



API 参考

# AWS IoT TwinMaker



API 版本 2021-11-29

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# AWS IoT TwinMaker: API 参考

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon 的商标和商业外观不得用于任何非 Amazon 的商品或服务，也不得以任何可能引起客户混淆、贬低或诋毁 Amazon 的方式使用。所有非 Amazon 拥有的其他商标均为各自所有者的财产，这些所有者可能附属于 Amazon、与 Amazon 有关联或由 Amazon 赞助，也可能不是如此。

# Table of Contents

Welcome .....	1
操作 .....	2
BatchPutPropertyValues .....	4
请求语法 .....	4
URI 请求参数 .....	5
请求体 .....	5
响应语法 .....	5
响应元素 .....	7
错误 .....	7
另请参阅 .....	8
CancelMetadataTransferJob .....	9
请求语法 .....	9
URI 请求参数 .....	9
请求体 .....	9
响应语法 .....	9
响应元素 .....	10
错误 .....	11
另请参阅 .....	12
CreateComponentType .....	13
请求语法 .....	13
URI 请求参数 .....	15
请求体 .....	15
响应语法 .....	18
响应元素 .....	18
错误 .....	19
另请参阅 .....	20
CreateEntity .....	21
请求语法 .....	21
URI 请求参数 .....	25
请求体 .....	26
响应语法 .....	28
响应元素 .....	28
错误 .....	29
另请参阅 .....	30

CreateMetadataTransferJob .....	31
请求语法 .....	31
URI 请求参数 .....	32
请求正文 .....	32
响应语法 .....	33
响应元素 .....	33
错误 .....	34
另请参阅 .....	35
CreateScene .....	36
请求语法 .....	36
URI 请求参数 .....	36
请求体 .....	36
响应语法 .....	38
响应元素 .....	39
错误 .....	39
另请参阅 .....	40
CreateSyncJob .....	41
请求语法 .....	41
URI 请求参数 .....	41
请求体 .....	42
响应语法 .....	42
响应元素 .....	43
错误 .....	43
另请参阅 .....	44
CreateWorkspace .....	46
请求语法 .....	46
URI 请求参数 .....	46
请求体 .....	46
响应语法 .....	48
响应元素 .....	48
错误 .....	48
另请参阅 .....	49
DeleteComponentType .....	51
请求语法 .....	51
URI 请求参数 .....	51
请求体 .....	51

响应语法 .....	51
响应元素 .....	52
错误 .....	52
另请参阅 .....	53
DeleteEntity .....	54
请求语法 .....	54
URI 请求参数 .....	54
请求体 .....	54
响应语法 .....	55
响应元素 .....	55
错误 .....	55
另请参阅 .....	56
DeleteScene .....	57
请求语法 .....	57
URI 请求参数 .....	57
请求体 .....	57
响应语法 .....	57
响应元素 .....	58
错误 .....	58
另请参阅 .....	58
DeleteSyncJob .....	60
请求语法 .....	60
URI 请求参数 .....	60
请求体 .....	60
响应语法 .....	60
响应元素 .....	61
错误 .....	61
另请参阅 .....	62
DeleteWorkspace .....	63
请求语法 .....	63
URI 请求参数 .....	63
请求体 .....	63
响应语法 .....	63
响应元素 .....	63
错误 .....	64
另请参阅 .....	65

ExecuteQuery .....	66
请求语法 .....	66
URI 请求参数 .....	66
请求正文 .....	66
响应语法 .....	67
响应元素 .....	68
错误 .....	68
另请参阅 .....	69
GetComponentType .....	71
请求语法 .....	71
URI 请求参数 .....	71
请求体 .....	71
响应语法 .....	71
响应元素 .....	74
错误 .....	77
另请参阅 .....	78
GetEntity .....	79
请求语法 .....	79
URI 请求参数 .....	79
请求体 .....	79
响应语法 .....	79
响应元素 .....	83
错误 .....	86
另请参阅 .....	86
GetMetadataTransferJob .....	88
请求语法 .....	88
URI 请求参数 .....	88
请求体 .....	88
响应语法 .....	88
响应元素 .....	90
错误 .....	92
另请参阅 .....	92
GetPricingPlan .....	94
请求语法 .....	94
URI 请求参数 .....	94
请求体 .....	94

响应语法 .....	94
响应元素 .....	95
错误 .....	95
另请参阅 .....	96
GetPropertyValue .....	97
请求语法 .....	97
URI 请求参数 .....	98
请求体 .....	98
响应语法 .....	101
响应元素 .....	102
错误 .....	103
另请参阅 .....	104
GetPropertyValueHistory .....	105
请求语法 .....	105
URI 请求参数 .....	106
请求体 .....	106
响应语法 .....	110
响应元素 .....	111
错误 .....	111
另请参阅 .....	112
GetScene .....	114
请求语法 .....	114
URI 请求参数 .....	114
请求体 .....	114
响应语法 .....	114
响应元素 .....	115
错误 .....	118
另请参阅 .....	118
GetSyncJob .....	120
请求语法 .....	120
URI 请求参数 .....	120
请求正文 .....	120
响应语法 .....	120
响应元素 .....	121
错误 .....	122
另请参阅 .....	123

GetWorkspace .....	125
请求语法 .....	125
URI 请求参数 .....	125
请求体 .....	125
响应语法 .....	125
响应元素 .....	126
错误 .....	127
另请参阅 .....	128
ListComponents .....	129
请求语法 .....	129
URI 请求参数 .....	129
请求体 .....	129
响应语法 .....	130
响应元素 .....	131
错误 .....	132
另请参阅 .....	132
ListComponentTypes .....	134
请求语法 .....	134
URI 请求参数 .....	134
请求体 .....	134
响应语法 .....	135
响应元素 .....	136
错误 .....	137
另请参阅 .....	137
ListEntities .....	139
请求语法 .....	139
URI 请求参数 .....	139
请求体 .....	139
响应语法 .....	140
响应元素 .....	141
错误 .....	141
另请参阅 .....	142
ListMetadataTransferJobs .....	143
请求语法 .....	143
URI 请求参数 .....	143
请求正文 .....	143

响应语法 .....	144
响应元素 .....	145
错误 .....	146
另请参阅 .....	146
ListProperties .....	148
请求语法 .....	148
URI 请求参数 .....	148
请求体 .....	148
响应语法 .....	150
响应元素 .....	152
错误 .....	152
另请参阅 .....	153
ListScenes .....	154
请求语法 .....	154
URI 请求参数 .....	154
请求体 .....	154
响应语法 .....	155
响应元素 .....	155
错误 .....	156
另请参阅 .....	156
ListSyncJobs .....	158
请求语法 .....	158
URI 请求参数 .....	158
请求体 .....	158
响应语法 .....	159
响应元素 .....	159
错误 .....	160
另请参阅 .....	161
ListSyncResources .....	162
请求语法 .....	162
URI 请求参数 .....	162
请求体 .....	163
响应语法 .....	164
响应元素 .....	164
错误 .....	165
另请参阅 .....	165

ListTagsForResource .....	167
请求语法 .....	167
URI 请求参数 .....	167
请求正文 .....	167
响应语法 .....	168
响应元素 .....	168
错误 .....	169
另请参阅 .....	169
ListWorkspaces .....	171
请求语法 .....	171
URI 请求参数 .....	171
请求正文 .....	171
响应语法 .....	172
响应元素 .....	172
错误 .....	173
另请参阅 .....	173
TagResource .....	175
请求语法 .....	175
URI 请求参数 .....	175
请求正文 .....	175
响应语法 .....	176
响应元素 .....	176
错误 .....	176
另请参阅 .....	177
UntagResource .....	178
请求语法 .....	178
URI 请求参数 .....	178
请求体 .....	178
响应语法 .....	178
响应元素 .....	179
错误 .....	179
另请参阅 .....	179
UpdateComponentType .....	180
请求语法 .....	180
URI 请求参数 .....	182
请求体 .....	182

响应语法 .....	184
响应元素 .....	185
错误 .....	186
另请参阅 .....	187
UpdateEntity .....	188
请求语法 .....	188
URI 请求参数 .....	192
请求体 .....	193
响应语法 .....	194
响应元素 .....	194
错误 .....	195
另请参阅 .....	196
UpdatePricingPlan .....	197
请求语法 .....	197
URI 请求参数 .....	197
请求正文 .....	197
响应语法 .....	198
响应元素 .....	198
错误 .....	199
另请参阅 .....	199
UpdateScene .....	201
请求语法 .....	201
URI 请求参数 .....	201
请求体 .....	202
响应语法 .....	203
响应元素 .....	203
错误 .....	203
另请参阅 .....	204
UpdateWorkspace .....	205
请求语法 .....	205
URI 请求参数 .....	205
请求体 .....	205
响应语法 .....	206
响应元素 .....	206
错误 .....	207
另请参阅 .....	208

数据类型 .....	209
BatchPutPropertyError .....	212
内容 .....	212
另请参阅 .....	212
BatchPutPropertyErrorEntry .....	214
内容 .....	214
另请参阅 .....	214
BundleInformation .....	215
内容 .....	215
另请参阅 .....	215
ColumnDescription .....	216
内容 .....	216
另请参阅 .....	216
ComponentPropertyGroupRequest .....	217
内容 .....	217
另请参阅 .....	217
ComponentPropertyGroupResponse .....	219
内容 .....	219
另请参阅 .....	219
ComponentRequest .....	221
内容 .....	221
另请参阅 .....	222
ComponentResponse .....	223
内容 .....	223
另请参阅 .....	225
ComponentSummary .....	226
内容 .....	226
另请参阅 .....	228
ComponentTypeSummary .....	229
内容 .....	229
另请参阅 .....	230
ComponentUpdateRequest .....	231
内容 .....	231
另请参阅 .....	232
CompositeComponentRequest .....	233
内容 .....	233

另请参阅 .....	234
CompositeComponentTypeRequest .....	235
内容 .....	235
另请参阅 .....	235
CompositeComponentTypeResponse .....	236
内容 .....	236
另请参阅 .....	236
CompositeComponentUpdateRequest .....	237
内容 .....	237
另请参阅 .....	238
DataConnector .....	239
内容 .....	239
另请参阅 .....	239
DataType .....	240
内容 .....	240
另请参阅 .....	241
DataValue .....	242
内容 .....	242
另请参阅 .....	244
DestinationConfiguration .....	245
内容 .....	245
另请参阅 .....	245
EntityPropertyReference .....	246
内容 .....	246
另请参阅 .....	247
EntitySummary .....	248
内容 .....	248
另请参阅 .....	250
ErrorDetails .....	251
内容 .....	251
另请参阅 .....	251
FilterByAsset .....	252
内容 .....	252
另请参阅 .....	253
FilterByAssetModel .....	254
内容 .....	254

另请参阅 .....	255
FilterByComponentType .....	256
内容 .....	256
另请参阅 .....	256
FilterByEntity .....	257
内容 .....	257
另请参阅 .....	257
FunctionRequest .....	258
内容 .....	258
另请参阅 .....	258
FunctionResponse .....	260
内容 .....	260
另请参阅 .....	261
InterpolationParameters .....	262
内容 .....	262
另请参阅 .....	262
lotSiteWiseSourceConfiguration .....	263
内容 .....	263
另请参阅 .....	263
lotSiteWiseSourceConfigurationFilter .....	264
内容 .....	264
另请参阅 .....	264
lotTwinMakerDestinationConfiguration .....	265
内容 .....	265
另请参阅 .....	265
lotTwinMakerSourceConfiguration .....	266
内容 .....	266
另请参阅 .....	266
lotTwinMakerSourceConfigurationFilter .....	267
内容 .....	267
另请参阅 .....	267
LambdaFunction .....	268
内容 .....	268
另请参阅 .....	268
ListComponentTypesFilter .....	269
内容 .....	269

另请参阅 .....	270
ListEntitiesFilter .....	271
内容 .....	271
另请参阅 .....	272
ListMetadataTransferJobsFilter .....	273
内容 .....	273
另请参阅 .....	273
MetadataTransferJobProgress .....	275
内容 .....	275
另请参阅 .....	275
MetadataTransferJobStatus .....	277
内容 .....	277
另请参阅 .....	277
MetadataTransferJobSummary .....	278
内容 .....	278
另请参阅 .....	279
OrderBy .....	280
内容 .....	280
另请参阅 .....	280
ParentEntityUpdateRequest .....	281
内容 .....	281
另请参阅 .....	281
PricingPlan .....	282
内容 .....	282
另请参阅 .....	283
PropertyDefinitionRequest .....	284
内容 .....	284
另请参阅 .....	285
PropertyDefinitionResponse .....	286
内容 .....	286
另请参阅 .....	288
PropertyFilter .....	289
内容 .....	289
另请参阅 .....	289
PropertyGroupRequest .....	291
内容 .....	291

另请参阅 .....	291
PropertyGroupResponse .....	292
内容 .....	292
另请参阅 .....	292
PropertyLatestValue .....	294
内容 .....	294
另请参阅 .....	294
PropertyRequest .....	295
内容 .....	295
另请参阅 .....	295
PropertyResponse .....	296
内容 .....	296
另请参阅 .....	296
PropertySummary .....	297
内容 .....	297
另请参阅 .....	298
PropertyValue .....	299
内容 .....	299
另请参阅 .....	300
PropertyValueEntry .....	301
内容 .....	301
另请参阅 .....	301
PropertyValueHistory .....	302
内容 .....	302
另请参阅 .....	302
Relationship .....	303
内容 .....	303
另请参阅 .....	303
RelationshipValue .....	304
内容 .....	304
另请参阅 .....	304
Row .....	305
内容 .....	305
另请参阅 .....	305
S3DestinationConfiguration .....	306
内容 .....	306

另请参阅 .....	306
S3SourceConfiguration .....	307
内容 .....	307
另请参阅 .....	307
SceneError .....	308
内容 .....	308
另请参阅 .....	308
SceneSummary .....	309
内容 .....	309
另请参阅 .....	310
SourceConfiguration .....	311
内容 .....	311
另请参阅 .....	311
Status .....	313
内容 .....	313
另请参阅 .....	313
SyncJobStatus .....	314
内容 .....	314
另请参阅 .....	314
SyncJobSummary .....	315
内容 .....	315
另请参阅 .....	316
SyncResourceFilter .....	317
内容 .....	317
另请参阅 .....	318
SyncResourceStatus .....	319
内容 .....	319
另请参阅 .....	319
SyncResourceSummary .....	320
内容 .....	320
另请参阅 .....	321
TabularConditions .....	322
内容 .....	322
另请参阅 .....	322
WorkspaceSummary .....	323
内容 .....	323

---

另请参阅 .....	324
常见参数 .....	325
常见错误 .....	328
.....	CCCXXxi

# Welcome

AWS IoT TwinMaker 是一项服务，您可以使用它来构建物理系统的可操作数字双胞胎。AWS IoT TwinMaker 叠加来自真实世界传感器、摄像头和企业应用程序的测量和分析，因此您可以创建数据可视化来监控您的物理工厂、建筑物或工业厂房。您可以使用此真实世界的数​​据来监控运营、进行诊断和纠正错误。

本文档最后一次发布于 2026 年 3 月 23 日。

# 操作

支持以下操作：

- [BatchPutPropertyValues](#)
- [CancelMetadataTransferJob](#)
- [CreateComponentType](#)
- [CreateEntity](#)
- [CreateMetadataTransferJob](#)
- [CreateScene](#)
- [CreateSyncJob](#)
- [CreateWorkspace](#)
- [DeleteComponentType](#)
- [DeleteEntity](#)
- [DeleteScene](#)
- [DeleteSyncJob](#)
- [DeleteWorkspace](#)
- [ExecuteQuery](#)
- [GetComponentType](#)
- [GetEntity](#)
- [GetMetadataTransferJob](#)
- [GetPricingPlan](#)
- [GetPropertyValue](#)
- [GetPropertyValueHistory](#)
- [GetScene](#)
- [GetSyncJob](#)
- [GetWorkspace](#)
- [ListComponents](#)
- [ListComponentTypes](#)
- [ListEntities](#)
- [ListMetadataTransferJobs](#)

- [ListProperties](#)
- [ListScenes](#)
- [ListSyncJobs](#)
- [ListSyncResources](#)
- [ListTagsForResource](#)
- [ListWorkspaces](#)
- [TagResource](#)
- [UntagResource](#)
- [UpdateComponentType](#)
- [UpdateEntity](#)
- [UpdatePricingPlan](#)
- [UpdateScene](#)
- [UpdateWorkspace](#)

