

Cosmos DB API	Cosmos DB Docs
Key Vault API	Key Vault API
Stream Analytics API	Stream Analytics Docs
Time Series Insights API	Time Series Insights Docs

[illegible]

--

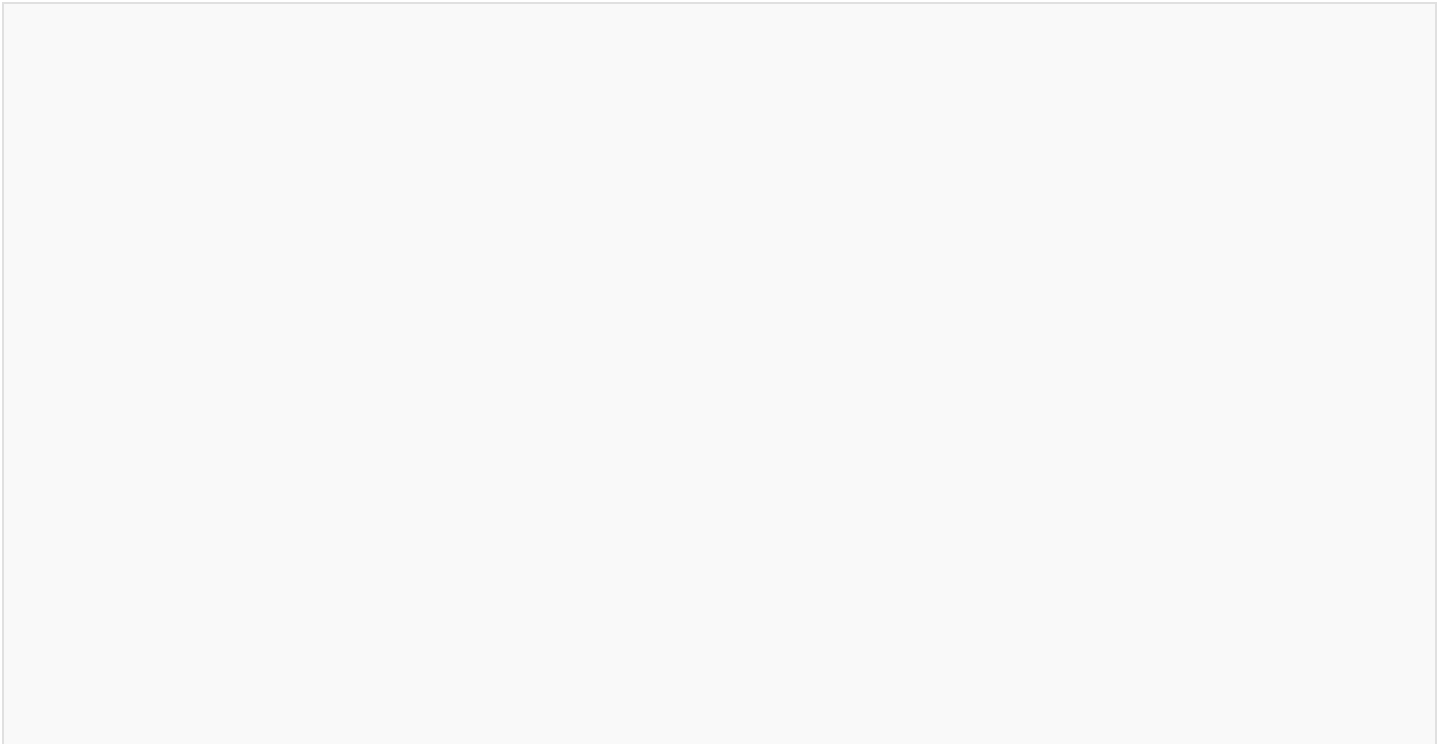
--

--

--

+) i > i. # 2 ' # +) 3 \ > \ \ ' BB' \ +


```
{
  title: 'tt
```



```
      "Val ue": "Yes"  
    }  
  ],  
  "Supporto
```

Authorization URL: <https://login.microsoftonline.com/orsitestab2connmicrosoft.com>

