 Windows系统文件夹。
斜体	表示参数、变量。	bae log listinstanceid
[] 或者 [alb]	表示可选项,至多选择一个。	ipconfig [-all -t]
{} 或者 {a b}	表示必选项,至多选择一个。	switch {act ive st and}

# 目录

1.定时关灯	05
2.天气信息控制加湿器开关	11
3.智能控制空气净化	19
4.使用树莓派摄像头实现人脸识别	22
5.温室监控大屏	33
6.监测不同地域作物土壤环境	39
7.监控仓库储备物的报警数据	43
8.按步骤提交设备故障信息	49
9.统计会议签到人数	59
10.室内温度监控	63
10.1. 概述	63
10.2. 创建项目和产品	63
10.3. 开发移动应用监控室内温度	66
10.4. 发布	70
11.IoT Studio使用LA自定义接口全链路开发	73
12.智能控制空调设备	80
12.1. 概述	80
12.2. 步骤一:在项目中创建产品	80
12.3. 步骤二: 创建控制空调的业务服务	83
12.4. 步骤三:调试与发布	88
13.使用IoT Studio搭建气象监测屏	91
13.1. 概述	91
13.2. 配置LoRa网关	91
13.3. 配置LoRa设备接入物联网平台	93
13.4. 使用IoT Studio开发监控大屏	98

# 1. 定时关灯

定时关灯服务适用于学生宿舍场景。在一个学生宿舍,每晚22点向智能灯发送关灯指令,随后查询灯的状态,并将灯的状态发送到宿舍管理员钉钉群中,告知是否关灯成功。

### 前提条件

已完成创建项目(例如: Project)。具体操作,请参见项目。

#### 创建产品与设备

- 登录物联网应用开发控制台,在页面左上角选择对应实例后,在左侧导航栏单击项目管理。
   您也可登录物联网平台控制台,在左侧导航栏选择增值服务,单击选购增值服务中的前往查看服务, 在增值服务中心,单击IoT Studio区域的立即试用,进入IoT Studio控制台。
- 2. 单击项目Project卡片,进入项目详情主页,单击左侧导航栏的产品。
- 3. 单击创建产品,配置参数并单击确认,如图所示。

有关产品及其物模型定义的具体操作,请参见产品。

本案例创建产品:智能灯,产品所属分类选择**智能生活/电工照明/灯**,物联网平台已为该类型产品预 定义了布尔值开关属性Light Switch。

/ 产品 / 创建产品		
← 创建产品		
* 产品名称		
智能灯		
* 所屋品类 💿		
智能生活 / 电工照明 / 灯	~	查看功能
* 节占光刑		
连网与数据		
* 连网方式 WiFi	~	
	$\sim$	
ICA 你在我的情况。(AIINK JSON)	•	
✔ 认证方式		
雨夕信自		
史夕に尽		
◇ 产品描述		

4. 添加一个智能灯设备(例如: smartLights)。具体操作,请参见添加设备。

5. 本案例使用设备模拟器的属性上报数据。具体操作,请参见设备模拟器。

## 创建业务服务

1. 在项目的主页选择业务逻辑页签。

/ 颉						项	目配置 发布为解决方面
≃品 关联 1	设备 关联 1	Web应用 1	移动应用 0	业务逻辑 1	数据资产 ⊕ 关联 420	数编(4 0	£9 0
<b>3开发</b> Web应用 移动应用	业务逻辑 数据资产 数据任务	5					历史功能
全部发布状态            SR         描述	输入内容 修改时间 小	V	发布状态 😡	发布时间 🔹	运行状态	操作	
	2021-08-02 15:	11:28	未发布			编辑 复制 删除	

- 2. 单击业务服务列表左上方的新建 > 从模板新建。
- 3. 在业务逻辑开发页面,找到并单击定时关灯模板卡片。

■ 公別版   IoT Studio   业务逻辑	11开没			从横板创建业务逻辑	×
业务逻辑开发				* <u>业务</u> 逻辑名称 () 语能入内容	
\$fight				* 新屬项目	新疆项目
					~ C
+				清脑入内容	
空白模板	煤气报警 Web	控制智能灯 环境政策存储	定时关灯 人脸识别		0/100
打开最近				機動信息 磁量量一个学生增合。需要每天晚上10点关闭宿会房间的 #2.25年2月1日。#201日本Autor25	助了,关闭之后童袍灯的状态,并在135万种中对宿舍
測試环境 > 提案	Q			BERQERS, ENBERTHRAD	
8#	所羅项目	描述	修改时间	③ 定时触发 → ■● 智能灯控制 →	11111111111111111111111111111111111111
Parat.	1000		2020-04-1		
Per la			2020-04-1		
and a second	1984		2020-04-1		
	1986		2020-04-1		
	1000		2020-04-1		
-	1000		2020-04-1		
-	1986		2020-04-1		
analise .	and age		2020-04-0		
-			2020-03-1	使用该糊版新聞 取消	

4. 在从模板创建业务逻辑面板中, 配置服务基本信息。

参数	说明
服务名称	自定义服务名称。 仅支持中文汉字、英文字母、数字、下划线(_)、连接号(-)和英文圆括号(()),且 必须以中文汉字、英文字母或数字开头,长度不超过30个字符(一个中文汉字算一个字 符)。
所属项目	显示服务所属的项目Project。
描述	描述服务的用途等信息。长度不超过100个字符(一个中文汉字算一个字符)。

#### 5. 单击使用该模板新建。

业务服务创建成功后,页面跳转至业务服务的编辑页面,并自动生成了一个业务流,您可根据实际需求 配置节点参数。

定时关灯 🗙			
<ul> <li>定时触发</li> </ul>	□ → <b>□</b>	 口—————————————————————————————————————	
			_

## 编辑服务流

1. 在业务服务编辑画布中,单击定时触发节点,进行配置。

参数	说明
节点名称	输入节点名称。
触发模式	选择为 <b>循环定时触发</b> 。
循环周期	选择为天。
触发时间	设置为22:00:00。
选择生效时间	设置该服务开始执行的时间。
选择结束时间	设置服务停止执行的时间。如果需要服务永久执行,请勾选 <b>永久生效</b> 。

2. 从左侧设备列表下,拖拽一个智能灯产品节点到画布上,替换空白的产品节点,将定时触发节点与智能 灯节点相连,并进行配置。

② 表示55A       ③ 表面55A       ③ 表面55A       ● 表面55A <th></th> <th>* 10点な中 ◎</th>		* 10点な中 ◎
参数	说明	
节点名称	输入节点名称。	
选择要控制的设备	选择该服务控制的设备: smartLights。	

参数	说明
选择操作类型	选择为 <b>设备动作执行</b> 。
下发数据	选择为 <b>属性</b> 。
主灯开关	选择为静态布尔值false,即关灯。

3. 替换第二个空白产品节点为智能灯产品节点,与前一个智能灯和后一个钉钉机器人节点相连,并进行配置。

	日田37 → ☆ 19191歳人 □     ・10点称 ◎ 20何度現は10点7     ・10点称 ◎ 20何度現は10点7     ・20点称 ◎ 20何度現は10点7     ・20点取り使用。     ・20点取用。     ・2
参数	说明
节点名称	输入节点名称。
选择要控制的设备	需与之前节点控制的设备一致: smartLights。
选择操作类型	选择为 <b>查询设备属性</b> 。
选择要查询属性	选择为 <b>主灯开关</b> 。
查询维度	勾选 <b>快照值</b> ,即查询开关状态。

4. 单击钉钉机器人节点进行配置。

	■ SEC	名称 ● 知時使用は节点? 月月人 注 一 わらよ ● パ// ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・
参数	说明	
节点名称	输入节点名称。	
Webhook	钉钉群机器人的Webhook地址。具体操作,请参见 <mark>创建钉钉机器人</mark>	0
配置方法	选择为 <b>使用模板</b> 。	

参数	说明
选择模板	选择为 <b>设备属性告警</b> 。
设备数据来源	选择为 <b>智能灯/主灯开关</b> 。
通知用户	根据您的业务需要选择:发送消息时,是否需要提醒群里所有人或某些人查看消 息。

5. 单击页面右上方保存按钮 🖻 ,保存设置。

## 调试与发布

- 1. 在业务服务编辑页面,单击右上方的部署调试按钮 🐟 ,部署并启动服务。
- 2. 调试服务。
  - i. 返回项目详情的**设备**页,进入该设备的在线调试页签,启动虚拟设备。
  - ii. 回到业务服务编辑页,单击右上方的**部署调试**按钮 🐢 ,在定时调试对话框中,输入时间为任意一

天的22:00:00,并单击**开始调试**。 钉钉群里将收到灯的开关状态信息。

**服务开发测试员** 机器人 6月2日 22:00 2019-06-02 22:00:00 设备 的主灯开关为0

3. 单击发布按钮 📢 发布服务。

业务需要发布到云端,才能被调用。

# 2.天气信息控制加湿器开关

本示例的服务将根据天气预报中的湿度信息,决定加湿器是否开启。如果湿度≤30则开启,否则不开启。

前提条件

- 已在阿里云云市场购买天气查询API。
- 已完成创建项目。具体操作,请参见项目。

### 步骤一: 创建产品与设备

- 登录物联网应用开发控制台,在页面左上角选择对应实例后,在左侧导航栏单击项目管理。
   您也可登录物联网平台控制台,在左侧导航栏选择增值服务,单击选购增值服务中的前往查看服务, 在增值服务中心,单击IoT Studio区域的立即试用,进入IoT Studio控制台。
- 2. 单击项目卡片,进入项目详情主页,单击左侧导航栏的产品。
- 3. 在产品页面,单击产品列表左上方的创建产品。
- 4. 在创建产品页面, 配置参数, 并单击确认, 如下图所示。

⑦ 说明 如果您已在物联网平台控制台创建了相关产品和设备,单击关联物联网平台产品,将 产品与该项目关联即可。

← 创建产品
* 产品名称
智能加湿器
* 所属品类 🕜
○ 标准品类 (●) 自定义品类
* 节点类型
● 直连设备 ● ◎ ◎ ◎ ◎ ◎ ◎ ◎ ◎ ◎ ◎ ◎ ◎ ◎ ◎ ◎ ◎ ◎ ◎
连网与数据
* 连网方式
Wi-Fi 🗸
* 数据格式 🕜
ICA 标准数据格式 (Alink JSON) V
✔认证方式
更多信息
▶ 产品描述
确认取消

- 5. 产品创建成功后,回到产品页面,在产品列表中,单击目标产品对应的查看。
- 6. 在产品详情页面,单击功能定义页签,再单击编辑草稿,选择添加自定义功能,自定义一个布尔值开关属性,单击确认,如下图所示

添加自定义功能	×
* 功能类型 ②	
属性 服务 事件	
* 功能名称 💿	
电源开关	
* 标识符 💿	
PowerSwitch	
* 数据类型	
bool (布尔型)	$\sim$
* 布尔值	
0- 关闭	
1 - 开启	
* 读写类型	
● 读写 ○ 只读	
描述	
请输入描述	
	0/100
确认	取消

- i. 物模型添加完成后,单击页面左下方的发布上线按钮,弹出发布物模型上线对话框。
- ii. 选中确认已查看当前版本与线上版本的比对结果复选框。
- iii. 单击**确定**,发布物模型。
- 7. 在项目详情页,单击左侧导航栏的**设备**。
- 8. 单击添加设备,添加智能加湿器设备,如图所示。

添加设备 💿 💦 🗙 🗙 🗙
<ul> <li>特别说明: DeviceName 可以为空,当为空时,阿里云会颁发产品下的 唯一标识符作为 DeviceName。</li> </ul>
产品
智能加湿器
DeviceName 💿
请输入 DeviceName
备注名称 🕜
请输入备注名称
确认取消

## 步骤二: 创建业务服务

1. 在项目的**主页**选择**业务逻辑**页签,单击**新建**并选择**新建**,创建一个业务服务。

84.95	/ 主页						项目数	置 发布为解决方案
产品 关联 1	设幅 1	F 关联	Web应用 1	移动应用 0	业与逻辑 1	数据资产 @ 关联 420	数据任务 0	Ø
<b>项目开发</b> Web应用	項目开発         Web应用         移动应用         业务逻辑         数据近产         数据任务         历史取編 ×							
新建全部	援布状态 > 総入内容	V						
名称	描述	修改时间 小		发布状态 😡	发布时间 🔹	运行状态	操作	
		2021-08-02 15:11:28		未发布		-	编辑 复制 删除	
							总计:1 〈 1 〉	每页显示: 10 🗸 🗸

2. 在业务逻辑开发页,单击左侧的**节点**按钮,拖拽一个HTTP请求节点到画布上,为节点做如下配置。

	* 参数名称 ③ cityld	* 节点名称 ⑧	如何使用该节点?
	* 炭型 String (字符型) ✓	HTTP请求 请求配置	
<mark>ĕ hite≴a d</mark>	* 是否必填	请求协议与方法需要在域名 理,请点击这里。	名管理处进行全项目的管 ×
	● 必嬪项 ○ 非必嬪项	* Action 🎯	
	默认值	QueryWeather	
	请输入默认值	Distance in the second s	
	描述	鉴权能适	
	请输入描述	<ul> <li>账号监权 ()</li> <li>访问限制 ()</li> </ul>	
		入参配置	*
	0/100	cityld	/ ×
	<b>御定</b> 取消	+ 181	īD入参

参数	说明
节点名称	输入节点名称。
Action	调用API时,参数Action的值。本示例中,设置为 <i>QueryWeather</i> 。
新增参数	设置调用API时的入参。本示例中,设置了一个必填入参cityld,类型为String。

3. 拖拽一个云市场API节点到画布上,将HTTP请求节点与该节点相连,并配置该节点。

|--|

★ HTTP请求 ● 至市场API	古点我際: 云市场API 五点教型: alyun,market_api 1D: node_Sbl44530
API週用結果 全局日志 节点日志	• 100% • 🗋 🗖 🗃
* "日志*:() 0 items	参数编写 1 ( "cityId":"((query.cityId))", "tokem":"46e1307aa0906072e3358c3b672a2ae4" 3 )
参数	说明
节点名称	输入节点名称。
请求方式	在云市场该API购买页,查看请求方式。本示例中,购买的API的请求方式为POST。
调用地址	在云市场该API购买页,查看该API的调用地址。
APPCODE	在云市场控制台,已购买的服务页列表中,查看已购买API的AppCode。
编码	选择编码方式。
设置请求头	根据对应接口的Header要求设置API的请求头。
参数填写	<ul> <li>根据云市场该API购买页的请求参数说明,填入请求参数。本示例中,需填入两个参数:</li> <li>cityld:城市ID,可填入: <ul> <li>静态参数,即填入一个固定值,如 "cityId":"20"。参数取值,可参见墨迹天气购买页提供的"cityid对照表"。</li> <li>动态参数,可填入 "cityId":"{{query.cityId}}",该结构是把API的入参作为子对象。</li> </ul> </li> <li>token:请在阿里云API网关控制台,选择调用API &gt; 已购买API,选择地域为华北2(北京),在购买的API详情中查看。具体的token查看步骤说明,请参见墨迹天气购买页提供的"token查询方式"。</li> </ul>

 4. 配置一个临时HTTP返回节点,用于调试、查看天气API的返回数据结构。
 因后续路径节点配置中,需调用湿度值,但天气API的返回数据结构未知,所以可以先配置一个HTTP返 回节点,并进行部署、调试。在调式结果日志中,查看返回数据结构。 i. 配置一个HTTP返回节点,输出选择为上一节点(payload)。

<ul> <li>         ・         部点名称         ・         </li> <li>         HTTP放回     </li> </ul>	如何使用该节点?
近日 - 始土 ◎ 田空道 ★日节点	5点 🖌 交量
上一节点(payload)	~
不婚写变量名则默认为整个返	间值

- ii. 单击**部署调试**按钮, 部署服务。
- iii. 部署成功后,再次单击部署调试按钮, 输入一个cityld值, 如20,进行调试。

可在API调用结果或HTTP返回节点的节点日志下,查看返回结果数据。

ľ		● 100% ●	п.		É.	
	A DUR READER				~	
	Af Heritstark			Q	^	
	"code" : 0				^	
	* "data" : { 2 items					
	condition : { y items					
	"temp": "]]"					
	"windDegrees" : "270"					1
	"windLevel" : "6"					
	"icon" : "0"					
	"humidity" : "11"					
	"windDir": "西风"					
	"updatetime" : "2020-04-21 13:30:08"					
	}					

返回数据结构显示, humidity在data下的condition下。因此, 路径配置中, 需使用 data.condition.humidity 调用天气API的返回结果中的湿度值。

iv. 调试成功后, 删除该HTTP返回节点。

5. 配置一个路径选择节点,与云市场API节点相连。

<mark>і итрах о⊸.00</mark> делали о́	1 H-259	<ul> <li>・ 日点名称 ●</li> <li>第日記述場</li> <li>・ 拡入 ●</li> <li>来目記市场API -</li> <li>第日記述 ●</li> <li>第日記述 ●</li></ul>	如何使用は15点? ・ ・ ・ ・ ・ ・ ・ 、 ・ 、 ・ 、 ・ ・ ・ ・ ・ ・ ・
参数	说明		
节点名称	输入节点名称。		
数据源	<ul> <li>。选择数据源为云市场API节点。</li> <li>输入变量为:上一步调式后,查看到的湿度值所在位置<i>data.condition.humidity</i>。</li> </ul>		

· # 479 0

参数	说明
新增路径	本示例中, 只需配置两个路径, 因此需删除第三条路径。 路径配置: • 路径1: ≤30。 • 路径2: >30。

6. 从左侧设备页签下,拖拽一个智能加湿器设备节点到画布上,与路径1相连。配置如下:

			* 节点名称 💿 如何使用读	(节点?
		智能加湿器		
			产品功能定义	
			* 远择要控制的设备 💿	
	HTTP请求 □ № 云市场API □ № K 路径送		G	$\sim$
			・选择操作类型 💿	
			设备动作执行	$\sim$
			* 下发数据 💿	
			● 屬性 ○ 服务	
			电源开关 💿	
			🛛 🖂 📈 未自节点 交量	
			布尔值	$\sim$
			true	~

参数	说明				
节点名称	输入节点名称。				
选择要控制的设备	选择该产品下的一个设备。				
选择操作类型	选择为 <b>设备动作执行</b> 。				
下发数据	勾选 <b>属性</b> ,设置电源开关值为true,即开启加湿器。				

7. 再配置一个智能加湿器, 与路径2相连。

该节点配置,设备与上一步选择一致,需将电源开关设置为false,即关闭加湿器。

		使用该节点?
	智能加湿器	
・ 別目表的の名 ●     ・ 別目表の的の名     ・       ・       ・	产品功能定义	
●         ●	* 选择要控制的设备 💿	
● 通報性的算器 ・ ● 通報性的算器 ・ ● 通報性的算器 ● ● 使物的时时行 ~ ● 下放放着 ● ● 常性 ● 服务 电研开炎 ● ■ 配行光 ● ■ 回行 ●	Collinson and and	~
	* 选择操作类型 🔘	
<ul> <li>下及数章 ●</li> <li>常性 ○勝み</li> <li>相野井 ○局</li> <li>和町市 受量</li> <li>市が間</li> <li>すか間</li> <li>すか</li> <li>すか</li> </ul>	设备动作执行	~
● 属性 ○ 服务 电振开关 ● 電磁通 未自节点 交量 本の局 ~ もkare ~	* 下发数据 ②	
<ul> <li>电振开关●</li> <li>●</li> <li>●</li></ul>	<ul> <li>■ 属性</li> <li>○ 服务</li> </ul>	
国空道 年前の点 交通 市外油 マ 16be マ マ	电源开关 🔘	
を印刷 ーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーー	固定值 🖌 来自节点	变量
faise V	布尔值	~
	false	~
	L	