# BatchPutPropertyValues

为多个时间序列属性设置值

## 请求语法

```
POST /workspaces/workspaceId/entity-properties HTTP/1.1
Content-type: application/json
```

```
{
  "entries": [
    {
      "entityPropertyReference": {
        "componentName": "string",
        "componentPath": "string",
        "entityId": "string",
        "externalIdProperty": {
          "string": "string"
        },
        "propertyName": "string"
      },
      "propertyValues": [
        {
          "time": "string",
          "timestamp": number,
          "value": {
            "booleanValue": boolean,
            "doubleValue": number,
            "expression": "string",
            "integerValue": number,
            "listValue": [
              "DataValue"
            ],
            "longValue": number,
            "mapValue": {
              "string": "DataValue"
            },
            "relationshipValue": {
              "targetComponentName": "string",
              "targetEntityId": "string"
            },
            "stringValue": "string"
          }
        }
      ]
    }
  ]
}
```

```
    }
  ]
}
]
```

## URI 请求参数

请求使用以下 URI 参数。

### [workspaceId](#)

包含要设置属性的工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### [entries](#)

将字符串映射到要设置的属性值条目的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：[PropertyValueEntry](#) 对象数组

数组成员：最少 1 个物品。最多 10 个物品。

是否必需：是

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "errorEntries": [
    {
```

```

    "errors": [
      {
        "entry": {
          "entityPropertyReference": {
            "componentName": "string",
            "componentPath": "string",
            "entityId": "string",
            "externalIdProperty": {
              "string": "string"
            },
            "propertyName": "string"
          },
          "propertyValues": [
            {
              "time": "string",
              "timestamp": number,
              "value": {
                "booleanValue": boolean,
                "doubleValue": number,
                "expression": "string",
                "integerValue": number,
                "listValue": [
                  "DataValue"
                ],
                "longValue": number,
                "mapValue": {
                  "string": "DataValue"
                },
                "relationshipValue": {
                  "targetComponentName": "string",
                  "targetEntityId": "string"
                },
                "stringValue": "string"
              }
            }
          ],
          "errorCode": "string",
          "errorMessage": "string"
        }
      }
    ]
  }
]

```

```
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [errorEntries](#)

在批量输入操作中导致错误的条目。

类型：[BatchPutPropertyErrorEntry](#) 对象数组

数组成员：最少 1 个物品。最多 10 个物品。

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# CancelMetadataTransferJob

取消元数据传输作业。

## 请求语法

```
PUT /metadata-transfer-jobs/metadataTransferJobId/cancel HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### [metadataTransferJobId](#)

元数据传输作业 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": string,
  "metadataTransferJobId": string,
  "progress": {
    "failedCount": number,
    "skippedCount": number,
    "succeededCount": number,
    "totalCount": number
  },
}
```

```
"status": {
  "error": {
    "code": "string",
    "message": "string"
  },
  "queuedPosition": number,
  "state": "string"
},
"updateDateTime": number
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### arn

元数据传输作业 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9_\-\.\:]+`

### metadataTransferJobId

元数据传输作业 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

### progress

元数据传输作业的进度。

类型：[MetadataTransferJobProgress](#) 对象

## status

元数据传输作业的状态。

类型：[MetadataTransferJobStatus](#) 对象

## updateDateTime

用于更新 DateTime 属性。

类型：时间戳

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### ConflictException

发生冲突。

HTTP 状态代码：409

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

## ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# CreateComponentType

创建组件类型。

## 请求语法

```
POST /workspaces/workspaceId/component-types/componentTypeId HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "componentTypeName": "string",
  "compositeComponentTypes": {
    "string" : {
      "componentTypeId": "string"
    }
  },
  "description": "string",
  "extendsFrom": [ "string" ],
  "functions": {
    "string" : {
      "implementedBy": {
        "isNative": boolean,
        "lambda": {
          "arn": "string"
        }
      }
    },
    "requiredProperties": [ "string" ],
    "scope": "string"
  }
},
"isSingleton": boolean,
"propertyDefinitions": {
  "string" : {
    "configuration": {
      "string" : "string"
    },
    "dataType": {
      "allowedValues": [
        {
          "booleanValue": boolean,
          "doubleValue": number,
          "expression": "string",
          "integerValue": number,
```

```
    "listValue": [
      "DataValue"
    ],
    "longValue": number,
    "mapValue": {
      "string" : "DataValue"
    },
    "relationshipValue": {
      "targetComponentName": "string",
      "targetEntityId": "string"
    },
    "stringValue": "string"
  }
],
"nestedType": "DataType",
"relationship": {
  "relationshipType": "string",
  "targetComponentTypeId": "string"
},
"unitOfMeasure": "string"
},
"defaultValue": {
  "booleanValue": boolean,
  "doubleValue": number,
  "expression": "string",
  "integerValue": number,
  "listValue": [
    "DataValue"
  ],
  "longValue": number,
  "mapValue": {
    "string" : "DataValue"
  },
  "relationshipValue": {
    "targetComponentName": "string",
    "targetEntityId": "string"
  },
  "stringValue": "string"
},
"displayName": "string",
"isExternalId": boolean,
"isRequiredInEntity": boolean,
"isStoredExternally": boolean,
```

```
    "isTimeSeries": boolean
  }
},
"propertyGroups": {
  "string" : {
    "groupType": "string",
    "propertyNames": [ "string" ]
  }
},
"tags": {
  "string" : "string"
}
}
```

## URI 请求参数

请求使用以下 URI 参数。

### componentTypeId

组件类型的 ID。

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

是否必需：是

### workspaceId

包含组件类型的工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

## componentTypeName

组件类型的友好名称。

类型：字符串

长度限制：最小长度为 0。最大长度为 256。

模式：`.*[^\u0000-\u001F\u007F]*.*`

必需：否

## compositeComponentTypes

这是一个将字符串映射到 `componentType` 的 `compositeComponentTypes` 的对象。`CompositeComponentType` 由 `componentTypeId` 引用。

类型：字符串到 [CompositeComponentTypeRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

## description

组件类型的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

必需：否

## extendsFrom

指定要扩展的父组件类型。

类型：字符串数组

长度约束：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：否

### [functions](#)

将字符串映射到组件类型中的函数的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [FunctionRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

### [isSingleton](#)

指定一个实体是否可以有多个此类型的组件的布尔值。

类型：布尔值

必需：否

### [propertyDefinitions](#)

将字符串映射到组件类型中的属性定义的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [PropertyDefinitionRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

### [propertyGroups](#)

类型：字符串到 [PropertyGroupRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

### [tags](#)

可用于管理组件类型的元数据。

类型：字符串到字符串映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 128。

键模式： $([\backslash p\{L}\backslash p\{Z}\backslash p\{N}\_ \cdot : / = + \backslash - @ ]^*)$

值长度限制：最小长度为 1。最大长度为 256。

价值模式： $.^*$

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "creationDateTime": number,
  "state": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [arn](#)

组件类型的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式： $arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\backslash a-zA-Z0-9\_ \- \. : ]^+$

## creationDateTime

创建实体的日期和时间。

类型：时间戳

## state

组件类型的当前状态。

类型：字符串

有效值：CREATING | UPDATING | DELETING | ACTIVE | ERROR

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### ConflictException

发生冲突。

HTTP 状态代码：409

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

### ThrottlingException

速率超过限制。

HTTP 状态代码 : 429

ValidationException

失败

HTTP 状态代码 : 400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# CreateEntity

创建一个实体。

## 请求语法

```
POST /workspaces/workspaceId/entities HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "components": {
    "string": {
      "componentTypeId": "string",
      "description": "string",
      "properties": {
        "string": {
          "definition": {
            "configuration": {
              "string": "string"
            },
            "dataType": {
              "allowedValues": [
                {
                  "booleanValue": boolean,
                  "doubleValue": number,
                  "expression": "string",
                  "integerValue": number,
                  "listValue": [
                    "DataValue"
                  ],
                  "longValue": number,
                  "mapValue": {
                    "string": "DataValue"
                  },
                  "relationshipValue": {
                    "targetComponentName": "string",
                    "targetEntityId": "string"
                  },
                  "stringValue": "string"
                }
              ],
              "nestedType": "DataType",
              "relationship": {
```

```
        "relationshipType": "string",
        "targetComponentTypeId": "string"
    },
    "type": "string",
    "unitOfMeasure": "string"
},
"defaultValue": {
    "booleanValue": boolean,
    "doubleValue": number,
    "expression": "string",
    "integerValue": number,
    "listValue": [
        "DataValue"
    ],
    "longValue": number,
    "mapValue": {
        "string" : "DataValue"
    },
    "relationshipValue": {
        "targetComponentName": "string",
        "targetEntityId": "string"
    },
    "stringValue": "string"
},
"displayname": "string",
"isExternalId": boolean,
"isRequiredInEntity": boolean,
"isStoredExternally": boolean,
"isTimeSeries": boolean
},
"updateType": "string",
"value": {
    "booleanValue": boolean,
    "doubleValue": number,
    "expression": "string",
    "integerValue": number,
    "listValue": [
        "DataValue"
    ],
    "longValue": number,
    "mapValue": {
        "string" : "DataValue"
    },
    "relationshipValue": {
```



```

    }
  ],
  "nestedType": "DataType",
  "relationship": {
    "relationshipType": "string",
    "targetComponentTypeId": "string"
  },
  "type": "string",
  "unitOfMeasure": "string"
},
"defaultValue": {
  "booleanValue": boolean,
  "doubleValue": number,
  "expression": "string",
  "integerValue": number,
  "listValue": [
    "DataValue"
  ],
  "longValue": number,
  "mapValue": {
    "string" : "DataValue"
  },
  "relationshipValue": {
    "targetComponentName": "string",
    "targetEntityId": "string"
  },
  "stringValue": "string"
},
"displayName": "string",
"isExternalId": boolean,
"isRequiredInEntity": boolean,
"isStoredExternally": boolean,
"isTimeSeries": boolean
},
"updateType": "string",
"value": {
  "booleanValue": boolean,
  "doubleValue": number,
  "expression": "string",
  "integerValue": number,
  "listValue": [
    "DataValue"
  ],
  "longValue": number,

```

```
    "mapValue": {
      "string": "DataValue"
    },
    "relationshipValue": {
      "targetComponentName": "string",
      "targetEntityId": "string"
    },
    "stringValue": "string"
  }
},
"propertyGroups": {
  "string": {
    "groupType": "string",
    "propertyNames": [ "string" ],
    "updateType": "string"
  }
}
},
"description": "string",
"entityId": "string",
"entityName": "string",
"parentEntityId": "string",
"tags": {
  "string": "string"
}
}
```

## URI 请求参数

请求使用以下 URI 参数。

### [workspaceId](#)

包含实体的工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### components

将字符串映射到实体中的组件的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [ComponentRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

### compositeComponents

这是一个将字符串映射到请求中 `compositeComponent` 更新的对象。映射的每个键都代表 `compositeComponent` 的 `componentPath`。

类型：字符串到 [CompositeComponentRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 2048。

密钥模式：`[a-zA-Z_\-0-9/]+`

必需：否

### description

实体的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

必需：否

### entityId

实体的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.:]*[a-zA-Z0-9]+`

必需：否

### entityName

实体名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[^\u0000-\u001F\u007F]+`

是否必需：是

### parentEntityId

实体的父实体 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`\\$ROOT|^[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.:]*[a-zA-Z0-9]+`

必需：否

### tags

可用于管理实体的元数据。

类型：字符串到字符串映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 128。

键模式：`([\\p{L}\\p{Z}\\p{N}_.:/=+\\-@]*)`

值长度限制：最小长度为 1。最大长度为 256。

价值模式：.\*

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "creationDateTime": number,
  "entityId": "string",
  "state": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### arn

实体的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\/a-zA-Z0-9\_\-\.\:]+

### creationDateTime

创建实体的日期和时间。

类型：时间戳

### entityId

实体的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.:]*[a-zA-Z0-9]+`

## state

实体的当前状态。

类型：字符串

有效值：CREATING | UPDATING | DELETING | ACTIVE | ERROR

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### ConflictException

发生冲突。

HTTP 状态代码：409

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

### ThrottlingException

速率超过限制。

HTTP 状态代码 : 429

ValidationException

失败

HTTP 状态代码 : 400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# CreateMetadataTransferJob

创建新的元数据传输作业。

## 请求语法

```
POST /metadata-transfer-jobs HTTP/1.1
Content-type: application/json

{
  "description": "string",
  "destination": {
    "iotTwinMakerConfiguration": {
      "workspace": "string"
    },
    "s3Configuration": {
      "location": "string"
    },
    "type": "string"
  },
  "metadataTransferJobId": "string",
  "sources": [
    {
      "iotSiteWiseConfiguration": {
        "filters": [
          { ... }
        ]
      },
      "iotTwinMakerConfiguration": {
        "filters": [
          { ... }
        ],
        "workspace": "string"
      },
      "s3Configuration": {
        "location": "string"
      },
      "type": "string"
    }
  ]
}
```

## URI 请求参数

该请求不使用任何 URI 参数。

## 请求正文

请求接受采用 JSON 格式的以下数据。

### description

元数据传输作业描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：.\*

必需：否

### destination

元数据传输作业目标。

类型：[DestinationConfiguration](#) 对象

是否必需：是

### metadataTransferJobId

元数据传输作业 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：否

### sources

元数据传输作业源。

类型：[SourceConfiguration](#) 对象数组

数组成员：固定数量为 1 项。

是否必需：是

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "creationDateTime": number,
  "metadataTransferJobId": "string",
  "status": {
    "error": {
      "code": "string",
      "message": "string"
    },
    "queuedPosition": number,
    "state": "string"
  }
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [arn](#)

元数据传输作业 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9_-\.\:]+`

### [creationDateTime](#)

元数据传输任务创建 DateTime 属性。

类型：时间戳

### [metadataTransferJobId](#)

元数据传输作业 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

### [status](#)

元数据传输作业响应状态。

类型：[MetadataTransferJobStatus](#) 对象

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### ConflictException

发生冲突。

HTTP 状态代码：409

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

## ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

## ThrottlingException

速率超过限制。

HTTP 状态代码：429

## ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# CreateScene

创建场景

## 请求语法

```
POST /workspaces/workspaceId/scenes HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "capabilities": [ "string" ],
  "contentLocation": "string",
  "description": "string",
  "sceneId": "string",
  "sceneMetadata": {
    "string" : "string"
  },
  "tags": {
    "string" : "string"
  }
}
```

## URI 请求参数

请求使用以下 URI 参数。

### workspaceId

包含场景的工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

## capabilities

场景用于进行自身渲染的功能列表。

类型：字符串数组

数组成员：最少 0 项。最多 50 项。

长度限制：最小长度为 0。最大长度为 256。

模式：.\*

必需：否

## contentLocation

指定内容定义文件位置的相对路径。

类型：字符串

长度限制：最小长度为 0。最大长度为 256。

模式：`[sS]3://[A-Za-z0-9._/-]+`

是否必需：是

## description

此场景的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：.\*

必需：否

## sceneld

场景的 ID

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

是否必需：是

### sceneMetadata

请求元数据。

类型：字符串到字符串映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

值长度限制：最小长度为 0。最大长度为 2048。

价值模式：`.*`

必需：否

### tags

可用于管理场景的元数据。

类型：字符串到字符串映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 128。

键模式：`([\p{L}\p{Z}\p{N}_.:/+\\-@]*)`

值长度限制：最小长度为 1。最大长度为 256。

价值模式：`.*`

必需：否

## 响应语法

HTTP/1.1 200

```
Content-type: application/json

{
  "arn": "string",
  "creationDateTime": number
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### arn

场景的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9\_\-\.:]+

### creationDateTime

创建场景的日期和时间。

类型：时间戳

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### ConflictException

发生冲突。

HTTP 状态代码：409

InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

ThrottlingException

速率超过限制。

HTTP 状态代码：429

ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# CreateSyncJob

此操作会创建 SyncJob。

## 请求语法

```
POST /workspaces/workspaceId/sync-jobs/syncSource HTTP/1.1
Content-type: application/json
```

```
{
  "syncRole": "string",
  "tags": {
    "string" : "string"
  }
}
```

## URI 请求参数

请求使用以下 URI 参数。

### syncSource

同步源。

#### Note

目前唯一支持的 syncSource 是 SITEWISE 。

模式：`[a-zA-Z_0-9]+`

是否必需：是

### workspaceId

工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### [syncRole](#)

I SyncJob AM 角色。使用此 IAM 角色从 SyncSource 中读取数据，以及创建、更新或删除相应的资源。 SyncJob

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iam::[0-9]{12}:role/.*`

是否必需：是

### [tags](#)

SyncJob 标签。

类型：字符串到字符串映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 128。

键模式：`([\p{L}\p{Z}\p{N}_.:/+\\-@]*)`

值长度限制：最小长度为 1。最大长度为 256。

价值模式：`.*`

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "creationDateTime": number,
```

```
"state": "string"  
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### arn

SyncJob ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9\_-\.\:]+

### creationDateTime

SyncJob 创建的日期和时间。

类型：时间戳

### state

SyncJob 响应状态。

类型：字符串

有效值：CREATING | INITIALIZING | ACTIVE | DELETING | ERROR

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

## ConflictException

发生冲突。

HTTP 状态代码：409

## InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

## ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

## ThrottlingException

速率超过限制。

HTTP 状态代码：429

## ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)

- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# CreateWorkspace

创建工作区。

## 请求语法

```
POST /workspaces/workspaceId HTTP/1.1
Content-type: application/json
```

```
{
  "description": "string",
  "role": "string",
  "s3Location": "string",
  "tags": {
    "string" : "string"
  }
}
```

