

## Web API request

# Handles

Returns a JSON object with all user-created handles in the current show.

<b>API</b>	Not in the online API documentation but in the <a href="#">Web API manual</a> .
<b>namespace</b>	
<b>usage</b>	<p>http://[ip]:4430/titan/handles</p> <p>can be extended by a handle group to show only these handles, e.g. http://[ip]:4430/titan/handles/Fixtures</p> <p>can be further extended by a page number of this handle group, e.g. http://[ip]:4430/titan/handles/Fixtures/1</p>
<b>example</b>	<a href="http://localhost:4430/titan/handles">http://localhost:4430/titan/handles</a> (link opens in new tab)
<b>action</b>	none
<b>returns</b>	JSON object with all user-created handles.
<b>remarks</b>	Lists only user-created handles. Some handle groups are not shown, e.g. any handles on the Mobile Wing.

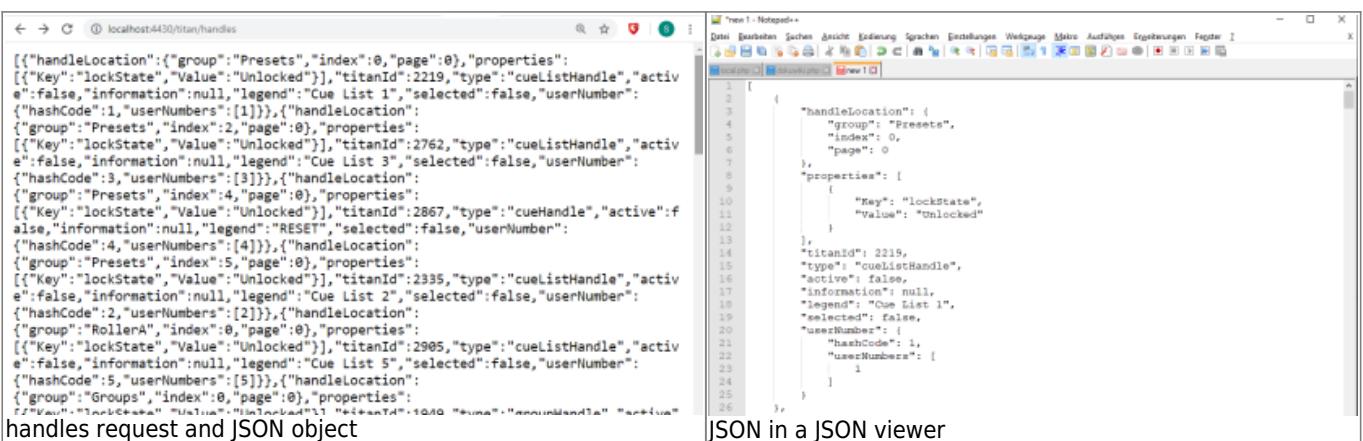
In order to view the JSON I recommend [Notepad++](#) with the JSON Viewer extension, but there are countless viewers available: online as well as standalone programs or for your preferred development platform..

In order to parse the JSON you might use e.g. `JSON.parse`, see [Titan IDs](#).

Some interesting fields:

- handleLocation: the location of the handle, e.g. which window, page, and index
- userNumber: the usernumber
- legend: the legend (if it is a string)
- active: whether a playback is active, i.e. fired (see [this facebook post](#))

## Screenshots



The screenshot shows two windows side-by-side. On the left is a web browser window titled 'localhost:4430/titan/handles' displaying the raw JSON response. On the right is a Notepad++ window titled 'HandleList.json' showing the same JSON data with syntax highlighting applied, making the code easier to read.

```

[{"handleLocation": {"group": "Presets", "index": 0, "page": 0}, "properties": [{"Key": "lockState", "Value": "Unlocked"}], "titanId": 2219, "type": "cueListHandle", "active": false, "information": null, "legend": "Cue List 1", "selected": false, "userNumber": {"hashCode": 1, "userNumbers": [1]}}, {"handleLocation": {"group": "Presets", "index": 12, "page": 0}, "properties": [{"Key": "lockState", "Value": "Unlocked"}], "titanId": 2762, "type": "cueListHandle", "active": false, "information": null, "legend": "Cue List 3", "selected": false, "userNumber": {"hashCode": 3, "userNumbers": [3]}}, {"handleLocation": {"group": "Presets", "index": 14, "page": 0}, "properties": [{"Key": "lockState", "Value": "Unlocked"}], "titanId": 2867, "type": "cueHandle", "active": false, "information": null, "legend": "RESET", "selected": false, "userNumber": {"hashCode": 4, "userNumbers": [4]}}, {"handleLocation": {"group": "Presets", "index": 15, "page": 0}, "properties": [{"Key": "lockState", "Value": "Unlocked"}], "titanId": 2335, "type": "cueListHandle", "active": false, "information": null, "legend": "Cue List 2", "selected": false, "userNumber": {"hashCode": 2, "userNumbers": [2]}}, {"handleLocation": {"group": "RollerA", "index": 0, "page": 0}, "properties": [{"Key": "lockState", "Value": "Unlocked"}], "titanId": 2905, "type": "cueListHandle", "active": false, "information": null, "legend": "Cue List 5", "selected": false, "userNumber": {"hashCode": 5, "userNumbers": [5]}}, {"handleLocation": {"group": "Groups", "index": 0, "page": 0}, "properties": [{"titanId": "TITANID", "userNumber": "1", "label": "Main Handle", "active": true}]}

```

handles request and JSON object

JSON in a JSON viewer

## Parsing the JSON (snippet/example)

```
...
// parse the JSON

var myArr = JSON.parse(this.responseText);
var field_un = "userNumber";
var field_le = "legend";
if (version < 12) {
    field_un = "UserNumber";
    field_le = "Legend";
}

// write number of handles somewhere

document.getElementById("num_handles").innerHTML = myArr.length;
...

// loop through the array and write its contents

for (var i = 0; i < myArr.length; i++) {

    // set the background colour based on the handle type
    bgcol = "#FFFFFF";
    if (myArr[i]["type"] == 'masterHandle') bgcol = "#FFdddd";
    ...

    // get handle usernumber
    handleUn = myArr[i][field_un].userNumbers;
    if (handleUn == null) handleUn = "";

    ...

    // get type and titanId and write it somewhere

    txt += "<td>" + handleUn + "</td>";
    txt += "<td>" + myArr[i]["type"] + "</td>";
    txt += "<td>" + myArr[i]["titanId"] + "</td>";

}
}
```

## Used In

- [Playback Masters](#)
- [Playback Timestamp](#)
- [Titan IDs](#)
- [Web API Demo](#)
- [Web-API requests](#)

From:

<https://www.avosupport.de/wiki/> - AVOSUPPORT



Permanent link:

<https://www.avosupport.de/wiki/webapi/requests/handles>

Last update: **2020/04/19 15:53**