8. 分别为两个设备节点配置HTTP返回节点。输出均设置为上一节点(payload)。

* 节点名称 🔘	如何使用该节点?
HTTP	
返回	
* 輸出 ②	
固定值	来自节点 🗸 交量
上一节点(payload)	` ·
不填写变量名则默认	为整个返回值

9. 单击页面右上角保存按钮 🖪 ,保存设置。

步骤三:调试与发布

- 1. 在业务服务编辑页面,单击右上方的部署调试按钮,,部署服务。
- 2. 部署完成后,再次单击**部署调试**按钮, 输入一个cityld值, 单击开始调试。

		9 5 H B 🔷 🛪 🖪 Ø
天气信息边 ×		
	<ul> <li>节点</li> <li>节点</li> <li>* 节点</li> <li>第級</li> <li>線入車</li> <li>数螺</li> <li>云tt</li> </ul>	後達 176日日 25 10月1日日 25 10月1日 25 10月11日 25 10月111日 25 10月111日 25 10月1111日 25 10月111111 25 10月11111111111111111111111111111111111
	data	.condition.humidity
	路径近 新増算 * 路径	時配置 19章
	<=	$\sim$
* chyld 20	数值 30	<u>周辺値 未</u> 自花点 <u>支</u> 量 122

全局日志显示当前湿度为11,所以服务执行了路径1,开启加湿器。

天气信息控 ×				
€				<b>F</b>
	● 100% ●			
APi调用结果 <b>全局日志</b> 节点日志	搜索日志关键字 Q	0	٢	×
<pre>'condition':{ 9 items 'vis':'30000' 'condition':ff' 'temp':'11' 'windDegrees':'270' 'windLevel':'6' 'cond':''f' 'humidity':'11' 'windDit':'fGX'</pre>				•

3. 单击**部署调试**右侧的**发布**按钮 📢 ,发布该服务。

业务需要发布到云端,才能被调用。

### 执行结果

该服务创建的API,可以用于Web可视化开发。

可右键单击服务名称,选择**服务调用帮助**,查看服务调用详情。

# 3.智能控制空气净化

本示例中,创建一个智能控制空气净化服务。当空气监测设备上报到当前空气中PM2.5浓度小于100时,关闭空气净化设备;大于等于100时,开启空气净化设备。

### 前提条件

已在项目(Project\_test)中完成空气检测(空气检测仪:airDetector)和净化(空气净化器:airPurifier) 产品、设备的创建。具体操作,请参见在项目中创建空气检测和净化的产品。

#### 创建控制空气净化器的业务服务

- 1. 在项目中创建业务服务(例如: 空气净化)。具体操作,请参见创建业务服务。
- 2. 在业务逻辑开发页,单击左侧的**节点**按钮,拖拽一个**设备触发**节点到画布上,并配置输入数据为空气 质量监测设备airDetector上报的属性: PM25浓度。

拖拽一个条件判断节点到画布上,与设备触发节点相连,并进行配置。
 配置条件1:设备触发的属性PM25值大于等于100。

			*节点名称 💿	如何使用该节点?
			条件判断	
			* 条件配置 📀	
没备触发 □		条件1 · 设备触发PM25浓度 >= 100	1	
			+ 条件	

拖拽两个空气净化器产品节点到画布上,分别与条件判断的满足条件和不满足条件点连接起来,进行配置。

오代하化 ×		* ひ点名称 ● 如何使用读节点7     空气争化器     字品为能意义     · 法研究作类型 ●     · 法研究作类型 ●     · 全部作用以行     · 下及效道 ●     · 予以效道 ●     · 予以效道 ●     · 原语 ● 服号     · · · · · · · · · · · · · · · · ·
参数	描述	
节点名称	设置节点名称。	
选择要控制的设备	选择设备airPurifier。	

参数	描述
选择操作类型	选择要操作的类型。可选:设备动作执行、查询设备属性、查询设备其他信息。本示例中需下发开关操作指令,所以选择为设备动作执行。
下发数据	本示例中,设置下发数据为电源开关属性,满足条件时,设置电源开关固定值为静态布尔值为true,即PM25值超过100时,开启空气净化器。不满足条件时,设置电源开关固定值为静态布尔值false,即关闭空气净化器。

5. 单击页面右上方的保存按钮 🛅 ,保存设置。

## 调试与发布

1. 在业务服务编辑页面,单击右上方的部署调试按钮 🐟 ,部署并启动服务。

- 2. 服务部署成功后,单击右上方的部署调试按钮 🐟 ,调试服务。
  - i. 单击调试 > 前往, 前往产品的设备模拟器页面。
  - ii. 单击启动模拟设备。
  - iii. 选择上行指令调试 > 属性上报。
  - iv. 在默认模块中推送调试指令。

下图示例中,模拟推送的PM25值是101,大于100,空气净化设备将收到开启指令。

设备模拟器			
请试设备: 空气检测仪 ✓ qzlAGjBqtDv9gxd ✓			
上行指令调试 下行指令调试		设备端日志查看云端日志	
自定义 Topic 圖性上級 事件上报	在线 🌑	樂型 / 时间	内容
欄決: 取込機決 ~ AQI	1	设备上损数据 2020/04/17 14:18:29	2020-04-17 14182-3109.48897271425148FA69C6868EE2D8C43, upstream - bo <sup>2</sup> typer-PROPERTY_REPORT_parames1'd1'123''.totd1'Cg655QBtrz.46CAH4Gp00010 07''entel/of' Thing-sent poerty poerty foarms' ("PAI25110] 'topoc'/logs 2020087537''.totd1'' 10'' Frastlanced Semasspecurces tracks by a 4''g1' d1''23'''ensuige'' huxces1''''entel of Thing-sent poerty poert' Ne Cendrug1''enteredent'''''''''''''''''''''''''''''''''''
谚输入参数 (int)			s10,"tatusLas10,"thingType1"DEVICE1.scriptDeta=[].traceId=0bc1Id59158
请继入参数 (text)			
地理位置			
Longitude			
请编入参数 (double)			
Latitude			
请编入参数 (double)			
Altitude			
请输入参数 (double)			
CoordinateSystem			
请选择参数 (enum)			
PM25			
101			
	-		
发送指令 量量			

3. 在项目的设备页,查看空气净化设备的运行状态下,电源开关为1(即开启状态)。

主页	设备管理 / 设备详情
产品	< 2mW4Mp6dMhP3600mP4jBg
设备	产品     空气争化器     查看       ProductKey     复制
账号	设备信息     Topic列表     运行状态     事件管理     服务调用       图表     表格     实时刷新
	电源开关 <b>1 (开启) ③</b> 2020/04/17 15:15:38

您可返回服务开发编辑页面,查看相关日志。

拉制空气争化器 ×	
	·
	🗢 100% 🚱 🔲 🛄
全局日志 节点日志	搜索日志关键字 Q 🚺 🛃 🗶 🗙
▼ "全节点日志":[ 3 items	
<b>0</b> : { 7 items	
"节点ID":"node_00ad80b0" "莱克尔教","邓森林华"	
「「二合心 · 」の目載及 「五古茶刑"・"input"	
▼ "节点输入":{ 2 items	
"deviceContext" : { 3 items	
"productKey" : "	
"deviceName" :	
"gmtCreate" : 1587108184669	
}	
"props" : { 1 item	
" 'PM25" : { 2 items	
"value" + 101	

4. 在服务开发编辑页,单击右上方**发布**按钮 **《**,发布服务。

业务需要发布到云端,才能被调用。

# 4.使用树莓派摄像头实现人脸识别

本文介绍在物联网应用开发(IoT Studio)平台使用树莓派摄像头实现人脸识别功能。即将树莓派摄像头采 集的人像,存储到阿里云对象存储(OSS)中,同时,通过设备属性上报获取图片,并调用人脸识别API进行 人脸识别后,在钉钉群中推送验证结果。

#### 流程图



#### 物料准备

硬件	说明
树莓派主板	将摄像头和红外线传感器连接到树莓派主板。
适用于树莓派的摄像头	<ul> <li>● 将摄像头连接线头插入到树莓派卡槽中。</li> </ul>
红外线传感器	● 将红外线传感器(PIR)连接线头插入到树每派GPIO引脚。

### 创建存储空间

本示例中,摄像头拍摄的图像将存储在阿里云对象存储中,因此需为图像存储创建一个存储空间Bucket。开发设备端SDK时,需要配置该Bucket信息。

1. 登录对象存储控制台。

若您还未开通对象存储服务,请单击**立即开通**,进入购买页开通服务。具体操作,请参见<mark>开通OSS服</mark> <mark>务</mark>。

2. 单击Bucket列表,然后单击创建Bucket。

您也可以单击概览,然后单击右上角的创建Bucket。

3. 在**创建Bucket**面板,按如下说明配置必要参数。其他参数保持默认配置,也可以在Bucket创建完成后单独配置。

参数	说明
Bucket名称	Bucket的名称。Bucket一旦创建,则无法更改其名称。 命名规则如下: • Bucket名称必须全局唯一。 • 只能包括小写字母、数字和短划线(-)。 • 必须以小写字母或者数字开头和结尾。 • 长度必须在3~63字节之间。

参数	说明
地域	选择您要创建的Bucket所在地域。Bucket一旦创建,则无法更改其所在地域。
存储类型	本示例中,选择 <b>标准存储</b> 。实际使用中,请根据您的实际需要选择。
同城冗余存储	本示例中,选择 <b>关闭</b> 。实际使用中,请根据您的实际需要选择。
读写权限	本示例中,选择 <b>公共读写</b> 。实际使用中,请根据您的实际需要选择。
实时日志查询	本示例中,选择 <b>不开通</b> 。实际使用中,请根据您的实际需要选择。

Bucket创建后,其概览页会显示Endpoint和域名。Endpoint信息需配置到设备端SDK中。

访问域名			
	EndPoint(地域节点) ②	Bucket 域名 ⑦	HTTPS
外网访问 ②	oss-	iots aliyuncs.com	支持
ECS 的经典网络访问(内网 ) ⑦	ossinternal.aliyuncs.com	iotsi	支持
ECS 的 VPC 网络访问(内网) ②	ossinternal.aliyuncs.com	iotsi-internal.aliyuncs.com	支持

- 4. 测试Bucket。
  - i. 在概览页, 单击左侧导航栏的文件管理, 然后单击上传文件。
  - ii. 在上传文件面板,参数保持默认配置,上传一张测试图片。
  - iii. 图片上传完成后,关闭上传任务面板,然后单击操作列的详情按钮,将图片URL复制到浏览器中。
     若浏览器中显示该图片,证明Bucket使用正常。

## 创建产品和设备

- 1. 登录物联网平台控制台。
- 2. 在实例概览页面,找到对应的实例,单击实例进入实例详情页面。

↓ 注意 在中国地域,目前仅华东2(上海)地域开通了公共实例服务。

物联网平台	企业版实例	运行中	2	即将到期 💿	۵	已到期 💿	٩
实例概范	2	2	~	0		0	~
产品文档 🖸							
增值服务	全部英例 >						图表: 设备数 消息TPS
			升配续费				升配续费
	「「「「」」「「」」」「「」」」「「」」」」「「」」」」「「」」」」」「」」」」			iot 7	基油型		
	♥ 运行中		10127	❷ 运行中			10 25 20
	ID: id		0	ID: ic			(1)
	到時时1月: 2021/10/02			型制制部引用J: 2021/10/01			
			升级企业版				
<	公共实例 试用				]买企业版实例		
	♥ 日井温					sentre i schols son orre,	
	ID: io		102TM1802 8		例关系的 快速入门		
	开通时间: 2021/08/31		$\smile$				

- 3. 在左侧导航栏,选择设备管理 > 产品,单击创建产品。
- 在新建产品页面,配置参数后,单击确认。
   本文示例的产品名称为摄像头,所属品类为自定义品类,其他参数使用默认值。

← 新建产品 (设备模型)
新建产品从设备中心新建产品
* 产品名称
摄像头
* 所属品类 💿
○ 标准品类 (●) 自定义品类
* 节点类型
直连设备   一 直连设备
连网与数据
* 连网方式
Wi-Fi ~
* 数据格式 ②
ICA 标准数据格式 (Alink JSON) V
* 数据校验级别 ②
<ul> <li>● 弱枝验</li> <li>○ 免校验</li> </ul>
へ收起
* 认证方式 💿
设备密钥 ✓
へ收起
更多信息
确认取消

5. 单击查看产品详情,在产品详情页,单击设备数后的前往管理。

6. 在设备页,单击添加设备。输入设备名称(DeviceName),设置设备备注名,单击确认。

## 为产品定义物模型

- 1. 在左侧导航栏,选择设备管理 > 产品。在产品列表中,单击产品对应的查看。
- 2. 在产品详情的功能定义页签下,定义物模型。

⑦ 说明 已发布的产品不能添加和更新功能。

本示例在物模型的默认模块中,添加以下属性。具体操作,请参见单个添加物模型。

标识符	数据类型	数据定义
people	bool	0: 没人; 1: 有人。
picURL	text	数据长度限制:2048字节。

#### 3. 发布物模型。

- i. 单击页面左下方的发布上线按钮, 弹出发布物模型上线对话框。
- ii. (可选)单击添加发布备注, 输入版本号和版本描述。

参数	说明
版本号	设置当前物模型版本号。后期可根据版本号管理物模型。 版本号支持英文字母、数字和英文句号(.),长度限制1~16个字符。
版本描述	描述当前版本物模型。支持中文汉字、英文字母、数字和特殊符号。长度限制为 100个字符。一个中文汉字算一个字符。

iii. 如存在已上线的版本,则需要进行比对,确认新版本的修改点。

单击**比对结果**,在**比对结果**对话框中查看修改点。确认无误后,单击**确认查阅**,回到**发布物模型** 上线对话框,自动勾选"确认已查看当前版本与线上版本的比对结果"。

iv. 单击确定,发布物模型。

⑦ 说明 物模型发布后,才会正式生效。

自定义功能属性配置完成,可在**功能定义**页签下查看。

有关物模型的更多信息,请参见物模型(功能定义)。

### 设备端SDK开发

因为树莓派基于Python语言,需要配置两个Python SDK: IoT设备端接入SDK和OSS文件上传SDK。

⑦ 说明 本次设备端SDK开发的设备接入版本为 aliyunsdkiotclient.AliyunIotMqttClient 。最 新版本可参考新的设备接入SDK进行适配。

#### 设备端SDK开发代码示例如下:

• 引入相关库。

```
import aliyunsdkiotclient.AliyunIotMqttClient as iot ##导入阿里云的设备MQTT库,如果import失败
需要先pip3 install
import json
import multiprocessing
import time
import random
import oss2 ##导入阿里云的OSS库,如果import失败需要先pip3 install oss2
from picamera import PiCamera ##树莓派的摄像头,系统自带
import RPi.GPIO as GPIO ##GPIO口,接红外PIR使用
```

• OSS Bucket访问授权。

```
auth = oss2.Auth('**YourAccessKeyId**','**YourAccessKeySecret**') ##Oss的授权,需要您的阿里
云账号AccessKey ID和AccessKey Secret,具体查看https://usercenter.console.aliyun.com/#/manage
/ak
bucket = oss2.Bucket(auth,'http://oss-*****.aliyuncs.com','**YourBucketName**') ##请在OSS
控制台, Bucket的概览页查看具体信息
global picURLtoIoT
camera = PiCamera()
camera.resolution = (800,600) ##拍照分辨率,越高越容易分析,但是上传速度越慢
```

• 初始化树莓派。

```
def init():
    GPIO.setwarnings(False)
    GPIO.setmode(GPIO.BOARD)
    GPIO.setup(3, GPIO.IN)
    pass
```

• 定义图像上传OSS Bucket。

```
def take_photo():
    ticks = int(time.time())
    fileName = 'test%s.jpg' % ticks ##在文件名加入时间戳作为简易加密手段
    filePath = '/home/pi/Pictures/%s' % fileName
    camera.capture(filePath)
    bucket.put_object_from_file('exampleobject.txt', 'D:\\localpath\\examplefile.txt')
## 填写Object完整路径和本地文件的完整路径。Object完整路径中不能包含Bucket名称。如果未指定本地路径,
```

```
则默认从示例程序所属项目对应本地路径中上传文件。
```

global picURLtoIoT picURLtoIoT = 'http://\*\*您的bucket名称\*\*.oss-\*\*\*\*\*.aliyuncs.com/bucket\_file\_name/%s' % fileName ##图片存储URL,需替换为您的bucket名称和bucket内文件名称

print(str(picURLtoIoT))

• 定义红外线感应器检测到有人时,摄像头拍照,否则休眠5秒。

```
def detectPeople():
   if GPIO.input(3) == True:
take photo()
    else:
    time.sleep(5)
options = {
    'productKey':'**设备的ProductKey**',
    'deviceName':'**设备的deviceName**',
    'deviceSecret':'**设备的deviceSecret**',
    'port':****,
    'host':'iot-as-mgtt.*****.aliyuncs.com' ##物联网平台域名
}
host = options['productKey'] + '.' + options['host']
def on message(client, userdata, msg):
   topic = '/' + productKey + '/' + deviceName + '/update'
    print(msg.payload)
def on connect(client, userdata, flags dict, rc):
   print("Connected with result code " + str(rc))
def on disconnect(client, userdata, flags dict, rc):
    print("Disconnected.")
```

#### • 定义设备上报数据到物联网平台。

```
def upload device(client):
    topic = '/sys/'+options['productKey']+'/'+options['deviceName']+'/thing/event/propert
y/post'
   while True:
       payload json = {
            'id': int(time.time()),
           'params': {
'people':1 ##物模型里布尔值以0和1的形式上报
                'picURL': picURLtoIoT,
           },
           'method': "thing.event.property.post"
           }
       print('send data to iot server: ' + str(payload json))
       client.publish(topic, payload=str(payload json))
if __name__ == '__main__':
   client = iot.getAliyunIotMqttClient(options['productKey'], options['deviceName'], opt
ions['deviceSecret'], secure mode=3)
   client.on connect = on connect
   client.connect(host=host, port=options['port'], keepalive=60)
   p = multiprocessing.Process(target=upload device, args=(client,))
   p.start()
   detectPeople()
   GPIO.cleanup()
   client.loop forever()
```

完整的示例代码,请参见本文结尾<mark>附录:设备端SDK代码示例</mark>。

#### 创建业务服务

1. 登录物联网应用开发控制台,在页面左上角选择对应实例后,在左侧导航栏单击项目管理。

2. 创建普通项目。具体操作,请参见普通项目。

- 3. 将已创建的摄像头产品和设备与项目关联。具体操作,请参见关联产品至普通项目。
- 4. 在项目详情**主页**,单击**业务服务**页签,单击应用列表上方的新建 > 新建,创建一个业务服务。
- 5. 在业务逻辑编辑页,单击左侧的节点按钮,服务开发的功能节点将展示在列表中。
- 6. 从**输入**页签下,拖拽一个**设备触发**节点到画布上,触发数据源配置为摄像头产品下设备的属性上报。

	* 市底高称 @	如何使用该节点?
② 定时触发	≦ 2842 0	
🔁 моттіјя	*5#*# 0	
	· · · · · · · · · · · · · · · · · · ·	/ ×
<ul> <li>HTTP请求</li> </ul>	<ul> <li>● 資源報告</li> </ul>	
		/ ×
	* 触发质件 ①	*
/ 输出	履住。全部履住	/ ×
HITP22		+ 事件上报 + 状态变更

7. 配置一个云市场API节点,并与设备触发节点相连。

* 节点名称 💿 如何使用该节点?
云市场API
前往阿里云市场
API配置
* 请求方式
POST V
* 调用地址 💿
New Yorkshop Control of Control of Control of
* APPCODE 💿
Lenes courternois
* 编码
请选择 イント
参数编写
<pre>1 { 2 "IMAGE":"{{query.props.picURL.value}}", 3 "IMAGE_TYPE":"1" 4 }</pre>