## URI 请求参数

请求使用以下 URI 参数。

### [workspaceId](#)

工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### [description](#)

工作区的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

必需：否

### role

与工作区关联的执行角色的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iam::[0-9]{12}:role/.*`

必需：否

### s3Location

存储与工作空间关联的资源的 Amazon 简单存储服务存储段的 ARN。

类型：字符串

长度限制：最小长度为 0。最大长度为 1024。

模式：`.*(^arn:((aws)|(aws-cn)|(aws-us-gov)):s3::)([a-zA-Z0-9_-]+$).*`

必需：否

### tags

可用于管理工作区的元数据

类型：字符串到字符串映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 128。

键模式：`([\p{L}\p{Z}\p{N}_.:/+\\-@]*)`

值长度限制：最小长度为 1。最大长度为 256。

价值模式：`.*`

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "creationDateTime": number
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### arn

工作区的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9_\-\.\:]+`

### creationDateTime

创建工作区的日期和时间。

类型：时间戳

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

## ConflictException

发生冲突。

HTTP 状态代码：409

## InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

## ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

## ThrottlingException

速率超过限制。

HTTP 状态代码：429

## ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)

- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# DeleteComponentType

删除组件类型。

## 请求语法

```
DELETE /workspaces/workspaceId/component-types/componentTypeId HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### componentTypeId

要删除的组件类型的 ID。

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

是否必需：是

### workspaceId

包含组件类型的工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200  
Content-type: application/json
```

```
{  
  "state": "string"  
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### state

要删除的组件类型的当前状态。

类型：字符串

有效值：CREATING | UPDATING | DELETING | ACTIVE | ERROR

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerError

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ThrottlingException

速率超过限制。

HTTP 状态代码 : 429

ValidationException

失败

HTTP 状态代码 : 400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# DeleteEntity

删除实体。

## 请求语法

```
DELETE /workspaces/workspaceId/entities/entityId?isRecursive=isRecursive HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### entityId

要删除的实体的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.:]*[a-zA-Z0-9]+`

是否必需：是

### isRecursive

指定操作是否删除子实体的布尔值。

### workspaceId

包含要删除实体的工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "state": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### state

已删除实体的当前状态。

类型：字符串

有效值：CREATING | UPDATING | DELETING | ACTIVE | ERROR

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码 : 402

ThrottlingException

速率超过限制。

HTTP 状态代码 : 429

ValidationException

失败

HTTP 状态代码 : 400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# DeleteScene

删除场景

## 请求语法

```
DELETE /workspaces/workspaceId/scenes/sceneId HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### [sceneId](#)

要删除的场景的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

是否必需：是

### [workspaceId](#)

工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
```

## 响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 200 响应。

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# DeleteSyncJob

删除 SyncJob。

## 请求语法

```
DELETE /workspaces/workspaceId/sync-jobs/syncSource HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### syncSource

同步源。

#### Note

目前唯一支持的 syncSource 是 SITEWISE 。

模式：`[a-zA-Z_0-9]+`

是否必需：是

### workspaceId

工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "state": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### state

SyncJob 响应状态。

类型：字符串

有效值：CREATING | INITIALIZING | ACTIVE | DELETING | ERROR

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerError

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码 : 402

ThrottlingException

速率超过限制。

HTTP 状态代码 : 429

ValidationException

失败

HTTP 状态代码 : 400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# DeleteWorkspace

删除工作区

## 请求语法

```
DELETE /workspaces/workspaceId HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### workspaceId

要删除的工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "message": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### message

指定工作区删除结果的字符串。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：.\*

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerError

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ExecuteQuery

运行查询，访问个人工作区中实体知识图谱中的信息。

## Note

该 ExecuteQuery 操作仅适用[AWS 于 Java SDK2](#)。ExecuteQuery 不适用于任何 AWS Java SDK 版本小于 2.x。

## 请求语法

```
POST /queries/execution HTTP/1.1
Content-type: application/json
```

```
{
  "maxResults": number,
  "nextToken": "string",
  "queryStatement": "string",
  "workspaceId": "string"
}
```

## URI 请求参数

该请求不使用任何 URI 参数。

## 请求正文

请求接受采用 JSON 格式的以下数据。

### [maxResults](#)

一次性返回的最大结果数。默认值为 50。

类型：整数

有效范围：最小值为 1。最大值为 100。

必需：否

## nextToken

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

必需：否

## queryStatement

查询语句

类型：字符串

长度限制：最小长度为 1。最大长度为 1000。

模式：[\\s\\S]+

是否必需：是

## workspaceId

工作区的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：[a-zA-Z\_0-9][a-zA-Z\_\\-0-9]\*[a-zA-Z0-9]+

必需：是

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "columnDescriptions": [
    {
```

```
    "name": "string",
    "type": "string"
  }
],
"nextToken": "string",
"rows": [
  {
    "rowData": [ JSON value ]
  }
]
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [columnDescriptions](#)

ColumnDescription 对象列表。

类型：[ColumnDescription](#) 对象数组

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：`.*`

### [rows](#)

代表查询结果中的一行。

类型：[Row](#) 对象数组

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

## AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

## InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

## QueryTimeoutException

查询超时异常。

HTTP 状态代码：400

## ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

## ThrottlingException

速率超过限制。

HTTP 状态代码：429

## ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)

- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# GetComponentType

检索有关组件类型的信息。

## 请求语法

```
GET /workspaces/workspaceId/component-types/componentTypeId HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### componentTypeId

组件类型的 ID。

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

是否必需：是

### workspaceId

包含组件类型的工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
```

```

"componentTypeId": "string",
"componentTypeName": "string",
"compositeComponentTypes": {
  "string" : {
    "componentTypeId": "string",
    "isInherited": boolean
  }
},
"creationDateTime": number,
"description": "string",
"extendsFrom": [ "string" ],
"functions": {
  "string" : {
    "implementedBy": {
      "isNative": boolean,
      "lambda": {
        "arn": "string"
      }
    },
    "isInherited": boolean,
    "requiredProperties": [ "string" ],
    "scope": "string"
  }
},
"isAbstract": boolean,
"isSchemaInitialized": boolean,
"isSingleton": boolean,
"propertyDefinitions": {
  "string" : {
    "configuration": {
      "string" : "string"
    },
    "dataType": {
      "allowedValues": [
        {
          "booleanValue": boolean,
          "doubleValue": number,
          "expression": "string",
          "integerValue": number,
          "listValue": [
            "DataValue"
          ],
          "longValue": number,
          "mapValue": {

```

```

        "string" : "DataValue"
    },
    "relationshipValue": {
        "targetComponentName": "string",
        "targetEntityId": "string"
    },
    "stringValue": "string"
}
],
"nestedType": "DataType",
"relationship": {
    "relationshipType": "string",
    "targetComponentTypeId": "string"
},
"type": "string",
"unitOfMeasure": "string"
},
"defaultValue": {
    "booleanValue": boolean,
    "doubleValue": number,
    "expression": "string",
    "integerValue": number,
    "listValue": [
        "DataValue"
    ],
    "longValue": number,
    "mapValue": {
        "string" : "DataValue"
    },
    "relationshipValue": {
        "targetComponentName": "string",
        "targetEntityId": "string"
    },
    "stringValue": "string"
},
"displayname": "string",
"isExternalId": boolean,
"isFinal": boolean,
"isImported": boolean,
"isInherited": boolean,
"isRequiredInEntity": boolean,
"isStoredExternally": boolean,
"isTimeSeries": boolean
}

```

```
  },
  "propertyGroups": {
    "string": {
      "groupType": "string",
      "isInherited": boolean,
      "propertyNames": [ "string" ]
    }
  },
  "status": {
    "error": {
      "code": "string",
      "message": "string"
    },
    "state": "string"
  },
  "syncSource": "string",
  "updateDateTime": number,
  "workspaceId": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### arn

组件类型的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\/a-zA-Z0-9\_\-\.]+

### componentTypeId

组件类型的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

### [componentTypeName](#)

组件类型名称。

类型：字符串

长度限制：最小长度为 0。最大长度为 256。

模式：`.*[^\u0000-\u001F\u007F]*.*`

### [compositeComponentTypes](#)

这是一个将字符串映射到 `componentType` 的 `compositeComponentTypes` 的对象。`CompositeComponentType` 由 `componentTypeId` 引用。

类型：字符串到 [CompositeComponentTypeResponse](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\.\-0-9]+`

### [creationDateTime](#)

创建组件类型的日期和时间。

类型：时间戳

### [description](#)

组件类型的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

### [extendsFrom](#)

此组件类型扩展的父组件类型的名称。

类型：字符串数组

长度约束：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

## [functions](#)

将字符串映射到组件类型中的函数的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [FunctionResponse](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

## [isAbstract](#)

指定组件类型是否为抽象的布尔值。

类型：布尔值

## [isSchemaInitialized](#)

指定组件类型是否具有架构初始化程序以及架构初始化程序是否已运行的布尔值。

类型：布尔值

## [isSingleton](#)

指定一个实体是否可以有多个此类型的组件的布尔值。

类型：布尔值

## [propertyDefinitions](#)

将字符串映射到组件类型中的属性定义的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [PropertyDefinitionResponse](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

## [propertyGroups](#)

一次性返回的最大结果数。默认值为 25。

有效范围：最小值为 1。最大值为 250。

类型：字符串到 [PropertyGroupResponse](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

### status

组件类型的当前状态。

类型：[Status](#) 对象

### syncSource

的 SyncSource SyncJob，前提是该实体是由创建的。 SyncJob

类型：字符串

模式：`[a-zA-Z_0-9]+`

### updateDateTime

上次更新组件的日期和时间。

类型：时间戳

### workspaceId

包含组件类型的工作区 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerError

出现意外错误。

HTTP 状态代码 : 500

ResourceNotFoundException

未找到资源。

HTTP 状态代码 : 404

ThrottlingException

速率超过限制。

HTTP 状态代码 : 429

ValidationException

失败

HTTP 状态代码 : 400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# GetEntity

检索有关实体的信息。

## 请求语法

```
GET /workspaces/workspaceId/entities/entityId HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### [entityId](#)

实体的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.]*[a-zA-Z0-9]+`

是否必需：是

### [workspaceId](#)

工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
```

```

"areAllComponentsReturned": boolean,
"arn": "string",
"components": {
  "string" : {
    "areAllCompositeComponentsReturned": boolean,
    "areAllPropertiesReturned": boolean,
    "componentName": "string",
    "componentTypeId": "string",
    "compositeComponents": {
      "string" : {
        "componentName": "string",
        "componentPath": "string",
        "componentTypeId": "string",
        "definedIn": "string",
        "description": "string",
        "propertyGroups": {
          "string" : {
            "groupType": "string",
            "isInherited": boolean,
            "propertyNames": [ "string" ]
          }
        },
        "status": {
          "error": {
            "code": "string",
            "message": "string"
          },
          "state": "string"
        },
        "syncSource": "string"
      }
    },
    "definedIn": "string",
    "description": "string",
    "properties": {
      "string" : {
        "areAllPropertyValuesReturned": boolean,
        "definition": {
          "configuration": {
            "string" : "string"
          },
          "dataType": {
            "allowedValues": [
              {

```

```

        "booleanValue": boolean,
        "doubleValue": number,
        "expression": "string",
        "integerValue": number,
        "listValue": [
            "DataValue"
        ],
        "longValue": number,
        "mapValue": {
            "string" : "DataValue"
        },
        "relationshipValue": {
            "targetComponentName": "string",
            "targetEntityId": "string"
        },
        "stringValue": "string"
    }
],
"nestedType": "DataType",
"relationship": {
    "relationshipType": "string",
    "targetComponentTypeId": "string"
},
"type": "string",
"unitOfMeasure": "string"
},
"defaultValue": {
    "booleanValue": boolean,
    "doubleValue": number,
    "expression": "string",
    "integerValue": number,
    "listValue": [
        "DataValue"
    ],
    "longValue": number,
    "mapValue": {
        "string" : "DataValue"
    },
    "relationshipValue": {
        "targetComponentName": "string",
        "targetEntityId": "string"
    },
    "stringValue": "string"
},
},

```

```

        "displayName": "string",
        "isExternalId": boolean,
        "isFinal": boolean,
        "isImported": boolean,
        "isInherited": boolean,
        "isRequiredInEntity": boolean,
        "isStoredExternally": boolean,
        "isTimeSeries": boolean
    },
    "value": {
        "booleanValue": boolean,
        "doubleValue": number,
        "expression": "string",
        "integerValue": number,
        "listValue": [
            "DataValue"
        ],
        "longValue": number,
        "mapValue": {
            "string" : "DataValue"
        },
        "relationshipValue": {
            "targetComponentName": "string",
            "targetEntityId": "string"
        },
        "stringValue": "string"
    }
}
},
"propertyGroups": {
    "string" : {
        "groupType": "string",
        "isInherited": boolean,
        "propertyNames": [ "string" ]
    }
},
"status": {
    "error": {
        "code": "string",
        "message": "string"
    },
    "state": "string"
},
"syncSource": "string"

```

```
    }
  },
  "creationDateTime": number,
  "description": string,
  "entityId": string,
  "entityName": string,
  "hasChildEntities": boolean,
  "parentEntityId": string,
  "status": {
    "error": {
      "code": string,
      "message": string
    },
    "state": string
  },
  "syncSource": string,
  "updateDateTime": number,
  "workspaceId": string
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### areAllComponentsReturned

此标记注明 API 响应中是否返回了所有组件。返回的最大组件数为 30。

类型：布尔值

### arn

实体的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9\_\-\.:]+

## components

将字符串映射到实体中的组件的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [ComponentResponse](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

## creationDateTime

创建实体的日期和时间。

类型：时间戳

## description

实体的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

## entityId

实体的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.:]*[a-zA-Z0-9]+`

## entityName

实体名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[^\u0000-\u001F\u007F]+`

## [hasChildEntities](#)

指定实体是否关联了子实体的布尔值。

类型：布尔值

## [parentEntityId](#)

该实体的父实体的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`\\$ROOT|^[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\\-0-9\\.]*[a-zA-Z0-9]+`

## [status](#)

实体的当前状态。

类型：[Status](#) 对象

## [syncSource](#)

同步作业的 syncSource ( 如果该实体是由同步作业创建的 )。

类型：字符串

模式：`[a-zA-Z_0-9]+`

## [updateDateTime](#)

上次更新实体的日期和时间。

类型：时间戳

## [workspaceId](#)

工作区的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\\-0-9]*[a-zA-Z0-9]+`

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)

- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# GetMetadataTransferJob

检索有关元数据传输任务的信息。

## 请求语法

```
GET /metadata-transfer-jobs/metadataTransferJobId HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### [metadataTransferJobId](#)

元数据传输作业 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "creationDateTime": number,
  "description": "string",
  "destination": {
    "iotTwinMakerConfiguration": {
      "workspace": "string"
    },
    "s3Configuration": {
      "location": "string"
    }
  }
}
```

```
    "type": "string"
  },
  "metadataTransferJobId": "string",
  "metadataTransferJobRole": "string",
  "progress": {
    "failedCount": number,
    "skippedCount": number,
    "succeededCount": number,
    "totalCount": number
  },
  "reportUrl": "string",
  "sources": [
    {
      "iotSiteWiseConfiguration": {
        "filters": [
          { ... }
        ]
      },
      "iotTwinMakerConfiguration": {
        "filters": [
          { ... }
        ],
        "workspace": "string"
      },
      "s3Configuration": {
        "location": "string"
      },
      "type": "string"
    }
  ],
  "status": {
    "error": {
      "code": "string",
      "message": "string"
    },
    "queuedPosition": number,
    "state": "string"
  },
  "updateDateTime": number
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [arn](#)

元数据传输作业 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9_-\.\:]+`

### [creationDateTime](#)

元数据传输任务的创建 DateTime 属性。

类型：时间戳

### [description](#)

元数据传输作业描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

### [destination](#)

元数据传输作业的目标。

类型：[DestinationConfiguration](#) 对象

### [metadataTransferJobId](#)

元数据传输作业 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

### [metadataTransferJobRole](#)

元数据传输作业的角色。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iam::[0-9]{12}:role/.*`

### [progress](#)

元数据传输作业的进度。

类型：[MetadataTransferJobProgress](#) 对象

### [reportUrl](#)

元数据传输作业的报告 URL。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`.*`

### [sources](#)

元数据传输作业的来源。

类型：[SourceConfiguration](#) 对象数组

数组成员：固定数量为 1 项。

### [status](#)

元数据传输作业的状态。

类型：[MetadataTransferJobStatus](#) 对象

### [updateDateTime](#)

元数据传输任务的更新 DateTime 属性。

类型：时间戳

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)

- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# GetPricingPlan

获取定价计划。

## 请求语法

