

sgapiList(%%surveyID, %%page, %%countperpage, %%filterstring)

Important Update to Custom Scripting

The CustomScript Action now supports the LUA programming language. Visit our [NEW Lua Scripting Resources!](#)

1. New accounts (created after October 29, 2018) will only have the option to use Lua in scripts.
2. As of October 29, 2018 Custom Scripting Actions will default to Lua as the scripting type in the Custom Scripting Action for accounts created before this date. You will be able to switch to the Legacy Custom Scripting; though we highly encourage using Lua.
3. In the long term, Legacy Custom Scripting Actions will be switched to read-only. The exact date on this is to be determined; we will send notifications well ahead of time.

This function returns the list of responses for the given survey. This is equivalent to a get list on the SurveyResponse object via the RestAPI. To learn more visit the [SurveyResponse API Object documentation](#).

Parameters

- **%%page** - the page of the results to return (default is 1)
- **%%countperpage** - the number of results per page to return (default is 50)
- **%%filterstring** - the filter to apply to the results returned (default is null)

To learn more about paging results and filtering visit our [Filtering documentation for the Rest API](#).

Example

In this example there are two surveys: Survey A and Survey B. Both surveys request the student ID number of the respondent. Survey B uses this function to check if the entered survey ID has been previously recorded on survey A.

See it in action (make sure to leave enough time for your first response to process):

- [Example Survey A](#)
- [Example Survey B](#)

```

//this script is located at the top of page 2 on survey B

%%id = 2; //2 is the id of the unique field
%%student = sgapiGetValue(%%id); //defines student as the ID

%%surveyA = 646222;

//check if the student completed survey A
%%status = sgapiList(%%surveyA,1,1,'&filter[field][0]=[question(2)]&filter[operator][0]=&filter[value][0]=' . %%student)
;

//set flag for whether or not Survey A has been completed
%%flag = 3; //3 is the ID of the completed question

if(sgapiCount(%%status['data']) == 0){
sgapiSetValue(%%flag,'No');
}else{
sgapiSetValue(%%flag,'Yes');
}

```

A Quirk You May Run Across When Using sgapiList

In Custom Scripting, if you use the sgapiList function and then try to "var dump" using sgapiPrint_R the resulting array, you'll end up with a bunch of strange errors that make it difficult to understand what's going on:

```

[responseID] => 1
[[question merge codes require an id="question id" parameter]] => example
[[question merge codes require an id="question id" parameter]] => ...
[[variable is not a valid merge code function]] => ...
etc.

```

What's happening is, these key names are identical to normal API output, like "[question(2)]" and so on. When displayed in a survey, our software thinks they are merge codes, and so it tries to process them as such! This of course fails (that's not a proper merge code), and so you get errors instead of your key names.

There are 2 options to address this:

- 1) Use the API itself to see what key names are available.
- 2) Or "sanitize" the output so it doesn't contain something our software will think is a merge code.

Example code:

```
// Get data
%%responses = sgapiList(1234567);
%%data = %%responses['data'];
%%record = %%data[1];

// Munge key names, stripping square brackets and
// replacing them with HTML entities as necessary
foreach (%%record as %%recordKey => %%recordValue) {
%%pre = %%post = "";
// If '[' is the first character, strip the first and the last character
if (0 === sgapiStrPos(%%recordKey, '[')) {
%%recordKey = sgapiSubStr(%%recordKey, 1, -1);
%%pre = '&l' . 'sqb;';
%%post = '&r' . 'sqb;';
}
%%output .= "Key: " . %%pre . %%recordKey . %%post . "
";
}
}
```

- Author: Zach Steveson
- Share This Article:
- Last updated: 10/11/2020 2:10 pm EDT