需在阿里云云市场购买人脸识别API。本示例购买的API是艾科瑞特(iCREDIT)\_智能人脸识别。

参数	说明
节点名称	输入节点名称。
请求方式	在云市场该API购买页,查看请求方式。本示例中,购买的API的请求方式为POST。
调用地址	在云市场该API购买页,查看该API的调用地址。
APPCODE	在云市场控制台,已购买的服务页列表中,查看已购买API的AppCode。
编码	选择编码方式。

参数	说明							
	根据云市场该API购买页的请求参数说明,填入请求参数。本示例中,需填入两个参 数:							
	○ IMAGE: 图像数据存储的URL。可输入:							
	■ 静态值:即输入图像在OSS中的存储地址URL。							
参数填写	■ 动态值:配置为动态获取值,本示例动态取值设置为: "IMAGE":"{{query							
	.props.picURL.value}}" ,即从属性picURL的数据中获取。							
	<ul> <li>IMAGE_TYPE: 0: 图像内容为图像数据BASE64编码; 1: 图像内容为图像文件的 URL。本示例采用URL,配置为 "IMAGE_TYPE":"1"。</li> </ul>							

#### 8. 配置钉钉机器人节点。

	* 节点名称 💿 如何使用该节点?
	钉钉机器人
	推送配置
	* Webhook 📀
È 设备触发 □→●■◎ 云市场API □→●■◎ 钉钉机器人 □	请输入推送配置的webhook
	配置方法
	* 配置方法
	○ 使用模版 ● 自定义
	* 消息类型 📀
	FeedCard类型 ~
	内容配置
	<pre>1   { 2 "feedCard": { 3  4  5  6  7  8  9  1  10  3  11  12  12  14  15  15  16  17  17  17  17  17  18  19  10  11  12  10  11  12  10  11  12  10  11  12  10  11  12  10  11  12  10  11  10  10</pre>

参数	说明
节点名称	输入节点名称
Webhook	钉钉群机器人的Webhook地址。具体操作,请参见 <mark>创建钉钉机器人</mark> 。
配置方法	选择为 <b>自定义</b> 。
消息类型	选择为FeedCard类型。
内容配置	<ul> <li>修改消息标题title。</li> <li>修改picURL值为{{query.props.picURL.value}}。</li> </ul>

### 9. 单击页面右上方保存按钮 🖻 ,保存设置。

## 调试与发布

- 1. 在业务服务编辑页面,单击右上方的部署调试按钮 🐢,部署并启动服务。
- 2. 部署完成后,再次单击右上方的部署调试按钮 🐢,调试服务。
  - i. 单击调试 > 前往, 前往产品的设备模拟器页面。
  - ii. 单击启动模拟设备。
  - iii. 选择上行指令调试 > 属性上报。
  - iv. 在默认模块中推送调试指令。

设备模拟器			
调试设备: 摄像头 >	/		
上行指令调试 下行指令调试		<b>设备端日志</b> 查看云端日志	
自定义 Topic 属性上报 事件上报	在线 🔵	类型 / 时间	内容
模块: 默认模块 ~ picURL(picURL)		云平台到设备消息	received topic=/sys/a1PhZCaNLFu/3S49e1T uofMZr8gsCo1l/thing/went/property/post_r epky, paplicad="(rocde":200."data")(L*d1"161 2002/0877"; "mesuage":"success", "metho d";"thing.event.property.post", "version";"1. 0"]
people(people) 有人-1 V 劳送指令 重置		设备到云平台消息	publish topic=/sys/a1PhZCaNLFu/3S49e1Tu

数据推送成功,钉钉群将收到图片。



3. 单击发布按钮,发布服务。

## 附录:设备端SDK代码示例

```
import aliyunsdkiotclient.AliyunIotMqttClient as iot ##导入阿里云的设备MQTT库,如果import失败需
要先pip3 install
import json
import multiprocessing
import time
import random
import oss2 ##导入阿里云的OSS库,如果import失败需要先pip3 install oss2
from picamera import PiCamera ##树莓派的摄像头,系统自带
import RPi.GPIO as GPIO ##GPIO口,接红外PIR使用
```

```
auth = oss2.Auth('**YourAccessKeyId**','**YourAccessKeySecret**') ##OSS的授权,需要您的阿里云
账号AccessKey ID和AccessKey Secret,具体查看https://usercenter.console.aliyun.com/#/manage/ak
bucket = oss2.Bucket(auth,'http://oss-*****.aliyuncs.com','**YourBucketName**') ##请在OSS控
制台, Bucket的概览页查看具体信息
global picURLtoIoT
camera = PiCamera()
camera.resolution = (800,600) ##拍照分辨率,越高越容易分析,但是上传速度越慢
##初始化树莓派
def init():
   GPIO.setwarnings (False)
   GPIO.setmode (GPIO.BOARD)
   GPIO.setup(3, GPIO.IN)
   pass
def take photo():
   ticks = int(time.time())
   fileName = 'test%s.jpg' % ticks ##在文件名加入时间戳作为简易加密手段
   filePath = '/home/pi/Pictures/%s' % fileName
   camera.capture(filePath)
   bucket.put object from file('bucket file name/%s', fileName) ##在这里改bucket名称
   global picURLtoIoT
   picURLtoIoT = 'http://**您的bucket名称**.oss-*****.aliyuncs.com/bucket file name/%s' %
fileName ##图片存储URL,需替换为您的bucket名称和bucket内文件名称
   print(str(picURLtoIoT))
##如果检测到有人则拍照,否则休眠5秒
def detectPeople():
   if GPIO.input(3) == True:
take photo()
   else:
   time.sleep(5)
options = {
    'productKey':'**设备的ProductKey**',
    'deviceName':'**设备的deviceName**',
   'deviceSecret':'**设备的deviceSecret**',
   'port':****,
   'host':'iot-as-mqtt.cn-shanghai.aliyuncs.com' ##物联网平台域名
}
host = options['productKey'] + '.' + options['host']
def on message(client, userdata, msg):
   topic = '/' + productKey + '/' + deviceName + '/update'
   print(msg.payload)
def on connect(client, userdata, flags dict, rc):
   print("Connected with result code " + str(rc))
def on disconnect(client, userdata, flags_dict, rc):
   print("Disconnected.")
##设备上报属性
def upload device(client):
   topic = '/sys/'+options['productKey']+'/'+options['deviceName']+'/thing/event/property/
post'
   while True:
       payload_json = {
           'id': int(time.time()),
           'params': {
'people':1 ##物模型里布尔值以0和1的形式上报
               'picURL': picURLtoIoT,
```

```
},
    'method': "thing.event.property.post"
    }
    print('send data to iot server: ' + str(payload_json))
    client.publish(topic, payload=str(payload_json))

if __name__ == '__main__':
    client = iot.getAliyunIotMqttClient(options['productKey'], options['deviceName'], optio
ns['deviceSecret'], secure_mode=3)
    client.on_connect = on_connect
    client.connect(host=host, port=options['port'], keepalive=60)
    p = multiprocessing.Process(target=upload_device, args=(client,))
    p.start()
    detectPeople()
    GPIO.cleanup()
    client.loop_forever()
```

参数	说明
productKey	
deviceName	您添加设备后,保存的设备证书信息,请梦见 <b>设备证书信息</b> 。 您可在物联网平台控制台中设备的设备详情页面查看。
deviceSecret	
port	设置为 1883 。
	接入域名。获取方法,请参见 <mark>查看实例终端节点</mark> 。
host	本示例( iot-as-mqtt.cn-shanghai.aliyuncs.com )为公共实例接入。
	cn-shanghai 为本示例所在地域。

# 5.温室监控大屏

本文介绍使用Web可视化开发工具,搭建一个农业生产环境的温室监控大屏。

#### 前提条件

- 已完成创建项目。具体操作,请参见项目。
- 已完成创建产品与设备。具体操作,请参见产品和设备。

#### 背景信息

在农业生产的场景中,使用监控大屏实时展示各智能监控设备上报的数据,您可随时了解温室内和温室外的 温度、湿度、光照度、土壤水分等信息。

监控大屏示例图:



### 操作步骤

- 1. 创建Web可视化应用。具体操作,请参见创建Web应用。
- 2. 在Web应用编辑页左侧导航栏,单击**页面**,然后配置应用页面背景、分辨率等信息。
  - 分辨率选择为1920\*1080(常见宽屏比例)。
  - • 在底部工具栏,选中自适应前的复选框,应用页面就可以自适应屏幕大小(等比缩放,宽度撑满屏 幕)。

	页面	0 100 200 300 400 500 600 700 800 900 1100 1100 1200 1300 1400 1500 1600 1700 1800 150	自定义新增页1配置
M	~ 导航布局	8	作为首页
Ý			隐藏布局信息
宮			免登录访问 🛛
0			访问限制 🔮
			背景颜色 #FFFFFF
	₽ <b>₽</b>		背景图像
	◇ 页面 +		$\sim$
	日定入利增贝)		上传图片
		0 <= >	页面分辨率(所有页面都会更新)
		8	1920x1080
			1024x768
			1366x768
			1280x800
			1440X900
		2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	✓ 1920x1080
			自定义
			如何配置页面属性
		视图 ☑ 标尺   网格   ☑ 参考线      适合画布 〔	

- 3. 布局。
  - i. 拖拽矩形组件到画布上。这些矩形组件将作为其他组件的背景。
  - ii. 单击选中矩形组件, 然后在右侧配置样式:大小、位置、填充颜色、边框颜色和粗细。



iii. 拖拽一个iframe组件到页面中间,调整组件大小、位置,和配置关联链接。iframe组件可用于展示 链接的页面内容,如企业介绍视频等。

≡	<b>測试-test04</b> 公測版   IoT St	01 udio   Web可	「視化开发									已自动保存 16:43:01	ち 撤销	€ 恢复	● 预览	✔ 保存	◀ 发布	<b>I</b> 14	2 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7	釛
	组件		✔ 提示浮层		100	200 300	400 500	600 7	700 800	900 100	0 1100 120	0 1300 1400 1500	đ	習链接					×	ć
0	Q 请输入组 新 1949.4t	1件名称搜索	~	300									Ŀ.							
8	→ 基础		Ť	0										因安全政 HTTP地 HTTP地	(策原因,目) 址,需要绑定	前只支持HT 自己的域名	TPS地址,並 才能够预览	0果您填写	7	
			_	-10						1		-0	뜒	接内容						
0	T	-	RO H					Ĭ				Ĭ						参3	故来源	
	X+		El/h	100									+	URL参数 🕖						
	500	矩形	椭圆形	500				-		+		-								
		+		300						NET YEARS										
	三角形	星形	多边形	400																
		$\odot$	12	> 500				-				-6								
	箭形	指示灯	卡片	109																
	C		0	200																
	时钟	iframe	地图	800																
	<b></b>			006																
	大屏			1000																
	~ 空间			1100																
	2			1200																
	设备地图			1300									Ŀ.,							
	~ 控制			 视图	☑ 标尺	网格	🗹 參考线	_		适合画布 门	缩放 - )	-		确定	取消					

设置目标链接为HTTPS链接后,组件中将展示目标页面的内容。

↓ 注意 如果设置为HTTP链接, 画布上不会显示链接页面内容。需单击上方预览按钮, 预览页面会展示HTTP链接页的内容。

=	0 100 200 300 400 500	9 <u>660 700 860 900 1000 1100 1200 1300 1400 1500 1600</u>	配置链接	×
100			③ 因安全政策原因,目前只支持HTTPS地址,如果您填 HTTP地址,需要绑定自己的城名才能够预览	写了
1000 1000 1000 200 100 - 200 100 100 100 100 100 100 100 100 100		WEREAR- MUNICIPAL OF THE CONTROL OF	HHR内容 https://market.aliyun.com/products/201196002 4 + URL争权 ●	●数米源
1200 1100			确定 取消	
视图	☑ 标尺 网格 ☑ 参考线	适合画布 门 缩放 ——		

4. 配置文字组件,用于展示大屏的标题。

从左侧**组件**中,拖拽一个**文字**组件到画布上。然后在右侧配置栏中,设置文字显示位置、文字内容、文字样式等。

	10 E : H	+ - T + J
	728 X	25 Y 0
	480 W	54 H 🛦
3.01%が大学 期期 用品 API かた取作 新華商 10%が6/15 15%が大学 15%が大学	组件名称	文字-03dw
智慧农业解决方案 服务上和400000	组件可见性	
開入を営行と金原ルが完正、は指数中年1221- 助成 最近の特徴品件     30533586     35533586     35533586     35535586	不透明度	100
₩ 100万分: 由市市会合 信紙: 001400	1.62.026	
1210001002: 9省 时 他 书	文字内容	农业生产大数据监控
8451- 595048258 X: 0211335	~ 文字样式	
1967年 第1995年1976年1976年1976年1976年1976年1976年1976年1976	字体	徽软雅黑 ~
	> 字号	36 ~
	颜色	#058BFC
	粗细	加粗、、
	行高	54
	动客	厚由
	~371	70 T
		山府民業文字
☑ 标尺   网格   ☑ 参考线	──	and a manager of

#### 5. 配置时钟组件,用于展示当前时间。

	0 700 800 900 1000 1100 1200	1300 1400 1500 1600 1700 1800 19	0 样式	交互
80 III III				+ - T + 1
	农业生产大数据监控大屏幕	2020-04-27 17:59:44 2020-04-27 17:59:44	1490 X 408 W	21 Y 0 ° 58 H A >
200 100 11111111111111111111111111111111	エロはの次 担所 連站 AFI の公议件 基础设件 新考察     SP5555、2001C03     著語交上編次方案 目示。		组件名称 组件可见性	时钟-73fn
000			不透明度	100%
100	日 単位の90300 単位によります。 単位によります。 単位によります。 単位の90300 一位の9030 一位の90300 一位の90300 一位の9030 一位の9030 一位の9030 一〇の 一〇の 一〇の 一〇の 一〇の 一〇の 一〇の 一〇の		展示格式 背景颜色 × 文字样式	HANDIB)
>	立意知识 20入房前车 N: on.com 長方公園		字号	32 🗸 🖲
100			颜色粗细	■ #333333 加粗 · · · ●
0			~ 边框 粗细	1
00 11 11 11 10			颜色	#333333
0			(#IL	
祝图   ☑ 标尺	适合画布 🗋	缩放 ──		NOTION CONTROL DATA

6. 配置数据展示组件。

设备上报的数据可使用多种组件展示,如文字组件、表格组件、各种图形组件。配置组件时,需将组件的数据源配置为设备上报的对应数据。

本文示例中,使用了以下组件:

- 配置文字组件,展示各区域的标题。
- 配置仪表盘组件,用于展示智能设备上报的属性数据。
  - 在仪表盘组件样式配置中,配置标题为该仪表盘展示的数据类名称,如湿度、室内温度、室外温度等。
  - 数据源配置为设备上报的对应属性。如,湿度的数据源为温湿度计上报的当前湿度。
#### 下图为展示温度数据的仪表盘配置。

式-test0401 I版 LINT Studio LWeb可加化开发			已自动保存 16:50:16	ち撤销 ⊂ 恢复	● 預览 ● 保存	🗲 发布 🗧 工单 📀 帮助
100 100 200 300 400 500 800	700 800 900 1000 11	00 1200 1300 14	00 1500 1600	展示数据-数据	据源配置	×
温室环境 (1,0) (1	CULLECT SUBJECT	这择设备 想示DeviceName 6TOcwB3IOVIHtbg17Edw 約2000	але	<ul> <li>遠拝数据源</li> <li>设备</li> <li>・产品</li> <li>温湿度作感過</li> <li>・设备</li> <li>・知定设备</li> <li>数据项</li> <li>④ 设备属性 ④</li> <li>・属性 ④</li> </ul>	接口   + 动态设备   这择属性   格式参考	・ * 空设备 ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・
				(		
		适合画布 🛄 缩放 —		确定取	消	帮助文档

• 配置文字组件,用于展示某类数据值。

本示例中,使用三个文字组件共同实现数据展示,组成数据类+数据+数据单位的组合。三个文字组件的文字内容分别为:

- 手动输入的数据类名称。
- 单击**文字内容**右侧**配置数据源**按钮,在数据源配置面板,配置设备上报的相关属性数据。
- 手动输入的属性数据单位。

下图为展示室外温度数据的配置。

≡	<b>测试-test0401</b> 公测版   IoT Studio   Web可视化开发			已自动保存	17:37:13	○撤销 ♂ 恢复 ● 預	宽 🕑 保存 📢 发行	布 🖬 工单 (	<b>0</b> 帮助
*		700 800 900 1000 11	00 1200 1300 14	1500	1600	文字内容-数据源	記置		×
0		<b>杰小开关于</b> 来开始来到				选择数据源	接口		
8	8		选择设备		×	*产品			,
٢	温室环境	DURING & DIVISION	搜索DeviceName	Q	• C	温湿度传感器			×
	A THE STATE OF THE DESIGN OF T	智慧农业解决方案 80月智慧农业解决方案。3月智慧化业有关方案。 2022/2008	6TOcwB3fOVIHtx917Edw		~	*设备 +指定设备	+ 动态设备	+ 空设备	
	$ \begin{array}{c} \begin{array}{c} \begin{array}{c} \end{array} \\ \end{array} $	v <b>O</b> RP(#9: ***** state(#2: %				数据项 ● 设备属性 @ ○ i			
	0 00 40 00 00 40 40	新教授科: 新想农业新永元集 美安时代: 书次 前務英之目和1年有效				•属性 @			
	8 1 (* 56, *	立即购买加入购效率					选择属性		
		RAGR					格式参考	验证数据格式	C
	室外环境								
	□ 温度:55℃ 3 温度:50% □ 0								
	9 元素() 1000 Lux 雨量: 0		设备管理	确定	取消				
				_					
			适合画布 🌅   缩放 🗕	0		确定取消		帮助文	楢

• 配置开关组件,用于展示温室内设备状态。

并大组件的奴据源希比直力对应的议留。
--------------------

	00 1200 1300 1400 1500 1600 1770 1800	开关数据-数据源配置	×
农业生产大数据监控大	异幕 2020-04-27 17:48:38	选择数据源 设备 接口	
E BARRE BARRE	#100+ #*# 营养液检测	*产品	1
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5	N & FRIENDS	温湿度传感器	×
「1 225 225 225 225 225 225 225 225 225 22		* 设备	
		6TOcwB3fOVIHtx917Edw	/ ×
8 07 N 07 21180.7x127: 9%	H V 67/ V5 67/ V. 97	数据项	
8884 HECONA78	▲ 40090300 ★ 02π1335 4 5 5 5 5 5 5 5 5 5 5 5 5 5	● 设备属性 ❷	
B 55. 5	status75027 st maxedstature (* 1., *) (* 55, *)	•属性 🛛	1
		湿度	×
※ 単一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一		✓ 转换整数型为布尔值	
R	P	True (真) : 60 🗸	
8	M         M	False (假) : 55	
-//m2		格式参考	验证数据格式
	喷灌1 喷灌2 喷灌3 喷灌4 喷灌5		
8	风扇1 风扇2 风扇3 风扇4 风扇5		
2			
8-			
☆ 二 夜田 🛛 杨尺 🗌 网络 💟 参考线	适合画布 💟 缩放 ——	與以注	帮助又相

○ 配置一个曲线图,展示一天内的光照度。

曲线图的数据源配置为光照监控设备;实时数据时间段选择为当天0点-24点。

	0  700  800  900  1000  1100  1200  1300	0 1400 1500 1600	展示数据-数据源配置	×
温室环境 (1、1、1、1、1、1、1、1、1、1、1、1、1、1、1、1、1、1、1、	COULD CALL AND	2020-04-2 养液检测 56 56 56 56 55 55 55 55 55 55	法持数据源 205 単设备多属性 ● ● 多设备单属性 ・产品 ・ご品 ・び品 ・び品 ・び合 ・び合 ・び合 ・び合 ・び合 ・び合 ・び合 ・び合 ・び合 ・び合 ・び合 ・びつ ・びの ・びつ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	
		1., 55, 55, 55, 55, 55, 55, 55, 55, 55, 5	数据项 ④ 设备历史数据 ⑧ * 属性 @	
空外环境 温度:55℃ 法度:50% 法用温射:15w/m2 光照度:1000Lux 雨量:0			温度 × 実时数据时间段 ● 1天 → 至現在 当天0点-24点 时间单位 1小时 → 格式参考 验证数据格式	
	适合画布 []]	缩放 —	确定 取消 帮助文档	