```
GET /pricingplan HTTP/1.1
```

## URI 请求参数

该请求不使用任何 URI 参数。

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "currentPricingPlan": {
    "billableEntityCount": number,
    "bundleInformation": {
      "bundleNames": [ "string" ],
      "pricingTier": "string"
    },
    "effectiveDateTime": number,
    "pricingMode": "string",
    "updateDateTime": number,
    "updateReason": "string"
  },
  "pendingPricingPlan": {
    "billableEntityCount": number,
    "bundleInformation": {
      "bundleNames": [ "string" ],
      "pricingTier": "string"
    },
    "effectiveDateTime": number,
    "pricingMode": "string",
```

```
    "updateDateTime": number,  
    "updateReason": "string"  
  }  
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [currentPricingPlan](#)

为当前计费周期选择的定价计划。

类型：[PricingPlan](#) 对象

### [pendingPricingPlan](#)

待定定价计划。

类型：[PricingPlan](#) 对象

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerError

出现意外错误。

HTTP 状态代码：500

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

## ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# GetPropertyValue

获取组件、组件类型、实体或工作区的属性值。

您必须指定 `componentName`、`componentTypeId`、`entityId` 或 `workspaceId` 的值，但不能同时指定两者。

## 请求语法

```
POST /workspaces/workspaceId/entity-properties/value HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "componentName": "string",
  "componentPath": "string",
  "componentTypeId": "string",
  "entityId": "string",
  "maxResults": number,
  "nextToken": "string",
  "propertyGroupName": "string",
  "selectedProperties": [ "string" ],
  "tabularConditions": {
    "orderBy": [
      {
        "order": "string",
        "propertyName": "string"
      }
    ],
    "propertyFilters": [
      {
        "operator": "string",
        "propertyName": "string",
        "value": {
          "booleanValue": boolean,
          "doubleValue": number,
          "expression": "string",
          "integerValue": number,
          "listValue": [
            "DataValue"
          ],
          "longValue": number,
          "mapValue": {
            "string" : "DataValue"
          }
        }
      }
    ]
  }
}
```

```
    },
    "relationshipValue": {
      "targetComponentName": "string",
      "targetEntityId": "string"
    },
    "stringValue": "string"
  }
}
]
```

## URI 请求参数

请求使用以下 URI 参数。

### [workspaceId](#)

操作返回其值的工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### [componentName](#)

操作返回其属性值的组件名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：否

## componentPath

此字符串指定从顶层组件开始的复合组件路径。

类型：字符串

长度限制：最小长度为 1。最大长度为 2048。

模式：`[a-zA-Z_\-0-9/]+`

必需：否

## componentTypeId

操作返回其属性值的组件类型的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：否

## entityId

操作返回其属性值的实体的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.:]*[a-zA-Z0-9]+`

必需：否

## maxResults

一次性返回的最大结果数。默认值为 25。

有效范围：最小值为 1。最大值为 250。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

### nextToken

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

必需：否

### propertyGroupName

属性组名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：[a-zA-Z\_\-0-9]+

必需：否

### selectedProperties

操作返回其值的属性。

类型：字符串数组

数组成员：最少 1 个物品。最多 10 项。

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

是否必需：是

### tabularConditions

表格条件。

类型：[TabularConditions](#) 对象

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "propertyValues": {
    "string" : {
      "propertyReference": {
        "componentName": "string",
        "componentPath": "string",
        "entityId": "string",
        "externalIdProperty": {
          "string" : "string"
        },
        "propertyName": "string"
      },
      "propertyValue": {
        "booleanValue": boolean,
        "doubleValue": number,
        "expression": "string",
        "integerValue": number,
        "listValue": [
          "DataValue"
        ],
        "longValue": number,
        "mapValue": {
          "string" : "DataValue"
        },
        "relationshipValue": {
          "targetComponentName": "string",
          "targetEntityId": "string"
        },
        "stringValue": "string"
      }
    }
  },
  "tabularPropertyValues": [
```

```
[
  {
    "string" : {
      "booleanValue": boolean,
      "doubleValue": number,
      "expression": "string",
      "integerValue": number,
      "listValue": [
        "DataValue"
      ],
      "longValue": number,
      "mapValue": {
        "string" : "DataValue"
      },
      "relationshipValue": {
        "targetComponentName": "string",
        "targetEntityId": "string"
      },
      "stringValue": "string"
    }
  }
]
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

### [propertyValues](#)

将字符串映射到响应中的属性和最新属性值的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [PropertyLatestValue](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

### [tabularPropertyValues](#)

属性值表。

类型：从字符串到 [DataValue](#) 对象映射的由数组组成的数组。

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### ConnectorFailureException

连接器发生故障。

HTTP 状态代码：424

### ConnectorTimeoutException

连接器超时。

HTTP 状态代码：424

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码 : 404

ThrottlingException

速率超过限制。

HTTP 状态代码 : 429

ValidationException

失败

HTTP 状态代码 : 400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# GetPropertyValueHistory

检索有关组件、组件类型、实体或工作区的时间序列属性值历史记录的信息。

您必须为 `workspaceId` 指定一个值。对于特定于实体的查询，请为 `componentName` 和 `entityId` 指定值。对于跨实体查询，请为指定一个值。 `componentTypeId`

## 请求语法

```
POST /workspaces/workspaceId/entity-properties/history HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "componentName": "string",
  "componentPath": "string",
  "componentTypeId": "string",
  "endDateTime": number,
  "endTime": "string",
  "entityId": "string",
  "interpolation": {
    "interpolationType": "string",
    "intervalInSeconds": number
  },
  "maxResults": number,
  "nextToken": "string",
  "orderByTime": "string",
  "propertyFilters": [
    {
      "operator": "string",
      "propertyName": "string",
      "value": {
        "booleanValue": boolean,
        "doubleValue": number,
        "expression": "string",
        "integerValue": number,
        "listValue": [
          "DataValue"
        ],
        "longValue": number,
        "mapValue": {
          "string": "DataValue"
        },
        "relationshipValue": {
```

```
        "targetComponentName": "string",
        "targetEntityId": "string"
    },
    "stringValue": "string"
}
],
"selectedProperties": [ "string" ],
"startTime": number,
"startDateTime": "string"
}
```

## URI 请求参数

请求使用以下 URI 参数。

### workspaceId

工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### componentName

组件名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：否

## componentPath

此字符串指定从顶层组件开始的复合组件路径。

类型：字符串

长度限制：最小长度为 1。最大长度为 2048。

模式：`[a-zA-Z_\-0-9/]+`

必需：否

## componentTypeId

组件类型的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：否

## endDateTime

该参数已弃用。

要返回的最新属性值的日期和时间。

类型：时间戳

必需：否

## endTime

要 ISO8601 DateTime 返回的最新属性值中的。

有关 ISO8601 DateTime 格式的更多信息，请参阅数据类型[PropertyValue](#)。

类型：字符串

长度约束：最小长度为 20。最大长度为 35。

必需：否

## entityId

实体的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.]*[a-zA-Z0-9]+`

必需：否

## interpolation

指定插值类型和插值数据间隔的对象。

类型：[InterpolationParameters](#) 对象

必需：否

## maxResults

一次性返回的最大结果数。默认值为 25。

有效范围：最小值为 1。最大值为 250。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

## nextToken

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：`.*`

必需：否

## [orderByTime](#)

在结果顺序中使用的时间方向。

类型：字符串

有效值：ASCENDING | DESCENDING

必需：否

## [propertyFilters](#)

筛选属性值历史请求的对象列表。

类型：[PropertyFilter](#) 对象数组

数组成员：最少 1 个物品。最多 10 个物品。

必需：否

## [selectedProperties](#)

请求检索其值历史记录的属性列表。

类型：字符串数组

数组成员：最少 1 个物品。最多 10 项。

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

是否必需：是

## [startDateTime](#)

该参数已弃用。

要返回的最早的属性值的日期和时间。

类型：时间戳

必需：否

## [startTime](#)

最早要返回 ISO8601 DateTime 的属性值中的一个。

有关 ISO8601 DateTime 格式的更多信息，请参阅数据类型[PropertyValue](#)。

类型：字符串

长度约束：最小长度为 20。最大长度为 35。

必需：否

## 响应语法

HTTP/1.1 200

Content-type: application/json

```
{
  "nextToken": "string",
  "propertyValues": [
    {
      "entityPropertyReference": {
        "componentName": "string",
        "componentPath": "string",
        "entityId": "string",
        "externalIdProperty": {
          "string": "string"
        },
        "propertyName": "string"
      },
      "values": [
        {
          "time": "string",
          "timestamp": number,
          "value": {
            "booleanValue": boolean,
            "doubleValue": number,
            "expression": "string",
            "integerValue": number,
            "listValue": [
              "DataValue"
            ],
            "longValue": number,
            "mapValue": {
              "string": "DataValue"
            },
            "relationshipValue": {
```

```
        "targetComponentName": "string",
        "targetEntityId": "string"
    },
    "stringValue": "string"
}
]
}
]
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

### [propertyValues](#)

将字符串映射到组件类型中的属性定义的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：[PropertyValueHistory](#) 对象数组

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

## ConnectorFailureException

连接器发生故障。

HTTP 状态代码：424

## ConnectorTimeoutException

连接器超时。

HTTP 状态代码：424

## InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

## ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

## ThrottlingException

速率超过限制。

HTTP 状态代码：429

## ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)

- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# GetScene

检索有关场景的信息。

## 请求语法

```
GET /workspaces/workspaceId/scenes/sceneId HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### sceneId

场景的 ID

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

是否必需：是

### workspaceId

包含场景的工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200  
Content-type: application/json
```

```
{
  "arn": "string",
  "capabilities": [ "string" ],
  "contentLocation": "string",
  "creationDateTime": number,
  "description": "string",
  "error": {
    "code": "string",
    "message": "string"
  },
  "generatedSceneMetadata": {
    "string" : "string"
  },
  "sceneId": "string",
  "sceneMetadata": {
    "string" : "string"
  },
  "updateDateTime": number,
  "workspaceId": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### arn

场景的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9\_-\.\:]+

### capabilities

场景用于渲染的功能列表。

类型：字符串数组

数组成员：最少 0 项。最多 50 项。

长度限制：最小长度为 0。最大长度为 256。

模式：`.*`

### contentLocation

指定内容定义文件位置的相对路径。

类型：字符串

长度限制：最小长度为 0。最大长度为 256。

模式：`[sS]3://[A-Za-z0-9._/-]+`

### creationDateTime

创建场景的日期和时间。

类型：时间戳

### description

场景的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

### error

SceneResponse 错误。

类型：[SceneError](#) 对象

### generatedSceneMetadata

生成的场景元数据。

类型：字符串到字符串映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

值长度限制：最小长度为 0。最大长度为 2048。

价值模式：`.*`

### sceneld

场景的 ID

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

### sceneMetadata

响应元数据。

类型：字符串到字符串映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

值长度限制：最小长度为 0。最大长度为 2048。

价值模式：`.*`

### updateDateTime

上次更新场景的日期和时间。

类型：时间戳

### workspaceId

包含场景的工作区 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)

- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# GetSyncJob

获取 SyncJob。

## 请求语法

```
GET /sync-jobs/syncSource?workspace=workspaceId HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### syncSource

同步源。

#### Note

目前唯一支持的 syncSource 是 SITEWISE 。

模式：`[a-zA-Z_0-9]+`

是否必需：是

### workspaceId

工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

## 请求正文

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
```

Content-type: application/json

```
{
  "arn": "string",
  "creationDateTime": number,
  "status": {
    "error": {
      "code": "string",
      "message": "string"
    },
    "state": "string"
  },
  "syncRole": "string",
  "syncSource": "string",
  "updateDateTime": number,
  "workspaceId": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### arn

同步工作 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9\_-\.\:]+

### creationDateTime

创建日期和时间。

类型：时间戳

### status

SyncJob 响应状态。

类型：[SyncJobStatus](#) 对象

### [syncRole](#)

同步 IAM 角色。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iam::[0-9]{12}:role/.*`

### [syncSource](#)

同步源。

#### Note

目前唯一支持的 syncSource 是 SITEWISE。

类型：字符串

模式：`[a-zA-Z_0-9]+`

### [updateDateTime](#)

更新日期和时间。

类型：时间戳

### [workspaceId](#)

包含同步任务的工作区 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

## AccessDeniedException

访问被拒绝。

HTTP 状态代码 : 403

## InternalServerErrorException

出现意外错误。

HTTP 状态代码 : 500

## ResourceNotFoundException

未找到资源。

HTTP 状态代码 : 404

## ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码 : 402

## ThrottlingException

速率超过限制。

HTTP 状态代码 : 429

## ValidationException

失败

HTTP 状态代码 : 400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)

- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# GetWorkspace

检索有关工作区的信息。

## 请求语法

```
GET /workspaces/workspaceId HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### [workspaceId](#)

工作区的 ID。

长度限制：最小长度为 1。最大长度为 2048。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+|^arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9_-]+`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "creationDateTime": number,
  "description": "string",
  "linkedServices": [ "string ],
  "role": "string",
  "s3Location": "string",
```

```
"updateDateTime": number,  
"workspaceId": "string"  
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### arn

工作区的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9_-\.\:]+`

### creationDateTime

创建工作区的日期和时间。

类型：时间戳

### description

工作区的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

### linkedServices

链接到工作区的服务列表。

类型：字符串数组

模式：`[a-zA-Z_0-9]+`

## role

与工作区关联的执行角色的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iam::[0-9]{12}:role/.*`

## s3Location

存储与工作空间关联的资源的 Amazon S3 存储桶的 ARN。

类型：字符串

长度限制：最小长度为 0。最大长度为 1024。

模式：`.*(^arn:((aws)|(aws-cn)|(aws-us-gov)):s3::)([a-zA-Z0-9_-]+)$.*`

## updateDateTime

上次更新工作区的日期和时间。

类型：时间戳

## workspaceId

工作区的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### InternalServerError

出现意外错误。

HTTP 状态代码：500

## ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

## ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

## ThrottlingException

速率超过限制。

HTTP 状态代码：429

## ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListComponents

此 API 列出了实体的组件。

## 请求语法

```
POST /workspaces/workspaceId/entities/entityId/components-list HTTP/1.1
Content-type: application/json
```

```
{
  "componentPath": "string",
  "maxResults": number,
  "nextToken": "string"
}
```

## URI 请求参数

请求使用以下 URI 参数。

### entityId

操作返回其元数据（组件/属性）的实体的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.]*[a-zA-Z0-9]+`

是否必需：是

### workspaceId

工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

## componentPath

此字符串指定从顶层组件开始的复合组件路径。

类型：字符串

长度限制：最小长度为 1。最大长度为 2048。

模式：[a-zA-Z\_\-0-9/]+

必需：否

## maxResults

一次性返回的最大结果数。默认值为 25。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

## nextToken

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "componentSummaries": [
    {
      "componentName": "string",
      "componentPath": "string",
```

```
    "componentTypeId": "string",
    "definedIn": "string",
    "description": "string",
    "propertyGroups": {
      "string" : {
        "groupType": "string",
        "isInherited": boolean,
        "propertyNames": [ "string" ]
      }
    },
    "status": {
      "error": {
        "code": "string",
        "message": "string"
      },
      "state": "string"
    },
    "syncSource": "string"
  }
],
"nextToken": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### componentSummaries

包含有关组件信息的对象列表。

类型：[ComponentSummary](#) 对象数组

### nextToken

指定下一页组件结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：`.*`

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)

- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListComponentTypes

列出工作区中的所有组件类型。

## 请求语法

```
POST /workspaces/workspaceId/component-types-list HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    { ... }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

## URI 请求参数

请求使用以下 URI 参数。

### [workspaceId](#)

工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### [filters](#)

筛选请求的对象列表。

类型：[ListComponentTypesFilter](#) 对象数组

必需：否

### [maxResults](#)

一次性返回的最大结果数。默认值为 25。

有效范围：最小值为 1。最大值为 250。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "componentTypeSummaries": [
    {
      "arn": "string",
      "componentTypeId": "string",
      "componentTypeName": "string",
      "creationDateTime": number,
      "description": "string",
      "status": {
        "error": {
          "code": "string",
          "message": "string"
        }
      },
    }
  ],
}
```

```
        "state": "string"
      },
      "updateDateTime": number
    }
  ],
  "maxResults": number,
  "nextToken": "string",
  "workspaceId": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [componentTypeSummaries](#)

包含有关组件类型信息的对象列表。

类型：[ComponentTypeSummary](#) 对象数组

### [maxResults](#)

指定最大结果显示数。

类型：整数

有效范围：最小值为 0。最大值为 200。

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

### [workspaceId](#)

工作区的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)

- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListEntities

列出工作区中的所有实体。

## 请求语法

```
POST /workspaces/workspaceId/entities-list HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    { ... }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

## URI 请求参数

请求使用以下 URI 参数。

### [workspaceId](#)

工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### [filters](#)

筛选请求的对象列表。

**Note**

有效输入仅接受一个对象。

类型：[ListEntitiesFilter](#) 对象数组

必需：否

**maxResults**

一次性返回的最大结果数。默认值为 25。

有效范围：最小值为 1。最大值为 250。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

**nextToken**

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：`.*`

必需：否

**响应语法**

```
HTTP/1.1 200
Content-type: application/json

{
  "entitySummaries": [
    {
      "arn": "string",
      "creationDateTime": number,
```

```
    "description": "string",
    "entityId": "string",
    "entityName": "string",
    "hasChildEntities": boolean,
    "parentEntityId": "string",
    "status": {
      "error": {
        "code": "string",
        "message": "string"
      },
      "state": "string"
    },
    "updateDateTime": number
  }
],
"nextToken": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### entitySummaries

包含有关实体信息的对象列表。

类型：[EntitySummary](#) 对象数组

### nextToken

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：`.*`

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

## InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

## ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

## ThrottlingException

速率超过限制。

HTTP 状态代码：429

## ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListMetadataTransferJobs

列出元数据传输作业。

## 请求语法

```
POST /metadata-transfer-jobs-list HTTP/1.1
Content-type: application/json

{
  "destinationType": "string",
  "filters": [
    { ... }
  ],
  "maxResults": number,
  "nextToken": "string",
  "sourceType": "string"
}
```

## URI 请求参数

该请求不使用任何 URI 参数。

## 请求正文

请求接受采用 JSON 格式的以下数据。

### [destinationType](#)

元数据传输作业的目标类型。

类型：字符串

有效值：s3 | iotsitewise | iottwinmaker

是否必需：是

### [filters](#)

筛选元数据传输作业的对象。

类型：[ListMetadataTransferJobsFilter](#) 对象数组

必需：否

### maxResults

一次性返回的最大结果数。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

### nextToken

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

必需：否

### sourceType

元数据传输作业的源类型。

类型：字符串

有效值：s3 | iotsitewise | iottwinmaker

是否必需：是

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "metadataTransferJobSummaries": [
    {
      "arn": "string",
```

```
    "creationDateTime": number,
    "metadataTransferJobId": "string",
    "progress": {
      "failedCount": number,
      "skippedCount": number,
      "succeededCount": number,
      "totalCount": number
    },
    "status": {
      "error": {
        "code": "string",
        "message": "string"
      },
      "queuedPosition": number,
      "state": "string"
    },
    "updateDateTime": number
  }
],
"nextToken": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [metadataTransferJobSummaries](#)

元数据传输作业摘要。

类型：[MetadataTransferJobSummary](#) 对象数组

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：`.*`

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)

- [AWS 适用于 Ruby V3 的 SDK](#)

# ListProperties

此 API 列出了组件的属性。

## 请求语法

```
POST /workspaces/workspaceId/properties-list HTTP/1.1
Content-type: application/json
```

```
{
  "componentName": "string",
  "componentPath": "string",
  "entityId": "string",
  "maxResults": number,
  "nextToken": "string"
}
```

## URI 请求参数

请求使用以下 URI 参数。

### workspaceId

工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### componentName

操作返回其属性的组件名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：否

### [componentPath](#)

此字符串指定从顶层组件开始的复合组件路径。

类型：字符串

长度限制：最小长度为 1。最大长度为 2048。

模式：`[a-zA-Z_\-0-9/]+`

必需：否

### [entityId](#)

操作返回其元数据（组件/属性）的实体的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.:]*[a-zA-Z0-9]+`

是否必需：是

### [maxResults](#)

一次性返回的最大结果数。默认值为 25。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：`.*`

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "propertySummaries": [
    {
      "areAllPropertyValuesReturned": boolean,
      "definition": {
        "configuration": {
          "string": "string"
        },
        "dataType": {
          "allowedValues": [
            {
              "booleanValue": boolean,
              "doubleValue": number,
              "expression": "string",
              "integerValue": number,
              "listValue": [
                "DataValue"
              ],
              "longValue": number,
              "mapValue": {
                "string": "DataValue"
              },
              "relationshipValue": {
                "targetComponentName": "string",
                "targetEntityId": "string"
              },
              "stringValue": "string"
            }
          ],
          "nestedType": "DataType",
          "relationship": {
            "relationshipType": "string",
            "targetComponentTypeId": "string"
          }
        }
      }
    }
  ]
}
```

```
    "type": "string",
    "unitOfMeasure": "string"
  },
  "defaultValue": {
    "booleanValue": boolean,
    "doubleValue": number,
    "expression": "string",
    "integerValue": number,
    "listValue": [
      "DataValue"
    ],
    "longValue": number,
    "mapValue": {
      "string" : "DataValue"
    },
    "relationshipValue": {
      "targetComponentName": "string",
      "targetEntityId": "string"
    },
    "stringValue": "string"
  },
  "displayName": "string",
  "isExternalId": boolean,
  "isFinal": boolean,
  "isImported": boolean,
  "isInherited": boolean,
  "isRequiredInEntity": boolean,
  "isStoredExternally": boolean,
  "isTimeSeries": boolean
},
"propertyName": "string",
"value": {
  "booleanValue": boolean,
  "doubleValue": number,
  "expression": "string",
  "integerValue": number,
  "listValue": [
    "DataValue"
  ],
  "longValue": number,
  "mapValue": {
    "string" : "DataValue"
  },
  "relationshipValue": {
```

```
        "targetComponentName": "string",
        "targetEntityId": "string"
    },
    "stringValue": "string"
}
]
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [nextToken](#)

指定下一页属性结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

### [propertySummaries](#)

包含属性信息的对象列表。

类型：[PropertySummary](#) 对象数组

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

ThrottlingException

速率超过限制。

HTTP 状态代码：429

ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListScenes

列出工作区中的所有场景。

## 请求语法

```
POST /workspaces/workspaceId/scenes-list HTTP/1.1
Content-type: application/json
```

```
{
  "maxResults": number,
  "nextToken": "string"
}
```

## URI 请求参数

请求使用以下 URI 参数。

### workspaceId

包含场景的工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### maxResults

指定最大结果显示数。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

## [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "sceneSummaries": [
    {
      "arn": "string",
      "contentLocation": "string",
      "creationDateTime": number,
      "description": "string",
      "sceneId": "string",
      "updateDateTime": number
    }
  ]
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

## [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：`.*`

### [sceneSummaries](#)

包含场景信息的对象列表。

类型：[SceneSummary](#) 对象数组

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListSyncJobs

全部列出 SyncJobs。

## 请求语法

```
POST /workspaces/workspaceId/sync-jobs-list HTTP/1.1
Content-type: application/json
```

```
{
  "maxResults": number,
  "nextToken": "string"
}
```

## URI 请求参数

请求使用以下 URI 参数。

### [workspaceId](#)

包含同步任务的工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### [maxResults](#)

一次性返回的最大结果数。默认值为 50。

有效范围：最小值为 0。最大值为 200。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "syncJobSummaries": [
    {
      "arn": "string",
      "creationDateTime": number,
      "status": {
        "error": {
          "code": "string",
          "message": "string"
        },
        "state": "string"
      },
      "syncSource": "string",
      "updateDateTime": number,
      "workspaceId": "string"
    }
  ]
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

### [syncJobSummaries](#)

列出的 SyncJob 摘要。

类型：[SyncJobSummary](#) 对象数组

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

## ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListSyncResources

列出同步资源。

## 请求语法

```
POST /workspaces/workspaceId/sync-jobs/syncSource/resources-list HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    { ... }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

## URI 请求参数

请求使用以下 URI 参数。

### syncSource

同步源。

#### Note

目前唯一支持的 syncSource 是 SITEWISE 。

模式：`[a-zA-Z_0-9]+`

是否必需：是

### workspaceId

包含同步任务的工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### [filters](#)

筛选请求的对象列表。

支持以下筛选条件：

- 使用状态筛选
- 使用 ResourceType 和筛选 ResourceId
- 使用 ResourceType 和筛选 ExternalId

类型：[SyncResourceFilter](#) 对象数组

必需：否

### [maxResults](#)

一次性返回的最大结果数。默认值为 50。

有效范围：最小值为 0。最大值为 200。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "syncResources": [
    {
      "externalId": "string",
      "resourceId": "string",
      "resourceType": "string",
      "status": {
        "error": {
          "code": "string",
          "message": "string"
        },
        "state": "string"
      },
      "updateDateTime": number
    }
  ]
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

### [syncResources](#)

同步资源。

类型：[SyncResourceSummary](#) 对象数组

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)

- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版软件开发工具包](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListTagsForResource

列出与资源关联的所有标签。

## 请求语法

```
POST /tags-list HTTP/1.1
Content-type: application/json

{
  "maxResults": number,
  "nextToken": "string",
  "resourceARN": "string"
}
```

## URI 请求参数

该请求不使用任何 URI 参数。

## 请求正文

请求接受采用 JSON 格式的以下数据。

### [maxResults](#)

一次性返回的最大结果数。默认值为 25。

有效范围：最小值为 1。最大值为 250。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：`.*`

必需：否

### [resourceARN](#)

资源的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\/a-zA-Z0-9_\-\.:]+`

必需：是

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "tags": {
    "string" : "string"
  }
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

## tags

可用于管理资源的元数据。

类型：字符串到字符串映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 128。

键模式：([\p{L}\p{Z}\p{N}\_.:/=\-\@]\*)

值长度限制：最小长度为 1。最大长度为 256。

价值模式：.\*

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)

- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListWorkspaces

检索当前账户中有关工作区的信息。

## 请求语法

```
POST /workspaces-list HTTP/1.1
Content-type: application/json
```

```
{
  "maxResults": number,
  "nextToken": "string"
}
```

## URI 请求参数

该请求不使用任何 URI 参数。

## 请求正文

请求接受采用 JSON 格式的以下数据。

### [maxResults](#)

一次性返回的最大结果数。默认值为 25。

有效范围：最小值为 1。最大值为 250。

类型：整数

有效范围：最小值为 0。最大值为 200。

必需：否

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "workspaceSummaries": [
    {
      "arn": "string",
      "creationDateTime": number,
      "description": "string",
      "linkedServices": [ "string" ],
      "updateDateTime": number,
      "workspaceId": "string"
    }
  ]
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [nextToken](#)

指定下一页结果的字符串。

类型：字符串

长度限制：最小长度为 0。最大长度为 17880。

模式：.\*

### [workspaceSummaries](#)

包含有关工作区信息的对象列表。

类型：[WorkspaceSummary](#) 对象数组

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)

- [AWS 适用于 Ruby V3 的 SDK](#)

# TagResource

为资源添加标签。

## 请求语法

```
POST /tags HTTP/1.1
Content-type: application/json

{
  "resourceARN": "string",
  "tags": {
    "string" : "string"
  }
}
```

## URI 请求参数

该请求不使用任何 URI 参数。

## 请求正文

请求接受采用 JSON 格式的以下数据。

### resourceARN

资源的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\/a-zA-Z0-9\_-\.\:]+

是否必需：是

### tags

要添加到此资源的元数据。

类型：字符串到字符串映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 128。

键模式：(`[\\p{L}\\p{Z}\\p{N}_.:/=+\\-@]*`)

值长度限制：最小长度为 1。最大长度为 256。

价值模式：`.*`

是否必需：是

## 响应语法

```
HTTP/1.1 200
```

## 响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 200 响应。

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### TooManyTagsException

标签数量超出限值。

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# UntagResource

删除资源的标签。

## 请求语法

```
DELETE /tags?resourceARN=resourceARN&tagKeys=tagKeys HTTP/1.1
```

## URI 请求参数

请求使用以下 URI 参数。

### resourceARN

资源的 ARN。

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\w/a-zA-Z0-9_-\.\:]+`

是否必需：是

### tagKeys

要从资源中删除的标签键名称列表。您没有指定该值。键及其关联的值都将删除。

数组成员：最少 0 个物品。最多 50 项。

长度限制：最小长度为 1。最大长度为 128。

模式：`([\p{L}\p{Z}\p{N}_./=+\-@]*)`

必需：是

## 请求体

该请求没有请求正文。

## 响应语法

```
HTTP/1.1 200
```

## 响应元素

如果此操作成功，则该服务会发送回带有空 HTTP 正文的 HTTP 200 响应。

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# UpdateComponentType

更新组件类型中的信息。

## 请求语法

```
PUT /workspaces/workspaceId/component-types/componentTypeId HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "componentTypeName": "string",
  "compositeComponentTypes": {
    "string" : {
      "componentTypeId": "string"
    }
  },
  "description": "string",
  "extendsFrom": [ "string" ],
  "functions": {
    "string" : {
      "implementedBy": {
        "isNative": boolean,
        "lambda": {
          "arn": "string"
        }
      }
    },
    "requiredProperties": [ "string" ],
    "scope": "string"
  }
},
"isSingleton": boolean,
"propertyDefinitions": {
  "string" : {
    "configuration": {
      "string" : "string"
    },
    "dataType": {
      "allowedValues": [
        {
          "booleanValue": boolean,
          "doubleValue": number,
          "expression": "string",
          "integerValue": number,

```

```

    "listValue": [
      "DataValue"
    ],
    "longValue": number,
    "mapValue": {
      "string" : "DataValue"
    },
    "relationshipValue": {
      "targetComponentName": "string",
      "targetEntityId": "string"
    },
    "stringValue": "string"
  }
],
"nestedType": "DataType",
"relationship": {
  "relationshipType": "string",
  "targetComponentTypeId": "string"
},
"unitOfMeasure": "string"
},
"defaultValue": {
  "booleanValue": boolean,
  "doubleValue": number,
  "expression": "string",
  "integerValue": number,
  "listValue": [
    "DataValue"
  ],
  "longValue": number,
  "mapValue": {
    "string" : "DataValue"
  },
  "relationshipValue": {
    "targetComponentName": "string",
    "targetEntityId": "string"
  },
  "stringValue": "string"
},
"displayName": "string",
"isExternalId": boolean,
"isRequiredInEntity": boolean,
"isStoredExternally": boolean,

```

```
    "isTimeSeries": boolean
  }
},
"propertyGroups": {
  "string" : {
    "groupType": "string",
    "propertyNames": [ "string" ]
  }
}
}
```

## URI 请求参数

请求使用以下 URI 参数。

### [componentTypeId](#)

组件类型的 ID。

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

是否必需：是

### [workspaceId](#)

工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### [componentTypeName](#)

组件类型名称。

类型：字符串

长度限制：最小长度为 0。最大长度为 256。

模式：`.*[^\u0000-\u001F\u007F]*.*`

必需：否

### [compositeComponentTypes](#)

这是一个将字符串映射到 `componentType` 的 `compositeComponentTypes` 的对象。`CompositeComponentType` 由 `componentTypeId` 引用。

类型：字符串到 [CompositeComponentTypeRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

### [description](#)

组件类型的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

必需：否

### [extendsFrom](#)

指定此组件类型扩展的组件类型。

类型：字符串数组

长度约束：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：否

### [functions](#)

将字符串映射到组件类型中的函数的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [FunctionRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

### [isSingleton](#)

指定一个实体是否可以有多个此类型的组件的布尔值。

类型：布尔值

必需：否

### [propertyDefinitions](#)

将字符串映射到组件类型中的属性定义的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [PropertyDefinitionRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

### [propertyGroups](#)

属性组。

类型：字符串到 [PropertyGroupRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json
```

```
{
  "arn": "string",
  "componentTypeId": "string",
  "state": "string",
  "workspaceId": "string"
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [arn](#)

组件类型的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\/a-zA-Z0-9\_-\.\.:]+

### [componentTypeId](#)

组件类型的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：[a-zA-Z\_\.\-0-9:]+

### [state](#)

组件类型的当前状态。

类型：字符串

有效值：CREATING | UPDATING | DELETING | ACTIVE | ERROR

### [workspaceId](#)

包含组件类型的工作区 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# UpdateEntity

更新实体。

## 请求语法

```
PUT /workspaces/workspaceId/entities/entityId HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "componentUpdates": {
    "string" : {
      "componentTypeId": "string",
      "description": "string",
      "propertyGroupUpdates": {
        "string" : {
          "groupType": "string",
          "propertyNames": [ "string" ],
          "updateType": "string"
        }
      },
    },
    "propertyUpdates": {
      "string" : {
        "definition": {
          "configuration": {
            "string" : "string"
          },
        },
        "dataType": {
          "allowedValues": [
            {
              "booleanValue": boolean,
              "doubleValue": number,
              "expression": "string",
              "integerValue": number,
              "listValue": [
                "DataValue"
              ],
              "longValue": number,
              "mapValue": {
                "string" : "DataValue"
              },
              "relationshipValue": {
                "targetComponentName": "string",

```