{
" { M

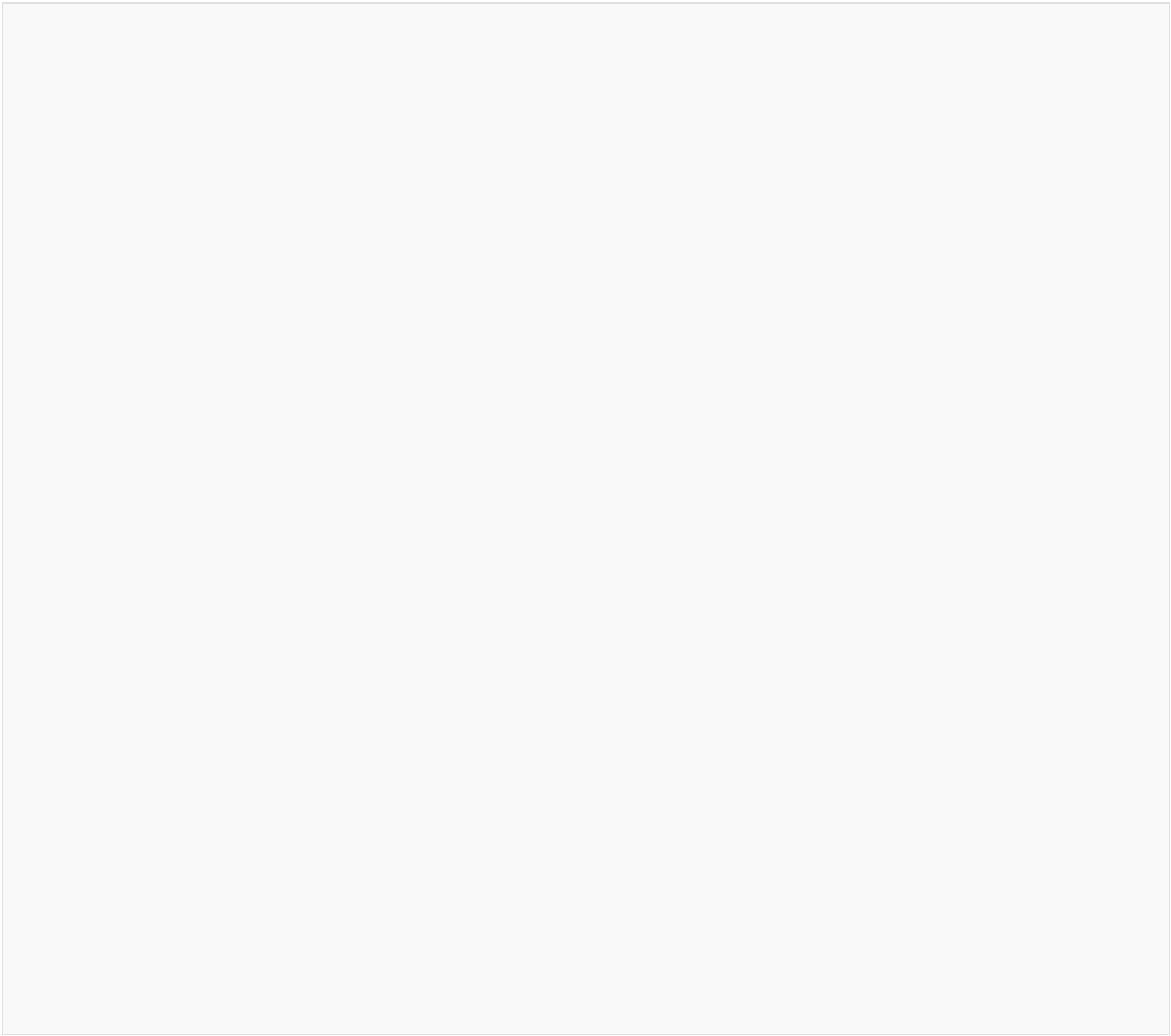
{ " | " k k {

1000


```
{  
  "id": "string"  
}
```



```
{  
  "subject":
```

```
{
  "$metadata": {
    "$modified": "11/02/2020 14: 07: 39 +00 00",
    "$uri": "ruri": ruri": ê4: 07: 39
```