7. 配置完成后,单击**预览**,查看和验证应用页面。

8. 单击发布,将应用发布到云端。

# 6.监测不同地域作物土壤环境

本示例创建一个设备地图大屏,通过物联网应用开发的空间功能监测并展示不同区域作物土壤的实时环境数据。

#### 前提条件

已完成创建项目(例如: project\_space)。具体操作,请参见普通项目。

#### 步骤一: 创建土壤传感器设备

- 登录物联网应用开发控制台,在页面左上角选择对应实例后,在左侧导航栏单击项目管理。
   您也可登录物联网平台控制台,在左侧导航栏选择增值服务,单击选购增值服务中的前往查看服务, 在增值服务中心,单击IoT Studio区域的立即试用,进入IoT Studio控制台。
- 2. 在普通项目列表中,找到项目: project\_space,单击项目卡片。
- 在项目详情页,单击左侧导航栏的产品,单击创建产品,完成参数配置,单击确认。
   本案例为产品命名为*土壤传感器*,选择自定义品类,其他参数使用默认值。
   参数说明的更多信息,请参见参数说明表。
- 在产品列表中,单击土壤传感器产品对应操作栏的查看,单击功能定义,为产品自定义物模型属性, 说明如下所示。

功能名称	标识符	数据类型	取值范围	单位	读写类型
土壤温度	soilTemperat ure	double(双精 度浮点型)	-10~100	摄氏度/℃	读写
土壤体积含水 率	Soilvolumetric water	double(双精 度浮点型)	0~100	百分比/%	读写
地理位置	GeoLocation	struct(结构 体)	<ul> <li>添加JSON对象,参数:</li> <li>经度 (Longitud e)</li> <li>纬度 (Latitude)</li> <li>数据类型为 double,取值 范围为- 180~180。</li> </ul>	度/°	读写

5. 在产品详情页,单击**设备数**右侧的**前往管理**。在**设备**页面,单击**添加设备**,添加3个土壤传感器设备。

DeviceName(备注名称)分别为: *soil\_sensor\_01(土壤传感器1号)、soil\_sensor\_02(土壤传感器2 号)、soil\_sensor\_03(土壤传感器3号*)。具体操作,请参见设备。

设备					
土壤传感器	设备总数 🚱 3	<ul> <li>激活设备 ②</li> <li>・当</li> <li>0</li> </ul>	前在线 🕢		局網
关联物联网平台设备 添加设	DeviceName 🗸 i	唐编入 DeviceName Q	请选择设备标签 💙		
DeviceName/酱注名称	设备所属产品	世美点节	状态	最后上线时间	摄作
soil_sensor_01 土壤传感器1号	土壤传感器	设备	◎ 未激活		查看   解除却定   删除
soil_sensor_03 二 土壤传感醋3号	土壤传感器	设备	◎ 未激活		查看   解除绑定   删除
soil_sensor_02	土壤传感藏	设备	● 未激活		查看   解除绑定   删除

 6. 本案例使用设备模拟器的属性上报数据。其中地理位置(可使用高德开放平台经纬度拾取工具)设置为 3个不同的值,其他属性值可自定义。具体操作,请参见设备模拟器。

#### 步骤二:新增空间

在项目详情页,单击左侧导航栏的空间,对应土壤传感器1号、2号、3号所在地理位置,分别新增*实验地1* 号、实验地2号、实验地3号空间。具体操作,请参见新增空间。

空间						
新增空间	关联空间 请输入空间名称或错注	Q				
	空间名称	备注	类型	缩略图	创建时间 小	操作
	实验地1号		地理空间	1	2020/06/09 15:46:36	编辑删除解绑
	实验地2号		地理空间		2020/06/09 15:45:08	编辑删除解绑
	实验地3号		地理空间		2020/06/09 11:41:12	编辑删除解绑

#### 步骤三:配置设备地图

- 1. 创建Web可视化应用(例如: 设备地图大屏)。具体操作,请参见创建Web应用。
- 2. 在Web应用编辑器中,单击最左侧的组件图标 ()。
- 3. 在画布左侧组件列表上方, 输入设备地图, 找到该组件, 然后将组件拖拽到中间画布。

有关组件添加的更多信息,请参见添加组件。



- 4. 在Web应用编辑器右侧样式中,单击编辑设备地图。
- 5. 在设备地图编辑器的设备数据页面,单击选择产品,将3个土壤传感器导入设备地图组件。

设备数据 🥥	×	E
+ 选择产	品	<u>0</u>
1个产品/3个设备	如何配置设备坐标	0)
♥ 土壌传感器	¢×	
支持设备控制 🙆		

6. 单击土壤传感器右侧的设置按钮 , 设置气泡展示设备属性值。

土壤传感器	×
3个设备	
自定义设备点样式 📀	
∨ 气泡属性展示	
配置属性	
配置属性	
配置属性	:: ×

### 步骤四:为设备添加地理空间

1. 返回设备地图编辑器,单击最右侧功能菜单的**地理空间**,添加已新增的*实验地1号、实验地2号、实验 地3号*空间。然后自定义地理空间样式,如图所示。具体操作,请参见(可选)配置地理空间。



- 2. 单击页面保存按钮,然后返回Web可视化编辑页面,刷新设备地图组件。
- 4击页面右上角预览按钮,查看设备展示效果。
   如图所示,您可在Web页面根据地图显示颜色,快速观察分析不同区域对应农作物的土壤环境。
   单击空间所在区域的任意位置,展示对应的地理空间信息;单击设备气泡点,展示该区域土壤的实时环境数据。



4. 单击发布按钮,将应用发布到云端后使用。

# 7.监控仓库储备物的报警数据

本案例创建两个业务服务和一个Web应用,用于展示不同地域仓库储备物品的报警数据,实现手动选择发出 报警信息的功能。

#### 前提条件

已完成创建项目(例如: project\_space)。具体操作,请参见普通项目。

#### 步骤一: 创建收集器设备

- 登录物联网应用开发控制台,在页面左上角选择对应实例后,在左侧导航栏单击项目管理。
   您也可登录物联网平台控制台,在左侧导航栏选择增值服务,单击选购增值服务中的前往查看服务, 在增值服务中心,单击IoT Studio区域的立即试用,进入IoT Studio控制台。
- 2. 在普通项目列表中,找到项目: project\_space,单击项目卡片。
- 在项目详情页,单击左侧导航栏的产品,单击创建产品,完成参数配置,单击保存。
   本案例为产品命名为收集器,选择自定义品类,其他参数使用默认值。
   参数说明的更多信息,请参见参数说明表。
- 4. 在产品列表中,单击收集器对应操作栏的**查看**,单击**功能定义**,自定义功能属性(信息集),用于记录发出的报警信息。

* 功能名称 💿
信息集
* 标识符 💿
MessageR
* 数据类型
array (数组) V
* 元素类型:
◯ int32 ◯ float ◯ double
* 元素个数:
100
100
* 读写类型
● 读写 ○ 只读

5. 在产品详情页,单击**设备数**右侧的**前往管理**。在**设备**页面,单击**添加设备**,添加一个收集器设备 (collector)。具体操作,请参见设备。

设备							
☆ ★ ☆ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★	设备总数 (a) 2	● 潮活设备 ◎ 1	● 当前在线 ◎ 1				C
关联物联网平台设备 添加设备	DeviceName ~	′ 请输入 DeviceName	Q 清淀	世界设备标签 >			
DeviceName/备注名称		设备所履产品	节点类型	状态/启用状态 💟	最后上线时间	操作	
collector		沙痛器	沿台	• ## <b>(</b> )	2020/07/22 17:12:22	吉香   和陸	

> 文档版本: 20220425

CONFECCION

本案例使用设备模拟器功能,调试默认模块中的属性,模拟设备发送数据,供业务服务和应用开发使用。具体操作,请参见设备模拟器。

94.00

• ux 🔪

EVENINIEE TITEEE

20.48 × 0.047

#### 步骤二:开发业务服务

1. 创建HTTP服务:获取报警信息,配置如下。具体操作,请参见公开API。

water

使用产品节点收集器,选择设备*collector*,设置查询设备属性*信息集*的**快照值**,作为HTTP服务的返回数据。具体操作,请参见公开API。

* 市局資料 🔘		如何使用该节点?
HTTP这回		
返回		
* 輸出 💿		
固定值	来自节点	变量
收集器		~
信息集		~
data.value		

- 2. 依次单击页面右上角的保存、部署调试、发布按钮,完成服务发布。
- 3. 创建设备触发服务: 仓库报警, 通过钉钉机器人发送报警信息。

节点配置如下所示。

	*节点名称 ② 如何使用该节点?
	NodeJS脚本
м <mark>оф</mark> (1717).#А. —	Nodejs (Node v6.10)
, i	*脚本 全屏 保存 扩展库管理 编码规范
	空白模板 🗸
	<pre>1/** 2 + Gencam (Object) payload 上一节者影響法 3 * Gencam (Object) payload 上一节者影響法 4 * Gencam (Object) payload 子君影物法 5 * Gencam (Object) context { gapks, papsecret } 5 * Gencam (Object) context { gapks, papsecret } 7 * module.exports = asyme function(payload, node, query, context, global) { 7 * war result = Mes_Join('Nn'); 11 * var result = Mes_Join('Nn'); 12 * return result; 13 * return result; 14 * oncole.lag(result); 15 * return result; 15 * return result; 16 * return result; 17 * return result; 17 * Gence * Context *</pre>

节点	配置
设备触发	选择产品收集器的设备collector,设置属性信息集上报时,触发服务。
NodeJS脚本	处理信息集的数组数据为字符串类型,配置如上图所示。 其中 node_f2572c70 为设备触发的节点ID。
条件判断	NodeJS脚本输出值不为空时,发出报警信息。否则使用HTTP返回节点结束服务 流。

创建业务服务的具体操作,请参见创建业务服务。

4. 依次单击页面右上角的保存、部署调试、发布按钮,完成服务发布。

#### 步骤三:开发可视化应用监控报警数据

- 1. 创建Web应用。具体操作,请参见创建Web应用。
- 2. 在Web应用编辑器中,添加一个穿梭框组件,四个文字和指示灯组件。具体操作,请参见添加组件。
- 3. 分别选中组件配置样式和数据源。
  - 穿梭框:本案例以下文静态数据为例,作为公司统计的待报警数据,在穿梭框左侧列表展示。

```
[
 {
  "label": "北京仓库报警",
  "value": "北京仓货物清空,请及时补仓! "
 },
 {
  "label": "南京仓库报警",
  "value": "南京仓货物满,请及时出仓! "
 },
 {
  "label": "杭州仓库报警",
  "value": "杭州仓货物满,请及时出仓! "
 },
 {
  "label": "天津仓库报警",
  "value": "天津仓货物清空,请及时补仓! "
 }
]
```

左侧列表对应的报警信息如下。当选中数据项移动到右侧列表框后,表示发出对应的报警信息。

```
[
"北京仓货物清空,请及时补仓!",
"南京仓货物满,请及时出仓!",
"杭州仓货物满,请及时出仓!",
"天津仓货物清空,请及时补仓!"
]
```

将发出的报警信息数据下发给设备 collector, 如图所示。具体操作, 请参见交互配置。

样式	交互
∨ 交互1	Ť
事件	值改变 🗸 🗸
动作	赋值给变量 🛛 🗸 🗸
变量名	变量值
rightM	参数: rightValue
配置	管理变量
✔ 交互2	Ŵ
事件	值改变 🗸 🗸
动作	设置设备属性 🗸 🗸
产品	收集器 编辑 清除
设备	collector
属性	信息集
属性值	动态来源

○ 文字组件:分别输入内容北京、南京、杭州、天津。

指示灯组件:绑定接口获取报警信息,通过返回的报警信息,设置指示灯状态。
 需通过数据过滤脚本,判断发出报警的地域,开启对应地域指示灯,如下图所示。

```
function _filter(data) {
    // 判断返回数组中,是否包含includes的内容。是,则输出true,表示开启指示灯,否则关闭指示灯。
    return data.includes('北京仓货物清空,请及时补仓! ') === true;
}
```

展示数据-数据源配置	>
选择数据源	
接口	~
接口来源	
服务开发工作台	~
* 选择接口	前往服务工作台 C
获取报警信息	~
请求参数 〇 静态参数 🖗 💿 动态参数 💿	
▼ 返回结果 🕜	
1 []	0 11 ()
✔ ▼ 数据过滤脚本 @	
1 function _filter(data) {  2 return data.includes('北京仓约 3 }	管 : ▶
	•
▶ 数据表配置 👔	
▶ 处理后结果 🕖	
	<b>*</b> 1
✓ 定时刷新 刷新频次 2	や

#### 发出报警信息时,仓库清空,指示灯显橘黄色;仓库货满,指示灯显红色。

							样式			交互
								+	- T	+
报警列表			发出报警		北京		712 X 48 W	48	Υ	0
胡莉人	Q		遺輸入	Q		6- <del>0-</del> 6	组件名称	78	(示灯-zp5	5m
北京会庫报警 南京会庫报警					南京	•	组件可见性			
机州位库报智 天津仓库报警		<	Not Found		杭州		小燈明度		同志が年	
						•	展示样式	8	他	
4.1%			0.15		天津		关闭		#3B81F	FF
4 M			0.90		- 41-1-		<b>HD</b>		#FERE	00

4. 单击**确定**。

#### 5. 单击右上角预览,调试应用。



钉钉机器人发送消息。



6. 单击右上方**发布**按钮 **《**,将应用发布到云端。

# 8.按步骤提交设备故障信息

本案例创建四个业务服务和一个Web应用,搭建一个设备故障上报的工单页面,引导使用Web应用的用户完成故障信息的提交。

#### 前提条件

已完成项目的创建(例如project\_space)。详细内容,请参见项目。

#### 步骤一:使用云数据库创建数据库表

- 1. 登录云数据库RDS版控制台。
- 2. 在云数据库管理页,单击创建实例,创建一个MySQL类型的数据库实例。
- 3. 在数据库实例列表中,单击该实例对应的管理操作按钮。
- 4. 在左侧导航栏中,单击账号管理,创建数据库用户账号(例如test save)和密码。
- 5. 在左侧导航栏中,单击**数据库管理**,创建数据库(例如test\_db),授权账号(例 如test\_account)。
- 6. 在左侧导航栏中,单击数据安全性,添加数据库IP白名单。具体操作,请参见设置白名单。
- 7. 在左侧导航栏中,单击基本信息,查看该数据库的基本信息,并申请外网地址。具体操作,请参见申请 外网地址。

<	<ul> <li>(运行中) </li> <li>(运行中) </li> <li>(运行中) </li> </ul>	⑦开始使用阿里云RDS
基本信息		
账号管理	基本信息	设置白名单 迁移可用区 ^
数据库管理	实例ID: m-	名称: rm-b
备份恢复	地域可用区 ❷:	类型及系列 : 常规实例 (高可用版)
数据库连接	存储类型 🛛 : 本地SSD 曲	
数据库代理	内网地址 ●: m-	内网端囗: 3306
监控与报警	外网地址 ❷: m-	外网端囗: 3306
数据安全性	读写分离地址	
服务可用性	注意:实例IP地址可能发生变化,连接串则始终不变,请使用以上连接串进行实例连接。	

- 8. 在页面右上方,单击登录数据库,输入用户账号(test\_save)和密码,登录数据库。
- 9. 在DMS页面,创建名为device\_server的数据库表,用于存储提交的工单数据,如图所示。

#### 分别设置id和time自动增长和更新。

tips:鼠标右键可查看更多操作哦~ ×	基本信息	+新増	□ □ 插入 ● 删除 □ 計上移 □ ↓ 下移					
e device_server	列信息		列名	类型	长度	可空	主键	备注
⊟ ≡ 29(4) ≡ id int(11)	索引信息	0	id	int	11			
i≡ user text	外键信息	1	user	text				联系人手机号
i≡ msg text i≕ time datetime		2	msg	text				问题描述
		3	time	datetime				工单提交时间
				默认	值 CURRENT_TIMESTAMP			
					✔ 根据当前时间戳更新			

#### 步骤二:开发业务服务

1. 登录物联网应用开发控制台,在左侧导航栏单击loT Studio > 项目管理,单击项目project\_space的 卡片。 2. 创建名为**更新步骤**的公开API服务,用于刷新步骤组件当前显示的步骤。配置如图所示。具体操作,请参见公开API。

	* 节点名称 💿	如何使用该节点?
	NodeJS脚本	
★ HTTP请求 ● → ◆ NodeJS脚本 ● → ◆ HTTP返回 ●	Node.js (Node v6.10)	
	*脚本	全屏 保存 扩展库管理 编码规范
	空白模板	~
	<pre>1 /** 2 *@param {Object} payLoad 上一 3 *@param {Object} node 描定菜小 4 *@param {Object} auery 观秀演] 5 *@param {Object} cantext { ap 6 */ 7 module.exports = async function 9 let step = payLoad.stepNum; 10 const butTXT = payLoad.butOpe 11 if(butTXT=="下一步"){ 12 l step = step+1; 13 }; 14 if(butTXT=="上一步"){ 15 l step = step-1; 16 } 17 18 return step; 19 }</pre>	Ÿ系的輸出 <sup>Ÿ</sup> 系的輸出 <sup>新一</sup> ↑ <sup>Ÿ</sup> 系的輸出 µpKey, appSecret } (payload, node, query, context) { pr;

节点	配置说明
HTTP请求	Action可自定义, 需配置入参: <ul> <li>stepNum: 当前步骤数。</li> <li>butOper: 当前更新步骤的操作。</li> </ul>
Node.JS脚本	编辑更新步骤值的脚本。
HTTP返回	选择 <b>来自节点</b> 的NodeJS <b>脚本</b> ,返回更新后的步骤值。

3. 创建名为手机号校验的公开API服务,用于校验当前输入的手机号码是否正确。配置如图所示。

		* 节点名称 📀		如何使用	脂该节点?
_		NodeJS脚本			
▲ HTTP请求	▶↓◇ NodeJS脚本 �ロ → ┣	HTTP返回 Node.js (Node v6.1	10)		
		* 脚本	全屏 保存	扩展库管理	编码规范
		空白模板			$\sim$
		1 /** 2 * @param {Ob 3 * @param {Ob 4 * @param {Ob 5 * @param {Ob 6 */ 7 * module.expor 8 const ipho 10 let warn = 11 if (iphone 12 warn = 13 // 4 if (!( 15 warn 16 } 17 } 18 return war	<pre>bject} payLoad 上一节点的输出 bject} node 指定某个节点的输出 bject query 服务近第一个节点的输出 bject context { appKey, appSecret tts = async function(payLoad, node, bone = payLoad.iphone; = '手机号不能为空!'; = &amp;&amp; iphone.length &gt; 0) {</pre>	} query, con	text) {

节点	配置说明
HTTP请求	Action可自定义,配置的入参 <b>iphone</b> ,设置为 <b>非必填项</b> ,用于指定输入 的手机号码。
Node.JS脚本	<pre>编辑检验手机号码的脚本。 module.exports = async function(payload, node, query, context, global) {    const iphone = payload.iphone;    let warn = '<b>手机号不能为空! ';</b>    if (iphone &amp;&amp; iphone.length &gt; 0) {      warn = '<b>校验通过! ';</b>      // 判断手机号是否合法      if (!(/^1[3456789]\d{9}\$/.test(iphone))) {         warn = '<b>当前输入不合法! ';</b>      }    }    return warn; };</pre>
HTTP返回	选择 <b>来自节点</b> 的NodeJS <b>脚本</b> ,返回校验结果。