```

        "targetEntityId": "string"
    },
    "stringValue": "string"
}
],
"nestedType": "DataType",
"relationship": {
    "relationshipType": "string",
    "targetComponentTypeId": "string"
},
"type": "string",
"unitOfMeasure": "string"
},
"defaultValue": {
    "booleanValue": boolean,
    "doubleValue": number,
    "expression": "string",
    "integerValue": number,
    "listValue": [
        "DataValue"
    ],
    "longValue": number,
    "mapValue": {
        "string" : "DataValue"
    },
    "relationshipValue": {
        "targetComponentName": "string",
        "targetEntityId": "string"
    },
    "stringValue": "string"
},
"displayName": "string",
"isExternalId": boolean,
"isRequiredInEntity": boolean,
"isStoredExternally": boolean,
"isTimeSeries": boolean
},
"updateType": "string",
"value": {
    "booleanValue": boolean,
    "doubleValue": number,
    "expression": "string",
    "integerValue": number,
    "listValue": [

```

```

        "DataValue"
      ],
      "longValue": number,
      "mapValue": {
        "string" : "DataValue"
      },
      "relationshipValue": {
        "targetComponentName": "string",
        "targetEntityId": "string"
      },
      "stringValue": "string"
    }
  }
},
"updateType": "string"
}
},
"compositeComponentUpdates": {
  "string" : {
    "description": "string",
    "propertyGroupUpdates": {
      "string" : {
        "groupType": "string",
        "propertyNames": [ "string" ],
        "updateType": "string"
      }
    }
  },
  "propertyUpdates": {
    "string" : {
      "definition": {
        "configuration": {
          "string" : "string"
        },
        "dataType": {
          "allowedValues": [
            {
              "booleanValue": boolean,
              "doubleValue": number,
              "expression": "string",
              "integerValue": number,
              "listValue": [
                "DataValue"
              ],
              "longValue": number,

```

```
        "mapValue": {
            "string": "DataValue"
        },
        "relationshipValue": {
            "targetComponentName": "string",
            "targetEntityId": "string"
        },
        "stringValue": "string"
    }
],
"nestedType": "DataType",
"relationship": {
    "relationshipType": "string",
    "targetComponentTypeId": "string"
},
"type": "string",
"unitOfMeasure": "string"
},
"defaultValue": {
    "booleanValue": boolean,
    "doubleValue": number,
    "expression": "string",
    "integerValue": number,
    "listValue": [
        "DataValue"
    ],
    "longValue": number,
    "mapValue": {
        "string": "DataValue"
    },
    "relationshipValue": {
        "targetComponentName": "string",
        "targetEntityId": "string"
    },
    "stringValue": "string"
},
"displayName": "string",
"isExternalId": boolean,
"isRequiredInEntity": boolean,
"isStoredExternally": boolean,
"isTimeSeries": boolean
},
"updateType": "string",
"value": {
```

```

        "booleanValue": boolean,
        "doubleValue": number,
        "expression": "string",
        "integerValue": number,
        "listValue": [
            "DataValue"
        ],
        "longValue": number,
        "mapValue": {
            "string" : "DataValue"
        },
        "relationshipValue": {
            "targetComponentName": "string",
            "targetEntityId": "string"
        },
        "stringValue": "string"
    }
}
},
"updateType": "string"
}
},
"description": "string",
"entityName": "string",
"parentEntityUpdate": {
    "parentEntityId": "string",
    "updateType": "string"
}
}
}

```

## URI 请求参数

请求使用以下 URI 参数。

### [entityId](#)

实体的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.]*[a-zA-Z0-9]+`

是否必需：是

## workspaceId

包含实体的工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### componentUpdates

将字符串映射为请求中组件更新的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [ComponentUpdateRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

### compositeComponentUpdates

这是一个将字符串映射到请求中 `compositeComponent` 更新的对象。映射的每个键都代表 `compositeComponent` 的 `componentPath`。

类型：字符串到 [CompositeComponentUpdateRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 2048。

密钥模式：`[a-zA-Z_\-0-9/]+`

必需：否

### description

实体的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：.\*

必需：否

### entityName

实体名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[^\u0000-\u001F\u007F]+`

必需：否

### parentEntityUpdate

描述父实体更新请求的对象。

类型：[ParentEntityUpdateRequest](#) 对象

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json
```

```
{
  "state": string,
  "updateDateTime": number
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

## state

实体更新的当前状态。

类型：字符串

有效值：CREATING | UPDATING | DELETING | ACTIVE | ERROR

## updateDateTime

上次更新实体的日期和时间。

类型：时间戳

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### ConflictException

发生冲突。

HTTP 状态代码：409

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码 : 402

ThrottlingException

速率超过限制。

HTTP 状态代码 : 429

ValidationException

失败

HTTP 状态代码 : 400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# UpdatePricingPlan

更新定价计划。

## 请求语法

```
POST /pricingplan HTTP/1.1
Content-type: application/json

{
  "bundleNames": [ "string" ],
  "pricingMode": "string"
}
```

## URI 请求参数

该请求不使用任何 URI 参数。

## 请求正文

请求接受采用 JSON 格式的以下数据。

### bundleNames

捆绑包名称。

类型：字符串数组

数组成员：最少 1 个物品。最多 10 项。

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

必需：否

### pricingMode

定价模型。

类型：字符串

有效值：BASIC | STANDARD | TIERED\_BUNDLE

是否必需：是

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "currentPricingPlan": {
    "billableEntityCount": number,
    "bundleInformation": {
      "bundleNames": [ "string" ],
      "pricingTier": "string"
    },
    "effectiveDateTime": number,
    "pricingMode": "string",
    "updateDateTime": number,
    "updateReason": "string"
  },
  "pendingPricingPlan": {
    "billableEntityCount": number,
    "bundleInformation": {
      "bundleNames": [ "string" ],
      "pricingTier": "string"
    },
    "effectiveDateTime": number,
    "pricingMode": "string",
    "updateDateTime": number,
    "updateReason": "string"
  }
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### [currentPricingPlan](#)

更新当前定价计划。

类型：[PricingPlan](#) 对象

[pendingPricingPlan](#)

更新待定定价计划。

类型：[PricingPlan](#) 对象

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)

- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# UpdateScene

更新场景。

## 请求语法

```
PUT /workspaces/workspaceId/scenes/sceneId HTTP/1.1  
Content-type: application/json
```

```
{  
  "capabilities": [ "string" ],  
  "contentLocation": "string",  
  "description": "string",  
  "sceneMetadata": {  
    "string" : "string"  
  }  
}
```

## URI 请求参数

请求使用以下 URI 参数。

### sceneId

场景的 ID

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

是否必需：是

### workspaceId

包含场景的工作区 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### capabilities

场景用于渲染的功能列表。

类型：字符串数组

数组成员：最少 0 项。最多 50 项。

长度限制：最小长度为 0。最大长度为 256。

模式：.\*

必需：否

### contentLocation

指定内容定义文件位置的相对路径。

类型：字符串

长度限制：最小长度为 0。最大长度为 256。

模式：[sS]3://[A-Za-z0-9.\_/-]+

必需：否

### description

此场景的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：.\*

必需：否

### sceneMetadata

场景元数据。

类型：字符串到字符串映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

值长度限制：最小长度为 0。最大长度为 2048。

价值模式：`.*`

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json

{
  "updateDateTime": number
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

### updateDateTime

上次更新场景的日期和时间。

类型：时间戳

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

ThrottlingException

速率超过限制。

HTTP 状态代码：429

ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# UpdateWorkspace

更新工作区。

## 请求语法

```
PUT /workspaces/workspaceId HTTP/1.1  
Content-type: application/json
```

```
{  
  "description": "string",  
  "role": "string",  
  "s3Location": "string"  
}
```

## URI 请求参数

请求使用以下 URI 参数。

### [workspaceId](#)

工作区的 ID。

长度限制：最小长度为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

## 请求体

请求接受采用 JSON 格式的以下数据。

### [description](#)

工作区的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

必需：否

### role

与工作区关联的执行角色的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iam::[0-9]{12}:role/.*`

必需：否

### s3Location

存储与工作区关联资源的 S3 桶的 ARN。

类型：字符串

长度限制：最小长度为 0。最大长度为 1024。

模式：`.*(^arn:((aws)|(aws-cn)|(aws-us-gov)):s3:::)([a-zA-Z0-9_-]+$).*`

必需：否

## 响应语法

