

# Custom Question Builder

## Important Update to Custom Scripting

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

### Legacy Custom Scripting Language Deprecation Plans

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.

Available on these licenses: [Full Access](#)

Under **Account > Libraries > Custom Questions** we provide a framework to build your own survey question from scratch. Below we'll cover a fairly basic custom question setup.

Before we get started, you should know that we created custom questions for advanced programmers to choose their own adventure within SurveyGizmo. Part of that adventure is support, as custom questions are beyond the scope of our support team.

In this tutorial we will cover the Net Promoter Score (NPS) question that we originally built as a custom question before adding it to the core application.

# About Tab

On the **About** tab enter the descriptive details of your custom question. You'll want to make sure you specify, at the very least, a **Question Type Name**, **Status** and **Description**.

Note: If the status of your Custom Question is **In Design**, the question cannot be added to a survey for testing purposes. If you need to test your question, set the status to **Active**.

There are also several fields for optional Image URLs. The Icon URL will display on the Build tab of your survey.

# Layout Tab

On the **Layout** tab you'll enter the HTML for your question. Below is the HTML for the NPS custom question. Note, you'll need to use SurveyGizmo the following elements within your HTML:

- SurveyGizmo Classes, e.g. sg-question-title, sg-question-number, etc.
- Custom question merge codes, e.g., %elementid%. [See Available Merge Codes](#).
- SurveyGizmo merge codes e.g., [question("number"), id="%elementid%"]  
[question("title"), id="%elementid%", displaytitle="true"]  
[question("required"), id="%elementid%"].

```
[question("number"), id="%elementid%"] [question("title"),  
id="%elementid%", displaytitle="true"] [question("required"), id=  
"%elementid%"]
```

## Custom Question Merge Codes

- **%elementid%** - the question or element id that will be dynamically populated when your question is created.
- **%pageid%** - the page id that will be dynamically populated when your question is created.
- **%surveyid%** - the survey id that will be dynamically populated when your survey and question is created.

## Script Tab

The Script tab should include the Javascript to run onload and onsubmit. Below is the Javascript from the Script tab for the NPS custom question. Note, you'll

need to use custom script functions, e.g., `getCustomProperty` and `getStorageValue`.

```
{
  array_flip: function(trans)
  {
    var count = 10;
    var key, tmp_ar = [];
    for (key in trans)
    {
      if (trans.hasOwnProperty(key))
      {
        tmp_ar[count] = trans[key];
        count--;
      }
    }
    return tmp_ar;
  },
  //onload - show the previous response (if any)
  onload: function()
  {
    //Rearrange order
    var labels = [];
    labels.push(
      '
      • Extremely Likely10
      ',
      '
      • 9
      ', '
      • 8
      ', '
      • 7
      ', '
      • 6
      ', '
      • 5
      ',
      '
      • 4
      ', '
      • 3
      ', '
      • 2
      ', '
      • 1
      ',
      '
      '
    );
  }
}
```

• Extremely Unlikely0

');

```
var values = [];
```

```
values.push(
```

```
'
```

•

','

```
'
```

•

','

```
'
```

•

','

```
'
```

•

','

```
'
```

•

','

```
'
```

•

','

```
'
```

•

','

```
'
```

•

','

```
'
```

•

','

```
'
```

•

','

```
'
```

•

');

```
if (this.getCustomProperty("order") == 'Ascending')
```

```
{
```

```
    //flip array
```

```
    labels = this.array_flip(labels);
```

```
    values = this.array_flip(values);
```

```
}
```

```
$("#nps-labels-%elementid%").html(labels.join(''));
```

```
$("#nps-values-%elementid%").html(values.join(''));
```

```
//Ability to set labels
```

```
var llabel = this.getCustomProperty("l_label");
```

```
var rlabel = this.getCustomProperty("r_label");
```

```
if (llabel != null && llabel != '')
```

```

    {
        $("#l1label-%elementid%").html(l1label);
    }
    if (rlabel != null && rlabel != '')
    {
        $("#r1label-%elementid%").html(rlabel);
    }
    if (!window.isEditor)
    {
        var value = this.getStorageValue();
        if (value != '')
        {
            this.getElement("nps-%elementid%" + value)[0].checked
= true;
        }
    }
},
//Store the value from the custom question into this survey element
onsubmit: function()
{
    var value = $("input[name=nps-%elementid%]:checked").val();
    this.setStorageValue(value);
}
}

```

## CSS/JS Tab

The CSS/JS tab should include CSS and any Client Side Javascript you wish to run in survey taking. Below is the CSS from the CSS/JS tab for the NPS custom question.

```
.netpromoterscore {
  display: inline-block;
  overflow:auto;
  margin-bottom:0px;
}
.netpromoterscore li {
  width:35px;
  float: left;
  display:block;
  text-align:center;
  padding: 3px;
}
.npsvalues li {
  width:35px;
  float: left;
  display:block;
  text-align:center;
  padding: 3px;
  margin-top:0px;
}
.netpromoterscore li span {
  display: block;
  margin-left: -7px;
  margin-top: -30px;
  position: absolute;
  text-align: center;
  width: 50px;
}
.nps-options {
  padding-top:35px;
}
```

## The Script Tab Vs The CSS/JS Tab

Both the JavaScript you place in the Question Script field as well as in the Client Side Javascript field will be loaded with your question in a survey, in the same