### 4. 创建名为存储数据的HTTP服务,将工单数据存储到已创建的数据表device\_server。配置如图所示。

HTTP请求 ●□▶□ ● 3 云数据库My ●□ ● ■ ● HTTP返回 ●	<ul> <li>* 节点名称 ◎ 如何使用该节点?</li> <li>云数据库MySQL</li> <li>参数</li> <li>* 用户名</li> <li>test_save</li> <li>* 密码</li> </ul>
● 100% ● □ □ □ □ 2	• 注接地址 ◎ m-bi * 数据库名 test_db • 被口号
	<pre>     [5]     [5]     [3306     * 操作类型     [插入</pre>

节点	配置说明
HTTP请求	Action可自定义, 需配置入参: · user: 联系人手机号。 · msg: 问题描述。
云数据库MySQL	设置插入数据的参数,详细配置请参见云数据库MySQL。 { "table": "device_server", "rows": [{ "user": "{{payload.user}}", "msg": "{{payload.msg}}" }] }
HTTP返回	选择 <b>来自节点的云数据库MySQL</b> ,返回插入数据的操作结果。

5. 创建名为提交结果的HTTP服务,返回提交工单的结果。配置如图所示。

#### 最佳实践·按步骤提交设备故障信息

▲	节点配置 节点日志
	* 节点名称 💿 如何使用该节点?
	NodeJS脚本
→ HTTP请求 → → ↓ ◇ NodeJS脚本 → ↓ ◇ HTTP返回	Node.js (Node v6.10)
	*脚本 全屏 保存 扩展库管理 编码规划
	空白模板 インジャング ション
	2 * @param {Ubject} payLoad エーヤ点的搬出
	5 * @param {Ubject} node 信定来了 P品の通信 4 & Granger (Object) guony 間条逆営一人芸人的報告
	F & Operan (Object) query XX 37 00 37 7 7 7 7 10 mm 40)
	6 */
	7; module.exports = async function(navload, node, query, context) {
	Reserves a synce reneered (payroad) nodely queryy concerces (
	<pre>9 const user = pavload.user:</pre>
	<pre>10 const msg = payload.msg;</pre>
	11 let result='提交丁单成功,请保持电话畅诵!'
	<pre>12 if(user.length ==0    msg.length == 0){</pre>
	13 result="提交工单失败,请刷新页面,填写完整信息!";
	14 }
	15
	16 return result;
	17 }

节点	配置说明
HTTP请求	Action可自定义, 需配置入参: • user: 联系人手机号。 • msg: 问题描述。
Node.JS脚本	<pre>编辑检验入参内容是否为空的脚本。 module.exports = async function(payload, node, query, context) {     const user = payload.user;     const msg = payload.msg;     let result='提交工单成功,请保持电话畅通! '     if(user.length ==0    msg.length == 0){         result='提交工单失败,请刷新页面,填写完整信息! ';     }     return result; }</pre>
HTTP返回	选择 <b>来自节点</b> 的NodeJS <b>脚本</b> ,返回检验结果。

6. 分别调试已配置的业务服务,并将服务发布到云端。

#### 步骤三:开发可视化应用提交工单

- 1. 创建Web应用。详细内容,请参见创建Web应用。
- 在Web应用编辑器中,执行以下步骤,完成如图所示配置。
   其中组件添加操作请参见添加组件,各组件详细配置请参见各组件帮助文档。
   您可在画布中选中组件后,通过拖动改变组件位置。

	提交工单	
描述问题	填写手机号	提交
清輸入联系人手机号		手机号不能为空!
<b>提交工单失败,</b> ○提交 ●返回	请刷新页面,填写完整(	記! 下-步 上-步 上-步 下-步

i. 按照下表依次添加组件,并配置组件名称和样式。

组件	组件名称	样式说明
矩形	背景	<ul> <li>自定义宽(W)和高(H)</li> <li>填充颜色: #FFFFFF</li> <li>边框颜色: #CEEBC2</li> </ul>
文字	标题	文字内容:提 <b>交工单</b>
步骤	步骤	<ul> <li>数据源:接口更新步骤,实现步骤切换。</li> <li>设置动态参数stepNum和butOper分别为变量:step和变量:butTXT,定时刷新频次2秒。</li> <li>其中变量step和butTXT需新增,分别设置默认值为0和下一步。</li> <li>步骤设置:配置3个步骤依次为描述问题、填写手机号、提交,默认选中第一步的描述问题。</li> <li>步骤样式:类型为箭头,组件背景为#CEEBC2。</li> </ul>
<b>立</b> 本框	手机号框	默认提示:请输入联系人手机号
文本性	问题描述框	默认提示:请输入问题具体内容
	验证提示	<ul> <li>文字样式:字号为16,颜色为#FF0000。</li> <li>文字内容为接口数据源:手机号校验。</li> <li>其中动态参数iphone为组件值:手机号框,定时刷新频次2秒。</li> </ul>

文字 <b>组件</b>	组件名称	样式说明
	提交	文字内容为接口数据源: <b>提交结果</b> 。 设置动态参数 <b>user为组件值:手机号框,msg为组件值:问题描述</b> 框,定时刷新频次2秒。
单选	单选	<ul> <li>数据源:静态数据源代码如下,展示提交和返回两个选项。         </li> <li></li></ul>
		■ 默认选中:返回。组件值改变时,表示提交工单。
	按钮4	按钮内容:上一步
按钮	按钮3	
17 11	按钮2	按钮内容:下 <b>一步</b>
	按钮1	

ii. 分别选中按钮组件,配置点击事件的交互动作,使按钮在对应步骤隐藏或显示。

如下图所示,序号①②③④分别对按钮1、按钮2、按钮3、按钮4的配置。

需设置变量step分别赋值为固定值1、2、3、2;变量butTXT赋值为对应的按钮内容text,实现通过按钮操作切换步骤。

<ol> <li>样式</li> </ol>	交互	✓ 交互1 2	Ť	3 样式	交互	4 样式	交互
∨ 交互1	Ŷ	事件	点击	✓ 交互1	Ŧ	∨ 交互1	Ŵ
事件	点击 >	动作	赋值给变量 🗸 🗸	事件	点击	事件	点击~
动作	赋值给变量 🗸 🗸	变量名	变量值	动作	賊債给变量 🗸 🗸	动作	赋值给变量 🗸 🗸 🗸
变量名	变量值	step	2	变量名	变量值	变量名	变量值
step	1	butTXT	参数: text	step	3	step	2
butTXT	参数: text	配置	管理变量	butTXT	参数: text	butTXT	参数: text
配置	管理变量	→ 交互2	Ŧ	配置	管理变量	配置	管理变量
∨ 交互2	ŵ	事件	点击  >	∨ 交互2	Ŵ	→ 交互2	Ŵ
事件	点击 >>	动作	展示隐藏 >	事件	点击	事件	点击
动作	展示隐藏 >	点击出现	单选 × V	动作用	展示隐藏 🗸 🗸	动作	展示隐藏
点击出现	手机号框 × × 验证提示 × 按钮2 ×	点击隐藏	按钮3 × 手机号框 × × 验证提示 ×	点击出现	手机号框 X Y 验证提示 X 按钮4 X	点击出现	问题描述框 X V 按钮1 X
点击隐藏	问题描述框 × × × 按钮1 ×		按钮2 X	点击隐藏	単选 × V 按钮3 ×	泉山通識	Folista × V 验证提示 × 按钮4 ×

有关交互动作的详细配置,请参见交互配置。

iii. 选中单选组件, 配置值改变的交互动作, 实现信息提交后的校验存储和结果展示。如图所示。

样式	交互		0.52			×
∨ 交互1		Ť	月反	券配直 		^
事件	值改变	$\sim$	接口	]来源 8条开发工作台		~
动作	调用其它服务	$\sim$				
接口来源	服务开发工作台 编辑 清除		* 进	择接口 P储数据	前往服	务工作台 C ~
服务名称	存储数据		请3 〇	\$参数 静态参数 🛯 💿 动态参数 🜑		
∨ 交互2		Ĩ	* us	er 🔞		
事件	值改变	$\sim$	值:	组件值: 手机号框	0	参数来源
动作	展示隐藏	$\sim$	* m	sg 😰		
点击出现	提交 🗙	$\sim$	值:	组件值: 问题描述框	8	参数来源
点击隐藏	按钮3 × 单选 ×	~	- 1			<b>0</b> 7 6
	步骤 X 标题 X			<pre>1 t 2 "affectedRows": 1, 3 "changedRows": 0, 4 "fieldCount": 0, 5 "insertId": 6, 6 "message": "", 7 "protocol41": true, 8 "serverStatus": 2, 9 "warningCount": 0 10 }</pre>		

3. 分别关闭组件手机号框、验证提示、提交、单选、按钮2、按钮3、按钮4的组件可见性,如图所示。

	提交工单	
描述问题	填写手机号	提交
请输入问题具体内容		
		下 <del>一步</del> 上一步

 调整矩形组件宽\*高(700\*250),使用组件通用样式的对齐功能,设置各组件位置。如图所示,详细操 作请参见样式配置。

	提交工单		
描述问题	填写手机号	$\geq$	提交
提交工单失败, 请输入问题具体内容			<b>吉息!</b> 手机号不能为空! 下 <del>一步</del>

5. 单击右上角**预览**按钮,调试应用。

-6				
2	提交	$\rightarrow$	填写手机号	描述问题
		\$		
		\$		

6. 前往已登录数据库(test\_db)的DMS页面,查询数据表中存储的数据。

表可编程对象	SQLConsole 编辑表.devi ×						
支持%模糊匹配表名称 ○ ∈	<u>     牧庁(F6)</u> 格式化(F10)     执行计划(F9)     常用SQL ✓ SQL诊断     显示段量     前往数倍开发     前往数据可视化     前指数						
tips:銀标右键可查看更多操作哦~ × 団 曲 device_server	SELECT * FR	OM `device_server` ;					
	L						
	单行详情新增	删除 提交修改 部署周期任务	数据可视化-New 🛆				
	序号	i≡ iq	1t 📄 user	lt 🖹 msg	Jr 💮 time		
	1	1		tst	2020-08-03 11:12:47		

7. 返回Web应用编辑页面,单击右上角**发布**按钮,将应用发布到云端。

# 9.统计会议签到人数

本示例创建一个业务服务,通过全局变量功能,实现会议签到人数的统计,并在钉钉群中推送签到人数的消息。

#### 前提条件

已完成项目的创建(例如project\_space)。具体操作,请参见项目。

#### 步骤一: 创建读卡器设备

- 1. 登录物联网应用开发控制台,在页面左上角选择对应实例后,在左侧导航栏单击 **项目管理**,然后单击 项目project\_space的卡片。
- 2. 在项目详情页,单击左侧导航栏的产品,单击创建产品,完成参数配置,单击保存。

本案例为产品命名为读卡器,选择自定义品类,其他参数使用默认值。

参数说明的更多信息,请参见创建产品。

- 3. 在产品列表中,单击读卡器产品对应操作栏的查看,单击功能定义,为产品自定义功能属性**卡号**,标识 符为CardID,数据类型为**字符串**,选择**读**写类型。
- 在产品详情页,单击设备数右侧的前往管理。在设备页面,单击新增设备,添加一个读卡器设备 (reader)。具体操作,请参见设备。

#### 步骤二: 创建业务服务

- 1. 创建业务服务。具体操作,请参见创建业务服务。
- 2. 在业务逻辑编辑器,添加数值型的全局变量count,设置默认值为0,如图所示。

具体操作,请参见变量配置。

添加变量	×
* 变量名称 💿	
count	
* 变量类型 📀	
● 全局变量 ○ 局部变量	
* 数据类型 💿	
Num (数值型)	
默认值	
0	
描述	
请输入描述	
加密 💿	
	_
确定取消	

3. 在业务逻辑编辑器,单击左侧的**节点**按钮,依次拖拽**设备触发、数值计算、变量设置**和**钉钉机器人**节 点到画布上,并进行连接、配置。

节点	配置
设备触发	选择产品读卡器和设备reader的属性上报类型。
数值计算	设置全局变量count作为数据源,设备触发后变量值加1。 数值计算节点以参数calcul_result输出最终计算结果。
变量设置	单击+变量,将变量count值设置为数值计算节点的结果calcul_result值,实现人数的 递增统计。 ⑦ 说明 变量设置节点输出结果的参数字段与其上一个节点输出的参数字段一致。
钉钉机器人	配置钉钉群机器人的Webhook地址。具体操作,请参见创建钉钉机器人。 配置方法为自定义的text。 { "msgtype": "text", "text": { "content": "签到次数为{{payload.calcul_result}}" } } 如guad.calcul_result 表示输出上一个变量设置节点的设置结果。

数值计算节点配置。

节点配置 节点日詞	±							
*节点名称 ② 如何使用该节点?								
数值计算	数值计算							
输入配置								
* 数据源 💿								
固定值	来自节点	变量 🗸						
count		~						
数值计算								
* 运算方法 👔								
相加 (輸出值为数据)	原加上各参数的结果)	$\checkmark$						
添加参数								
* 参数1		删除						
固定值 🗸	来自节点	变量						
数值型		$\sim$						
1								

#### 变量设置节点配置。

变量设置	×
将变量	
count	$\sim$
* 设置为 固定值 来自节点	变量
上一节点(payload)	$\sim$
calcul_result	
确定	取消

4. 单击页面右上方保存按钮 🖪 ,保存设置。

### 步骤三:调试与发布

- 1. 在业务服务编辑页面,单击右上方的部署调试按钮 🐟 ,部署并启动服务。
- 2. 部署完成后,再次单击右上方的部署调试按钮,,调试服务。

i. 单击调试 > 前往, 前往产品的设备模拟器页面。

- ii. 单击启动模拟设备。
- iii. 选择上行指令调试 > 属性上报。
- iv. 在默认模块中推送调试指令。

本示例中,模拟用户刷卡签到三次,即推送CardID的值三次。

设备模拟器		
講成设备: 法中職 ~		
上行描令调试 下行描令调试	设备端日志 查看云端日志	
自定义 Topic 属性上报 事件上报 在线 C	) <u>类型</u> /时间	内容
懐染: 默认機会 >>	设备上报数据 2020/04/20 15:36:40	2020-64-20 15:16-64.178 274425986C0946C89F8A257474608.6 upstream - bc1ypes PROPERTY_REPORT parames ["d"1123", 'otd'12DF2Q_ENty[1148E0180010 1" 'memod' Thing avertapioper post" parame"(_uzdd' 1/011) "app(-") 2060/11412864/_ version"10", rest_prode/200 menagescures tapica?
91011		19x1/http://www.cateure/19x1/19x1/00000000000000000000000000000
	沿離 HR世祖 2020/04/20 15:36:27	2020-64-20 15:1621059 2744299800949618F8A3873476086. upstream - bolyges PROPERTY_REPORT parames ("d"1123"):otd3112022.04 RHy1148E0160010 11" method" Thing evertproperty post" parames ("d"1578 T topic") ''ope 18/0123481.424", ''operation' 10" parameters ("d"1578 T topic") ''operation' parameters ("d"1123"):otd3112022.04 RHy1148E0160010 8/0123481.424", ''operation' 10" parameters ("d"123"):otd3112022.04 RHy1148E0160010 9/0123481.424", ''operation' 10" Parameters ("d"123"):otd3112042.04 RHy114801484 9/012341.424", ''operation' 10" Parameters ("d"123"):otd31120481.44", ''operation' 10" Parameters ("d"123"):otd31120 9/0123481.44", ''operation' 10" Parameters ("d"123"):otd31120481.44", ''operation' 10" Parameters ("d"123"):otd31120481.44", ''operation' 10" Parameters ("d"123"):otd31120481.44", ''operation' 10" Parameters ("d"123"):otd31120 9/0123481.44", ''operation' 10" Parameters ("d"123"):otd31120 9/0123481.44", ''operation' 10" Parameters ("d"123"):otd31120 9/0123481.44", ''operation' 10" Parameters ("d"123"):otd31120 9/0123
	役称⊥1号認知報 2020/04/20 15:36:08	2020-04-20 151008001 274429980094058F26426742180080, upixeem - fu/2yee2P60P871_4EP0018 pwm=1/u1/23*/uU/272P20.1PHy114801100010 1*****************************

钉钉消息群接收到提示消息。

	机器人
0	签到次数为1
	机器人
	签到次数为2
	机器人
	签到次数为3

- 3. 单击发布按钮 🦪 ,发布该服务。
- 4. 业务需要发布到云端,才能被调用。

# 10.室内温度监控

## 10.1. 概述

阿里云IoT Studio全新推出移动可视化开发工具。本文介绍通过使用该工具,快速搭建室内温度监控应用, 监测室内温湿度和空气质量,并手动控制空调和空气净化器设备,调节室内温湿度和空气质量。

#### 背景信息

移动可视化开发是物联网应用开发(IoT Studio)提供的开发工具。无需写代码,只需在编辑器中,拖拽组件到画布上,再配置组件显示样式、数据源和动作。您可以快速搭建设备控制、设备状态展示、数据展示等物联网场景下的移动应用。

#### 操作步骤

- 1. 创建项目和产品。
- 2. 开发移动应用监控室内温度。
- 3. 发布。

## 10.2. 创建项目和产品

本文介绍在物联网应用开发(loT Studio)上创建项目,并为该项目创建产品和设备。

#### 步骤一:创建项目

- 1. 登录物联网应用开发控制台,在页面左上角选择公共实例后,在左侧导航栏单击项目管理。
- 2. 在项目管理页面,单击新建项目。

普通项目							
新建项目	搜索		Q				
创建于 20	)19/11/11 12:34	1:34	:	创建于 20	19/11/11 12:34	:34	:
<b>12</b> 设备	5 <sub>Web应用</sub>	<b>6</b> 业务逻辑		<b>12</b> 设备	5 <sub>Web应用</sub>	<b>6</b> 业务逻辑	

3. 在新建项目页面,移动鼠标至新建空白项目区域,单击创建空白项目。

- 4. 在新建空白项目对话框, 输入项目名称(室内温度监控)和描述信息。
- 5. 单击确认。

#### 步骤二: 创建产品和设备

1. 在项目主页,单击左侧导航栏的产品。

2. 在产品页面,单击产品列表左上方的创建产品。

室内温度监控 / 产品			
产品			
关联物联网平台产品	创建产品 请输入产品	洛称查询 Q	请选择产品标签 💙

3. 在**创建产品**页面,输入产品名称,选择**自定义品类**,其他参数使用默认值,单击**确认**。重复此操作分 别创建五个产品(客厅温湿度、卧室温湿度、厨房温湿度、空气检测器、开关遥控器)。

产品创建成功后,回到产品页面,可在产品列表中查看到已创建的产品。

室内温度监控 / 产品 / 创建产品
← 创建产品
* 产品名称
到影響
* 所属品类 💿
○ 标准品类 (●) 自定义品类
* 节点类型
直连设备     水 网关子设备     水 网关子设备     水 网关子设备     水 网关设备     水
连网与数据
* 连网方式
Wi-Fi 🗸 🗸
* 数据格式 🔞
ICA 标准数据格式 (Alink JSON) V
◇认证方式
更多信息
▼ 产品描述
确认 取消

- 4. 在产品列表中,找到目标产品,单击产品对应的查看。
- 在产品详情页面,单击功能定义页签,再单击编辑草稿,选择添加自定义功能。
   本示例添加以下自定义属性。具体操作,请参见单个添加物模型。

