

Web API request

Handles

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

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

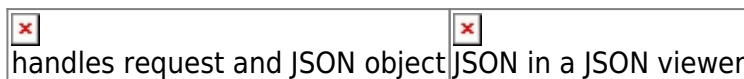
In order to view the JSON I recommend [Notepad++](#) with the JSON Viewer extension, but there are countless viewers available: online as well as standalone programs or for your preferred development platform..

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 user number
- `legend`: the legend (if it is a string)
- `active`: whether a playback is active, i.e. fired (see [this facebook post](#))

Screenshots



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://avosupport.de/wiki/> - **AVOSUPPORT**

Permanent link:
<https://avosupport.de/wiki/webapi/requests/handles?rev=1587310071>

Last update: **2020/04/19 15:27**