```
HTTP/1.1 200
Content-type: application/json
```

```
{
  "updateDateTime": number
}
```

## 响应元素

如果此操作成功，则该服务将会发送回 HTTP 200 响应。

服务以 JSON 格式返回以下数据。

## updateDateTime

当前更新的日期和时间。

类型：时间戳

## 错误

有关所有操作的常见错误的信息，请参阅[常见错误](#)。

### AccessDeniedException

访问被拒绝。

HTTP 状态代码：403

### InternalServerErrorException

出现意外错误。

HTTP 状态代码：500

### ResourceNotFoundException

未找到资源。

HTTP 状态代码：404

### ServiceQuotaExceededException

已超出服务限额。

HTTP 状态代码：402

### ThrottlingException

速率超过限制。

HTTP 状态代码：429

### ValidationException

失败

HTTP 状态代码：400

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 命令行界面 V2](#)
- [AWS 适用于 .NET 的 SDK V4](#)
- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Go v2 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS JavaScript V3 版 SDK](#)
- [AWS 适用于 Kotlin 的 SDK](#)
- [AWS 适用于 PHP 的 SDK V3](#)
- [AWS Python 软件开发工具包](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# 数据类型

AWS IoT TwinMaker API 包含各种操作使用的多种数据类型。本节详细描述每种数据类型。

## Note

不能保证数据类型结构中每个元素的顺序。应用程序不应假设特定的顺序。

支持以下数据类型：

- [BatchPutPropertyError](#)
- [BatchPutPropertyErrorEntry](#)
- [BundleInformation](#)
- [ColumnDescription](#)
- [ComponentPropertyGroupRequest](#)
- [ComponentPropertyGroupResponse](#)
- [ComponentRequest](#)
- [ComponentResponse](#)
- [ComponentSummary](#)
- [ComponentTypeSummary](#)
- [ComponentUpdateRequest](#)
- [CompositeComponentRequest](#)
- [CompositeComponentTypeRequest](#)
- [CompositeComponentTypeResponse](#)
- [CompositeComponentUpdateRequest](#)
- [DataConnector](#)
- [DataType](#)
- [DataValue](#)
- [DestinationConfiguration](#)
- [EntityPropertyReference](#)
- [EntitySummary](#)

- [ErrorDetails](#)
- [FilterByAsset](#)
- [FilterByAssetModel](#)
- [FilterByComponentType](#)
- [FilterByEntity](#)
- [FunctionRequest](#)
- [FunctionResponse](#)
- [InterpolationParameters](#)
- [lotSiteWiseSourceConfiguration](#)
- [lotSiteWiseSourceConfigurationFilter](#)
- [lotTwinMakerDestinationConfiguration](#)
- [lotTwinMakerSourceConfiguration](#)
- [lotTwinMakerSourceConfigurationFilter](#)
- [LambdaFunction](#)
- [ListComponentTypesFilter](#)
- [ListEntitiesFilter](#)
- [ListMetadataTransferJobsFilter](#)
- [MetadataTransferJobProgress](#)
- [MetadataTransferJobStatus](#)
- [MetadataTransferJobSummary](#)
- [OrderBy](#)
- [ParentEntityUpdateRequest](#)
- [PricingPlan](#)
- [PropertyDefinitionRequest](#)
- [PropertyDefinitionResponse](#)
- [PropertyFilter](#)
- [PropertyGroupRequest](#)
- [PropertyGroupResponse](#)
- [PropertyLatestValue](#)
- [PropertyRequest](#)

- [PropertyResponse](#)
- [PropertySummary](#)
- [PropertyValue](#)
- [PropertyValueEntry](#)
- [PropertyValueHistory](#)
- [Relationship](#)
- [RelationshipValue](#)
- [Row](#)
- [S3DestinationConfiguration](#)
- [S3SourceConfiguration](#)
- [SceneError](#)
- [SceneSummary](#)
- [SourceConfiguration](#)
- [Status](#)
- [SyncJobStatus](#)
- [SyncJobSummary](#)
- [SyncResourceFilter](#)
- [SyncResourceStatus](#)
- [SyncResourceSummary](#)
- [TabularConditions](#)
- [WorkspaceSummary](#)

# BatchPutPropertyError

BatchPutProperty 操作返回的错误。

## 内容

### entry

包含 BatchPutProperty 操作返回的错误相关信息的对象。

类型：[PropertyValueEntry](#) 对象

必需：是

### errorCode

错误代码。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

必需：是

### errorMessage

错误消息。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

必需：是

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)

- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# BatchPutPropertyErrorEntry

包含 BatchPutProperty 操作返回的错误相关信息的对象。

## 内容

errors

包含 BatchPutProperty 操作返回的错误相关信息的对象列表。

类型：[BatchPutPropertyError](#) 对象数组

数组成员：最少 1 个物品。最多 10 个物品。

必需：是

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# BundleInformation

有关定价捆绑包的信息。

## 内容

### bundleNames

捆绑包名称。

类型：字符串数组

数组成员：最少 1 个物品。最多 10 项。

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

必需：是

### pricingTier

定价套餐。

类型：字符串

有效值：TIER\_1 | TIER\_2 | TIER\_3 | TIER\_4

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ColumnDescription

查询结果中列的描述。

## 内容

### name

列描述的名称。

类型：字符串

模式：.\*

必需：否

### type

列描述的类型。

类型：字符串

有效值：NODE | EDGE | VALUE

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ComponentPropertyGroupRequest

组件属性组请求。

## 内容

### groupType

组类型。

类型：字符串

有效值：TABULAR

必需：否

### propertyNames

属性名称。

类型：字符串数组

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：否

### updateType

更新类型。

类型：字符串

有效值：UPDATE | DELETE | CREATE

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)

- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ComponentPropertyGroupResponse

组件属性组响应。

## 内容

### groupType

组类型。

类型：字符串

有效值：TABULAR

必需：是

### isInherited

指定是否从父实体继承属性组的布尔值

类型：布尔值

必需：是

### propertyNames

属性名称

类型：字符串数组

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：是

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)

- [AWS 适用于 Ruby V3 的 SDK](#)

# ComponentRequest

设置组件类型创建或更新请求相关信息的对象。

## 内容

### componentTypeId

组件类型的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：否

### description

组件请求的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

必需：否

### properties

将字符串映射到要在组件类型中设置的属性的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [PropertyRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

### propertyGroups

属性组。

类型：字符串到 [ComponentPropertyGroupRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ComponentResponse

返回组件类型创建或更新请求相关信息的对象。

## 内容

### areAllCompositeComponentsReturned

此标记标注是否在 API 响应中返回所有 `compositeComponents`。

类型：布尔值

必需：否

### areAllPropertiesReturned

此标记标注是否在 API 响应中返回组件的所有属性。返回的最大属性数是 800。

类型：布尔值

必需：否

### componentName

组件名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：否

### componentTypeId

组件类型的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：否

## compositeComponents

这列出了包含 compositeComponents 相关信息的对象。

类型：字符串到 [ComponentSummary](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

## definedIn

请求中设置的属性定义的名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`.*`

必需：否

## description

组件类型的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

必需：否

## properties

将字符串映射到要在组件类型中设置的属性的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [PropertyResponse](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

## propertyGroups

属性组。

类型：字符串到 [ComponentPropertyGroupResponse](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

## status

组件类型的状态。

类型：[Status](#) 对象

必需：否

## syncSource

同步作业的 syncSource ( 如果该实体是由同步作业创建的 ) 。

类型：字符串

模式：`[a-zA-Z_0-9]+`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ComponentSummary

返回组件摘要相关信息的对象。

## 内容

### componentName

组件名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：是

### componentTypeId

组件类型的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：是

### status

组件类型的状态。

类型：[Status](#) 对象

必需：是

### componentPath

此字符串指定从顶层组件开始的复合组件路径。

类型：字符串

长度限制：最小长度为 0。最大长度为 2048。

模式：`[a-zA-Z_\-0-9/]+`

必需：否

#### definedIn

请求中设置的属性定义的名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`.*`

必需：否

#### description

组件请求的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

必需：否

#### propertyGroups

属性组。

类型：字符串到 [ComponentPropertyGroupResponse](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

#### syncSource

同步作业的 syncSource ( 如果该实体是由同步作业创建的 ) 。

类型：字符串

模式：`[a-zA-Z_0-9]+`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ComponentTypeSummary

包含组件类型相关信息的对象。

## 内容

arn

组件类型的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9_\-\.\:]+`

必需：是

componentTypeId

组件类型的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：是

creationDateTime

创建组件类型的日期和时间。

类型：时间戳

必需：是

updateDateTime

上次更新组件类型的日期和时间。

类型：时间戳

必需：是

#### componentTypeName

组件类型名称。

类型：字符串

长度约束：最小长度为 0。最大长度为 256。

模式：`.*[^\u0000-\u001F\u007F]*.*`

必需：否

#### description

组件类型的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

必需：否

#### status

组件类型的当前状态。

类型：[Status](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ComponentUpdateRequest

组件更新请求。

## 内容

### componentTypeId

组件类型的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：[a-zA-Z\_\.\-0-9:]+

必需：否

### description

组件类型的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：.\*

必需：否

### propertyGroupUpdates

属性组更新。

类型：字符串到 [ComponentPropertyGroupRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：[a-zA-Z\_\-0-9]+

必需：否

### propertyUpdates

将字符串映射到要在组件类型更新中设置的属性的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [PropertyRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：`[a-zA-Z_\-0-9]+`

必需：否

## updateType

组件更新请求的更新类型。

类型：字符串

有效值：`CREATE` | `UPDATE` | `DELETE`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# CompositeComponentRequest

设置复合组件更新请求相关信息的对象。

## 内容

### description

组件类型的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：.\*

必需：否

### properties

这是一个将字符串映射到要在组件类型中设置的属性的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [PropertyRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：[a-zA-Z\_\-0-9]+

必需：否

### propertyGroups

属性组。

类型：字符串到 [ComponentPropertyGroupRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：[a-zA-Z\_\-0-9]+

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# CompositeComponentTypeRequest

设置有关组件类型复合组件类型的信息的对象

## 内容

`componentTypeId`

这就是 `compositeComponentType` 所指的 `componentTypeId`。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# CompositeComponentTypeResponse

返回组件类型复合组件类型相关信息的对象。

## 内容

### componentTypeId

这就是此 `compositeComponentType` 所指的 `componentTypeId`。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：否

### isInherited

此布尔值表示此 `compositeComponentType` 是否继承自其父项。

类型：布尔值

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# CompositeComponentUpdateRequest

设置复合组件更新请求相关信息的对象。

## 内容

### description

组件类型的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：.\*

必需：否

### propertyGroupUpdates

属性组更新。

类型：字符串到 [ComponentPropertyGroupRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：[a-zA-Z\_\-0-9]+

必需：否

### propertyUpdates

将字符串映射到要在组件类型更新中设置的属性的对象。映射中的每个字符串对于该对象都必须是唯一的。

类型：字符串到 [PropertyRequest](#) 对象的映射

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：[a-zA-Z\_\-0-9]+

必需：否

### updateType

组件更新请求的更新类型。

类型：字符串

有效值：CREATE | UPDATE | DELETE

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# DataConnector

数据连接器。

## 内容

### isNative

一个布尔值，用于指定数据连接器是否为原生 AWS IoT TwinMaker。

类型：布尔值

必需：否

### lambda

与此数据连接器关联的 Lambda 函数。

类型：[LambdaFunction](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# Data Type

指定属性的数据类型的对象。

## 内容

### type

数据类型的基础类型。

类型：字符串

有效值：RELATIONSHIP | STRING | LONG | BOOLEAN | INTEGER | DOUBLE | LIST  
| MAP

必需：是

### allowedValues

此数据类型的允许值。

类型：[DataValue](#) 对象数组

数组成员：最少 0 个物品。最多 50 项。

必需：否

### nestedType

数据类型中的嵌套类型。

类型：[DataType](#) 对象

必需：否

### relationship

一个组件与另一个组件之间的关联关系。

类型：[Relationship](#) 对象

必需：否

### unitOfMeasure

此数据类型中使用的度量单位。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# DataValue

指定属性值的对象。

## 内容

### booleanValue

布尔值。

类型：布尔值

必需：否

### doubleValue

双精度值。

类型：双精度

必需：否

### expression

生成值的表达式。

类型：字符串

长度限制：长度下限为 1。最大长度为 316。

模式： $(^\\$\\{Parameters\\. [a-zA-z]+([a-zA-z_0-9]^*)\\})\\$)$

必需：否

### integerValue

整数值。

类型：整数

必需：否

### listValue

多个值的列表。

类型：[DataValue](#) 对象数组

数组成员：最少 0 个物品。最多 50 项。

必需：否

### longValue

长型值。

类型：长整型

必需：否

### mapValue

将字符串映射到多个 DataValue 对象的对象。

类型：字符串到 [DataValue](#) 对象的映射

映射条目：最低 0 项。最多 50 项。

密钥长度限制：最小长度为 1。最大长度为 256。

密钥模式：.\*

必需：否

### relationshipValue

将一个组件关联到另一个组件的值。

类型：[RelationshipValue](#) 对象

必需：否

### stringValue

字符串值。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# DestinationConfiguration

[操作链接] 元数据传输作业目标配置。

## 内容

### type

目标类型。

类型：字符串

有效值：s3 | iotsitewise | iottwinmaker

必需：是

### iotTwinMakerConfiguration

元数据传输任务 AWS IoT TwinMaker 配置。

类型：[iotTwinMakerDestinationConfiguration](#) 对象

必需：否

### s3Configuration

元数据传输作业 S3 配置。[需要添加 S3 实体]

类型：[S3DestinationConfiguration](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# EntityPropertyReference

唯一标识实体属性的对象。

## 内容

### propertyName

属性的名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：是

### componentName

组件名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：否

### componentPath

此字符串指定从顶层组件开始的复合组件路径。

类型：字符串

长度限制：最小长度为 0。最大长度为 2048。

模式：`[a-zA-Z_\-0-9/]+`

必需：否

### entityId

实体的 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.]*[a-zA-Z0-9]+`

必需：否

### externalIdProperty

外部 IDs 到属性名称的映射。外部 IDs 唯一标识来自外部数据存储的属性。

类型：字符串到字符串映射

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`.*`

值长度限制：最小长度为 1。最大长度为 256。

价值模式：`.*`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# EntitySummary

包含有关实体信息的对象。

## 内容

arn

实体的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\/a-zA-Z0-9_\-\.\:]+`

必需：是

creationDateTime

创建实体的日期和时间。

类型：时间戳

必需：是

entityId

实体的 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.]*[a-zA-Z0-9]+`

必需：是

entityName

实体名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[^\u0000-\u001F\u007F]+`

必需：是

#### status

实体的当前状态。

类型：[Status](#) 对象

必需：是

#### updateDateTime

上次更新实体的日期和时间。

类型：时间戳

必需：是

#### description

实体的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

必需：否

#### hasChildEntities

指定实体是否有子实体的最终布尔值。

类型：布尔值

必需：否

#### parentEntityId

父实体 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`\\$ROOT|^[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\\-0-9\\.]*[a-zA-Z0-9]+`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ErrorDetails

错误详细信息。

## 内容

### code

错误代码。

类型：字符串

有效值：VALIDATION\_ERROR | INTERNAL\_FAILURE | SYNC\_INITIALIZING\_ERROR | SYNC\_CREATING\_ERROR | SYNC\_PROCESSING\_ERROR | SYNC\_DELETING\_ERROR | PROCESSING\_ERROR | COMPOSITE\_COMPONENT\_FAILURE

必需：否

### message

错误消息。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# FilterByAsset

按资产筛选。在导出 AWS IoT SiteWise 内容时，可以包含多个筛选条件。请参阅《AWS 物联网 SiteWise 用户指南》中的[导出元数据示例](#)。

## 内容

### assetExternalId

资产的外部 ID 属性。

类型：字符串

长度限制：最小长度为 2。最大长度为 128。

模式：`.*[a-zA-Z0-9_][a-zA-Z_\-0-9.:]*[a-zA-Z0-9_]+.*`

必需：否

### assetId

按资产 ID 筛选。

类型：字符串

长度限制：固定长度为 36。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}`

必需：否

### includeAssetModel

设置为 true 可包括指定资产使用的资产模型。

类型：布尔值

必需：否

### includeOffspring

设置为 true 可在指定资产模型的层次结构中包含所有关联的资产模型。

类型：布尔值

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# FilterByAssetModel

按资产模型筛选。

## 内容

### assetModelExternalId

资产模型的外部 ID 属性。

类型：字符串

长度限制：最小长度为 2。最大长度为 128。

模式：`.*[a-zA-Z0-9_][a-zA-Z_\-0-9.]*[a-zA-Z0-9_]+.*`

必需：否

### assetModelId

资产模型的 ID。

类型：字符串

长度限制：固定长度为 36。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}`

必需：否

### includeAssets

设置为 true 可包括使用指定资产模型的所有资产。

类型：布尔值

必需：否

### includeOffspring

设置为 true 可在指定资产模型的层次结构中包括所有关联的资产模型。

类型：布尔值

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# FilterByComponentType

按组件类型筛选。

## 内容

componentTypeId

组件类型 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：是

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# FilterByEntity

按实体筛选。

## 内容

entityId

实体 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.:]*[a-zA-Z0-9]+`

必需：是

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# FunctionRequest

函数请求正文。

## 内容

implementedBy

数据连接器。

类型：[DataConnector](#) 对象

必需：否

requiredProperties

函数的必需属性。

类型：字符串数组

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：否

scope

函数的范围。

类型：字符串

有效值：ENTITY | WORKSPACE

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)

- [AWS 适用于 Ruby V3 的 SDK](#)

# FunctionResponse

函数响应。

## 内容

### implementedBy

数据连接器。

类型：[DataConnector](#) 对象

必需：否

### isInherited

表示是否继承此函数。

类型：布尔值

必需：否

### requiredProperties

函数的必需属性。

类型：字符串数组

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：否

### scope

函数的范围。

类型：字符串

有效值：ENTITY | WORKSPACE

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# InterpolationParameters

指定如何在列表中插入数据的对象。

## 内容

### interpolationType

插值类型。

类型：字符串

有效值：LINEAR

必需：否

### intervalInSeconds

插值时间间隔（以秒为单位）。

类型：长整型

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# lotSiteWiseSourceConfiguration

元数据传输任务 AWS IoT SiteWise 源配置。

## 内容

### filters

AWS 物联网 SiteWise 源配置筛选器。

类型：[lotSiteWiseSourceConfigurationFilter](#) 对象数组

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# lotSiteWiseSourceConfigurationFilter

AWS 物联网 SiteWise 源配置筛选器。

## 内容

### Important

由于此数据类型为 UNION，因此在使用或返回时只能指定以下成员之一。

### filterByAsset

按资产筛选。

类型：[FilterByAsset](#) 对象

必需：否

### filterByAssetModel

按资产模型筛选。

类型：[FilterByAssetModel](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# IoT TwinMaker Destination Configuration

元数据传输任务 AWS IoT TwinMaker 目标配置。

## 内容

### workspace

物联网 TwinMaker 工作空间。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9_\-\.]+`

必需：是

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# lotTwinMakerSourceConfiguration

元数据传输任务 AWS IoT TwinMaker 源配置。

## 内容

### workspace

物联网 TwinMaker 工作空间。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9_\-\.]+`

必需：是

### filters

元数据传输任务 AWS IoT TwinMaker 源配置筛选器。

类型：[lotTwinMakerSourceConfigurationFilter](#) 对象数组

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# IoT TwinMaker Source Configuration Filter

元数据传输任务 AWS IoT TwinMaker 源配置筛选器。

## 内容

### Important

由于此数据类型为 UNION，因此在使用或返回时只能指定以下成员之一。

### filterByComponentType

按组件类型筛选。

类型：[FilterByComponentType](#) 对象

必需：否

### filterByEntity

按实体筛选。

类型：[FilterByEntity](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# LambdaFunction

Lambda 函数。

## 内容

arn

Lambda 函数的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):lambda:[a-z0-9-]+:[0-9]{12}:function:[\a-zA-Z0-9_-]+`

必需：是

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListComponentTypesFilter

用于筛选组件类型列表中的条目的对象。

## Note

有效输入仅接受一个对象。

## 内容

## Important

由于此数据类型为 UNION，因此在使用或返回时只能指定以下成员之一。

### extendsFrom

列表中的组件类型扩展的组件类型。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：[a-zA-Z\_\.\-0-9:]+

必需：否

### isAbstract

指定列表中组件类型是否为抽象的布尔值。

类型：布尔值

必需：否

### namespace

列表中组件类型所属的命名空间。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListEntitiesFilter

筛选实体列表中的条目的对象。

## 内容

### Important

由于此数据类型为 UNION，因此在使用或返回时只能指定以下成员之一。

### componentTypeId

列表实体中组件类型的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\.\-0-9:]+`

必需：否

### externalId

组件的外部 ID 属性。外部 ID 属性是外部存储系统的主键。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`.*`

必需：否

### parentEntityId

列表中实体的父实体。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`\\$ROOT|^[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.]*[a-zA-Z0-9]+`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ListMetadataTransferJobsFilter

ListMetadataTransferJobs 过滤器。

## 内容

### Important

由于此数据类型为 UNION，因此在使用或返回时只能指定以下成员之一。

### state

筛选条件状态。

类型：字符串

有效值：VALIDATING | PENDING | RUNNING | CANCELLING | ERROR | COMPLETED | CANCELLED

必需：否

### workspaceId

工作区 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)



# MetadataTransferJobProgress

元数据传输作业的进度。

## 内容

### failedCount

在此过程中失败的资产数量。

类型：整数

必需：否

### skippedCount

在此过程中跳过的资产数量。

类型：整数

必需：否

### succeededCount

成功转移的资产数量。

类型：整数

必需：否

### totalCount

转移过程中的资产总数。

类型：整数

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)

- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# MetadataTransferJobStatus

元数据传输作业状态。

## 内容

### error

元数据传输作业错误。

类型：[ErrorDetails](#) 对象

必需：否

### queuedPosition

排队的位置。

类型：整数

必需：否

### state

元数据传输作业状态。

类型：字符串

有效值：VALIDATING | PENDING | RUNNING | CANCELLING | ERROR | COMPLETED | CANCELLED

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# MetadataTransferJobSummary

元数据传输作业摘要。

## 内容

arn

元数据传输作业摘要 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9_\-\.\:]+`

必需：是

creationDateTime

元数据传输任务摘要创建 DateTime 对象。

类型：时间戳

必需：是

metadataTransferJobId

元数据传输作业摘要 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

status

元数据传输作业摘要状态。

类型：[MetadataTransferJobStatus](#) 对象

必需：是

updateDateTime

元数据传输任务摘要更新 DateTime 对象

类型：时间戳

必需：是

progress

元数据传输任务摘要进度。

类型：[MetadataTransferJobProgress](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# OrderBy

对返回输出进行排序的筛选标准。可以按升序或降序对其进行排序。

## 内容

### propertyName

属性名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

必需：是

### order

筛选结果的设置顺序。

类型：字符串

有效值：ASCENDING | DESCENDING

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# ParentEntityUpdateRequest

父实体更新请求。

## 内容

### updateType

更新的类型。

类型：字符串

有效值：UPDATE | DELETE

必需：是

### parentEntityId

父实体 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`\\$ROOT|^[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\\-0-9\\.]*[a-zA-Z0-9]+`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PricingPlan

定价方案。

## 内容

### effectiveDateTime

定价方案的生效日期和时间。

类型：时间戳

必需：是

### pricingMode

定价模型。

类型：字符串

有效值：BASIC | STANDARD | TIERED\_BUNDLE

必需：是

### updateDateTime

更新定价方案的设定日期和时间。

类型：时间戳

必需：是

### updateReason

更改定价方案的更新原因。

类型：字符串

有效值：DEFAULT | PRICING\_TIER\_UPDATE | ENTITY\_COUNT\_UPDATE | PRICING\_MODE\_UPDATE | OVERWRITTEN

必需：是

### billableEntityCount

可计费实体数量。

类型：长整型

必需：否

### bundleInformation

定价方案的捆绑包信息。

类型：[BundleInformation](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertyDefinitionRequest

用于设置有关属性的信息的对象。

## 内容

### configuration

指定关于属性的配置信息的映射。使用此字段指定从外部源读取和向外部源写入的信息。

类型：字符串到字符串映射

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

值模式：`.*`

必需：否

### dataType

包含数据类型信息的对象。

类型：[DataType](#) 对象

必需：否

### defaultValue

包含默认值的对象。

类型：[DataValue](#) 对象

必需：否

### displayName

属性的友好名称。

类型：字符串

长度约束：最小长度为 0。最大长度为 256。

模式：`.*[^\u0000-\u001F\u007F]*.*`

必需：否

#### isExternalId

指定属性 ID 是否来自外部数据存储的布尔值。

类型：布尔值

必需：否

#### isRequiredInEntity

指定是否需要属性的布尔值。

类型：布尔值

必需：否

#### isStoredExternally

指定是否在外部分存储属性的布尔值。

类型：布尔值

必需：否

#### isTimeSeries

指定属性是否由时间序列数据组成的布尔值。

类型：布尔值

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertyDefinitionResponse

包含来自属性定义请求的响应数据的对象。

## 内容

### dataType

包含数据类型信息的对象。

类型：[DataType](#) 对象

必需：是

### isExternalId

指定属性 ID 是否来自外部数据存储的布尔值。

类型：布尔值

必需：是

### isFinal

指定是否可以更新属性定义的布尔值。

类型：布尔值

必需：是

### isImported

指定是否从外部数据存储导入属性定义的布尔值。

类型：布尔值

必需：是

### isInherited

指定属性定义是否继承自父实体的布尔值。

类型：布尔值

必需：是

## isRequiredInEntity

指定某个实体内是否需要属性的布尔值。

类型：布尔值

必需：是

## isStoredExternally

指定是否在外部存储属性的布尔值。

类型：布尔值

必需：是

## isTimeSeries

指定属性是否由时间序列数据组成的布尔值。

类型：布尔值

必需：是

## configuration

指定关于属性的配置信息的映射。

类型：字符串到字符串映射

密钥长度限制：最小长度为 1。最大长度为 256。

键模式：`[a-zA-Z_\-0-9]+`

值模式：`.*`

必需：否

## defaultValue

包含默认值的对象。

类型：[DataValue](#) 对象

必需：否

## displayName

属性的友好名称。

类型：字符串

长度约束：最小长度为 0。最大长度为 256。

模式：`.*[^\u0000-\u001F\u007F]*.*`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertyFilter

筛选属性请求返回的条目的对象。

## 内容

### operator

与此属性筛选条件关联的运算符。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

必需：否

### propertyName

与此属性筛选条件关联的属性名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

必需：否

### value

与此属性筛选条件关联的值。

类型：[DataValue](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)

- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertyGroupRequest

## 内容

### groupType

组类型。

类型：字符串

有效值：TABULAR

必需：否

### propertyNames

属性名称。

类型：字符串数组

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertyGroupResponse

属性组响应

## 内容

groupType

组类型。

类型：字符串

有效值：TABULAR

必需：是

isInherited

指定是否从父实体继承属性组的布尔值

类型：布尔值

必需：是

propertyNames

属性名称。

类型：字符串数组

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：是

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)

- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertyLatestValue

属性的最新值。

## 内容

### propertyReference

用于指定有关属性的信息的对象。

类型：[EntityPropertyReference](#) 对象

必需：是

### propertyValue

属性的值。

类型：[DataValue](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertyRequest

用于设置有关属性的信息的对象。

## 内容

### definition

用于指定有关属性的信息的对象。

类型：[PropertyDefinitionRequest](#) 对象

必需：否

### updateType

更新属性请求的更新类型。

类型：字符串

有效值：UPDATE | DELETE | CREATE

必需：否

### value

属性的值。

类型：[DataValue](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertyResponse

包含属性响应相关信息的对象。

## 内容

### areAllPropertyValuesReturned

此标记标注是否在 API 响应中返回列表或映射类型属性的所有值。每个属性返回值的最大数量为 50。

类型：布尔值

必需：否

### definition

用于指定有关属性的信息的对象。

类型：[PropertyDefinitionResponse](#) 对象

必需：否

### value

属性的值。

类型：[DataValue](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertySummary

这是一个包含属性信息的对象。

## 内容

### propertyName

这是属性名称。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：是

### areAllPropertyValuesReturned

此标记标注是否在 API 响应中返回列表或映射类型属性的所有值。每个属性返回值的最大数量为 50。

类型：布尔值

必需：否

### definition

这是属性的架构。

类型：[PropertyDefinitionResponse](#) 对象

必需：否

### value

这是属性的值。

类型：[DataValue](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertyValue

包含有关时间序列属性值的信息的对象。

## 内容

### value

指定时间序列属性值的对象。

类型：[DataValue](#) 对象

必需：是

### time

ISO86时间序列属性的值的 01 DateTime 。

以 ISO 8601 格式记录属性值的时间：YYYY-MM-DDThh:mm:ss[.SSSSSSSSSS][z/±HH:mm]。

- [YYYY]：年
- [MM]：月
- [DD]：日
- [hh]：时
- [mm]：分
- [ss]：秒
- [.SSSSSSSSSS]：在保持优先级的情况下，提升精度 例如：[.573123] 等于 573123000 毫微秒。
- Z：默认时区 UTC
- ± HH:mm：时区偏移（以小时和分钟为单位）。

必填子字段:: mm:ss 和 [z/±HH:mm] YYYY-MM-DDThh

类型：字符串

长度约束：最小长度为 20。最大长度为 35。

必需：否

### timestamp

该成员已弃用。

时间序列属性值的时间戳。

类型：时间戳

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertyValueEntry

用于指定时间序列属性值相关信息的对象。该对象由[BatchPutPropertyValues](#)操作使用和使用。

## 内容

### entityPropertyReference

包含具有该属性的实体的相关信息对象。

类型：[EntityPropertyReference](#) 对象

必需：是

### propertyValues

指定时间序列属性值的对象列表。

类型：[PropertyValue](#) 对象数组

数组成员：最少 1 个物品。最多 10 个物品。

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# PropertyValueHistory

时间序列属性值的历史记录。

## 内容

### entityPropertyReference

唯一标识实体属性的对象。

类型：[EntityPropertyReference](#) 对象

必需：是

### values

包含时间序列属性历史记录值相关信息的对象列表。

类型：[PropertyValue](#) 对象数组

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# Relationship

指定与其他组件类型之间关系的对象。

## 内容

### relationshipType

关系的类型。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：.\*

必需：否

### targetComponentTypeId

与此关系关联的目标组件类型的 ID。

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：[a-zA-Z\_\.\-0-9:]+

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# RelationshipValue

关联组件和实体的值。

## 内容

### targetComponentName

与此关系值关联的目标组件名称

类型：字符串

长度限制：最小长度为 1。最大长度为 256。

模式：`[a-zA-Z_\-0-9]+`

必需：否

### targetEntityId

与此关系值关联的目标实体的 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|^[a-zA-Z0-9][a-zA-Z_\-0-9.:]*[a-zA-Z0-9]+`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# Row

代表查询结果中的一行。

## 内容

rowData

一行查询结果中的数据。

类型：JSON 值的数组。

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# S3DestinationConfiguration

S3 目标配置。

## 内容

location

S3 目标配置位置。

类型：字符串

模式：`.*(^arn:((aws)|(aws-cn)|(aws-us-gov)):s3:::)([/a-zA-Z0-9_-]+$).*`

必需：是

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# S3SourceConfiguration

S3 目标源配置。

## 内容

location

S3 目标源配置位置。

类型：字符串

模式：`.*(^arn:((aws)|(aws-cn)|(aws-us-gov)):s3:::)([a-zA-Z0-9_-]+)\`  
`([/a-zA-Z0-9_-]+$).*`

必需：是

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# SceneError

场景错误。

## 内容

code

SceneError 代码。

类型：字符串

有效值：MATTERPORT\_ERROR

必需：否

message

SceneError 消息。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# SceneSummary

包含场景信息的对象。

## 内容

arn

场景的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\a-zA-Z0-9_\-\.\:]+`

必需：是

contentLocation

指定内容定义文件位置的相对路径。

类型：字符串

长度约束：最小长度为 0。最大长度为 256。

模式：`[sS]3://[A-Za-z0-9._/-]+`

必需：是

creationDateTime

创建场景的日期和时间。

类型：时间戳

必需：是

sceneId

场景的 ID

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

updateDateTime

上次更新场景的日期和时间。

类型：时间戳

必需：是

description

场景描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：`.*`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# SourceConfiguration

数据源配置。

## 内容

### type

源配置类型。

类型：字符串

有效值：`s3` | `iotsitewise` | `iottwinmaker`

必需：是

### iotSiteWiseConfiguration

源配置物联网 SiteWise 配置。

类型：[iotSiteWiseSourceConfiguration](#) 对象

必需：否

### iotTwinMakerConfiguration

源配置物联网 TwinMaker 配置。

类型：[iotTwinMakerSourceConfiguration](#) 对象

必需：否

### s3Configuration

源配置 S3 配置。

类型：[S3SourceConfiguration](#) 对象

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# Status

表示实体、组件、组件类型或工作区状态的对象。

## 内容

### error

错误消息。

类型：[ErrorDetails](#) 对象

必需：否

### state

实体、组件、组件类型或工作区的当前状态。

类型：字符串

有效值：CREATING | UPDATING | DELETING | ACTIVE | ERROR

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# SyncJobStatus

SyncJob 状态。

## 内容

error

SyncJob 错误。

类型：[ErrorDetails](#) 对象

必需：否

state

状态 SyncJob 状态。

类型：字符串

有效值：CREATING | INITIALIZING | ACTIVE | DELETING | ERROR

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# SyncJobSummary

摘 SyncJob 要。

## 内容

arn

SyncJob 摘要 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\w/a-zA-Z0-9_-\.\:]+`

必需：否

creationDateTime

创建日期和时间。

类型：时间戳

必需：否

status

SyncJob 摘要状态。

类型：[SyncJobStatus](#) 对象

必需：否

syncSource

同步源。

类型：字符串

模式：`[a-zA-Z_0-9]+`

必需：否

## updateDateTime

更新日期和时间。

类型：时间戳

必需：否

## workspaceId

包含同步任务的工作区 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# SyncResourceFilter

同步资源筛选条件。

## 内容

### Important

由于此数据类型为 UNION，因此在使用或返回时只能指定以下成员之一。

### externalId

外部 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：否

### resourceId

同步资源筛选条件资源 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：否

### resourceType

同步资源筛选条件资源类型

类型：字符串

有效值：ENTITY | COMPONENT\_TYPE

必需：否

## state

同步资源筛选条件的状态。

类型：字符串

有效值：INITIALIZING | PROCESSING | DELETED | IN\_SYNC | ERROR

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# SyncResourceStatus

同步资源状态。

## 内容

error

状态错误。

类型：[ErrorDetails](#) 对象

必需：否

state

同步资源状态的状态。

类型：字符串

有效值：INITIALIZING | PROCESSING | DELETED | IN\_SYNC | ERROR

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# SyncResourceSummary

同步资源摘要。

## 内容

### externalId

外部 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：否

### resourceId

资源 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：否

### resourceType

资源类型。

类型：字符串

有效值：ENTITY | COMPONENT\_TYPE

必需：否

### status

同步资源摘要状态。

类型：[SyncResourceStatus](#) 对象

必需：否

updateDateTime

更新日期和时间。

类型：时间戳

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# TabularConditions

表格条件。

## 内容

### orderBy

对输出进行排序的筛选标准。可以按升序或降序对其进行排序。

类型：[OrderBy](#) 对象数组

数组成员：最少 1 个物品。最多 10 个物品。

必需：否

### propertyFilters

您可以使用各种逻辑运算符和键值格式来筛选请求。例如：