客厅温湿度、卧室温湿度、厨房温湿度产品

功能名称	标识符	数据类型	取值范围	单位	读写类型
湿度	Humidity	float(单精度 浮点型)	0~100	百分比(%)	读写
温度	Temperature	float(单精度 浮点型)	-20~50	摄氏度(℃)	读写
空气检测器产品					
功能名称	标识符	数据类型	取值范围	单位	读写类型
PM25浓度	PM25	float(单精度 浮点型)	取值范围: 0~10000	微克每立方米	读写
开关遥控器产品					
功能名称	标识符	数据类型	取值范围	单位	读写类型
卧室灯开关	RoomLightSw itch	bool (布尔 型)	。 0: 关 。 1: 开	不涉及	读写
客厅灯开关	LivingLightSwi tch	bool (布尔 型)	。 0: 关 。 1: 开	不涉及	读写
空调开关	AirConditionin gSwitch	bool (布尔 型)	。 0: 关 。 1: 开	不涉及	读写
空气净化器开 关	AirCleanerSwi tch	bool (布尔 型)	。 0: 关 。 1: 开	不涉及	读写

6. 在左侧导航栏,单击**设备**。在**设备**页面,分别为五个产品创建设备,如下图所示。

具体操作,请	参见添加设备。					
主页	智能空调 / 设备					
产品	设备					
设备	全部产品 🗸	设备总数 ◎ ● 激流 2 2	舌设备 🔍	● 当前在线 <sup>◎</sup> 1		
空间	关联物联网平台设备 添加设备	DeviceName V 请输入 De	eviceName	Q 请选择设备标签	£ ~	
账号	DeviceName/备注名称	设备所属产品	节点类型	状态	最后上线时间	操作
	Airconditioner	智能空调	设备	● 在线	2021/08/25 17:29:39	查看   解除关联   删除
	Tempdevice	温度检测仪	设备	● 商线	2021/08/25 10:12:36	查看   解除关联   删除
	删除					

- 7. 本案例使用设备模拟器的属性上报数据。更多信息,请参见设备模拟器。
  - i. 登录物联网平台控制台。
  - ii. 在**实例概览**页面, 单击公共实例进入实例详情页面。
  - iii. 在左侧导航栏,选择**监控运维 > 设备模拟器**。
  - iv. 选择本次模拟的设备(例如空气检测器),单击启动设备模拟器。
  - v. 选择属性上报功能, 推送调试指令, 如下图所示。

设备模拟器			
講試役音: 空气检測器 ~ Air-detector ~			
上行擔令调试 下行指令调试	i	设备端日志 查看云端日志	
面定义 Topic         層性上級         事件上級	在线 🌑	英型 / 时间	内容
機快: 取込機快 >		云平台到设备消息 2021/04/27 11:52:33.251	received topics/gy// s1°method1*thing.e
90		设备到云平台消息 2021/04/27 11:52:33.179	publish topic=/iys/a hing event property/
		云平台到设备消息 2021/04/27 11:52:33.105	received topics/ys/ s <sup>*</sup> /method <sup>*</sup> /thing.e
		设备到云平台消息 2021/04/27 11:52:33.44	publish topicz/tys/a hing event property j
		云平台到设备消息 2021/04/27 11:52:32:934	received topics/ys/ s1/method1/thing.e
		设备到云平台消息 2021/04/27 11:52:32:875	publish topicx/gy/a hing event property
		云平台到设备消息 2021/04/27 11:52:32.812	received topics/ys// st/method/thing.e
		设督到云平台消息 2021/04/27 11:52:32.740	publish topicx/gs/a hing event property (
发送指令 領略推送 重量			

推送数据后,可在页面右侧设备端日志下查看设备端日志。

有关产品和设备的更多信息,请参见产品和设备。

#### 后续步骤

开发移动应用监控室内温度。

# 10.3. 开发移动应用监控室内温度

本文介绍搭建室内温度监控移动应用的开发过程。

#### 前提条件

已完成创建项目和产品。

步骤一: 创建移动应用

- 1. 登录物联网应用开发控制台,在页面左上角选择公共实例后,在左侧导航栏单击项目管理。
- 2. 在普通项目列表中,找到室内温度监控项目,单击项目卡片。
- 3. 在项目主页页面的项目开发下,单击移动应用页签。
- 4. 在移动应用页签,单击应用列表上方的新建。
- 5. 在新建移动应用对话框中, 输入应用名称(室内温度监控)和描述, 单击确认。
- 6. 移动应用创建完成后, 会自动进入移动应用编辑器。
- 7. 在移动应用页面的画布左侧,单击编辑按钮 🥕 ,编辑页面标题。

#### 步骤二: 配置组件样式和数据源

- 1. 在移动应用编辑器中,单击最左侧的组件图标 💿 。
- 2. 从左侧组件列表中,拖拽一个图片组件到画布上方。

在移动应用编辑器右侧样式中,设置图片样式并单击**上传图片**,上传室内布局图,如下图所示。更多信息,请参见<mark>图片</mark>。

-100 0 100 200 300 400 500 600 700	样式	交互
	组件名称 ✔ 组件可见性	图片-dok7
室内温度控制	不透明度	100%
客庁大丁 空中で少子化設置 BF 家大丁	上边距     下边距       0     0       宽度     高度	左边距 右边距 0 0 350 150 150 100 100 100 100 100 100 100 1
	图片	
×		重置 上传图片 S

 拖拽一个选项卡组件到图片组件下方,设置三个选项卡,分别为客厅温湿度、卧室温湿度、厨房温湿 度,对应展示不同内容。在每个选项卡中添加三个卡片组件和一个柱状图组件,展示卧室温湿度、客厅 温湿度、厨房温湿度、空气质量的状态,如下图所示。



- i. 分别选中温度、湿度、PM2.55浓度卡片,配置卡片样式和数据源。本案例以**客厅温湿度**页签为 例。
  - a. 卡片组件以一定的界面样式展示常用数据,设置显示大小、标题、单位、背景颜色。更多信息,请参见卡片。
  - b. 在移动应用编辑器右侧**样式**中,单击**展示数据**后的**配置数据源**。数据源参数配置如下,详细 配置说明,请参见配置数据源。

配置项	温度卡片	湿度卡片	PM25浓度卡片
选择数据 源	选择 <b>设备</b> 。		
产品	本案例选择 <b>客厅温湿度</b> 。	本案例选择 <b>空气检测器</b> 。	
设备	本案例选择 <b>指定设备</b> 为roor	本案例选择指定设备 为Air-detector。	
数据项	作为卡片组件数据源的数据现		
属性	选择 <b>温度</b> 属性。	选择 <b>湿度</b> 属性。	选择 <b>PM25浓度</b> 属性。

ii. 选中柱状图组件,在移动应用编辑器右侧**样式**中,设置柱状图的显示大小,并单击**展示数据**后的**配置数据源**。数据源参数配置如下。更多信息,请参见配置数据源。

配置项	说明		
选择数据源	选择 <b>数据表资源</b> 。		
数据表类型	选择 <b>设备数据表</b> 。		
产品	本案例选择客厅温湿度。		
设备选择方式	选择项目下指定设备。		
选择设备	本案例选择 <b>指定设备</b> 为room4。		
刷新设置	选中 <b>定时刷新</b> 复选框,并设置刷新频次为1分钟。 ⑦ 说明 如果不选中,数据将不会自动更新,为保证性能,刷新频次间隔最低为1分 钟。		

 拖拽四个自由卡片组件到选项卡组件下方,在自由卡片组件中添加文字组件和开关组件,控制客厅灯、 卧室灯、空调、空气净化器的开关状态,如下图所示。



- i. 双击自由卡片进入编辑页面。从左侧**组件**列表中,拖拽一个开关和一个文字组件到自由卡片中,设 置标题和样式。
  - 自由卡片组件作为背景容器,设置填充颜色为#FAFAFA。更多信息,请参见自由卡片。
  - 文字组件作为开关标题(例如**卧室灯**)。更多信息,请参见文字。
  - 开关组件作为灯开关状态的显示,分别为ON和OFF状态上传对应图片(您可自定义)。更多信 息,请参见<del>开关</del>。

ii. 选中开关组件,在移动应用编辑器右侧样式中,单击开关数据后的配置数据源。以卧室灯开关为例,数据源参数配置如下。更多信息,请参见配置数据源。

配置项	说明
选择数据源	选择 <b>设备</b> 。
产品	本案例选择开关遥控器。
设备	本案例选择 <b>指定设备</b> 为switch。
数据项	作为开关组件数据源的数据项。默认为 <b>设备属性</b> 。
属性	选择 <b>卧室灯开关</b> 属性。

5. 单击编辑器上方保存按钮,保存设置。

#### 后续步骤

发布

## 10.4. 发布

在移动应用编辑器页面,完成组件配置后,需将应用发布到云端,以供使用。本文介绍应用发布和集成使用 的具体内容。

#### 前提条件

已完成移动应用可视化界面的搭建。

#### 步骤一:预览与调试

- 1. 在移动应用编辑器中,单击页面右上方的预览,查看并调试组件的显示效果。
- 2. 在应用预览页面右侧,进行选择机型。
- 3. 使用手机扫描二维码, 在手机端调试应用的功能。



步骤二:发布

- 1. 回到移动应用编辑器,单击最左侧导航栏的应用设置图标③,在域名管理页签添加域名。
- 2. 单击最左侧导航栏的页面图标参, 单击页面右上角发布图标 4。
- 3. 单击确定。
- 4. (可选)配置应用鉴权。具体操作,请参见应用鉴权。

开启账号或Token鉴权后,只有使用账号登录或Token加签的访问者才允许访问应用。

5. 返回物联网应用开发平台IoT Studio > 应用开发页面,或应用所在项目的主页,单击移动应用页签, 单击已发布应用操作栏的发布地址,可查并复制应用访问地址。

项目开发					-		
Web应用	移动应用业务服务	数据资产 数据任	2		手60a: 2834	1960月及中国星	历史功能 🗸
+ 新建	▽ 搜索				1252		G
名称	发布状态		描述	修改时间小	E-16224		
室内温度监控	✓ 已发布	ī	-	2021/04/2611:22:41	编辑 预范 发布	5地址   删除	

发布结果

* 🕐 🏵 اللہ 🕆 🖾		I[[ (BI]) 11:26
×	室内温度监控	
	室内温度监控	
空气净化空调	/器 卧室灯	
客厅温湿度	卧室温湿度	厨房温湿度
<b>23 ℃</b> 温度	<b>30</b> % 湿度	<b>90 µg/m3</b> PM25浓度
柱状图 ● 温度		• 湿度
40 30 20 10 8:00 12	00 16: 00	20: 00 00: 00
卧室灯 8开8 🤤	2	客厅灯 E#8 📿
空调	÷ ع	2 <b>气净化器</b> 3开启 📀

有关手机兼容性的使用说明,请参见手机兼容性说明。
# 11.IoT Studio使用LA自定义接口全链路 开发

本文介绍如何基于物联网数据分析创建自定义数据指标,并输出到IoT Studio的报表组件中。

#### 前提条件

- 已购买数据型实例。具体操作,请参见购买物联网实例。
- 已创建产品和设备。具体操作,请参见创建产品和设备。
- 已定义物模型。具体操作,请参见为产品定义物模型。
- 已完成设备上报数据。具体操作,请参见用设备模拟器调试设备。

### 步骤一:创建项目

- 1. 登录物联网应用开发控制台,在页面左上角选择已购买的数据型实例后,在左侧导航栏单击项目管理。
- 2. 在项目管理页面,单击新建项目。

普通项目							
新建项目	搜索		Q				
创建于 20	19/11/11 12:34	i:34	0 0	创建于 20	 19/11/11 12:34	:34	:
<b>12</b> 设备	5 <sub>Web应用</sub>	<b>6</b> 业务逻辑		<b>12</b> 设备	5 <sub>Web应用</sub>	<b>6</b> 业务逻辑	

- 3. 在新建项目页面,移动鼠标至新建空白项目区域,单击创建空白项目。
- 4. 在新建空白项目对话框, 输入项目名称(例如: studio\_link\_la\_test)和描述信息。
- 5. 单击确认。

## 步骤二:关联已创建的产品和设备

- 1. 在项目主页,单击左侧导航栏的产品。
- 2. 在产品页面,单击产品列表左上方的关联物联网平台产品。
- 3. 在**关联物联网产品**面板,选中已创建的产品(例如:温湿度传感器),并选中**关联产品同时关联其**下 所有设备复选框。

■ 案例   V2019-11-26,开发内部	親悟   loT Studi					关联物联网产品					×
主页						您可以关联 物联网干台 · 设备管理 中	的任意产品			更多期	b
产品	产品					请输入产品名称查询	搜索				
设备	关联物联网平台产品	创建产品	请输入产品名称查询			名称		ProductKey	节点类型	创建时间	
空间				节点类型	自然此时间	12 注思度伸紧器			05	2021-05-25 16:07:24	
账号	這思度传感器			设备	2021/05,	Mar. 2006.14718/148				2021/03/23 10:07:24	
										10.00	10.04

4. 单击确定。

可在该项目的产品页面的产品列表中查看已关联产品。

#### 步骤三:使用数据分析备份产品数据

- 1. 登录物联网平台控制台, 在实例概览页, 找到并进入已购买的数据型实例。
- 2. 在左侧导航栏,选择数据分析 > 数据管理。
- 3. 在产品列表中,找到已创建的产品(例如:温湿度传感器),并对其进行备份。
  - 单个备份

单击产品操作栏的**备份**。

• 批量备份

选中产品名称前的复选框,单击列表下方的备份。

4. 在备份对话框中,设置备份类型和备份周期,单击确定。

参数	说明
备份类型	默认同地域,不支持修改。
备份周期	可选1个月、2个月、3个月、4个月、5个月、6个月、7个月、8个月、9个 月、1年、2年、3年、5年、10年、永久(102年)。

5. (可选)在产品列表中,找到目标产品,单击编辑、停止或开启,修改备份设置。

## 步骤四:基于分组创建自定义指标

1. 为产品导入原始定义指标。具体操作,请参见导入原始定义指标。

2. 新增三个原子词(例如:最高温度、最低温度、平均温度)。具体操作,请参见新增原子词。

← 原于	<b>词库管理</b>						
新増	原子词 从产品屬性批量导入 全部指标域 ∨	请搜索原子词名称 Q					全部 已选择
	原子词名称/描述	标识符	主题域	度量类型	单位	已衍生指标数 🚱	操作
	平 <b>均温度</b> 平均温度	avg_temperature	分组	计数	摄氏度	0	编辑 删除
	<b>秦氏温度</b> 最任温度	min_temperature	分组	计数	摄氏度	0	编辑 删除
	<b>最高温度</b> 最高温度	max_temperature	分组	计数	摄氏度	0	編輯 删除

- 3. 为新增的原子词配置对应衍生指标并发布上线。
  - i. 在左侧导航栏,选择**数据分析 > 数据资产**。
  - ii. 在指标管理页签,单击分组视角。
  - iii. 单击已创建的IoT Studio分组名称(例如: studio\_link\_la\_test),进入**分组详情**页面。

⑦ 说明 步骤一:创建项目后,会同步在分组视角中创建loT Studio分组,分组名称即已创建的项目名 称。 物联网平台 / 数据分析 / 数据图 数据资产 指标管理 Topic 管理 域管理 所有指标 产品视角 设备视角 分组视角 空间视角 活动视角 间车管理 指标任务 新建分组 全部未遵 ✓ 清箱入分组名称搜索 Q 全部 日田橋 分组名称/分组/D 来源 创建时间 1100 studio\_link\_la\_test IoT Studio 2021/05/25 15:40:10 žē -

iv. 在分组详情页面, 单击指标列表, 然后单击新建指标。

#### v. 在新建指标页面,配置指标定义参数,如下表和下图所示。

参数	说明
指标域	选择 <b>分组</b> 。
所属分组	选择已创建的IoT Studio分组(例如: studio_link_la_test)。
衍生范围	选择 <b>本实体</b> 。
· 바뀌지지 @ 응 호암바루 · 문성지바루 · · · · · · · · · · · · · · · · · · ·	・計算的点の         19730           ● 系計計算         ● 数式計算           ● (新生活算法)         ● 5179 2. 始経計算規制           ・計算力(面)         ● 5179 2. 始経計算規制           ・計算力(面)         ● 5179 2. 始経計算用           ● 第四計算         ● 第位計算

- vi. 配置完成后,单击完成并上线。
- vii. 您可在分组详情 > 指标列表中查看到新增的指标。

物联网平台 / 数据	分析 / 数据资产 / 分组详情							帮助文档
← studio	division in the second							
分组信息								
分担 ID			分组圈述					
创建时间	2021/05/25 15:40:10		设备数 2			空间数	0	
原始建义指标数	0		衍生定义指标数			衍生指标数	0	
内容管理 空间列表 该 条建后标 詳	操列表 操行列表	٩						250 EBS
□ 指标名称	指标 Code	指标类型	會是类型	数据处型	単位	描述	操作 😡	
< 最近1天平均	22	衍生贈标	计数	浮槽数 (double)	摄氏盘	时间修饰词:最近1	<b>立后:编辑:</b> 下线: 克隆   删除	
量近1天景低	皇宠	衍生瘤标	计数	译·希数 (double)	項氏室	时间修饰词:最近1	查番   编辑   下线   克隆   勤除	
最近1天最高	9x	衍生增标	计数	浮槽数 (double)	揚氏室	时间修饰词:最近1	宣誓 编辑 下线 克隆 删除	

4. 回刷指标。具体操作,请参见指标回刷。

物联网数据分析采用T+1日方式统计衍生指标数据,您可通过指标回刷功能,回刷历史数据,快速获取 衍生指标数据。

## 步骤五:新建自定义服务API

- 1. 在左侧导航栏,选择数据分析 > 数据服务。
- 2. 在API列表,单击自定义服务API,然后单击新建API。

3. 在新建API页面,完成API基本信息的配置,然后单击下一步。

1 API 基本信息
* API 名称
studio
* API Path 🕥
API 标签 ②
请输入标签内容,按回车后生成标签。
studio X
API 描述
请描述新建的 API
返回类型
JSON
下一步(归方)即当

4. 在配置参数并测试页面,完成参数配置。

All and all all all all all all all all all al												
station	数据源											
side	数据源类型											
minu	指标											
ind	* 指标域											
state       state <td< td=""><td>分组</td><td></td><td>~</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></td<>	分组		~									
state          State       State         State </td <td>* 数据集 💿</td> <td></td>	* 数据集 💿											
Aligned Control       Aligned Control	studio		$\sim$									
State State       State	2000 do 804											
Rtail:       Rtai:       Rtai:       Rtail:       Rtail:			10 mm + 100 / 10 + 100 /									
syste distribute         getsy       getsy <t< td=""><td>成功改進() stud</td><td>NO_IINK_IA_test () 10038038</td><td>(11)1111111111111111111111111111111111</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></t<>	成功改進() stud	NO_IINK_IA_test () 10038038	(11)1111111111111111111111111111111111									
flot              flot <th>* 请求参数(</th> <th></th>	* 请求参数(											
	绑定字段	参数名		参数类型	操作符 (	0	必填	示例值	参数语述			
Attic         Attic <th< td=""><td>业务时</td><td>lil V date</td><td>time</td><td>VARCHAR</td><td>BETV</td><td>v ~</td><td></td><td>20210201 ~ 20210228</td><td>请输入参</td><td>ROMER</td><td></td><td>Ô</td></th<>	业务时	lil V date	time	VARCHAR	BETV	v ~		20210201 ~ 20210228	请输入参	ROMER		Ô
State         State <th< td=""><td>+原加参索</td><td>τ</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>共有</td><td>1項</td></th<>	+原加参索	τ									共有	1項
依照         参数名         参数名         参数名         所述         所述         参数名         所述           正         最近 (宋平)、         Genskyls         D008.2         □         面20.30%	• 返回参数 (											
●       ●	优先级	绑定字段	参数名	参数类型	用于排序	示例值				参数描述		
三     近発対用 v     Gdd_time     VdRCHAR     □     20210201     面積人登和版任     首       - 形成学校     形成学校     新生     新生     新生     ### 2.41	=	最近1天平均 ∨	dehna8kyhs	DOUBLE		遺輸入	示例這			清编入参数描述		Ō
• 添加橡軟	=	业务时间 ~	date_time	VARCHAR		20210	201			清编入参数描述		Ô
	+液加酸素	t									共有	2 項