Sample Response

```
{
  "Name": "Config",
  "Status": {
    "IsHealthy": true,
    "H
```


TYPE	NAME	DESCRIPTION
System.String	version	
List<System.String>	tags	
IFormFile	package	

Returns

TYPE	DESCRIPTION
System.Threading.Tasks.Task<PackageApiModel>	

RemoveTagAsync(String, String)

Declaration

```
public Task<ActionResult<PackageApiModel>> RemoveTagAsync(string id, string tag)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	id	
System.String	tag	

Returns

TYPE	DESCRIPTION
System.Threading.Tasks.Task<ActionResult<PackageApiModel>>	

RemoveTagsAsync(String)

Declaration

```
public Task<ActionResult<PackageApiModel>> RemoveTagsAsync(string id)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	id	

Returns

TYPE	DESCRIPTION
System.Threading.Tasks.Task<ActionResult<PackageApiModel>>	

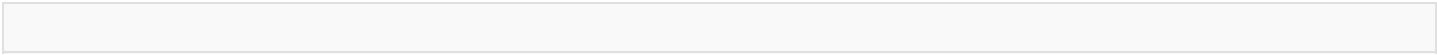
UploadFileAsync(IFormFile)

Declaration

```
public Task<UploadFileServiceModel> UploadFileAsync(IFormFile uploadedFile)
```

Parameters


```
``json
{
  "Items": [``json
```


```
{  
  "Items": [  
    {  
      "C
```



```
{  
  "Items": [  
    {  
      "C
```



```
{  
  "Items"
```



```
{
  "items": [
    {
      "name": "sample",
      "blobName": "sample",
      "uploadedBy": "sampleDevice",
      "uploadedOn": "date",
      "size": 0
    }
  ]
}
```

