

Show a Random Set of Text/Instruction Fields

Last Modified on 07/15/2016 4:44pm EDT

You may have noticed that when setting up your survey page to [randomly show a set of questions](#), text/instruction fields are not included! But of course, there's always a way to hack it with some Javascript, so here we go!

Check it out in an [example survey](#)!

OR

[Add a survey](#) with this setup to your account!

Necessary Components

You can add an other textbox to your row headers for the following table questions:

- [Text/Instructions](#) fields
- [JavaScript Action](#)

Setup

If you have multiple text/instruction fields and need to randomly show some specific number of those to the respondent, just add the following script to a JavaScript action on the same page!

```

$(document).ready(function() {
  var numToShow = 2; //Change this number to how many fields we need to show

  /*-----Don't touch anything below this line-----*/
  var textEls = $('.sg-type-instruction');
  var randIndices = [];

  for (var i = 0; i < numToShow; i++) {
    var randIndex = Math.floor(Math.random() * textEls.length);
    do {
      randIndex = Math.floor(Math.random() * textEls.length);
    } while (randIndices.indexOf(randIndex) !== -1);

    randIndices.push(randIndex);
  }
  //console.log(randIndices);
  textEls.each(function(i, el) {
    if (randIndices.indexOf(i) === -1) {
      el.remove();
      //console.log('set for removal: #' + i);
    }
  });
});

```

Now all you have to do is specify the number of elements you want to show in the line up at the top, here:

```

var numToShow = 2; //Change this number to how many fields we need to show

```

In this case 2 random text/instructions fields will be shown to the respondent.

Record which text element was shown

If you'd like to also know which element was shown to the respondent, use the following Javascript instead.

```

$(document).ready(function() {
    var numToShow = 1; //Change this number to how many fields we need to show

    /*-----Don't touch anything below this line-----*/
    var textEls = $('.sg-type-instruction');
    var randIndices = [];
    var shownArr = [];
    var shownStr;

    for (var i = 0; i < numToShow; i++) {
        var randIndex = Math.floor(Math.random() * textEls.length);
        do {
            randIndex = Math.floor(Math.random() * textEls.length);
        } while (randIndices.indexOf(randIndex) !== -1);

        randIndices.push(randIndex);
    }
    //console.log(randIndices);
    textEls.each(function(i, el) {
        if (randIndices.indexOf(i) === -1) {
            $(this).remove();
            //console.log('set for removal: #' + i);
        } else {
            var classArr = $(this).attr('class').split(' ');
            //Fix for Preview
            if (classArr.indexOf('sg-cc-hook') !== -1) classArr.splice(classArr.indexOf('sg-cc-hook'), 1);
            var shownClass = classArr[classArr.length - 1];
            shownArr.push(shownClass);
        }
    });
    shownStr = shownArr.join(',');
    $('.shown').find('input[type="text"]').val(shownStr);
});

```

In this case you will also need to add:

- A unique [CSS Class Name](#) to each text element (this is what will show up in your reporting).
- A [textbox question](#) with a CSS Class Name of `sg-hide shown`.

`sg-hide` will hide the question on the page, and `shown` will be used by the script so we can write to it. What we're actually doing is printing a comma-separated list of the custom classes of the text

elements that were shown to the respondent.

In my example survey we're writing the present duck to the textbox and then using a merge code on the last page to reveal the duck to the respondent.

Scripting and Other Out-of-the-Box Customizations

We're always happy to help you debug any documented script. That said, we do not have the resources to write scripts on demand.

If you have customization ideas that you haven't figured out how to tackle, we're happy to be a sounding board for SurveyGizmo features and functionality ideas that might meet your customization. Beyond this, you might want to consult with someone on our [Programming Services Team](#); these folks might have the scripting chops to help you to achieve what you are looking for!

@plans @pro @ent