5. 完成配置后,在测试API区域内,输入请求参数的测试值后,单击开始测试。

测试 API						
参数名称 pageNum pageSize date_time	参称然后到 INTEGER INTEGER VARCHAR	提否必構 約 格 約	BRYERG = BET	Jinda 0	(第次)評措 返回元例	
开始制成 《	) 剥试成功 API调 F 发布	用时间: 677 m 取消	5			

6. 单击**发布**。

# 步骤六:在IoT Studio中使用自定义API

1. 返回IoT Studio项目主页,在数据服务页签可以查看到已发布的自定义服务API。

■ studic ■	and a strategy of the strategy					I# #80 🧐
主页	studio 主页					
产品	studio 📠 🖌 📼	and the second second			如何使用数据型实例? 项目	2018年1月1日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日
设备	产品 关联	设备关联	Web应用	移动应用	业务逻辑	
空间	2	2	1	0	0	
账号	项目开发					
	Web应用 移动应用 数据服务					
	你可以通过以下方式创建数据选产:1.通过可识化分析或者 如需进行全量数据选产管理,比如搜索,通进入数据分析创	ISQL分析创建自定义数编服务,前往 可现化分析控制的 300编版产页。	台 或 SQL分析控制台。2.通过添加外部数据源以增加外部	数据表,前往外部数据源,3.前往业务模型数据源。创建可以使	用的业务模型数据表。	
	数据服务					
	自定义数 >> 清编入数据服务名称	Q				C
	名称	與些		更新时间	操作@	
	studio	用户	自定义API	2021/05/25 21:13:29	查看	
					总计:1 < 1 >	每页显示: 10 💙

- 2. 在项目主页页面的项目开发下,选择Web应用。
- 3. 单击应用列表上方的新建。
- 在新建Web应用对话框中,填入应用名称和描述,单击确认。
   创建应用完成后,会自动打开Web应用编辑器。
- 5. 添加表格组件。
  - i. 在Web应用编辑器中,单击最左侧的组件图标 😚。
  - ii. 在画布左侧组件列表上方, 输入表格, 找到该组件, 然后将组件拖拽到中间画布。
  - iii. 选中组件,在右侧样式中单击展示数据后的配置数据源。
  - iv. 在数据源配置面板,选择已创建的自定义API,完成配置。

数据源配置	>	×
选择数据源		Ì
接口	~	
接口来源		
数据分析服务	$\sim$	
*接口	1	
studio_	×	
请求参数		
● 静态参数 🔿 动态参数 ◎		
1 { 2 "date_time": null, 3 "pageNum": null, 4 "pageSize": null 5 }		
▼ 返回结果 Ø		
<pre>1 { 2 "data": [ 3 { 4 "date_time": "20210525", 5 "avg_temperature": 31.98 6 }, 7 { 8 "date_time": "20210523" 9 }, 10 { </pre>		
▶ 叙語过滤剧本 🕜		
刷新设置 🔕		
确定 取消	帮助文档	

- v. 配置数据展示。具体操作,请参见表格。
- vi. 完成配置后,自定义API中的数据即可展示在表格组件中。

date_time	avg_temperature	
20210525	31.98	
20210523		
20210522		

# 12.智能控制空调设备

# 12.1. 概述

本文介绍使用IoT Studio的业务逻辑功能,开发一个控制空调的智能服务。该服务可根据当前温度的变化,控制空调自动开启或关闭。

# 背景信息

业务逻辑是IoT Studio提供的开发工具,支持通过编排服务节点的方式快速完成业务服务开发。

本示例以智能控制空调设备为场景,创建一个智能空调业务服务。

- 当温度检测仪上报当前温度>30℃时,开启空调制冷模式。
- 当温度检测仪上报当前温度<15℃时,开启空调制热模式。
- 当温度检测仪上报当前温度≥15℃,≤30℃时,关闭空调。

### 操作步骤

- 步骤一: 在项目中创建产品
- 步骤二: 创建控制空调的业务服务
- 步骤三: 调试与发布

# 12.2. 步骤一: 在项目中创建产品

本文介绍在IoT Studio创建项目,并为该项目创建产品和设备。

## 步骤一:创建项目

- 1. 登录物联网应用开发控制台,在页面左上角选择公共实例后,在左侧导航栏单击项目管理。
- 2. 在项目管理页面,单击新建项目。

普通项目							
新建项目	搜索		Q				
创建于 2	019/11/11 12:34	:34	0 0	创建于 20	19/11/11 12:34	:34	*
<b>12</b> 设备	5 <sub>Web应用</sub>	<b>6</b> 业务逻辑		<b>12</b> 设备	5 <sub>Web应用</sub>	<b>6</b> 业务逻辑	

- 3. 在新建项目页面,移动鼠标至新建空白项目区域,单击创建空白项目。
- 4. 在新建空白项目对话框, 输入项目名称(例如: 智能空调)和描述信息。
- 5. 单击确认。

完成创建项目后,会直接进入项目详情页面的主页。

主页	智能空调 / 主页	<b>16/2词</b> / 主页						
产品	智能空调						项目配置 发布为解决方案	
设备	产品 关联 0	设备 <del>关联</del> O	Web应用 O	移动应用 0	业务逻辑 0	数据资产 @ 关联 1588	数据任务 @ 0	
账号	项目开发							
	Web应用 移动应用	业务服务 数据资产 数据	19				历史功能 🗸	
	+ 新建 搜索	$\nabla$					G	
	名称	泼	布状态	描述	修改时间 小	操作		

## 步骤二: 创建产品和设备

- 1. 在项目主页, 单击左侧导航栏的产品。
- 2. 在产品页面,单击产品列表左上方的创建产品。

室内温度监控 / 产品				
产品				
关联物联网平台产品	创建产品 请输入	产品名称查询	Q 请选择产品标签	~

 • 在创建产品页面,创建产品名称为温度检测仪,选择自定义品类,其他参数使用默认值,单击确认。

○ 在创建产品页面,创建产品名称为智能空调,选择自定义品类,其他参数使用默认值,单击确定。

- 3. 产品创建成功后,回到产品页面,在产品列表中,单击目标产品对应的查看。
- 在产品详情页面,单击功能定义页签,再单击编辑草稿,选择添加自定义功能。
   本示例添加以下自定义属性。具体操作,请参见单个添加物模型。
  - 温度检测仪产品

功能名称	标识符	数据类型	取值范围	单位	读写类型
当前温度	CurrentTemp erature	int 32(整数 型)	-10~50	摄氏度(℃)	读写
功能类型	功能名称(全部) 🔽	标识符	数据类型	数据定义	操作
廣性	当前温度(自定义)	CurrentTemperature	int32 (整数型)	取值范围: -10 ~ 50	查看

智能空调产品

功能名称	标识符	数据类型	取值范围	单位	读写类型
空调开关	Switch	bool (布尔 型)	<ul><li>0: false</li><li>1: true</li></ul>	不涉及	读写

#### 最佳实践·智能控制空调设备

功能名称	标识符	数据类型	取值范围	单位	读写类型
风速	WindSpeed	enum(枚举 型)	<ul> <li>0:自动</li> <li>1:静音档</li> <li>2:低档</li> <li>3:中档</li> <li>4:高档</li> <li>5:最高档</li> </ul>	不涉及	读写
工作模式	WorkMode	bool (布尔 型)	<ul> <li>0:自动</li> <li>1:制冷</li> <li>2:制热</li> <li>3:通风</li> <li>4:除湿</li> </ul>	不涉及	读写
温度设置	temperature_ setting	double(双精 度浮点型)	18 ~ 30	摄氏度(℃)	读写
					17 A
功能类型	功能名称(全部) 🔽	标识符	数据类型	数据定义	操作
属性	空调开关 自定义	Switch	bool (布尔型)	布尔值: 0 - false	查看

属性	空调开关(自定义)	Switch	bool (布尔型)	布尔值: 0 - false 1 - true	查看
属性	风速(自定义)	WindSpeed	enum (发举型)	牧举语: 0 - 自动 1 - 静音档 2 - 低档 3 - 中档 4 - 高档 5 - 最高档	查音
属性	I作模式 ( <u>自定义</u> )	WorkMode	enum (夜举型)	枚半信: 0 - 自动 1 - 朝冷 2 - 朝於 3 - 通风 4 - 除语	查看
属性	温度设置(自定义)	temperature_setting	double (双精度浮点型)	取值范围: 18 ~ 30	查看

a. 物模型添加完成后, 单击页面左下方的发布上线按钮, 弹出发布物模型上线对话框。

b. 选中确认已查看当前版本与线上版本的比对结果前复选框。

c. 单击确定,发布物模型。

5. 在左侧导航栏,单击**设备**。在**设备**页面,分别为**温度检测仪**产品和**智能空调**产品创建设备,如下图所 示。

#### 具体操作,请参见添加设备。

主页	智能空调 / 设备					
7 <sup>00</sup> 20	设备					
设备	全部产品 >	设备总数	● 激活设备 <sup>©</sup> 2	● 当前在线 <sup>◎</sup> 1		
空间	关联物联网平台设备 添加设备	DeviceName ~ 请将	輸入 DeviceName	Q 请选择设备标	× ×	
账号	DeviceName/备注名称	设备所属产品	节点类型	状态	最后上线时间	操作
	Airconditioner	智能空调	设备	<ul> <li>         在线     </li> </ul>	2021/08/25 17:29:39	查看   解除关联   删除
	Tempdevice	温度检测仪	设备	● 离线	2021/08/25 10:12:36	查看   解除关联   删除
	的种					

- 本案例使用设备模拟器的属性上报数据。温度检测仪产品和智能空调产品的设备必须同时在线。更多 信息,请参见设备模拟器。
  - i. 登录物联网平台控制台。
  - ii. 在**实例概览**页面, 单击公共实例进入实例详情页面。
  - iii. 在左侧导航栏,选择监控运维 > 设备模拟器。
  - iv. 选择本次模拟的设备(例如温度检测器),单击启动设备模拟器。
  - v. 选择属性上报功能, 推送调试指令, 如下图所示。

推送数据后,可在页面右侧设备端日志下查看设备端日志。

i					
1	Nit设备: 温度检测仪 ~ Tempdevice ~	•			
	上行指令调试 下行指令调试		设备端日志查看云端日志		
	自定义 Topic 属性上报 專件上报	在线 💽	尚型 / 时间	内容	
ŧ	快: 默以標块 ~		云平台到设备满息 2021/08/25 17:41:40.575	received topic= s*,"method":"th	("code":200,"data":(),"id":"1629884500500","message":"succes
	前還箴(CurrentTemperature) 10		设备到云平台消息 2021/08/25 17:41:40.500	publish topic=/ 0°,"method":"th	29884500500,"params";("CurrentTemperature":10),"version":"1.
<			云平台到设备满息 2021/08/25 17:41:40.402	received topics s","method":"th	("code")200,"data")(),"id";"1629884500340","message";"succes
			设备到云平台海息 2021/08/25 17:41:40.341	publish topic=/ 0°,"method":ftf	29884500340,"params":("CurrentTemperature":10),"version":"1.
			云平台到设备消息 2021/08/25 17:41:40.266	received topic= s","method":"th	("code":200,"data":(),"id":"1629884500181","message":"succes
			设备到云平台湍息 2021/08/25 17:41:40.181	publish topica/ 0°,"method":ft/	29884500181,"params":("CurrentTemperature":10),"version":"1.
			云平台到设备满息 2021/08/25 17:41:40.139	received topic= s*,"method":*th	("code":200,"data 10,"id11"1629884500039","message":"succes
			云平台到设备消息 2021/08/25 17:41:40.85	received topic= s","method":"thing:even:property.post", version : 1.0-;	("code":200,"data":(),"id":"16298844999000", "message": "succes

有关产品和设备的更多信息,请参见产品和设备。

#### 后续步骤

步骤二: 创建控制空调的业务服务。

# 12.3. 步骤二: 创建控制空调的业务服务

您可在业务逻辑编辑器中拖拽功能节点到画布,配置节点名称、参数等,以可视化的方式开发业务服务。本 文介绍创建控制空调业务逻辑的开发过程。

#### 前提条件

已完成在项目中创建产品。具体操作,请参见步骤一:在项目中创建产品。

#### 步骤一: 创建业务服务

- 1. 登录物联网应用开发控制台,在页面左上角选择公共实例后,在左侧导航栏单击项目管理。
- 2. 在普通项目列表中,找到智能空调项目,单击项目卡片。
- 3. 在项目主页页面的项目开发下,单击业务逻辑页签。
- 4. 在业务逻辑页签, 单击列表上方的新建 > 新建。
- 5. 在新建业务服务对话框, 输入业务服务名称(例如: 智能控制空调)和描述信息。
- 6. 单击确认。

创建业务服务完成后,会自动打开业务逻辑编辑器。各区域功能说明,请参见业务逻辑编辑器。

≡	智能空调-智能控制空调 IoT Studio 业务逻辑开发			
Ð	服务列表	+ *	智能统制空调 ×	欢迎使用业务编排工作台
91 ©	<ul> <li>根理部列会換表</li> <li>              新聞会社会社会      </li> </ul>	Q ::		<ul> <li>○以完一个业务逻辑的开发,包需要进行以下4个参数:</li> <li>● 2015年2015年10月1日</li> <li>● 2015年2015年10月1日</li> <li>● 2015年2015年10月1日</li> <li>● 2015年2015年2015年2015年2015年2015年2015年2015年</li></ul>
				女な作用 単純成成成点由定波を、即可知道会参考編の开発、種紙を可以 出入血控品を測定者利益行作及成が利益行何の开发 如果更多和助、清約往春数本心
			and X	
			© 100% © 🗉 🖃 🗐	

## 步骤二: 配置业务逻辑节点

在业务逻辑开发页面,单击左侧的节点图标 
 在节点列表中,拖拽一个设备触发节点到画布上,并
 进行配置。更多详细配置,请参见设备触发。

智能控制空调 ×		部告诉法	
		* 节点名称 💿 如何使用该节点?	
		(注意文日 の)	
		温度检测仪	/ ×
		* 选择设备 💿	
		Tempdevice	/ ×
🞽 设备触发 📭		* 触发条件 📀 🍵	
		属性 -当前温度	/ ×
		+ 属性上报 + 事件上报	+ 状态变更
配置项	说明		
节点名称	设备触发。		

配置项	说明
选择产品	选择 <b>温度检测仪</b> 产品。
选择设备	选择产品下的指定设备Tempdevice。
触发条件	设置指定的属性上报:当前温度。

2. 在节点列表中,拖拽一个条件判断节点到画布上,与设备触发节点相连,并进行配置。更多详细配置,请参见条件判断。

#### 配置条件:设置当前温度>30。

智能控制空调 ×		
	•	* 节点名称 🔘 如何使用该节点?
		条件判断
		*条件配置 💿
		条件1 · 设备触发当前温度 > 30 /
		+ 条件

3. 在节点列表中,拖拽一个**智能空调**产品节点到画布上,与条件判断的满足条件点连接起来,进行配置。

配置项	描述
节点名称	智能空调。
选择要控制的设备	选择设备Airconditioner。
选择操作类型	选择为 <b>设备动作执行</b> 。
下发数据	本示例中, 满足条件时(当前温度值超过30), 即下 发数据: • 空调开关: true • 风速: 3(中档) • 工作模式: 1(制冷) • 温度设置: 23

*       *					
발생순명            * 전자 방송·文         * 造得要控制的设备 ●           · 造得要控制的设备 ●         Arconditioner           · 造得操作操型 ●         · 造得操作模型 ●					
P 品が能定义     ・ 造得要控制的设备      の     Arconditioner     ・ 造得操作典型     の					
· 遠澤要控制的设备 ● Arconditoner     · 遠澤要控制的设备 ● Arconditoner     · 遠澤操作典型 ●					
Arconditioner ・ 選邦最代典型  ●					
1 話を空神 ・ 法持续代表型 ●	~				
	*选择操作类型 🛞				
	~				
- 下汝奴道 ①					
<ul> <li>         ・         ・         ・</li></ul>					
空視开关 ◎					
東京道 2 東南市点 名並					
わり道	$\sim$				
true	~				
风速 ()					
国治道 余田市点 交量					
致迫型	$\sim$				
3					
工作掲式 ◎					
国会権   大会市点   文学					
数值型	~				
IRACE 0					
◆ 100% ● □ □ □ ■ □ ■ □ ■ □ ■ □ ■ □ ■ □ ■ □ ■ □					

在节点列表中拖拽一个路径选择节点,与条件判断的不满足条件点连接起来。更多详细配置,请参见路径选择。

路径节点设置为**设备触发-当前温度**,包含两个路径:

- 路径1: 当前温度值<15。
- 路径2: 其他(当前温度值≥15)。

智能控制空调 ×	*	* 节点名称 ② 如何便用该节点?	
▲ 12条数发 □	I	訊包选择 • 轮入 ◎ 来自设备就友 • 当前退度 • 話记配置 ◎ 會 路径1 - < 15	, , , , ,
		羅径2 - 其他 → 勝径	^

5. 在节点列表中,拖拽一个智能空调产品节点,与路径选择节点路径1相连,进行配置。

#### 最佳实践·智能控制空调设备

智能投動空洞 ×	* 节点名称 💿 如何使用该节点?	
	智能空调	
	产品功能定义 •选择要控制的设备 💿	
	Airconditioner	~
	*选择操作类型 💿	
	设备动作执行	~
	* 下发数据 💿	
	<ul> <li>属性</li> <li>服务</li> </ul>	
	空调开关 ②	
	固定値 🗸 来目わ点 変量	
	布尔值	$\sim$
	true	~
	风速 💿	
	国定值 🗸 来自节点 变量	
	数值型	$\sim$
	4	
	工作模式 💿	
	固定值 🖌 来自节点 变量	
	数值型	$\sim$
	2	
-	温度设置 💿	
• 100% • [] ·	国定值 🖌 来自节点 变量	

配置项	描述
节点名称	智能空调。
选择要控制的设备	选择设备Airconditioner。
选择操作类型	选择为 <b>设备动作执行</b> 。
下发数据	本示例中,满足条件时(前温度值小于15),即下发数据: • 空调开关:true • 风速:4(高档) • 工作模式:2(制热) • 温度设置:26

6. 在节点列表中,继续拖拽一个**智能空调**产品节点,与路径选择节点路径2相连,进行配置。

智能控制空调 🗙	
	<ul> <li>・ 5月在名称 ● 如何使用度55点?</li> <li>● 転空福</li> <li>● 第二</li> <li< th=""></li<></ul>
配置项	描述
节点名称	智能空调。

选择更控制的设备 进择设备Airconditioner

处开女江即归以田	远往设宙All Conditionel。
选择操作类型	选择为 <b>设备动作执行</b> 。
下发数据	本示例中, 满足条件时(当前温度值大于等于15), 即下发数据: • 空调开关: false • 风速: 0(自动) • 工作模式: 0(自动) • 温度设置: 25

7. 配置完成后,单击页面右上方的保存按钮 🖻 ,保存设置。

## 后续步骤

步骤三:调试与发布。

# 12.4. 步骤三: 调试与发布

本文介绍使用设备模拟器对智能控制空调设备服务进行调试。

# 前提条件

已完成业务逻辑的配置。具体操作,请参见步骤二:创建控制空调的业务服务。

#### 操作步骤

1. 在业务服务编辑页面,单击右上方的部署调试按钮 👧,部署并启动服务。

- 2. 服务部署成功后,单击右上方的部署调试按钮 👧,调试服务。
  - i. 在弹出的对话框中, 单击前往, 前往产品的设备模拟器页面。
  - ii. 单击启动设备模拟器。
  - iii. 选择上行指令调试 > 属性上报。
  - iv. 在默认模块中推送调试指令。

下图示例中,模拟推送的当前温度值是10,小于15,智能空调设备将收到指令。

物联网平台 / 监控运维 / 设备模拟器		
设备模拟器		
IIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII		
上行描令调试下行描令调试	设备端日志查看云端日志	
自定义 Topic 屬性上报 專件上报 在线 🌑	类型 / 时间	内容
標決: 就以機快 >	云平台到设备消息 2021/08/26 17:05:54.223	received topic v (sys/ code" 200 "deta" () "rd"1" 62/966/754165","message" "succes s" ("method" 1"thing e-
当新温读(CurrentTemperature)	设备到云平台消息 2021/08/26 17:05:54.166	publish topic://gu/a 9968754163/"params"\"CurrentTemperature"110;"version"^11. 0"["method"1hing=
	云平台到设备满意 2021/08/26 17:05:54.80	received topics/sys/ s1/method11thing =

在智能空调的设备模拟器页面,设备端日志区域可查看到接收的指令。即业务逻辑编排中需要进行调整的设备参数被设备(模拟器)接收到了,调用和执行成功。

received topic=/sys/al\*\*\*\*\*/Airconditioner/thing/service/property/set, payload={"meth od":"thing.service.property.set","id":"10\*\*\*\*\*\*\*","params":{"Switch":1,"WindSpeed":4, "temperature setting":26,"WorkMode":2},"version":"1.0.0"}

- Switch(空调开关):1(true)
- WindSpeed (风速): 4 (高档)
- temperature\_setting (温度设置): 26 (摄氏度)
- WorkMode(工作模式):2(制热)

设备模拟器		
満試设备: 智能空调 / Airconditioner /		
上行指令调试 下行指令调试	设备端日志 查看云端日志	
自定义 Topic 属性上报 事件上报 在线	業型/时间	內容
標時: 数以機会 ~	云平台到设备消息 2021/08/26 17:08:28.726	received topic=/ss/
空陽开关(Switch) 遠远呼参数 (bool)	云平台到设备消息 2021/08/26 17:08:28.58	and the second state of th
风速(VindSpeed) 请选择参数 (enum)	云平台到设备湍息 2021/08/26 17:08:27.756	
工作模式(WorkMode) 请选择参数 (enum)	云平台到设备消息 2021/08/26 17:08:27.718	
温度设置(temperature_setting)		
诺编入参数 (double)	云半台到设备消息 2021/08/26 17:08:27.676	terite of the state of the stat
	云平台到设备消息 2021/08/26 17:08:27.671	
	云平台到设备消息 2021/08/26 17:08:27.664	to repair choose wells out for
	云平台到设备满意 2021/08/26 17:08:27.641	EPRODUCTION EVENING THE TAXABLE CONTRACTOR

返回业务逻辑编辑页面,查看相关日志。



4. 在业务服务编辑页面,单击右上方的发布按钮,发布服务。

业务服务需要发布到云端,才能被调用。

# 13.使用IoT Studio搭建气象监测屏

# 13.1. 概述

本实践案例中使用LoRa气象监测设备监测气象信息,上报温度、湿度、大气压、经度、纬度等数据,并使用 loT Studio平台搭建监控大屏,展示气象监测设备最新上报的数据和历史数据曲线图。

# 架构图

本案例的架构图如下。



## 方案设计

实现过程:

- 1. 自主搭建气象站的LoRa网络。
- 2. 配置LoRa气象监测设备接入物联网平台。
- 3. 在IoT Studio平台搭建监控大屏。

# 物料准备

购买LoRa网关和LoRa气象监测设备硬件。

购买已通过Link WAN认证的产品(内置Link WAN密钥),可访问广域物联网或阿里云IoT元器件馆。

# 13.2. 配置LoRa网关

使用LoRa设备之前,您需在物联网络管理平台上配置LoRa网关,搭建物联网所需的网络服务。

# 前提条件

已开通物联网络管理平台。

## 操作步骤

- 1. 登录物联网络管理平台控制台
- 2. 在左侧导航栏,选择快速入门。
- 3. 选择搭建与分享网络对应的开始搭建。

= (-)阿里	22	\$\$\$2 (上現) ·	Q, 18.8	费用 工单	<b>BR</b> 22	支持与服务	⊡ ¢	. н	2	简体中文	0
物联网络管理平台	8	快速入门									
快速入门											
仪表盘			~		41				6-1		
网络管理	~	66	-		11			1	5		
节点管理											
认证实检查		搭建与	分享网络	使用	网络			生态	伙伴		
通知		快速搭建自主的物	联网私有小网络,并	把设备接到	阿里云服务上		平台的	硬件合作伙	ff (Alba	ba Cloud	
产品文档		774 162							APT UNICASIN	onj	
推荐硬件			2.06.18	21.8	1904			ALC/9	(X1#		

- 4. 单击**开始体验**。
- 注册网关,填入您的网关基本信息和位置信息后,单击下一步。
   网关的GwEUI、PIN Code和频段信息,请在您网关设备的标签上查看。
  - 如下图所示。

物联网络管理平台	快速入门 > 搭建与分享网络
快速入门	搭建与分享网络 / 第一步: 注册网关(共三步)
<ul> <li>(父表盘)</li> <li>网络管理</li> <li>节点管理</li> <li>认证实验室</li> <li>通知</li> <li>产品文档</li> <li>推荐硬件</li> </ul>	<b>3.</b> 新聞美麗語

6. 将网关通电、连网。

稍等片刻之后,网关状态显示为在线,则表示网关连网上线成功。

主册成功		×
✓ 网关注册 您可以从1	<b>成功!</b> 污方看到所注册的网关连线状态:	
网关名称	RAK_LoRa网关	
在线状态	● 在线 刷新	
注: 您需要将网关	进行通电,并接上网线后,才会检测出在线状态。	

7. 添加入网凭证,单击下一步。

凭证名称:				
好记的凭证名				
\$Q.IQ :				
CN470 罪频	~			
Class:				
A	~			
PocDelay:				
15	~	0		
			2	20
		共道	4.2	**

8. 将凭证授权给自己,单击完成。

搭建与分享网络 / 第三步:分享网络(共三步)		
添加完凭证后,您可将此凭证授权给其他用户来使用所组的 网络 (注:您也可以稍后于入网开通凭证列表上再进行授权)		
凭证名称:好记的凭证名 JoinEUI:d89660eff000000		
<ul> <li>● 授权給自己</li> <li>● 授权給其他用户</li> </ul>		
	-	<b>86</b> 11111
jer <b>ya</b>		

# 执行结果

将凭证授权给自己后,在物联网平台上使用该凭证创建连网方式为LoRaWAN的产品。

# 13.3. 配置LoRa设备接入物联网平台

配置LoRa网关后,您需要在物联网平台上创建LoRa产品和设备,定义物模型,编写、提交LoRa设备的数据 解析脚本。

## 创建产品和设备

- 1. 登录物联网平台控制台。
- 2. 在实例概览页面,找到对应的实例,单击实例进入实例详情页面。

↓ 注意 在中国地域,目前仅华东2(上海)地域开通了公共实例服务。

物联网平台 <b>实例概范</b>	企业版实例 <b>2</b>	>	运行中 <b>2</b>	\$	即将到期 ● 0	\$	已到期 @ O	Ŷ
产品文档 13 增值服务	全部实例							图表: 设备数 消息TPS
*********				升配 续费	iot Het			升配 续费
	● 該行中 ID: id 至同時時1時: 2021/10/02			设新数 0	<ul> <li>● 运行中</li> <li>ID: id</li> <li>至J期时间: 2021/10/01</li> </ul>			设新数 1
	▲ 公共実例 Ⅲ冊 ● これ番 用面目前:20216831			HRANK OBER 8		化版实例 提供更丰富的功能,更好的数3 例 快速入门	服稿者,更高的 SLA 保障。	

- 3. 在左侧导航栏,选择设备管理 > 产品。
- 4. 在**产品**页,单击**创建产品**,创建一个连网方式为LoRaWAN的产品。

参数	说明
产品名称	自定义产品名称。
所属品类	选择为 <b>自定义品类</b> 。
节点类型	选择 <b>直连设备</b> 。
连网方式	选择为LoRaWAN。
入网凭证	选择您在物联网络平台中创建并已授权的入网凭证。
数据格式	选择为 <b>透传/自定义</b> 。
认证方式	选择为 <b>设备密钥</b> 。

- 5. 产品创建成功后,单击**添加设备**栏下的**前往添加**,添加一个设备。 设备的DevEU和PIN Code,请在您的设备标签上查看。
- 6. 测试设备连接物联网平台。

按照设备上的标识,为设备连接天线、GPS天线、电池或电源。



设备上电约2分钟后,在物联网平台控制台对应的实例下的**设备**页的**设备列表**中,该设备的状态会显示为在线。

#### 定义物模型

物模型是将物理空间中的实体进行数字化,并在云端构建该实体的数据模型。在物联网平台中,定义物模型 即定义产品功能(包括属性、事件、服务)。完成功能定义后,系统将自动生成该产品的物模型。本示例 中,气象监测设备上报温度、湿度、气压、地理位置坐标等信息。因此,先在物联网平台上,为这些信息定 义数据模型,即定义对应的属性。

- 1. 在物联网平台控制台对应实例下的左侧导航栏,选择设备管理 > 产品。
- 2. 在产品页,找到之前创建的产品,单击对应的查看。
- 3. 在产品详情页功能定义页签下,选择编辑草稿 > 添加自定义功能,添加以下自定义功能。

属性名	标识符	类型	取值范围	步长	单位	读写类型
温度	T emperat u re	double	-99~100	0.01	°C	读写
湿度	Humidity	double	1~100	0.01	%	读写
大气压	Atmospher e	float	550 ~1060	0.01	hPa	读写
经度	Longitude	double	-180~180	0.01	o	读写
纬度	Latitude	double	-90~90	0.01	o	读写
海拔	Altitude	float	0~9999	0.01	m	读写
X加速度	Acceleratio n_X	float	- 1000~1000	0.01	mg	读写
Y加速度	Acceleratio n_Y	float	- 1000~1000	0.01	mg	读写
Z加速度	Acceleratio n_Z	float	- 1000~1000	0.01	mg	读写
运行速度	Speed	float	-10000 ~10000	0.01	Km/h	读写
电池电压	Battery_vol tage	float	0~100000	0.01	V	读写
气体阻力	Gas_resista nce	float	-10000 ~10000	0.01	无	读写

新增物模型的详细操作说明,请参见单个添加物模型。

4. 单击发布上线将物模型发布为正式版。

#### 编写数据解析脚本

本示例中,LoRa设备上报的数据是二进制格式,如 01880537A5109D5A00846C 。其中1、2字节为数据标 识码 01 88 ; 3、4、5字节为海拔数据 altitude:339m ; 6、7、8字节为纬度数 据 latitude:34.1925 ; 9、10、11字节为经度数据 longitude:108.8858 。 阿里云物联网平台的标准数据格式为Alink JSON格式,不能直接使用二进制数据进行业务处理;并且物联网 平台下发的数据也是Alink JSON格式。您需要根据您的设备数据格式和定义的物模型,编写数据解析脚本, 提交到物联网平台,以供物联网平台调用来解析上下行数据。

- 1. 登录物联网平台控制台,在对应实例的产品详情页,选择数据解析页签。
- 2. 在编辑脚本输入框中, 输入解析脚本。

⑦ 说明 脚本代码中属性的标识符必须与定义物模型时定义的一致。

详细的数据解析脚本编写指导,请参见LoRaWAN设备数据解析。

本示例的数据解析脚本如下:

```
// var COMMAND REPORT = 02;
// var COMMAND SET = 01;
var ALINK_PROP_REPORT_METHOD = 'thing.event.property.post'; //标准ALink JSON格式Topic,设
备上传属性数据到云端。
var ALINK PROP SET METHOD = 'thing.service.property.set';
var ALINK VERSION = "1.1";
function rawDataToProtocol(bytes) {
   var uint8Array = new Uint8Array(bytes.length);
   for (var i = 0; i < bytes.length; i++) {</pre>
       uint8Array[i] = bytes[i] & 0xff;
    }
   var dataView = new DataView(uint8Array.buffer, 0);
   var jsonMap = \{\};
    // var fHead = uint8Array[0]; // 第0个BYTE为上报协议。// if (fHead == COMMAND REPORT)
    {
       jsonMap['method'] = ALINK PROP REPORT METHOD; //ALink JSON格式 - 属性上报。
       jsonMap['version'] = ALINK VERSION; //ALink JSON格式 - 协议版本号固定字段。
       jsonMap['id'] = '' + 12345; //ALink JSON格式 - 标示该次请求id值。
       var params = {};
       switch (dataView.getInt16(0)) {
        case 0x0267:
           params['Temperature'] = Math.floor(dataView.getInt16(2) * 0.1 * 10) / 10;//
保留两位小数。
           params['Humidity'] = Math.floor(100 * dataView.getUint8(6) * 0.01 / 2 * 10)
/ 10;
           params['Atmosphere'] = Math.floor(dataView.getInt16(9) * 0.1 * 10) / 10;
           break:
       case 0x0188:
           var buffer = new Uint8Array(4);
           buffer[0] = 0;
           buffer[1] = uint8Array[2];
           buffer[2] = uint8Array[3];
           buffer[3] = uint8Array[4];
           var latitude = new DataView(buffer.buffer, 0);
           params['Latitude'] = Math.floor(latitude.getInt32(0) * 0.0001 * 10000) / 10
000;
           buffer[0] = 0;
           buffer[1] = uint8Array[5];
           buffer[2] = uint8Array[6];
           buffer[3] = uint8Array[7];
           var longitude = new DataView(buffer.buffer, 0);
```

```
params['Longitude'] = Math.floor(longitude.getInt32(0) * 0.0001 * 10000) /
10000;
           buffer[0] = 0;
           buffer[1] = uint8Array[8];
           buffer[2] = uint8Array[9];
           buffer[3] = uint8Array[10];
           var altitude = new DataView(buffer.buffer, 0);
           params['Altitude'] = Math.floor(altitude.getInt32(0) * 0.01 * 100) / 100;
           break;
        case 0x0371:
           params['Acceleration X'] = dataView.getInt16(2);
           params['Acceleration Y'] = dataView.getInt16(4);
           params['Acceleration_Z'] = dataView.getInt16(6);
           break;
       case 0x0702:
           params['Battery voltage'] = dataView.getInt16(2)/10;
           params['Speed'] = Math.floor(dataView.getInt16(6) * 0.01 * 100) / 100;
           break;
        case 0x0902:
           params['Gas resistance'] = dataView.getInt16(2);
           break;
        }
       jsonMap['params'] = params; //ALink JSON 格式 - params 标准字段 }
       return jsonMap;
   function protocolToRawData(bytes) {
       var method = json['method'];
       var id = json['id'];
       var version = json['version'];
       var payloadArray = [];
       return payloadArray;
   }
}
```

3. 测试脚本。

i. 选择模拟类型为设备上报数据。

ii. 在模拟输入下的输入框中,输入一个模拟数据: 01880537A5109D5A00846C。

ⅲ. 单击执行。

解析结果显示在运行结果栏中。

模拟输入	运行结果				
模拟类型:	设备接收数据				
<pre>{     "method": "thing.event.property.post",     "id": "12345",     "params": {         "Latitude": 34.1925,         "Longitude": 108.8858,         "Altitude": 339     },     "version": "1.1" }</pre>					
提交	▶ 执行				

4. 确认脚本能正确解析数据后,单击提交,将脚本提交到物联网平台系统。

? 说明 物联网平台不能调用草稿状态的脚本,只有已提交的脚本才会被调用来解析数据。

设备上报的属性数据经脚本成功解析后,您可以在该设备的设备详情页物模型数据 > 运行状态页签 下,查看设备上报的属性数据。

# 13.4. 使用IoT Studio开发监控大屏

IoT Studio平台,即物联网开发平台。您可以使用IoT Studio中的Web应用编辑器可搭建监控大屏,用于展 示设备上报的数据。

#### 操作步骤

- 1. 在物联网平台控制台左侧导航栏,选择loT Studio > 项目管理,在普通项目下单击新建项目,新建一 个普通项目,具体操作参见普通项目。 创建成功,自动进入该项目。
- 2. 在项目左侧导航栏,选择产品,再单击关联物联网平台产品,将已创建的气象监测产品与该项目关 联,具体操作参见关联产品至普通项目。
- 3. 在项目左侧导航栏,选择**设备**,再单击关联物联网平台设备,将要监控数据的来源设备与该项目关 联,具体操作参见关联设备至普通项目。
- 4. 在项目左侧导航栏,选择主页,单击Web应用 > 新建,新建一个Web应用,具体操作参见创建Web应 用。
- 5. 在Web应用编辑器中,搭建实时气象数据监控面板。
  - i. 选择自定义新增页,设置页面标题和背景颜色等面板页面显示效果。
  - ii. 在左侧导航栏中,选择 14件,打开组件列表。
  - iii. 从组件列表中,拖拽一个矩形组件到画布上,并配置组件样式,具体操作参见矩形。
  - iv. 从组件列表中,拖拽一个文字组件重叠于矩形组件上,再配置文字组件的数据源为气象监测设备的 温度属性,具体操作参见文字。 设置完成后,该文字组件将显示气象监测设备上报的温度值。

- v. 从组件列表中,拖拽一个文字组件重叠于矩形组件上,文字内容设置为*温度(℃)*,作为温度显示 组件的标题。
- vi. 选中配置好的三个组件,单击鼠标右键,选择成组,将这三个组件组成组件组。



vii. 根据要展示的属性数量,复制多个组。

复制组件组时,各组件的显示效果配置和数据源配置同时被复制。



viii. 对复制的组件组单击鼠标右键,选择解散组。

复制的组件组所有配置均相同。需先解散组,才能重新配置组件数据源等信息。

ix. 分别将数据源设置为该产品的其他属性,并设置对应的属性名称和单位。

x. 如有需要,还可在页面上增加其他组件,如图片组件等,参见基础组件使用说明。
 控制面板效果参考图如下。

实况数据:	(juan	模块	
<b>(</b> )	<b>(</b> )	<b>通</b>	<b>(</b> )
37.5	31	999.8	<b>79</b>
温度(c)	湿症 (%)	大气压 (hPa)	海疫 (m)
(つ)	()	⑦	()
0	0	0	1.12
X地加速度 (mg)	Y轴加速度 (mg)	Z输加速度 (mg)	运行速度 (Km/h)
©	8)	()	⑦
120.0803	30.1321	415	7247
⊮≋ (*)	16.02 (*)	电池电压 (mV)	低限力

xi. 所有组件配置完成后,单击页面上方的 🔿 预览,预览和测试应用页面。

- 在Web应用编辑器中,新建空白页面,配置属性数据曲线展示图。
   以配置温度数据展示曲线图为例。
  - i. 在左侧导航栏,选择 😒 页面,再单击新建符号+,新增空白页面。
  - ii. 在左侧导航栏,选择 3 组件,拖拽一个实时曲线组件到画布上,并配置实时曲线组件的数据源为
     气象监测设备的温度属性,具体操作参见实时曲线(旧)。
  - iii. 配置曲线图的显示样式。

调整曲线图大小、坐标,设置是否显示时间选择器,设置系列名称为温度等。

⑦ 说明 如果选中时间选择器前的复选框,表示曲线图上显示时间选择器。应用发布后,可以设置时间,查看对应时间段的温度数据。

iv. 配置完成后,单击页面上方的 🔿 预览,预览和测试应用页面。

7. 单击页面上方的 《发布,发布应用。

#### 后续步骤

应用发布后,在左侧导航栏中选择。可以开启应用Token验证,为应用绑定您自己的域名等。

更多Web应用可视化开发操作指导,请参见Web可视化开发文档。