Security

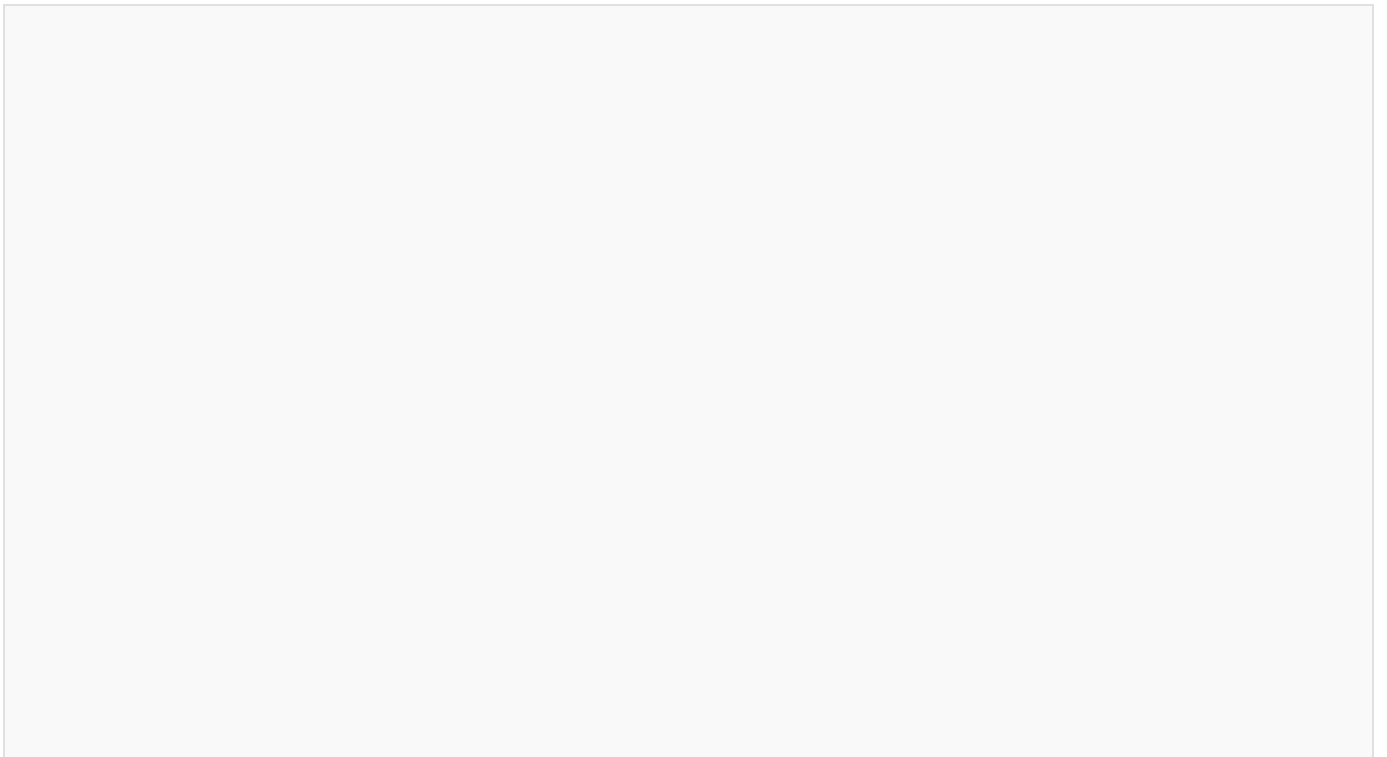
Type: OpenID Connect

Flow: implicit

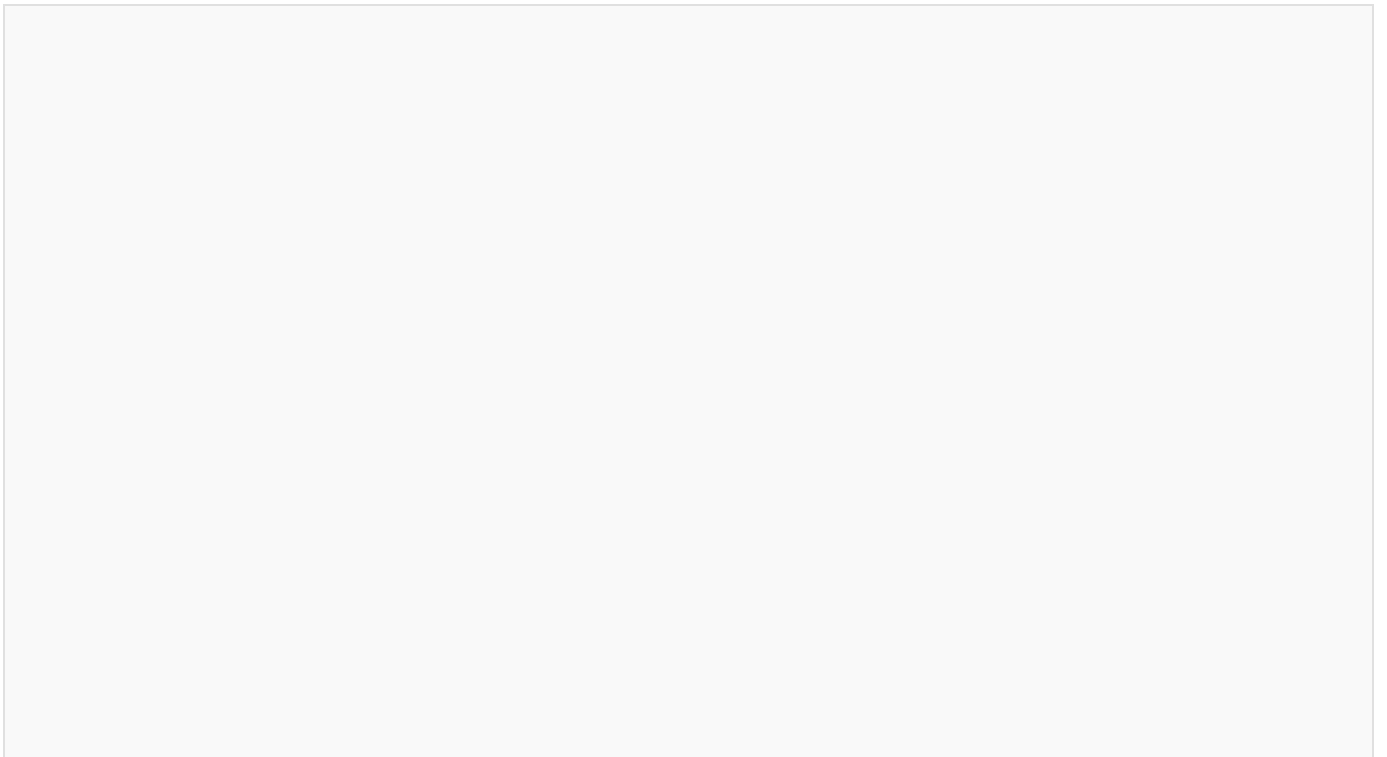
Authorization UR: [i](#) F



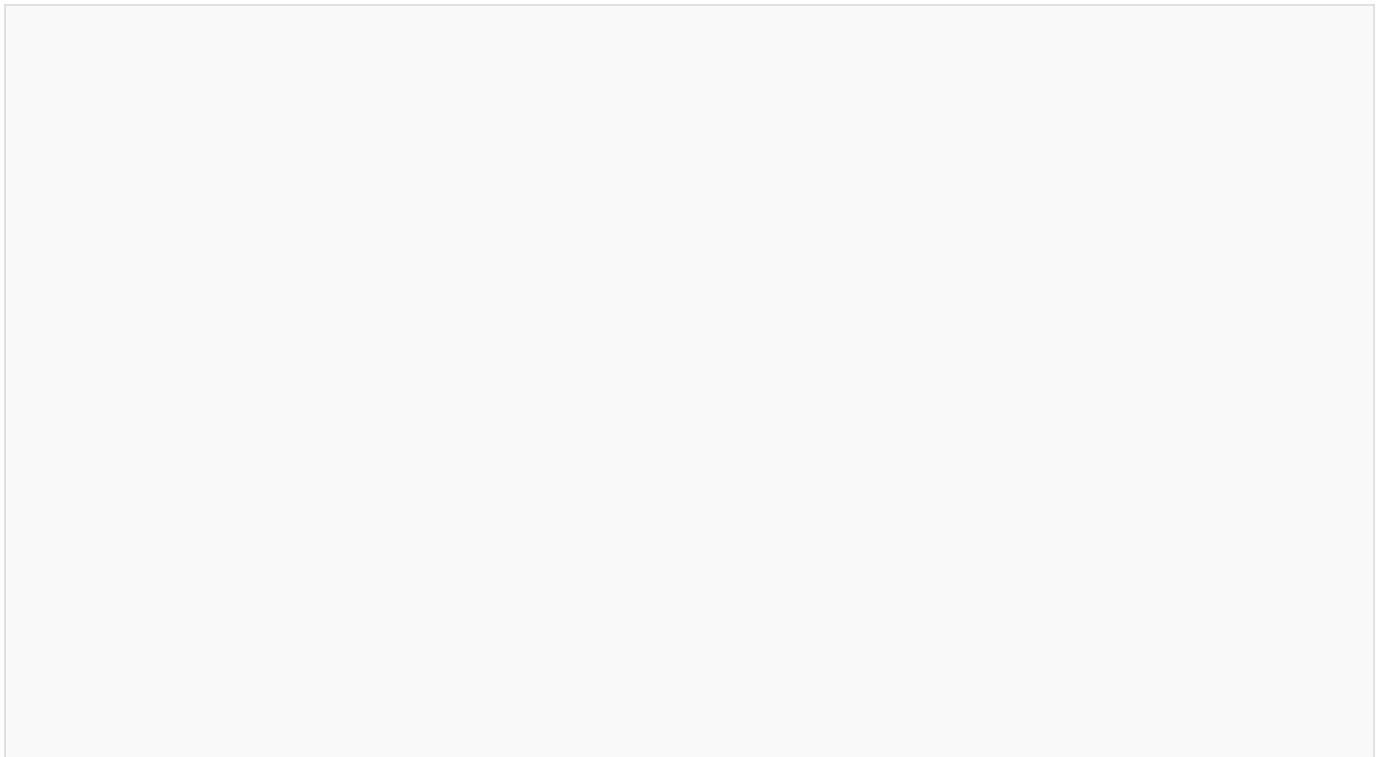

```
{
  "items": [
    {
      r l
```




```
{
  "items": [
    {
      r l
```




```
{  
  "iteme
```




```
{
  "eTag": "string",
  "id": "string",
  "name": "string",
  "dateCreated": "string",
  "dateModified": "string",
  "enabled": "string",
}
```



```
{
  "eTag": "string",
  "id": "string",
  "name": "string",
  "dateCreated": "string",
  "dateModified": "string",
  "enabled": "string",
}
```



```
{
  "items": [
    {
      "ed      u      u uu      u
```

```
{  
  "Items": [  
    {  
      "E"
```



```
{
  "eTag": "string",
  "id": "string",
  "name": "string",
  "dateCreated": "string",
  "dateModified": "string",
  "enabled": "string",
}
```