```
{"key": "serverType", "value": "webServer"}
```

类型：[PropertyFilter](#) 对象数组

数组成员：最少 1 个物品。最多 10 个物品。

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

# WorkspaceSummary

包含工作区相关信息的对象。

## 内容

arn

工作区的 ARN。

类型：字符串

长度约束：最小长度为 20。最大长度为 2048。

模式：`arn:((aws)|(aws-cn)|(aws-us-gov)):iottwinmaker:[a-z0-9-]+:[0-9]{12}:[\w/a-zA-Z0-9_-\.\.]+`

必需：是

creationDateTime

创建工作区的日期和时间。

类型：时间戳

必需：是

updateDateTime

上次更新工作区的日期和时间。

类型：时间戳

必需：是

workspaceId

工作区的 ID。

类型：字符串

长度限制：长度下限为 1。最大长度为 128。

模式：`[a-zA-Z_0-9][a-zA-Z_\-0-9]*[a-zA-Z0-9]+`

必需：是

#### description

工作区的描述。

类型：字符串

长度约束：最小长度为 0。最大长度为 2048。

模式：.\*

必需：否

#### linkedServices

链接到工作区的服务列表。

类型：字符串数组

模式：[a-zA-Z\_0-9]+

必需：否

## 另请参阅

有关以特定语言之一使用此 API 的更多信息 AWS SDKs，请参阅以下内容：

- [AWS 适用于 C++ 的 SDK](#)
- [AWS 适用于 Java 的 SDK V2](#)
- [AWS 适用于 Ruby V3 的 SDK](#)

## 常见参数

以下列表包含所有操作用于使用查询字符串对 Signature Version 4 请求进行签名的参数。任何特定于操作的参数都列在该操作的主题中。有关签名版本 4 的更多信息，请参阅 IAM 用户指南中的[签署 AWS API 请求](#)。

### Action

要执行的操作。

类型：字符串。

必需：是

### Version

请求所针对的 API 版本，以格式表示 YYYY-MM-DD。

类型：字符串。

必需：是

### X-Amz-Algorithm

您用于创建请求签名的哈希算法。

条件：当您在查询字符串中而不是 HTTP 授权标头中包括身份验证信息时，请指定此参数。

类型：字符串

有效值：AWS4-HMAC-SHA256

必需：条件

### X-Amz-Credential

凭证范围值，该值是一个字符串，其中包含您的访问密钥、日期、您要定位的区域、您请求的服务以及终止字符串（“aws4\_request”）。值采用以下格式表示：access\_key/YYYYMMDD/region/service/aws4\_request。

有关更多信息，请参阅 IAM 用户指南中的[创建已签名的 AWS API 请求](#)。

条件：当您在查询字符串中而不是 HTTP 授权标头中包括身份验证信息时，请指定此参数。

类型：字符串

必需：条件

### X-Amz-Date

用于创建签名的日期。格式必须为 ISO 8601 基本格式 (YYYYMMDD'T'HHMMSS'Z')。例如，以下日期时间是有效 X-Amz-Date 值：20120325T120000Z。

条件：X-Amz-Date 对所有请求而言是可选的；它可以用于覆盖对请求签名所使用的日期。如果日期标题以 ISO 8601 基本格式指定，则 X-Amz-Date 不是必填项。使用 X-Amz-Date 时，它总是会覆盖 Date 标题的值。有关更多信息，请参阅 IAM 用户指南中的 [AWS API 请求签名要素](#)。

类型：字符串

必需：条件

### X-Amz-Security-Token

通过调用 AWS Security Token Service (AWS STS) 获得的临时安全令牌。有关支持来自 AWS STS 的临时安全凭证的服务列表，请参阅《IAM 用户指南》中的 [使用 IAM 的 AWS 服务](#)。

条件：如果您使用的是中的临时安全证书 AWS STS，则必须包含安全令牌。

类型：字符串

必需：条件

### X-Amz-Signature

指定从要签名的字符串和派生的签名密钥计算的十六进制编码签名。

条件：当您在查询字符串中而不是 HTTP 授权标头中包括身份验证信息时，请指定此参数。

类型：字符串

必需：条件

### X-Amz-SignedHeaders

指定作为规范请求的一部分包含的所有 HTTP 标头。有关指定签名标头的更多信息，请参阅 IAM 用户指南中的 [创建已签名的 AWS API 请求](#)。

条件：当您在查询字符串中而不是 HTTP 授权标头中包括身份验证信息时，请指定此参数。

类型：字符串

必需：条件

## 常见错误

本节列出了所有 AWS 服务的 API 操作中常见的错误。对于特定于此服务的 API 操作的错误，请参阅该 API 操作的主题。

### AccessDeniedException

您没有足够的访问权限，无法执行该操作。

HTTP 状态代码：403

### ExpiredTokenException

请求中包含的安全令牌已到期。

HTTP 状态代码：403

### IncompleteSignature

请求签名不符合 AWS 标准。

HTTP 状态代码：403

### InternalFailure

由于未知错误、异常或故障，请求处理失败。

HTTP 状态代码：500

### MalformedHttpRequestException

HTTP 级别的请求问题，例如，我们无法根据内容编码指定的解压缩算法对正文进行解压缩。

HTTP 状态代码：400

### NotAuthorized

您无权执行此操作。

HTTP 状态代码：401

### OptInRequired

AWS 访问密钥 ID 需要订阅该服务。

HTTP 状态代码：403

## RequestAbortedException

当请求在回复发送前中止请求时（例如客户端关闭连接）可以使用的便捷异常。

HTTP 状态代码：400

## RequestEntityTooLargeException

HTTP 级别的请求问题。请求实体太大。

HTTP 状态代码：413

## RequestExpired

请求在请求的日期戳后超过 15 分钟或请求到期日期（例如预签名 URLs）超过 15 分钟到达服务，或者请求上的日期戳超过未来 15 分钟。

HTTP 状态代码：400

## RequestTimeoutException

HTTP 级别的请求问题。读取请求超时。

HTTP 状态代码：408

## ServiceUnavailable

由于服务器发生临时故障而导致请求失败。

HTTP 状态代码：503

## ThrottlingException

由于请求限制而导致请求被拒绝。

HTTP 状态代码：400

## UnrecognizedClientException

我们的记录中不存在提供的 X.509 证书或 AWS 访问密钥 ID。

HTTP 状态代码：403

## UnknownOperationException

所请求的操作无效。确认正确键入了操作。

HTTP 状态代码：404

## ValidationError

输入无法满足 AWS 服务指定的约束。

HTTP 状态代码 : 400

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。