--



ConfigType	Type which specifies the type of configuration	Yes, If Padka

--	--	--

{

"Devi ceGrG

```
{  
  "deploymentId": "string",  
  "name": "string",  
  "deployment": {}  
}
```

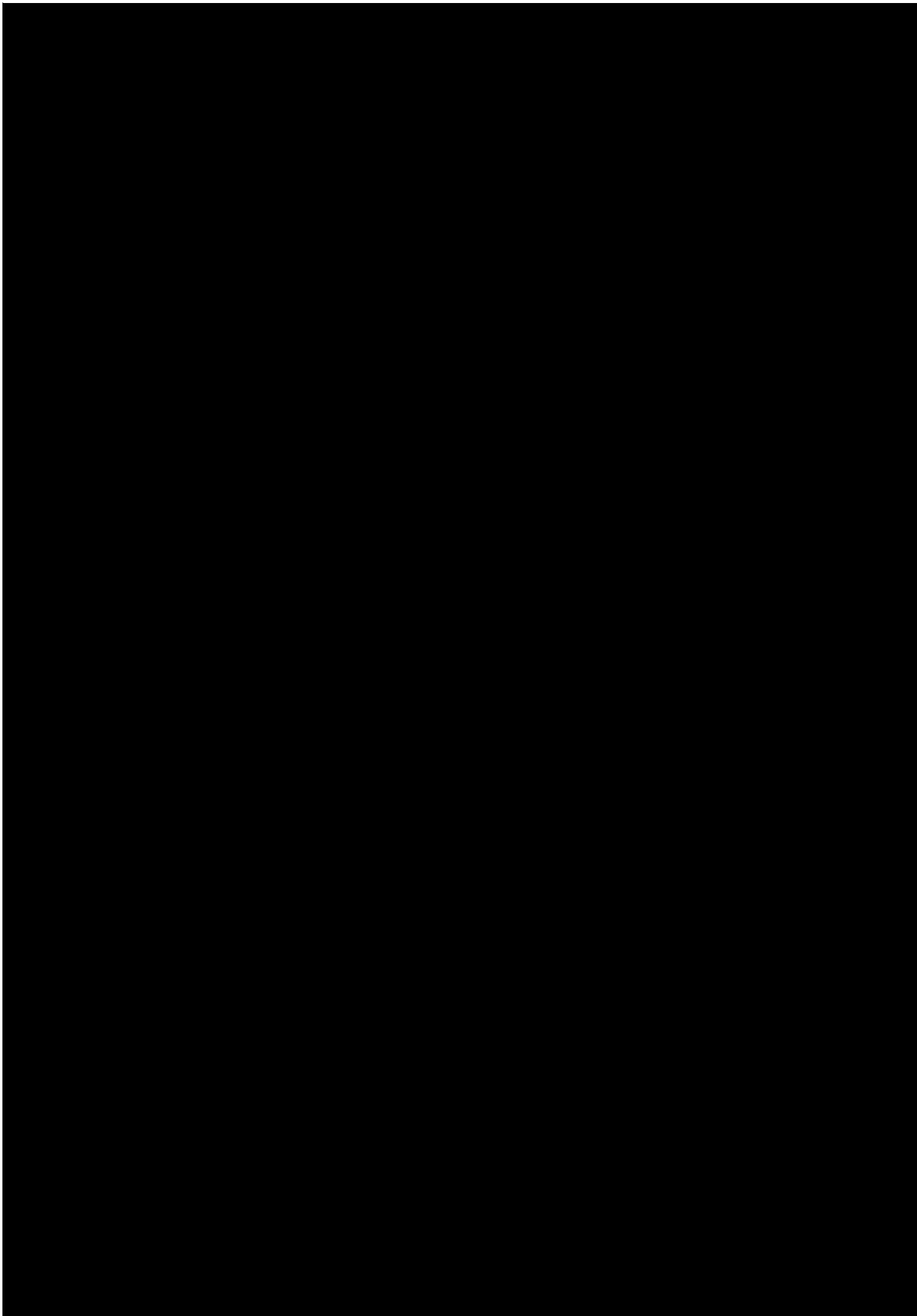

{

"Devi ceGr

PackageType : DeviceConfiguration ,

"ConfigType": "Firmware",

"rii"MiniCPFWare"



Reactivate Deployment

Creates a Device Configuration using the data froFing t F ing ti Conffing

Response

```
{
  "Metadata": {
    "$type": "Devicelist"
  }
}
```

```
{
  "$metadata": {
    "$type": "DeviceList;1",
    "$uri": "/v1/devices"
  },
  "ContinuationToken": null,
  "I
```


{
"ne"e"e"e"


```
{  
  "Metadata": {}  
}
```

```
{  
  "$metadata": {  
    "$type": "Devicelist; H
```



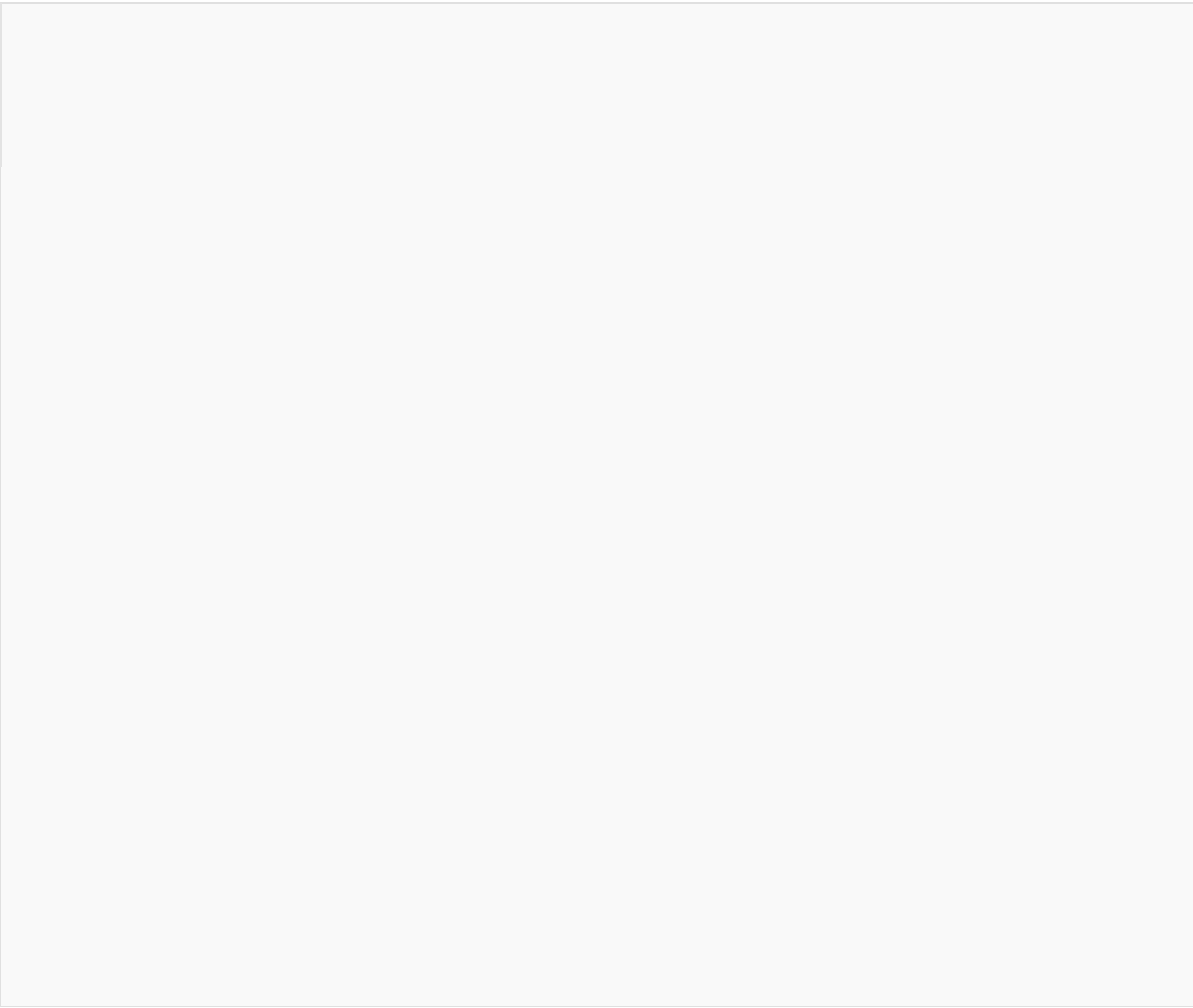
```
{
  "metadata": {
    "additional Prop1": "string",
    "additional Prop2": "string",
    "additional Prop3": "string"
  },
  "continuationToken": "string",
  "items": [
    {
      "eTag": "string",
      "id": "string",
      "c2dMessageCount": 0,
      "lastActivity": "2020-10-30T04:20:41.681Z",
      "connectorId": "2020-10-30T04:20:41.681Z",
    }
  ]
}
```



```
{
  "$metadata": {
    "$type": "DeviceList;1",
    "$uri": "/v1/devices"
  },
  "Content":
```



```
{  
  "metadata": {  
    "additional Prop1": "string",ting",tin
```



--


```
{
  "$metadata": {
    "$type": "DeviceList;1",
    "$uri": "/v1/device {
```



```
{
  "tag": "string",
  "id": "string",
  "base64": "c3R5bnQ="
}
```


--

$$\{ \text{"neee" e"} \}$$


```
{
  "eTag": "string",
  "id": "string",
  "c2dmMessageCount": 0,
  "lastActivity": "2020-10-30T05:41:54.391Z",
  "constStage": true,
  "enabled": true,
  "lastSelected": true,
}
```


Response Schema

```
{
  "eTag": "string",
  "id": "string",
  "c2dmMessageCount": 0,
  "lastActivity": "2020-10-30T05:41:54.391Z",
  "connected": true,
  "ena
```



```
{
  "ETag": "\"NDE1M1Q3OTM\\\" | AAAAAAAAAAE=",
  "Id": "M1TestDevice",
  "MessageCount": 0,
  "Lad": "k"
```


Response Schema

```
{
  "eTag": "string",
  "id": "string",
  "c2dmMessageCount": 0,
  "lastU
```



```
"tags": {  
  "addi u
```


{
"ne" e"

{
"ne" e"

```
{"/devices": {  
  "metadata": {  
    "type": "DeviceList;1",  
    "uri": "/v1/devices"  
  },  
}
```

```
{
  "$metadata": {
    "$type": "DeviceList;1",
    "$uri": "/v1/devices"
  },
  "ContinuationToken": null,
  "Items": [
    {
      "Reported": {},
      "Desired": {
        "schemaVersion": "ZZZZZZZZZZ",
        "suki": "fed": {}
      }
    }
  ]
}
```


Response


```
{  
  "Name": "Tenant Manager",  
  "Status": {  
    "IsHe nager",
```




Sample Response

```
{
  "Name": "ASA Manager",
  "Status": {
    "IsHe
```


Type:

