

LimeSurvey-Tutorial

Matrizen

August 2021 – November 2023

Inhaltsverzeichnis

Vorbemerkung	4
1. Matrix(Texte) mit DropDown bzw. DatePicker-Spalten	5
1.1. Matrix(Texte) mit DropDown-Spalten.....	5
1.1.1. Anwendungsbeispiel.....	5
1.1.2. Implementierung.....	6
1.2. Matrix(Texte) mit DatePicker und abhängigen Spalten.....	8
1.2.1. Anwendungsbeispiel.....	8
1.2.2. Implementierung.....	8
2. Verschiedene Spaltenbreiten	11
2.1.1. Anwendungsbeispiel.....	11
2.1.2. Implementierung.....	11
3. Matrizen mit „Sonstigen“-Zeilen	13
3.1. Matrix (Texte/Zahlen).....	13
3.1.1. Anwendungsbeispiel.....	13
3.1.2. Implementierung.....	14
3.2. „Normale“ Matrix.....	16
3.2.1. Anwendungsbeispiel.....	16
3.2.2. Implementierung.....	16
3.2.3. Was spricht dagegen?.....	19
4. Zusätzliche Spalten in Matrizen	20
4.1. Matrizen mit „Kommentar“-Spalten.....	20
4.1.1. Anwendungsbeispiel (Kommentar hinter Matrix).....	20
4.1.2. Implementierung.....	20
4.1.3. Anwendungsbeispiel (Kommentar/e vor Matrix).....	22
4.1.4. Implementierung.....	22
4.1.5. Was spricht dagegen?.....	23
4.2. Matrizen mit „Sonstige“-Spalten.....	24
4.2.1. Anwendungsbeispiel.....	24
4.2.2. Implementierung.....	24
4.2.3. Was spricht dagegen?.....	25
5. Matrizen mit zusätzlichen Kopfzeilen und Zwischenzeilen	26
5.1. Matrix(Zahlen).....	26
5.1.1. Anwendungsbeispiel.....	26
5.1.2. Implementierung.....	26
5.2. Matrix.....	28
6. „Dynamische“ Anzeige der Zeilen einer Matrix	29
6.1.1. Anwendungsbeispiel.....	29
6.1.2. Implementierung.....	31
7. Matrix(Texte) mit senkrechten Radio-Buttons in Zellen	34
7.1.1. Anwendungsbeispiel.....	34
7.1.2. Implementierung.....	34
7.1.3. Was spricht dagegen?.....	36
8. Matrix(Zahlen) im Checkbox-Layout mit letzter Spalte exklusiv	37
8.1.1. Anwendungsbeispiel.....	37
8.1.2. Implementierung.....	37
9. Matrix(Texte) mit Checkbox-Layout, Sonstigen und Exklusiven Optionen	39

9.1.1. Anwendungsbeispiel.....	39
9.1.2. Implementierung.....	39
10. Matrizen mit randomisierten Spalten.....	42
10.1.1. Anwendungsbeispiel.....	42
10.1.2. Implementierung.....	43
11. Matrix nach Spalte als MaxDiff.....	45
11.1.1. Anwendungsbeispiel.....	45
11.1.2. Implementierung.....	45
12. Semantisches Differential mit „KA“.....	47
12.1.1. Anwendungsbeispiel.....	47
12.1.2. Implementierung.....	48
13. Matrizen mit ausgeblendeten Zellen.....	50
13.1. Matrix.....	50
13.1.1. Anwendungsbeispiel.....	50
13.1.2. Implementierung.....	51
13.2. Matrix(Zahlen) mit Checkbox-Layout.....	52
13.2.1. Anwendungsbeispiel.....	52
13.2.2. Implementierung.....	52
13.3. Matrix(Texte).....	53
13.3.1. Anwendungsbeispiel.....	53
13.3.2. Implementierung.....	54
14. Matrix(Zahlen/Texte) mit Filterung der Spalten.....	55
14.1.1. Anwendungsbeispiel.....	55
14.1.2. Implementierung.....	56
15. Dual Matrix – Zweite Spalte gefiltert nach erster Spalte.....	59
15.1.1. Anwendungsbeispiel.....	59
15.1.2. Implementierung.....	61
16. Textausrichtung.....	65
16.1.1. Anwendungsbeispiel.....	65
16.1.2. Implementierung.....	66
17. Reset-Button in Matrix.....	67
17.1.1. Anwendungsbeispiel.....	67
17.1.2. Implementierung.....	69
18. Art „semantisches Differential“, aber Items in der Mitte.....	70
18.1.1. Anwendungsbeispiel.....	70
18.1.2. Implementierung.....	71
19. Anhang.....	73
19.1.1. Iss-Export einer Beispiel-Datei.....	73

Vorbemerkung

Diese kleinen Tutorials sollen Möglichkeiten aufzeigen, durch Einsatz von javascript und css Effekte bei der Darstellung von Fragen zu zeigen, die den Teilnehmern die Beantwortung erleichtert.

Obwohl einige der Beispiele auch mit plugins bzw. zu installierenden Frage-Designs realisierbar wären, habe ich bewusst darauf verzichtet.

Viele der studentischen Nutzer haben weder Rechte, plugins zu installieren noch am Theme Änderungen vorzunehmen.

Daher sind alle Beispiele alleine durch Einfügen von javascript bzw. css im Fragetext erstellt.

Der größte Teil dieser scripts stammt ursprünglich von Tony Partner, hier im Forum als „tpartner“ unterwegs. In einem weiteren Tutorial werde ich mir dann etwas untreu und einige seiner Fragenlayouts vorstellen.

Die dargestellten Codes sollten nicht aus diesem Text in LimeSurvey kopiert werden.

Es könnten sich noch Reste der Formatierung im Text befinden, der den Code dann unbrauchbar macht.

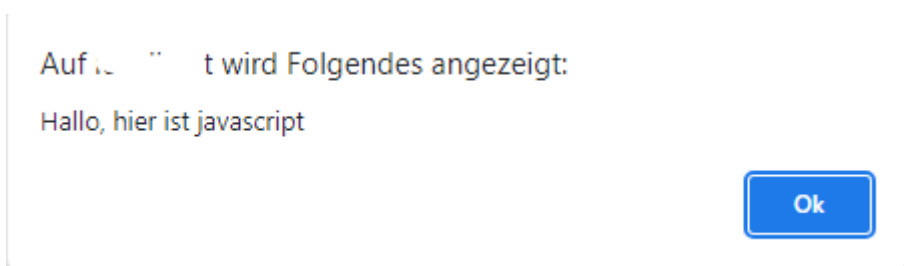
Daher bitte diese Codestücke immer aus der Beispielstudie übernehmen.

Da nahezu alle gezeigten Beispiele mit javascript realisiert werden, ist natürlich die Grundvoraussetzung, dass dieses eingesetzt werden kann.

Ein einfacher Test ist dies in den Quellcode der ersten Frage einzufügen

```
<script type="text/javascript" charset="utf-8">
$(document).on('ready pjax:scriptcomplete',function(){
    alert("Hallo, hier ist javascript");
});
</script>
```

Wenn dann dieses Fenster aufploppt, ist alle in Ordnung; wenn nicht, ...



Und dann noch dieses:

Solutions, code and workarounds presented in these text are given without any warranty, implied or otherwise.

Lösungen, Code und „Work-arounds“, die in diesem Text präsentiert werden, werden ohne jegliche stillschweigende oder sonstige Gewährleistung gegeben.

1. Matrix(Texte) mit DropDown bzw. Datepicker-Spalten

1.1. Matrix(Texte) mit DropDown-Spalten

1.1.1. Anwendungsbeispiel

Normalerweise werden in diesem Typ – wie der Name sagt – Texte eingegeben.

Aus Gründen einer leichteren Auswertbarkeit ist es aber oft angebracht, vordefinierte Antworten anzubieten.

Dazu ein Beispiel.

	Stadtteil	Sportart	Altersklasse	Bemerkungen
Adalbert Ameise	<input type="text" value="Osdorf"/>	<input type="text" value="Laufen"/>	<input type="text" value="B-Jugend"/>	<input type="text"/>
Berta Biber	<input type="text" value="Hmb-Barmbek"/>	<input type="text" value="Leichtathletik"/>	<input type="text" value="16-18"/>	<input type="text"/>
Claas Chamäleon	<input type="text" value="Barmbek-Nord"/>	<input type="text" value="Golf"/>	<input type="text" value="U18"/>	<input type="text"/>
Doris Dachs	<input type="text" value="Barmbek"/>	<input type="text" value="Schwimmen"/>	<input type="text"/>	<input type="text"/>

Man sieht, die Teilnehmer haben den Stadtteil in verschiedenen Schreibweisen eingetragen (auch mit Konfusion bzgl. Stadtbezirk und Stadtteil), bei der Sportart Ober- und Unterbegriffe gemischt, und auch bei der Altersklasse war nicht klar, wie der Eintrag erfolgen sollte.

Bei der Auswertung muss nun alles gelesen und umkodiert werden.

Daher bietet sich hier eine Drop-Down-Option an.

	Stadtteil	Sportart	Altersklasse	Bemerkungen
Adalbert Ameise	<input type="text" value="Bahrenfeld"/>	<input type="text" value="Bitte auswählen..."/>	<input type="text" value="Bitte auswählen..."/>	<input type="text"/>
Berta Biber	<input type="text" value="Bitte auswählen..."/>	<input type="text" value="Fußball"/>	<input type="text" value="U8 (2014 und jünger)"/>	<input type="text"/>
Claas Chamäleon	<input type="text" value="Bitte auswählen..."/>	<input type="text" value="Karate"/>	<input type="text" value="U10 (2012-2013)"/>	<input type="text"/>
Doris Dachs	<input type="text" value="Bitte auswählen..."/>	<input type="text" value="Leichtathletik"/>	<input type="text" value="U12 (2010-2011)"/>	<input type="text"/>

Stadtteil
Altona
Altona-Allstadt
Altona-Nord
Bahrenfeld
Blankenese
Bergedorf
Allermöhe
Altengamme
Billwerder

Sportart
Bitte auswählen...
Fußball
Handball
Karate
Leichtathletik
Schwimmen
Volleyball
Bitte auswählen...

Altersklasse
Bitte auswählen...
U8 (2014 und jünger)
U10 (2012-2013)
U12 (2010-2011)
U14 (2008-2009)
U16 (2006-2007)
U18 (2004-2005)
Bitte auswählen...

Damit erreicht man nun wohl definierte Antworten, die dadurch leicht auszuwerten sind.

Der Stadtteil hat als Oberbegriff den zugehörigen Stadtbezirk, die Sportarten enthalten festgelegte Begriffe (z.B. nur die Sportarten, die der Verein anbietet), die Altersklassen sind ebenfalls eindeutig.

1.1.2. Implementierung

Zunächst: Ich bevorzuge bei Matrizen folgende Codierung:

- Q1_Y001, Q1_Y002 für die y-Achse
- Q1_X001, Q1_X002 für die x-Achse

Daher tauchen im folgenden script die Bezeichnungen X001, X002, usw. für die verschiedenen Spalten auf.

Zur Implementierung wird in den Fragetext der Matrix(Texte) folgendes javascript eingefügt.
Javascript wird immer im Quellcode-Modus eingefügt.

```
<script type="text/javascript" charset="utf-8">
$(document).on('ready ajax:scriptcomplete',function(){
    var thisQuestion = $('#question{QID}');

    // Insert selects
    $('.answer-item.answer_cell_X001', thisQuestion).addClass('with-select').append('<select class="inserted-select form-control list-question-select">\
        <option value="">...</option>\
        <optgroup label="Altona">\
        <option value="1">Altona-Altstadt</option>\
        <option value="2">Altona-Nord</option>\
        <option value="3">Bahrenfeld</option>\
        <option value="4">Blankenese</option>\
        </optgroup>\
        <optgroup label="Bergedorf">\
        <option value="5">Allermöhe</option>\
        <option value="6">Altengamme</option>\
        <option value="7">Billwerder</option>\
        </optgroup>\
    </select>');
    $('.answer-item.answer_cell_X002', thisQuestion).addClass('with-select').append('<select class="inserted-select form-control list-question-select">\
        <option value="">...</option>\
        <option value="1">Fußball</option>\
        <option value="2">Handball</option>\
        <option value="3">Karate</option>\
        <option value="4">Leichtathletik</option>\
        <option value="5">Schwimmen</option>\
        <option value="6">Volleyball</option>\
    </select>');
    $('.answer-item.answer_cell_X003', thisQuestion).addClass('with-select').append('<select class="inserted-select form-control list-question-select">\
        <option value="">...</option>\
        <option value="1">U8 (2014 und jünger)</option>\
        <option value="2">U10 (2012-2013)</option>\
        <option value="3">U12 (2010-2011)</option>\
        <option value="4">U14 (2008-2009)</option>\
        <option value="5">U16 (2006-2007)</option>\
        <option value="6">U18 (2004-2005)</option>\
    </select>');
```

Um die Werte zu speichern, die in „value='x'“ eingetragen sind, füge man diesen Code ein

```
// Listeners
$('.inserted-select', thisQuestion).on('change', function(i) {
  if($(this).val() != '') {
    $(this).closest('.answer-item').find('input:text').val($('option:selected',
this).val()).trigger('change');
  }
  else {
    $(this).closest('.answer-item').find('input:text').val('').trigger('change');
  }
});
// Returning to page
$('.with-select input:text', thisQuestion).each(function(i) {
  var thisCell = $(this).closest('.answer-item');
  var inputText = $.trim($(this).val());
  $('select.inserted-select', thisCell).val(inputText);
});
```

Wenn man aber den Text speichern möchte, muss dieser Code genommen werden.

```
// Listeners on select elements
$('.inserted-select', thisQuestion).on('change', function(i) {
  if($(this).val() != '') {
    $(this).closest('.answer-item').find('input:text').val($.trim($('option:selected',
this).text())).trigger('keyup');
  }
  else {
    $(this).closest('.answer-item').find('input:text').val('').trigger('keyup');
  }
});
// Returning to page
$('.with-select input:text', thisQuestion).each(function(i) {
  var thisCell = $(this).closest('.answer-item');
  var inputText = $.trim($(this).val());
  var selectval = $('select.inserted-select option', thisCell).filter(function () { return $(this).html()
== inputText; }).val();
  $('select.inserted-select', thisCell).val(selectval);
});
```

Der Rest ist dann wieder für beide gleich

```
// Clean-up styles
$('select.inserted-select', thisQuestion).css({
  'max-width': '100%'
});
$('.with-select input:text', thisQuestion).css({
  'position': 'absolute',
  'left': '-9999em'
});
});
</script>
```

1.2. Matrix(Texte) mit Datepicker und abhängigen Spalten

1.2.1. Anwendungsbeispiel

Natürlich kann man statt eines einfachen Drop-Downs auch einen Datepicker in eine Matrix-Zelle einfügen. Zusätzlich wird hier gezeigt, wie man eine Spalte in Abhängigkeit von einer anderen anzeigt.

Q1b				
	Eintrittsdatum	Sportart	Altersklasse	Bemerkungen
Adalbert Ameise	29.06.2020 	Handball <input type="text"/>	U10 (2012-2013) <input type="text"/>	<input type="text"/>
Berta Biber	03.07.2019 	Karate <input type="text"/>	... <input type="text"/>	<input type="text"/>
Carlo Chinchilla	<input type="text"/>	... <input type="text"/>	... <input type="text"/>	<input type="text"/>
Doris Dachs	<input type="text"/>	... <input type="text"/>	... <input type="text"/>	<input type="text"/>

Hier sehen wir in der ersten Spalte den Datepicker und die dritte Spalte ist nur aktiv, wenn die Sportart „Fußball“ oder „Handball“ ist.

1.2.2. Implementierung

Zunächst der Datepicker. Am Anfang des Scripts werden einige Bibliotheken aus der LimeSurvey-Installation geladen.

```
<script type="text/javascript" charset="utf-8">
  $(document).on('ready pjax:scriptcomplete',function(){

    var rootPath = location.pathname.split('index.php')[0];

    // Identify this question
    var thisQuestion = $('#question{QID}');

    $('head').append('<link rel="stylesheet"
href="'+rootPath+'assets/packages/bootstrap/plugins/datetimerpicker/build/css/bootstrap-datetimerpicker.min.css"
type="text/css" />');

    $.getScript(rootPath+'assets/packages/bootstrap/plugins/datetimerpicker/build/js/bootstrap-
datetimerpicker.min.js')

    .done(function(script, textStatus) {

      // Insert the date-time-pickers
      $('.answer-item.answer_cell_X001 input:text', thisQuestion).each(function(i) {

        var thisID = $(this).attr('id');

        $(this).addClass('date-control date datetimerpicker')
```



```

.wrap('<div id="datetimepicker_'+thisID+'" class="input-group date date-timepicker-group" />')
.after('<div class="input-group-addon datetimepicker-addon btn btn-primary">\
    <i class="fa fa-calendar" aria-hidden="true"></i><span class="sr-only"></span>\
</div>');

$('#datetimepicker_'+thisID+').datetimepicker({
    useCurrent:true,
    allowInputToggle: true,
    format: 'DD.MM.YYYY'
});
});
})
.fail(function( jqxhr, settings, exception ) {
    console.log(exception);
});
});
</script>

<script type="text/javascript">
$(document).ready(function(){
    $(".datetimepicker").datetimepicker();
});
</script>

```

Das script für die Dropdowns ist ja bereits bekannt. Allerdings kommt hier der Teil hinzu, der die dritte Spalte auf „readonly“ setzt, wenn in der zweiten Spalte weder „Fußball“ noch „Handball“ gewählt wurde.

```

<script type="text/javascript" charset="utf-8">
$(document).on('ready ajax:scriptcomplete',function(){
    var thisQuestion = $('#question{QID}');

    $('#answer-item.answer_cell_X002', thisQuestion).addClass('with-select').append('<select class="inserted-select form-control list-question-select">\
        <option value="">...</option>\
        <option value="1">Fußball</option>\
        <option value="2">Handball</option>\
        <option value="3">Karate</option>\
        <option value="4">Leichtathletik</option>\
        <option value="5">Schwimmen</option>\
        <option value="6">Volleyball</option>\
    </select>');

    $('#answer-item.answer_cell_X003', thisQuestion).addClass('with-select').append('<select class="inserted-select form-control list-question-select">\
    <option value="">...</option>\
        <option value="1">U8 (2014 und jünger)</option>\
        <option value="2">U10 (2012-2013)</option>\
        <option value="3">U12 (2010-2011)</option>\
        <option value="4">U14 (2008-2009)</option>\
        <option value="5">U16 (2006-2007)</option>\
        <option value="6">U18 (2004-2005)</option>\
    </select>');

```

```

</select>');

// Listeners
$('.inserted-select', thisQuestion).on('change', function(i) {
  if($(this).val() != '') {
    $(this).closest('.answer-item').find('input:text').val($('option:selected',
this).val()).trigger('change');
  }
  else {
    $(this).closest('.answer-item').find('input:text').val('').trigger('change');
  }
}

```

Hier ist nun der neue Teil

```

// 3rd column conditional on 2nd column
if($(this).closest('.answer-item').hasClass('answer_cell_X002')) {
  handleColumnX003($(this));
}
});

function handleColumnX003(thisSelect) {
  var thisRow = $(thisSelect).closest('tr.subquestion-list');
  var item3 = $('.answer_cell_X003', thisRow);
  if($(thisSelect).val() < '3') {
    $('.inserted-select', item3).prop('disabled', false);
  }
  else {
    $('.inserted-select', item3).val('').prop('disabled', true);
    $('input:text', item3).val('').trigger('change');
  }
}
}

```

Ab hier wieder altbekannt

```

// Returning to page
$('.with-select input:text', thisQuestion).each(function(i) {
  var thisCell = $(this).closest('.answer-item');
  var inputText = $.trim($(this).val());
  $('select.inserted-select', thisCell).val(inputText);
});

// Clean-up styles
$('select.inserted-select', thisQuestion).css({
  'max-width': '100%'
});
$('.with-select input:text', thisQuestion).css({
  'position': 'absolute',
  'left': '-9999em'
});
});
</script>

```

2. Verschiedene Spaltenbreiten

2.1.1. Anwendungsbeispiel

Ebenfalls bei einer Matrix(Texte) kommt es oft vor, dass die Einträge in den Spalten eine verschiedenen Länge haben. In der einen Spalte werden nur Einträge mit wenigen Zeichen erwartet, wie Postleitzahlen, Automarken, in anderen dagegen recht lange.

Im Normalfall teilt LimeSurvey den für die Antworten bereitstehenden Raum gleichmäßig auf.

	PLZ	Straße	E-Mail	Alter
Adalbert Ameise	<input type="text" value="63128"/>	<input type="text" value="Bettina-von-Arnim-Allee"/>	<input type="text" value="meier.transportunternel"/>	<input type="text" value="34"/>
Berta Biber	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Claas Chamäleon	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Doris Dachs	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Dadurch wird zum Einen Platz verschwendet, zum Anderen werden Antworten optisch abgeschnitten.

Besser wäre es wohl so:

	PLZ	Straße	E-Mail	Alter
Adalbert Ameise	<input type="text" value="63128"/>	<input type="text" value="Bettina-von-Arnim-Allee 123a"/>	<input type="text" value="meinelange.emailadresse@meinlangerProvider.de"/>	<input type="text" value="34"/>
Berta Biber	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Claas Chamäleon	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Doris Dachs	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

2.1.2. Implementierung

In den Fragetext wird im Quellcode-Modus folgendes kleine javascript-snippet eingefügt.

Dieses macht nichts anderes, als jeder Spalte eine zusätzliche Klasse mit einer fortlaufenden Nummer zuzufügen.

Zu beachten ist, dass die erste Spalte (laufende Nummer 0) die Spalte mit den Teilfragen ist.

Die Einstellung in „Teilfragenbreite“ wird also überschrieben.

```

<script type="text/javascript" charset="utf-8">
$(document).on('ready pjax:scriptcomplete',function(){
var thisQuestion = $('#question{QID}');
// Add a question class
thisQuestion.addClass('custom-array');

// Column-specific classes
$('table.subquestion-list tr', thisQuestion).each(function(i) {
  $('th, td', this).each(function(i) {
    $(this).addClass('column-'+i);
  });
});
});
</script>

```

Danach kommt nun das Entscheidende, die Definition der Spaltenbreiten mittels css.

```

<style type="text/css">
.custom-array table.subquestion-list col {
width: auto !important;
}

.custom-array table.subquestion-list thead .column-0 { width: 15%; }
.custom-array table.subquestion-list thead .column-1 { width: 10%; }
.custom-array table.subquestion-list thead .column-2 { width: 30%; }
.custom-array table.subquestion-list thead .column-3 { width: 38%; }
.custom-array table.subquestion-list thead .column-4 { width: 7%; }
</style>

```

Zu beachten: Die Summe der Spaltenbreiten muss – natürlich – 100% ergeben.

Dies ist nicht auf Matrix(Texte) beschränkt, sondern bei allen Matrix-Typen einsetzbar.

Da dieses Tutorial primär das Layout behandelt, wird nicht auf die Validierung eingegangen (dass die PLZ aus fünf Ziffern besteht und das Alter in einem vernünftigen Bereich liegt).

3. Matrizen mit „Sonstigen“-Zeilen

3.1. Matrix (Texte/Zahlen)

3.1.1. Anwendungsbeispiel

Oft taucht der Wunsch auf, neben den vorgegebenen Teilfragen eine Option „Sonstige“ anzubieten.

Beispiele:

a. Matrix(Texte)

Q2			
	Fellart	Anzahl der Beine	Geschmack
Ameise	<input type="text"/>	<input type="text"/>	<input type="text"/>
Biber	<input type="text"/>	<input type="text"/>	<input type="text"/>
Chamäleon	<input type="text"/>	<input type="text"/>	<input type="text"/>
Dachs	<input type="text"/>	<input type="text"/>	<input type="text"/>
Sonstiges Tier 1 <input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Sonstiges Tier 2 <input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

b. Matrix(Zahlen) mit Checkbox-Darstellung

Q1			
	gegenwärtig	zukünftig	weder noch
Ameise	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Biber	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Chamäleon	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Dachs	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sonstiges Tier 1 <input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sonstiges Tier 2 <input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

3.1.2. Implementierung

Das Format der Studie muss „Gruppe für Gruppe“ sein.

Nach der Matrixfrage werden so viele Fragen von Typ „Kurzer Text“ erzeugt, wie man „Sonstige“ anzeigen will.

```
graph TD; G1[G1] --- Q1["[Q1] > Q1"]; Q1 --- Q1Other1["[Q1Other1] > Other 1:"]; Q1 --- Q1Other2["[Q1Other2] > Other 2:"];
```

In der Matrixfrage sind die beiden „Sonstigen“ bereits als Teilfragen angelegt

Code	Teilfrage
SQ001	Ameise
SQ002	Biber
SQ003	Chamäleon
SQ004	Dachs
SQ005	Sonstiges Tier 1
SQ006	Sonstiges Tier 2

Da die Matrix(Zahlen) nicht sehr verschieden von einer Matrix(Texte) ist (dahinterliegende Struktur) ist das script auch für beide identisch.

```

<script type="text/javascript" charset="utf-8">
  $(document).ready(function() {

    // Identify the questions
    var thisQuestion = $('#question'+{QID}+'');
    var nextQuestion1 = $(thisQuestion).nextAll('.text-short:eq(0)');
    var nextQuestion2 = $(thisQuestion).nextAll('.text-short:eq(1)');
    var nextQuestions = $(nextQuestion1).add(nextQuestion2);
    var nextLength = nextQuestions.length;
    var sqLength = ('tr.answers-list', thisQuestion).length;

    // Hide the short-text questions
    $(nextQuestions).hide();

    // Move the hidden text inputs into the array
    for (i = 0; i < nextLength; i++) {
      var workingIndex = (sqLength - 1) - (nextLength - i);
      var nextQ = nextQuestions[i];
      $('th.answertext:eq('+workingIndex+')', thisQuestion).append($('input[type="text"]',
nextQ)).closest('tr').addClass('otherRow');
    }

    // Some styling...
    $('input[type="text"]', thisQuestion).css({
      'width': '100%'
    });
  });
</script>

```

Dieses script wird in den Fragetext der Matrixfrage (im Quellcode-Modus) eingefügt.

Die rot markierten Zeilen sind wieder abhängig von der Anzahl der einzublendenden „Sonstigen“

Hier sind es zwei „Sonstige; der Aufbau ist wohl ersichtlich, und „eq(x)“ beginnt die Zählung bei 0.

3.2. „Normale“ Matrix

3.2.1. Anwendungsbeispiel

Q3					
	sehr gut	gut	mittel	schlecht	sehr schlecht
Ameise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Biber	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Chamäleon	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Sonstiges Tier 1 <input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Sonstiges Tier 2 <input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

3.2.2. Implementierung

Da eine „normale“ Matrix in der x-Achse keine Teilfragen, sondern Antwortoptionen hat, ist hier das script ein wenig anders.

Außerdem wird am Ende getestet, ob ein Text in den „Sonstigen“ steht, wenn ein Button angeklickt wurde, und ebenfalls umgekehrt

```
<script type="text/javascript" charset="utf-8">
  $(document).ready(function() {

    // Identify the questions
    var thisQuestion = $('#question'+{QID}+'');
    var nextQuestion1 = $(thisQuestion).nextAll('.text-short:eq(0)');
    var nextQuestion2 = $(thisQuestion).nextAll('.text-short:eq(1)');
    var nextQuestions = $(nextQuestion1).add(nextQuestion2);
    var nextLength = nextQuestions.length;
    var sqLength = ('tr.answers-list', thisQuestion).length;

    // Hide the short-text questions
    $(nextQuestions).hide();

    // Move the hidden text inputs into the array
    for (i = 0; i < nextLength; i++) {
      var workingIndex = (sqLength - 1) - (nextLength - i);
      var nextQ = nextQuestions[i];
      $('th.answertext:eq('+workingIndex+')', thisQuestion).append($('input[type="text"]',
nextQ)).closest('tr').addClass('otherRow');
    }
  }
}
```



```

// Some styling...
$('input[type="text"]', thisQuestion).css({ 'width': '100%' });

// Handle the "Other" radios
$('input[type="text"]', thisQuestion).on('keyup change',function(event){
    event.stopPropagation();

    var thisRow = $(this).closest('tr.answers-list');
    if($.trim($(this).val()) == '') {
        $('input:radio[value!=""]', thisRow).prop('checked',false);
        $('input:radio[value=""]', thisRow).click();
    }
    else {
        $('input:radio[value=""]', thisRow).prop('checked',false);
    }
});

// Handle the "Other" text inputs
$('.otherRow input.radio', thisQuestion).on('click',function(event){
    var thisRow = $(this).closest('tr.answers-list');
    if($(this).attr('value') == '') {
        $('input[type="text"]', thisRow).val('');
    }
});

// Validate the "Other" text inputs on submit
if($('#ls-button-submit').attr('data-inserted-other') != 'true') { // We're only doing this once on this
page
{
    $('#ls-button-submit').attr('data-inserted-other', 'true').on('click.insertedOther', function (event)
{

    var otherError = 0;

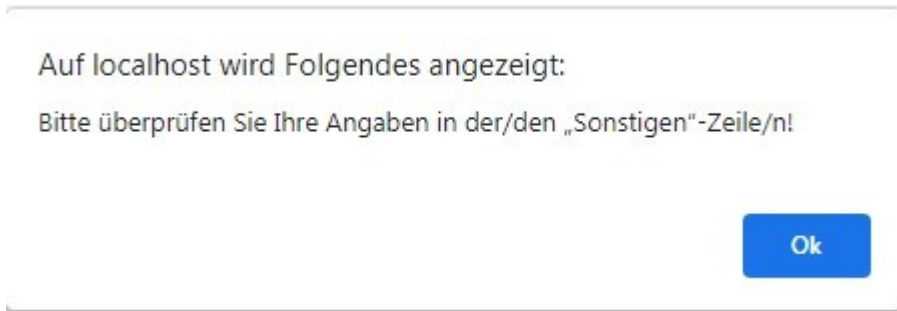
    $('.array-flexible-row .otherRow').each(function(i) {

        if(($('input:radio[value!=""]:checked', this).length > 0 && $('input[type="text"]',
this).val() == '') || ($('input:radio[value!=""]:checked', this).length == 0 && $('input[type="text"]',
this).val() != '')) {
            otherError = 1;
        }
    });

    if(otherError == 1) {
        alert('Bitte überprüfen Sie die Angaben in den "Sonstigen"-Zeilen!');
        return false;
    }
});
}
});
</script>

```

Im Falle einer Eingabe in den „Sonstigen“, aber ohne Bewertung erscheint dann folgende Meldung:



Falls man dies nicht möchte, sollten die obigen „roten“ Zeilen gelöscht werden.

3.2.3. Was spricht dagegen?

Da bereits in der Matrixfrage die Teilfragen für die – optionalen - „Sonstigen“ angelegt sind, kann diese Frage nicht als „Pflichtfrage“ deklariert sein; dann würde immer eine Eingabe dort verlangt werden.

Daher muss also mit der Fragenvalidierung sichergestellt werden, dass die „normalen“ Teilfragen beantwortet sind.

Eine Matrix wird ja meist verwendet, um die Teilfragen in irgendeiner Weise zu bewerten. Und in der Auswertung sollen dann Unterschiede zwischen den einzelnen Teilfragen erkannt werden.

- Dazu wird wohl zunächst einmal ein „t-test für abhängige Gruppen“ angewendet (es sind dieselben Personen, die unterschiedliche Items bewerten).
Man wird aber nur relativ wenige Nennungen eines sonstigen Items erhalten, dass ein Vergleich mit den vorgegebenen Items unsinnig wird; es können ja nur die Personen verglichen werden, die beide Items beantwortet haben.
- Sollten mehrere „Sonstige“ (wie hier im Beispiel) angeboten werden, verteilen sich die Nennungen auch noch (der eine Teilnehmer schreibt „Ente“ in das erste „Sonstige“, der andere schreibt „Ente“ in das zweite); es muss also noch sehr viel umkodiert und umstrukturiert werden.
- Es wird immer Nennungen geben, die vom Fragebogenersteller eigentlich als vorgegeben betrachtet wurden.
Beispiel: Vorgegebenes Item „Leichtathletik“, Eintrag in „Sonstiges“: Weitsprung.
Oder: Vorgegebenes Item „Geflügel“, Eintrag in „Sonstiges“: Ente.
- Bei einer Bewertung der Wichtigkeit oder des Gefallens werden nur solche Items als Sonstige hinzugefügt, die wichtig sind, oder die gut gefallen.
Dadurch entsteht gegenüber den übrigen Items ein Bias.

Da ja jede Umfrage ein bestimmtes Ziel verfolgt, werden die vorgegebenen Items ja diesem entsprechen.

Wenn also ein Vergleich deutschlandweit erscheinender Tageszeitungen gemacht werden soll, wird man diese vorgeben (wobei der Umfrageersteller definiert, welche dies sind). Das heißt, von den Teilnehmern in „Sonstige“ eingetragenen Nennungen sind zwar gut und schön, haben aber keinen zusätzlichen Wert. Und man bekommt Nennungen wie „Offenbach Post“ oder das „Miesbacher Tageblatt“.

Sollte die Frage aber zum Beispiel lauten „Wie wichtig bewerten Sie die folgenden Maßnahmen zum Lärmschutz an der B124a?“, wurden die wichtigsten möglichen Maßnahmen ja sicher bereits in einem Pretest ermittelt. Diese sollten dann in der Matrix bewertet werden.

Da es wenig plausibel ist, dass sonstige Maßnahmen genannt werden, aber mit der Bewertung „überhaupt nicht wichtig“, sollte man solche weiteren Nennungen nicht in die Matrixfrage einschließen, sondern eine weitere Frage stellen „Gibt es weitere für Sie wichtige Maßnahmen?“

4. Zusätzliche Spalten in Matrizen

4.1. Matrizen mit „Kommentar“-Spalten

4.1.1. Anwendungsbeispiel (Kommentar hinter Matrix)

Manchmal hätte man in einer Matrix, in welcher Objekte bewertet werden, gleich eine Begründung dieser Bewertung.

Dies ist also ein zusätzliches optionales Feld; zusätzlich zur Wahl eines der Radio-Buttons kann noch eine Begründung gegeben werden..

Beispiel:

Q1						
	1	2	3	4	5	Begründung
Adalbert Ameise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="text"/>
Berta Biber	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="text"/>
Carlo Chamäleon	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="text"/>
Doris Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="text"/>

4.1.2. Implementierung

Das Format muss „Gruppe für Gruppe“ sein.

Nach der Matrix-Frage wird eine Frage vom Typ „Mehrfache kurze Texte“ erzeugt mit derselben Anzahl Teilfragen wie die Matrixfrage.

Der Fragetext dieser Frage wird später zur Überschrift („Begründung“).

Dann wird in den Fragetext der Matrixfrage folgendes script eingefügt (im Quellcode-Modus)

```
<script type="text/javascript" charset="utf-8">
  $(document).on('ready ajax:scriptcomplete',function(){
    // Identify the questions
    var qArrayID = {QID};
    var qArray = $('#question'+qArrayID);
    var qMultiText = qArray.nextAll('.multiple-short-txt:eq(0)');

    // Hide the multi-short-text question
    $(qMultiText).hide();

    var tableWidth = $('table.questions-list:eq(0)', qArray).width();
    var answerWidth = $('col.ls-col-odd:eq(0)', qArray).width();
    var answerWidthPercent = (answerWidth/tableWidth)*100;
    var answersLength = $('col.ls-col-odd, col.ls-col-even', qArray).length;
    var answerWidthPercent2 = (answerWidthPercent*answersLength)/(answersLength+1)
    $('table.questions-list col', qArray).removeAttr('width');
```

```

    $('table.questions-list col:not(.col-answers)', qArray).css('width', 'auto');
    $('table.questions-list thead tr', qArray).append( '<th class="answer-text inserted-column-label"
/></th>');
    $('table.questions-list thead tr th.answer-text.inserted-column-label', qArray).css('width','30%');
    $('tr.answers-list', qArray).each(function(i) {
        $(this).append('<td class="answer-item text-item">\
        </td>\
        ');
    });
    // Load the column label for the text inputs
    $('.inserted-column-label:eq(0)', qArray).text($('.ls-label-question', qMultiText).text());
    $('.text-item', qArray).append('<label class="ls-label-xs-visibility">'+$('.ls-label-question',
qMultiText).text()+</label>');
    // Loop through the multi-short-text sub-questions
    $('li.answer-item', qMultiText).each(function(i) {
        // Move the text inputs into the array
        $('input[type="text"]', this).appendTo($('tr.answers-list:eq('+i+') .text-item', qArray));
    });
});
</script>

```

Wie im Beispiel zu sehen ist, kann man auch hier das bereits gezeigte script zur Änderung der Spaltenbreite anschließen.

4.1.3. Anwendungsbeispiel (Kommentar/e vor Matrix)

Es kann auch die Idee geben, freie Felder an den Anfang zu setzen und die Matrix am Ende.

Beispiel: Tierbeobachtung im heimischen Garten. Oder was auch immer.

Q4b		Datum	Tierart	rot	gelb	grün	blau	schwarz
1	<input type="text" value="12.1.2022"/>	<input type="text" value="Ameise"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
2	<input type="text" value="12.1.2022"/>	<input type="text" value="Biber"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
3	<input type="text" value=" "/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
4	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	

Hier wäre es sicher von Vorteil, die dynamische Erzeugung neuer Zeilen zusätzlich einzubauen. (siehe Kap. 6)

4.1.4. Implementierung

Das Format muss „Gruppe für Gruppe“ sein.

Nach der Matrix-Frage wird so viele Fragen vom Typ „Mehrfache kurze Texte“ erzeugt wie man Textfelder pro Zeile benötigt (mit derselben Anzahl Teilfragen wie die Matrixfrage).

Die Fragetexte dieser Fragen werden später zur Überschrift („Datum / Tierart“).

Dann wird in den Fragetext der Matrixfrage folgendes script eingefügt (im Quellcode-Modus)

Da hier zwei offene Felder angezeigt werden, gibt es die beiden Variablen „qMultiText“ und „qMultiTextBis“.

Sollte nur ein Feld angezeigt werden soll, müssen sämtliche Verweise, die „bis“ enthalten, entfernt werden (hier rot markiert)

```
<script type="text/javascript" data-author="Tony Partner">

$(document).on('ready pjax:scriptcomplete',function(){

    // Identify the questions
    var qArrayID = {QID};
    var qArray = $('#question'+qArrayID);
    var qMultiText = qArray.nextAll('.multiple-short-txt:eq(0)');

    // Nur falls ein zweites Feld angezeigt werden soll
    var qMultiTextBis = qMultiText.nextAll('.multiple-short-txt:eq(0)');

    // Hide the multi-short-text question
    $(qMultiText).hide();

    // Nur falls ein zweites Feld angezeigt werden soll
    $(qMultiTextBis).hide();
});
```

```

// Remove the core column widths
$('table.questions-list col:not(.col-answers)', qArray).css('width', 'auto');

// Insert the header cells
$('table.questions-list tthead td:eq(0)', qArray).after('<th class="answer-text inserted-column-label" /></th><th class="answer-text-bis inserted-column-label-bis" /></th>');
$('.inserted-column-label, .inserted-column-label-bis', qArray).css('width', '30%');

// Insert the answer row cells
$('tr.answers-list', qArray).each(function(i) {
    $('li.answer-text', this).after('<td class="answer-item text-item"></td><td class="answer-item text-item-bis"></td>');
});

// Load the column label for the text inputs
$('.inserted-column-label:eq(0)', qArray).text($('.ls-label-question', qMultiText).text());
$('.inserted-column-label-bis:eq(0)', qArray).text($('.ls-label-question', qMultiTextBis).text());

// Loop through the multi-short-text sub-questions
$('li.answer-item', qMultiText).each(function(i) {
    // Move the text inputs into the array
    $('input[type="text"]', this).appendTo($('tr.answers-list:eq('+i+') .text-item', qArray));
});

$('li.answer-item', qMultiTextBis).each(function(i) {
    // Move the text inputs into the array
    $('input[type="text"]', this).appendTo($('tr.answers-list:eq('+i+') .text-item-bis', qArray));
});

});
</script>

```

4.1.5. Was spricht dagegen?

Es ist nicht möglich, die Items der Matrix in randomisierter Reihenfolge anzuzeigen, da die Felder der Texteingabe diese Randomisierung nicht mitmachen und dadurch später in der Auswertung keine Zuordnung zwischen Bewertung und Begründung möglich ist.

4.2. Matrizen mit „Sonstige“-Spalten

4.2.1. Anwendungsbeispiel

Genauso gibt es die Möglichkeit, statt einer Begründung hier ein „Sonstiges“ eintragen zu lassen.

Der Unterschied zur vorherigen Lösung ist, dass diese „Sonstigen“-Spalte genau wie einer der Radio-Buttons wirkt, also hier „Genau eine Option der 5 muss angeklickt bzw. gefüllt sein“.

Beispiel:

Q1...					
	Zu Fuß	Mit dem Fahrrad	Mit Bus, Bahn, usw.	Mit dem Auto	Sonstiges
Adalbert Ameise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="text"/>
Berta Biber	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="text"/>
Carlo Chamäleon	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="text"/>
Doris Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="text"/>

4.2.2. Implementierung

Wie im vorigen Fall wird auch hier wieder hinter der Matrixfrage eine Frage von Typ „Mehrfache kurze Texte“ erzeugt.

Mit diesem script wird nun das Layout erzeugt und gleichzeitig sichergestellt, dass entweder ein „radio“ angeklickt wird oder ein Eintrag im Textfeld gemacht wird (nicht beides)

```
<script type="text/javascript" charset="utf-8">
  $(document).on('ready ajax:scriptcomplete',function(){
    // Identify the questions
    var q1ID = {QID};
    var thisQuestion = $('#question'+q1ID);
    var nextQuestion = thisQuestion.nextAll('.multiple-short-txt:eq(0)');
    var q2ID = $(nextQuestion).attr('id').replace(/question/, '');

    //Hide the multiple-short-text
    nextQuestion.hide();

    // Move the text inputs
    $('tr.answers-list', thisQuestion).each(function(i) {
      var thisCode = $(this).attr('id').split('X')[2].split(q1ID)[1];
      $('td.answer-item:last input[type="radio"]', this).css({
        'position': 'absolute',
        'left': '-9999em'
      });
      $('td.answer-item:last', this).removeClass('radio-item').addClass('inserted-text-item').append($
('input[type="text"][id$="X'+q2ID+thisCode+'"]', nextQuestion));
    });
  });
</script>
```



```

});

// Listeners on the text inputs
$('input[type="text"]', thisQuestion).on('keyup change', function(e) {
    var thisRadio = $(this).closest('td').find('input[type="radio"]');
    var thisRadioVal = thisRadio.val();
    if($.trim($(this).val()) != '') {
        $(thisRadio).trigger('click');
    }
    else {
        $(thisRadio).prop('checked', false);
        thisRadioVal = '';
    }
    // Reset Expression manager
    checkconditions(thisRadioVal, $(thisRadio).attr('name'), 'radio', 'click');
});

// Listeners on the radios
$('input[type="radio"]', thisQuestion).on('click', function(e) {
    if(!$(this).closest('td').hasClass('inserted-text-item')) {
        $(this).closest('tr').find('input[type="text"]').val('');
    }
});

});
</script>

```

Wie im Beispiel zu sehen ist, kann man auch hier das bereits gezeigte script zur Änderung der Spaltenbreite anschließen.

4.2.3. Was spricht dagegen?

Es ist nicht möglich, die Items der Matrix in randomisierter Reihenfolge anzuzeigen, da die Felder der Texteingabe diese Randomisierung nicht mitmachen und dadurch später in der Auswertung keine Zuordnung zwischen Items und sonstigen Nennungen möglich ist.

5. Matrizen mit zusätzlichen Kopfzeilen und Zwischenzeilen

5.1. Matrix(Zahlen)

5.1.1. Anwendungsbeispiel

Hier möchte ich einmal zeigen, wie man in einer Matrix sowohl die Spalten als auch die Zeilen durch mehrere Header und Zwischenüberschriften besser strukturieren kann.

Diese Schuhe sind im Besitz ...							
	Blaue Schuhe		Gelbe Schuhe			Rote Schuhe	
	Option 1	Option 2	Option 1	Option 2	Option 3	Option 1	Option 2
Buchstaben A-B							
Adalbert Ameise	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Berta Biber	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Buchstaben C-D							
Carlo Chamäleon	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Doris Dachs	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

5.1.2. Implementierung

Zunächst fügen wir die zusätzlichen Header der y-Achse hinzu.

Und zwar gehen wir von unten nach oben vor. Zunächst die Zeile der blauen, gelben und roten Schuhe; danach die obere Zeile.

Man erkennt dies leicht an der Funktion „prepend“, also „davorsetzen“.

```
<script type="text/javascript" charset="utf-8">
$(document).on('ready pjax:scriptcomplete',function(){
  // Insert the column categories
  $('#question{QID} table.subquestion-list thead tr:eq(0) td:eq(0)').remove();
  $('#question{QID} table.subquestion-list thead').prepend('<tr class="ls-heading">\
  <td rowspan="2" colspan="1" style="border-top:0 !important;"></td>\
  <th class="answer-text inserted-header" colspan="2">Blaue Schuhe</th>\
  <th class="answer-text inserted-header" colspan="3">Gelbe Schuhe</th>\
  <th class="answer-text inserted-header" colspan="2">Rote Schuhe</th>\
  </tr>');
  $('#question{QID} table.subquestion-list thead').prepend('<tr class="ls-heading">\
  <td rowspan="1" colspan="1" style="border-bottom:0 !important;background-color:silver"></td>\
  <th class="answer-text inserted-header" colspan="7" style="text-align:left;font-size:14pt;color:maroon;background-color:silver">Diese Schuhe sind im Besitz ...</th>\
  </tr>');
});
</script>
```

Danach nun das script für die Zwischenzeilen.

Wichtig in diesem script sind die Stellen eq(0), eq(2). Diese geben an, wo die Zeilen eingefügt werden.

eq(0): Vor der ersten Zeile (man fängt immer bei 0 an zu zählen)

eq(2): Vor der dritten Zeile.

So kann man also beliebig Zwischenzeilen einfügen.

Diese werden in der Definition nach Wunsch mit einem „style“ versehen:

```
var subHeading1 = '<span style="color:maroon;font-size:12pt;font-weight:bold"> Buchstaben A-B</span>';
```

Hier also: „Textfarbe „maroon“, Zeichengröße „12 Punkt“, und „fett“.

```
<script type="text/javascript" charset="utf-8">
  $(document).ready(function() {
    // Identify this question
    var thisQuestion = $('#question{QID}');
    // Define the sub-heading text strings
    var subHeading1 = '<span style="color:maroon;font-size:12pt;font-weight:bold">Buchstaben A-B</span>';
    var subHeading2 = '<span style="color:maroon;font-size:12pt;font-weight:bold">Buchstaben C-D</span>';
    var columnsLength = $('tr.subquestion-list:eq(0) > *', thisQuestion).length;
    // Insert the new rows
    $('tr.subquestion-list:eq(0)', thisQuestion).before('<tr class="sub-header-row"><th
colspan="'+columnsLength+'">'+subHeading1+'</th></tr>');
    $('tr.subquestion-list:eq(2)', thisQuestion).before('<tr class="sub-header-row"><th
colspan="'+columnsLength+'">'+subHeading2+'</th></tr>');
    // Fix up the row classes
    var rowClass = 1;
    $('table.subquestions-list tbody tr', thisQuestion).each(function(i) {
      if($(this).hasClass('sub-header-row')) {
        rowClass = 1
      }
      else {
        rowClass++;
        $(this).removeClass('array1 array2')
        if(rowClass % 2 == 0) {
          $(this).addClass('array2');
        }
        else {
          $(this).addClass('array1');
        }
      }
    });
  });
</script>
```

5.2. Matrix

Die Implementierung für eine Matrix ist völlig analog.

Man muss nur bedenken, dass eine Matrix in der x-Achse keine Teilfragen (subquestion) sondern Antwortoptionen (answer) hat.

Damit genügt es, diese beiden Zeilen des scripts

```
var columnsLength = $('tr.subquestion-list:eq(0) > *', thisQuestion).length;
// Insert the new rows
$('tr.subquestion-list:eq(0)', thisQuestion).before('<tr class="sub-header-row"><th colspan="'+columnsLength+'">'+subHeading1+'</th></tr>');
```

in dies zu ändern.

```
var columnsLength = $('tr.answers-list:eq(0) > *', thisQuestion).length;
// Insert the new rows
$('tr.answers-list:eq(0)', thisQuestion).before('<tr class="sub-header-row"><th colspan="'+columnsLength+'">'+subHeading1+'</th></tr>');
```

Wie beurteilen Sie die Kreationen der Teilnehmer beim Kochwettbewerb?

	sehr gut	gut	weder noch	schlecht	sehr schlecht
La bandera dominicana					
Doris Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Berta Biber	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Carlo Chinchilla	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Adalbert Ameise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Mondongo an Mofongo					
Doris Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Berta Biber	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Doris Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Carlo Chinchilla	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

6. „Dynamische“ Anzeige der Zeilen einer Matrix

6.1.1. Anwendungsbeispiel

Wenn man die Eingabe einer unbekannt Anzahl von Objekten erfragt

- Welche Medikamente nehmen Sie zur Zeit?
- An welchen Orten haben Sie bereits gewohnt?

so kann man eine Matrix mit einer großen Anzahl Zeilen direkt auf dem Bildschirm anzeigen.

Q6					
	Name	Stadtteil	Sportart	Altersklasse	Bemerkungen
1.	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
2.	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
3.	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
4.	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
5.	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
6.	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
7.	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Doch ist dies sicher nicht optimal. Der Beantworter könnte denken, er müsse alle Zeilen ausfüllen.

Eine Verbesserung ist dann schon, dass die nächste Zeile erst eingeblendet wird, wenn die aktuelle Zeile gefüllt ist; dazu wird dann die Teilfragenrelevanz benutzt.

Q6					
	Name	Stadtteil	Sportart	Altersklasse	Bemerkungen
1.	<input type="text" value="Adalbert Ameise"/>	<input type="text" value="Dudenhofen"/>	<input type="text" value="Schwimmen"/>	<input type="text" value="U12"/>	<input type="text" value="Tralala"/>
2.	<input type="text" value="Berta Biber"/>	<input type="text" value="Weißkirchen"/>	<input type="text" value="Voileyball"/>	<input type="text"/>	<input type="text"/>

Q6					
	Name	Stadtteil	Sportart	Altersklasse	Bemerkungen
1.	<input type="text" value="Adalbert Ameise"/>	<input type="text" value="Dudenhofen"/>	<input type="text" value="Schwimmen"/>	<input type="text" value="U12"/>	<input type="text" value="Tralala"/>
2.	<input type="text" value="Berta Biber"/>	<input type="text" value="Weißkirchen"/>	<input type="text" value="Voileyball"/>	<input type="text" value="U14"/>	<input type="text"/>
3.	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Für die dritte Zeile wäre dann einzutragen:

```
!is_empty(Q1_Y002_X001) AND !is_empty(Q1_Y002_X002) AND !is_empty(Q1_Y002_X003) AND !  
is_empty(Q1_Y002_X004)
```

dass also alle Felder der zweiten Zeile gefüllt sind (wir gehen davon aus, dass in diesem Beispiel das Feld „Bemerkungen“ leer bleiben darf)

Dies ist schon besser, doch auch hier hat die leere Zeile einen gewissen Aufforderungscharakter.

Daher gibt es diese Lösung. Der Teilnehmer fügt nur dann Zeilen hinzu, falls es notwendig ist. Fälschlicherweise hinzugefügte Zeilen lassen sich auch wieder entfernen.

Q6					
	Name	Stadtteil	Sportart	Altersklasse	Bemerkungen
1.	<input type="text" value="Adalbert Ameise"/>	<input type="text" value="Dudenhofen"/>	<input type="text" value="Schwimmen"/>	<input type="text" value="U12"/>	<input type="text" value="Tralala"/>
2.	<input type="text" value="Berta Biber"/>	<input type="text" value="Weißkirchen"/>	<input type="text" value="Volleyball"/>	<input type="text" value="U14"/>	<input type="text"/>

Vom „Look&Feel“ für den Teilnehmer scheint diese Lösung sehr gut zu sein. Hier liegt es in seiner Hand, weitere Zeilen hinzuzufügen, oder nicht.

Natürlich werden diese Zeilen nicht „live“ in der Datenbank hinzugefügt. Es muss vorher die Maximalzahl an Teilfragen angelegt sein.

Das script sorgt dafür, dass es nicht möglich ist, bei Erreichen der Maximalzahl noch weiter auf „zufügen“ zu klicken.

6.1.2. Implementierung

In den Fragetext (im Quellcode-Modus) wird folgendes script eingefügt:

```
<script>
$(document).ready(function() {

    // A function to add or remove rows of an Array (Multi Flexible)(Text) question
    function varLengthArray(qID) {

        if ($('#question'+qID+'.').length > 0) {

            // The HTML content of the Add/Remove elements - modify as you wish
            var addContent = '[+] Zeile zufügen';
            var removeContent = '[-] Zeile löschen';

            // Create the Add and Remove elements & insert them
            // Adjust colors by using other bootstrap classes
            // „btn-primary“, „btn-success“, „btn-info“, „btn-warning“, „btn-danger“
            var el1 = document.createElement('div');
            el1.setAttribute('id','addButton'+qID);
            el1.setAttribute('class','btn btn-success');
            document.body.appendChild(el1);
            var el2 = document.createElement('div');
            el2.setAttribute('id','removeButton'+qID);
            el2.setAttribute('class','btn btn-danger');
            document.body.appendChild(el2);

            // Move them to after the array
            $( 'div#addButton'+qID ).appendTo( $( '#question' + qID + ' table.ls-answers' ).parent() );
            $( 'div#removeButton'+qID ).appendTo( $( '#question' + qID + ' table.ls-answers' ).parent() );

            // Insert their HTML
            $( 'div#addButton'+qID ).html( addContent );
            $( 'div#removeButton'+qID ).html( removeContent );

            // Style the elements - you can modify here if you wish
            $( 'div#addButton'+qID ).css({
                'margin':'10px 0 10px 10px',
                'padding':'1px',
                'text-align':'center',
                'width':'auto',
                'cursor':'pointer',
                'float':'left'
            });

            $( 'div#removeButton'+qID ).css({
                'margin':'10px 0 10px 10px',
                'padding':'1px',
                'text-align':'center',
```

```

        'width':'auto',
        'cursor':'pointer',
        'float':'left'
    });

    // Initially hide the Remove element
    $( 'div#removeButton'+qID ).hide();

    // Call the functions below when clicked
    $( 'div#addButton'+qID ).click(function (event) {
        addRow(qID);
    });
    $( 'div#removeButton'+qID ).click(function (event) {
        removeRow(qID);
    });

    // Function to add a row, also shows the Remove element and hides the
    //Add element if all rows are shown
    function addRow(qID) {
        var arrayRow = '#question' + qID + ' table.ls-answers tr.subquestion-list';
        var rowCount = $( arrayRow ).size() - 1;
        $( arrayRow + '[name="hidden"]:first' ).attr('name', 'visible').show();
        $( 'div#removeButton'+qID ).show();
        if ( $( arrayRow + ':eq(' + rowCount + ')' ).attr('name') == 'visible' ) {
            $( 'div#addButton'+qID ).hide();
        }
    }

    // Function to remove a row, also clears the contents of the removed row,
    // shows the Add element if the last row is hidden and hides the Remove
    // element if only the first row is shown
    function removeRow(qID) {
        var arrayRow = '#question' + qID + ' table.ls-answers tr.subquestion-list';
        var rowCount = $( arrayRow ).size() - 1;
        $( arrayRow + '[name="visible"]:last input[type="text"]' ).val('');
        $( arrayRow + '[name="visible"]:last' ).attr('name', 'hidden').hide();
        $( 'div#addButton'+qID ).show();
        if ( $( arrayRow + ':eq(1)' ).attr('name') == 'hidden' ) {
            $( 'div#removeButton'+qID ).hide();
        }
    }

    // Just some initialization stuff
    var arrayRow = '#question' + qID + ' table.ls-answers tr.subquestion-list';
    var rowCount = '';

    // Initially hide all except first row or any rows with populated inputs
    $( arrayRow ).each(function(i) {
        if ( i > 0 ) {
            // We also need to give the hidden rows a name cause IE doesn't

```



```
// recognize jQuery :visible selector consistently
$( this ).attr('name', 'hidden').hide();

$('input[type=text]', this).each(function(i) {
    if ($(this).attr('value') != '') {
        $(this).parents('tbody:eq(0)').attr('name', 'visible').show();
        $( 'div#removeButton'+qID ).show();
    }
});
rowCount = i;
}
});
}
}
// Call the function with a question ID
varLengthArray({QID});
});
</script>
```

7. Matrix(Texte) mit senkrechten Radio-Buttons in Zellen

7.1.1. Anwendungsbeispiel

Um den oft geäußerten Wunsch nach einer „Dual-Matrix“ mit mehr als zwei Skalen in etwa zu erfüllen, kann man eine Matrix(Texte) nehmen und in jeder Zelle senkrechte Radio-Buttons einbauen.

Q7			
	Schwimmen	Farbe	Häufigkeit
Adalbert Ameise	<input type="radio"/> gut <input type="radio"/> mittel <input type="radio"/> schlecht	<input type="radio"/> Gelb <input type="radio"/> Blau <input type="radio"/> Rot	<input type="radio"/> Sehr oft <input type="radio"/> Oft <input type="radio"/> Manchmal <input type="radio"/> Selten <input type="radio"/> Nie
Berta Biber	<input type="radio"/> gut <input type="radio"/> mittel <input type="radio"/> schlecht	<input type="radio"/> Gelb <input type="radio"/> Blau <input type="radio"/> Rot	<input type="radio"/> Sehr oft <input type="radio"/> Oft <input type="radio"/> Manchmal <input type="radio"/> Selten <input type="radio"/> Nie
Carlo Chinchilla	<input type="radio"/> gut <input type="radio"/> mittel <input type="radio"/> schlecht	<input type="radio"/> Gelb <input type="radio"/> Blau <input type="radio"/> Rot	<input type="radio"/> Sehr oft <input type="radio"/> Oft <input type="radio"/> Manchmal <input type="radio"/> Selten <input type="radio"/> Nie

7.1.2. Implementierung

Der folgende javascript-Code stammt von einem polnischen Kollegen; daher ist die Syntax ein wenig anders.

```
<style type="text/css">.big-radio {
  width: 16px !important;
  height: 16px !important;
}
.clickable-span:hover {
  cursor: pointer;
  text-decoration: underline;
}
td.with-select input.form-control { display:none;}
.ls-answers .ls-heading th {
  text-align: left;
}
</style>
<script type="text/javascript" charset="utf-8">
  var checkClosest = function(elem) {
    elem.firstChild.click();
  };

  $(document).on('ready ajax:scriptcomplete', function(){
    var thisQuestion = $('#question{QID}');
```

```

// Insert radios
if(!document.querySelector(".inserted-radio")){
    $(' .answer-item.answer_cell_X001', thisQuestion).addClass('with-select').append(
        '<div style="text-align: left" class="list-question-select">\
            <span onclick="checkClosest(this)">\
                <input value="1" class="inserted-radio" type="radio" name="radio_" id="radio-1" /> gut
</span><br />\
            <span onclick="checkClosest(this)">\
                <input value="2" class="inserted-radio" type="radio" name="radio_" id="radio-2" /> mittel
</span><br />\
            <span onclick="checkClosest(this)">\
                <input value="3" class="inserted-radio" type="radio" name="radio_" id="radio-3" />
schlecht </span> \
            </div>'
    );
    $(' .answer-item.answer_cell_X002', thisQuestion).addClass('with-select').append(
        '<div style="text-align: left" class="list-question-select">\
            <span onclick="checkClosest(this)">\
                <input value="1" class="inserted-radio" type="radio" name="radio_" id="radio-1" /> Gelb
</span><br />\
            <span onclick="checkClosest(this)">\
                <input value="2" class="inserted-radio" type="radio" name="radio_" id="radio-2" /> Blau
</span><br />\
            <span onclick="checkClosest(this)">\
                <input value="3" class="inserted-radio" type="radio" name="radio_" id="radio-3" /> Rot \
            </div>'
    );
    $(' .answer-item.answer_cell_X003', thisQuestion).addClass('with-select').append(
        '<div style="text-align: left" class="list-question-select">\
            <span onclick="checkClosest(this)">\
                <input value="1" class="inserted-radio" type="radio" name="radio_" id="radio-1" /> Sehr
oft </span><br />\
            <span onclick="checkClosest(this)">\
                <input value="2" class="inserted-radio" type="radio" name="radio_" id="radio-2" />
Oft </span><br />\
            <span onclick="checkClosest(this)">\
                <input value="3" class="inserted-radio" type="radio" name="radio_" id="radio-3" />
Manchmal </span><br />\
            <span onclick="checkClosest(this)">\
                <input value="4" class="inserted-radio" type="radio" name="radio_" id="radio-4" />
Selten </span><br />\
            <span onclick="checkClosest(this)">\
                <input value="5" class="inserted-radio" type="radio" name="radio_" id="radio-5" /> Nie
</span> \
            </div>'
    );
}

document.querySelectorAll("input[id^='answer']").forEach( function(inputElem){
    var currID = inputElem.id.substr(6);
    inputElem.parentElement.querySelector("div.list-question-select").querySelectorAll('input').forEach(
function(radioNeedsName){
    radioNeedsName.name += currID;
    // Łapiemy zmiany i uzupełniamy oryginalny element
    radioNeedsName.addEventListener("change", function(){
        inputElem.value = event.target.value;
    });
});
}

```

```

    })
  })
});

// Clean-up styles
$('input.inserted-radio').addClass('big-radio');
$('.list-question-select').find('span').addClass('clickable-span')
$('.with-select input:text', thisQuestion).css({
  'position': 'absolute',
  'left': '-9999em'
});
});
</script><script>
// Zmiana atrybutu name radiosów tak, aby działała wyłączość per subpytanie (Y)
function applyCustomRadios() {
  document.querySelectorAll("input[id^='answer']").forEach( function(inputElem){
    var currID = inputElem.id.substr(6);
    inputElem.parentElement.querySelector("div.list-question-
select").querySelectorAll('input').forEach( function(radioNeedsName){
      radioNeedsName.name += currID;
      // łapiemy zmiany i uzupełniamy oryginalny element
      radioNeedsName.addEventListener("change", function(){
        inputElem.value = event.target.value;
      })
    });
    inputElem.parentElement.querySelector("div.list-question-
select").querySelectorAll('label').forEach( function(radioNeedsName){
      radioNeedsName.name += currID;
      // łapiemy zmiany i uzupełniamy oryginalny element
      radioNeedsName.addEventListener("change", function(){
        inputElem.value = event.target.value;
      })
    })
  });
}

document.addEventListener('ready pjax:scriptcomplete', applyCustomRadios)
document.addEventListener('pjax:scriptcomplete', applyCustomRadios)
</script>

```

7.1.3. Was spricht dagegen?

Obwohl wir Radio-Buttons anzeigen und auch ein numerischer „value“ gespeichert wird, wird der Typ der Variablen beim Export nach SPSS trotzdem „Text“ sein.

Um also vernünftig in SPSS arbeiten zu können, muss man hier das Format der Variablen von „string“ auf „numerisch“ ändern; aber das ist ja kein Beinbruch, geht mit „Copy&Paste“ blitzschnell.

8. Matrix(Zahlen) im Checkbox-Layout mit letzter Spalte exklusiv

8.1.1. Anwendungsbeispiel

Bei einer Frage mit einer Mehrfachnennungs-Matrix, wozu in LimeSurvey eine Matrix(Zahlen) mit Checkbox-Layout genutzt wird, ist es oft unerlässlich, auch eine Antwortmöglichkeit „keine (davon)“ anzubieten.

Q8	Handball	Karate	Leichtathletik	Schwimmen	Volleyball	Keine
Adalbert Ameise	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Berta Biber	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Carlo Chinchilla	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Doris Dachs	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Diese letzte Spalte muss natürlich exklusiv sein.

Das folgende script stellt die Exklusivität sicher und löscht bei Klick auf „keine“ eventuell andere angewählte Felder dieser Zeile.

8.1.2. Implementierung 1

Wie immer wird das folgende script in den Fragetext (im Quellcode-Modus) eingefügt.

```
<script type="text/javascript" charset="utf-8">
$(document).ready(function() {
    // Call the exclude function using question ID
    excludeOpt({QID});
});

// A function to make the last option in each array row exclusive
function excludeOpt (qID) {

    var thisQuestion = $('#question'+qID)

    // Add some classes to the checkbox cells
    $('td.checkbox-item', thisQuestion).addClass('normal-item');
    $('tr.subquestion-list', thisQuestion).each(function(i) {
        $('.normal-item:last', this).removeClass('normal-item').addClass('exclusive-item')
    });

    // A listener on the checkboxes
    $('input[type="checkbox"]', thisQuestion).on('change', function (event) {
        handleExclusive($(this).closest('td'));
    });
};
```

```

function handleExclusive(thisCell) {

    var thisRow = $(thisCell).closest('tr');

    // Uncheck the appropriate boxes in a row
    if ($(thisCell).hasClass('normal-item')) {
        $('.exclusive-item input[type="checkbox"]', thisRow).prop('checked', false);
    }
    else {
        $('.normal-item input[type="checkbox"]', thisRow).prop('checked', false);
    }

    // Check conditions (relevance)
    $('td.checkbox-item', thisRow).each(function(i) {
        var thisValue = '';
        if($('input[type="checkbox"]', this).is(':checked')) {
            thisValue = 1;
        }
        var thisSGQA = $('input[type="checkbox"]', this).attr('id').replace(/cbox_/ , '');

        $('input[type="hidden"]', this).attr('value', thisValue);
        fixnum_checkconditions(thisValue, thisSGQA, 'hidden');
    });
}
}
</script>

```

Um stattdessen die erste Spalte exklusiv zu machen, genügt es, diese Zeile (oben rot)

`$('.normal-item:last', this).removeClass('normal-item').addClass('exclusive-item')`

durch

`$('.normal-item:first', this).removeClass('normal-item').addClass('exclusive-item')`

zu ersetzen.

8.1.3. Implementierung 2

Eine neue, kürzere, flexiblere Methode ist diese.

Hier ist die Anzahl der exklusiven Spalten am Ende frei wählbar (im script). Allerdings sind diese Spalten nur am Ende möglich, nicht am Anfang.

```
<script type="text/javascript" data-author="Tony Partner">
  $(document).on('ready pjax:scriptcomplete',function(){

    var exclusiveNumber = 2; // Hier wird die Anzahl der exklusiven Spalten festgelegt

    var thisQuestion = $('#question{QID}');

    // Add some classes to the checkbox cells
    $('td.checkbox-item', thisQuestion).addClass('normal-item');
    $('tr.subquestion-list', thisQuestion).each(function(i) {
      $('td.checkbox-item', this).slice('-'+exclusiveNumber+').removeClass('normal-
item').addClass('exclusive-item');
    });

    // Listener on the checkboxes
    $('td.checkbox-item :checkbox', thisQuestion).on('change', function(e){

      if($(this).is(':checked')) {
        var thisRow = $(this).closest('tr');
        var thisCell = $(this).closest('td');

        // Identify checkbox items to uncheck
        var excludedItems = $('td.exclusive-item', thisRow);
        if ($(thisCell).hasClass('exclusive-item')) {
          excludedItems = $('td.checkbox-item', thisRow).not(thisCell);
        }

        // Uncheck the boxes
        $(excludedItems).each(function(i) {
          $('input[type="hidden"]', this).val('');
          $(':checkbox', this).prop('checked', false).trigger('change');
        });
      }
    });
  });
</script>
```

9. Matrix(Texte) mit Checkbox-Layout, Sonstigen und Exklusiven Optionen

9.1.1. Anwendungsbeispiel

Dies ist eigentlich eine Erweiterung des vorigen Beispiels. Allerdings wird hier eine Matrix(Texte) als Basis genommen.

Dadurch lassen sich sehr leicht Textfelder für „Sonstige“ Nennungen einbauen. Im Grunde sind dies die ursprünglichen Felder; die Checkboxes werden mittels javascript eingefügt.

Ein Beispiel könnte sein „Mit welchen Verkehrsmitteln gelangen Sie zu Ihrer Arbeitsstätte?“

Hierbei sind die letzten beiden Spalten exklusiv. Ein Klick in diese löscht alle eventuellen Angaben in den anderen Spalten.

	zu Fuß	Fahrrad	Bus, Bahn	Auto	Sonstiges (angeben)	Sonstiges (angeben)	Gar nicht	Keine Angabe
Adalbert Ameise	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>
Berta Biber	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Carlo Chinchilla	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text" value="Tretroller"/>	<input type="text" value="Draisine"/>	<input type="checkbox"/>	<input type="checkbox"/>
Doris Dachs	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text" value="Hubschrauber"/>	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>

Gleichzeitig wurden hier im Beispiel auch die Spaltenbreiten angepasst; die Texte benötigen ja mehr Platz als die Häkchen (dazu siehe Kapitel 2.)

9.1.2. Implementierung

Es wird nur diese eine Frage benötigt; alle Texte werden als Teilfragen eingegeben.

Dann wird dieses javascript in den Fragetext (im Quellcode-Modus) eingefügt.

```
<script type="text/javascript" charset="utf-8">

$(document).on('ready pjax:scriptcomplete',function(){

    // Identify this question
    var thisQuestion = $('#question{QID}');

    // Column-specific classes
    $('tr.subquestion-list', thisQuestion).each(function(i) {
        $('th, td', this).each(function(i) {
            $(this).addClass('column- '+i);
        });
    });
});
```



```

// Insert checkboxes
$('.answer-item.column-1, .answer-item.column-2, .answer-item.column-3, .answer-item.column-4, .answer-
item.column-7, .answer-item.column-8', thisQuestion).addClass('custom-checkbox-item');
$('.custom-checkbox-item', thisQuestion).each(function(i) {
    var thisID = $('input:text:eq(0)', this).attr('id');
    $('label', this).before('<input class="" id="'+thisID+'" value="Y" type="checkbox"
name="'+thisID.replace(/answer/, '')+'" />');
    if($('input:text:eq(0)', this).val() == 'Y') {
        $('input:checkbox:eq(0)', this).prop('checked', true);
    }
    $(this).removeClass('text-item').addClass('checkbox-item');
    $('input:text:eq(0)', this).remove();
});

// Identify exclusive items
$('.answer-item.column-1, .answer-item.column-2, .answer-item.column-3, .answer-item.column-4, .answer-
item.column-5, .answer-item.column-6', thisQuestion).addClass('non-exclusive-item');
$('.answer-item.column-7, .answer-item.column-8', thisQuestion).addClass('exclusive-item');

// Listeners for exclusive items
$('.non-exclusive-item input:checkbox', thisQuestion).on('change', function(e) {
    if($(this).is(':checked')) {
        $(this).closest('tr.subquestion-list').find('.exclusive-item input:checkbox').prop('checked',
false);
    }
});
$('.non-exclusive-item input:text', thisQuestion).on('keyup change', function(e) {
    if($.trim($(this).val()) != '') {
        $(this).closest('tr.subquestion-list').find('.exclusive-item input:checkbox').prop('checked',
false);
    }
});
$('.exclusive-item input:checkbox', thisQuestion).on('change', function(e) {
    if($(this).is(':checked')) {
        var thisItem = $(this).closest('.answer-item');
        $(this).closest('tr.subquestion-list').find('answer-
item').not(thisItem).find('input:checkbox').prop('checked', false);
        $(this).closest('tr.subquestion-list').find('input:text').val('');
    }
});
});
</script>

```

Hier folgt nun das bekannte script zur Anpassung der Spaltenbreiten.

```

<script type="text/javascript" charset="utf-8">
$(document).on('ready ajax:scriptcomplete',function(){
    var thisQuestion = $('#question{QID}');
    // Add a question class
    thisQuestion.addClass('custom-array');
    // Column-specific classes
    $('table.subquestion-list tr', thisQuestion).each(function(i) {
        $('th, td', this).each(function(i) {

```

```
        $(this).addClass('column-'+i);
    });
});
});
</script>
```

Und der css-Teil

```
<style type="text/css">.custom-array table.subquestion-list col {
    width: auto !important;
}

.custom-array table.subquestion-list thead .column-0 { width: 16%; }
.custom-array table.subquestion-list thead .column-1 { width: 7%; }
.custom-array table.subquestion-list thead .column-2 { width: 7%; }
.custom-array table.subquestion-list thead .column-3 { width: 7%; }
.custom-array table.subquestion-list thead .column-4 { width: 7%; }
.custom-array table.subquestion-list thead .column-5 { width: 21%; }
.custom-array table.subquestion-list thead .column-6 { width: 21%; }
.custom-array table.subquestion-list thead .column-7 { width: 7%; }
.custom-array table.subquestion-list thead .column-8 { width: 7%; }
</style>
```

10. Matrizen mit randomisierten Spalten

10.1.1. Anwendungsbeispiel

Manchmal ist es sinnvoll, die Antwortoptionen einer (normalen) Matrix in zufälliger Reihenfolge anzuzeigen. Natürlich nicht, wenn es sich um eine Skala handelt.

Aber es möge eine Frage sein wie „Welche Automarke bevorzugen diese Personen?“

Codiert sind die Marken in alphabetischer Reihenfolge. Es würde die Frage also immer so dargestellt werden.

Q10					
	AUDI	BMW	Chrysler	Dacia	FIAT
Adalbert Ameise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Berta Biber	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Carlo Chinchilla	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Doris Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Um nun jede Beeinflussung durch die Reihenfolge auszuschließen, bietet man die Antwortoptionen eben in dieser Weise an.

Q10					
	AUDI	FIAT	Dacia	BMW	Chrysler
Adalbert Ameise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Berta Biber	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Carlo Chinchilla	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Doris Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

10.1.2. Implementierung

Wie immer wird ein kleines javascript snippet in den Fragetext (im Quellcode-Modus) geschrieben.

```
<script type="text/javascript" charset="utf-8">
function randomArrayCols(qID){

    var thisQuestion = $('#question'+qID);

    // Index the array columns
    $('table.subquestion-list tr', thisQuestion).each(function(i) {
        $('> *', this).each(function(i) {
            $(this).attr('data-index', i);
        });
    });
    if($('table.subquestion-list tr > :last-child', thisQuestion).length > 0) {
        $('table.subquestion-list tr > :last-child', thisQuestion).addClass('col-no-answer');
    }

    // Create an array of the columns
    var columns = [];
    $('table.subquestion-list tr.answers-list:eq(0) td.answer-item', thisQuestion).each(function(i) {
        columns.push($(this).attr('data-index'));
    });

    // A function to shuffle elements or contents of arrays
    (function($){
        $.fn.shuffle = function() {
            return this.each(function(){
                var items = $(this).children();
                return (items.length)
                    ? $(this).html($.shuffle(items))
                    : this;
            });
        }

        $.shuffle = function(arr) {
            for(
                var j, x, i = arr.length; i;
                j = parseInt(Math.random() * i),
                x = arr[--i], arr[i] = arr[j], arr[j] = x
            );
            return arr;
        }
    })(jQuery);

    // Shuffle the columns array
    columns = $.shuffle(columns);
}
```

```
// Reposition the row elements
$('table.subquestion-list tr', thisQuestion).each(function(i) {
  var thisRow = $(this);
  $(columns).each(function(i, val) {
    thisRow.append($('
```

11. Matrix nach Spalte als MaxDiff

11.1.1. Anwendungsbeispiel

Dies ist ein Sonderfall einer Matrix nach Spalten.

- Es gibt nur zwei „Spalten“.
- Die Antworten in den beiden Spalten dürfen nicht identisch sein.

Als Musterbeispiel dient die Bestimmung des „besten“ und des „schlechtesten“ aus einer Liste von Objekten.

Ein kleines MaxDiff-Beispiel		
am schlechtesten		am besten
<input type="radio"/>	Adalbert Ameise	<input type="radio"/>
<input checked="" type="radio"/>	Berta Biber	<input type="radio"/>
<input type="radio"/>	Doris Dachs	<input type="radio"/>
<input type="radio"/>	Carlo Chinchilla	<input type="radio"/>

Hier sieht man, dass eine doppelte Eingabe in derselben Zeile gesperrt ist.

11.1.2. Implementierung

Auch hier wird der javascript Code in den Fragetext (im Quellcode-Modus) geschrieben.

```
<script type="text/javascript" charset="utf-8">
function maxDiff(qID, randomize) {

    // Identify some elements
    var thisQuestion = $('#question'+qID);
    var thisTable = $('table.subquestion-list:eq(0)', thisQuestion);

    // Assign a new question class
    $(thisQuestion).addClass('max-diff-array');

    // Move the columns
    $('thead tr:eq(0)', thisTable).prepend($('thead tr:eq(0) th:eq(1)', thisTable));
    $('tr.answers-list', thisTable).each(function(i){
    $('td.answer-item:eq(0)', this).prependTo(this);
    });

    // Random rows
    if(randomize) {
        var rowsArr = [];
        $('tr.answers-list', thisTable).each(function(i){
```

```

        $(this).attr('data-index', i);
        rowsArr.push(i);
    });
    shuffleArray(rowsArr);
    $(rowsArr).each(function(i){
        $('tbody', thisTable).append($('tr[data-index="'+this+'"]', thisTable));
    });
}

// Prevent clicking twice in the same row
$('input[type="radio"]', thisQuestion).on('click', function () {

    $('input[type="radio"]', thisQuestion).prop('disabled', false);
    $('input[type="radio"]:checked', thisQuestion).each(function(i) {
        var thisRow = $(this).closest('tr.answers-list');
        $('input[type="radio"]', thisRow).not(this).prop('disabled', true);
    });
});

// Fix up the row classes
var rowClass = 1;
$('tr.answers-list', thisTable).each(function(i) {
    $(this).addClass('array'+(2-(i%2)));
});
}
function shuffleArray(array) {
    for (var i = array.length - 1; i > 0; i--) {
        var j = Math.floor(Math.random() * (i + 1));
        var temp = array[i];
        array[i] = array[j];
        array[j] = temp;
    }
    return array;
}

$(document).ready(function(){
    // Call the maxDiff() function
    // Set the second parameter to true for randomized rows
    maxDiff({QID}, true);
});
</script>

```

In der letzten Zeile „maxDiff({QID}, true|false“ kann man wählen, ob die Objekte in randomisierter Reihenfolge angezeigt werden sollen („true“), oder nicht („false“).

Dann gibt es noch eine Einstellung mittels css (damit die Objekte in der Mitte zentriert angezeigt werden)

```

<style type="text/css">.ls-answers tbody .answertext {
    text-align: center;
}
</style>

```

12. Semantisches Differential mit „KA“

12.1.1. Anwendungsbeispiel

Wenn man ein „semantisches Differential“ und dabei auch eine Spalte „keine Angabe“ anzeigen möchte, sieht man diese Unschönheit.

Q12									
	+++	++	+	0	+	++	+++	Keine Angabe	
lang	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	kurz
dick	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	dünn
hoch	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	tief

Die „Keine Angabe“-Spalte liegt innerhalb des Differentials; besser sie wäre außerhalb der beiden Polpaare. Dazu könnte man nun die Frage als „Nicht-Pflichtfrage“ anlegen, was dieses ergibt:

Q12										
	+++	++	+	0	+	++	+++		Keine Antwort	
lang	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	kurz	<input checked="" type="radio"/>	
dick	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	dünn	<input checked="" type="radio"/>	
hoch	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	tief	<input checked="" type="radio"/>	

Ist möglich, sollte aber vermieden werden. Semantische Differentiale beinhalten normalerweise Itempaare, deren Zuordnung nicht auf Wissen, sondern auf Gefühlen und Meinungen beruht. Daher kann dies auch immer beantwortet werden, und die Frage daher als Pflichtfrage angelegt werden.

Mit ein wenig javascript Code erhalten wir dann so etwas:

Q12									
	+++	++	+	0	+	++	+++		keine Angabe
lang	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	kurz	<input type="radio"/>
dick	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	dünn	<input type="radio"/>
hoch	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	tief	<input type="radio"/>

Hier sind zusätzlich die Spaltenbreiten angepasst, siehe Kapitel 2., und die „KA“-Spalte abgesetzt.

12.1.2. Implementierung

Wie immer wird dieser javascript Code in den Fragetext (im Quellcode-Modus) eingefügt.

```
<script type="text/javascript" charset="utf-8">

$(document).on('ready ajax:scriptcomplete',function(){

    // Define this question
    var thisQuestion = $('#question{QID}');

    // Move the last column
    $('col:nth-last-child(2)', thisQuestion).before($('col:last-child', thisQuestion));

    $('table.subquestion-list tr', thisQuestion).each(function(i) {
        $('> *:nth-last-child(2)', this).before($('> *:last-child', this));
    });

    // Dieser Teil ist nur interessant, wenn man individuelle Spaltenbreiten haben möchte
    // Add a question class
    thisQuestion.addClass('custom-array');

    // Column-specific classes
    $('table.subquestion-list tr', thisQuestion).each(function(i) {
        $('th, td', this).each(function(i) {
            $(this).addClass('column-'+i);
        });
    });
});
</script>
```

Und hier einige Zeilen css, die aber nur notwendig sind, wenn man individuelle Spaltenbreiten oder die „KA“-Spalte farblich absetzen möchte.

```
<style type="text/css">
/* Spaltenbreiten definieren, Summe muss natürlich 100% ergeben */
.custom-array table.subquestion-list col {
    width: auto !important;
}

.custom-array table.subquestion-list thead .column-0 { width: 15% !important; }
.custom-array table.subquestion-list thead .column-1 { width: 8% !important; }
.custom-array table.subquestion-list thead .column-2 { width: 8% !important; }
.custom-array table.subquestion-list thead .column-3 { width: 8% !important; }
.custom-array table.subquestion-list thead .column-4 { width: 8% !important; }
.custom-array table.subquestion-list thead .column-5 { width: 8% !important; }
.custom-array table.subquestion-list thead .column-6 { width: 8% !important; }
.custom-array table.subquestion-list thead .column-7 { width: 8% !important; }
.custom-array table.subquestion-list thead .column-8 { width: 15% !important; }
.custom-array table.subquestion-list thead .column-9 { width: 14% !important; }
```

```
.custom-array th.answertext.control-label.column-0 {
  text-align:right !important;
}

/* Zelle für "keine Angabe" farblich absetzen, Trennlinie einfügen
   td.answer_cell_8 -> 8 = code dieser Antwort */
td.answer_cell_8 {
  background-color: #efefef !important;
  border-left: 2px solid #000 !important;
}
</style>
```

13. Matrizen mit ausgeblendeten Zellen

13.1. Matrix

13.1.1. Anwendungsbeispiel

Manchmal ist es so, dass einige Zellen einer Matrix von vornherein nicht auswählbar sein sollen.

Nehmen wir als Beispiel die Vorstellungen des Kasperle-Theaters in der nächsten Woche. Der Teilnehmer soll sich eine der Vorstellungen aussuchen.

Am Montag finden gar keine Frühvorstellungen statt, am Samstag keine Spätvorstellungen und am Sonntag nur eine Vorstellung um 15:00 Uhr.

Dann wäre es unsinnig, diese Optionen als Auswahlmöglichkeit zuzulassen.

Matrix					
	10:00 Uhr	13:00 Uhr	15:00 Uhr	18:00 Uhr	21:00 Uhr
Montag			<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Dienstag	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Mittwoch	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Donnerstag	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Freitag	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Samstag	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>		
Sonntag			<input type="radio"/>		

13.1.2. Implementierung

In den Fragetext (im Quellcode-Modus) werden diese wenigen Zeilen javascript-Code eingefügt.

```
<script type="text/javascript" charset="utf-8">
  $(document).ready(function() {
    // Remove in row 1
    $('#question{QID} tr[id^="jvatbd"]:eq(0) .radio-item:eq(0) *').remove();
    $('#question{QID} tr[id^="jvatbd"]:eq(0) .radio-item:eq(1) *').remove();
    // Remove in row 6
    $('#question{QID} tr[id^="jvatbd"]:eq(5) .radio-item:eq(3) *').remove();
    $('#question{QID} tr[id^="jvatbd"]:eq(5) .radio-item:eq(4) *').remove();
    // Remove in row 7
    $('#question{QID} tr[id^="jvatbd"]:eq(6) .radio-item:eq(0) *').remove();
    $('#question{QID} tr[id^="jvatbd"]:eq(6) .radio-item:eq(1) *').remove();
    $('#question{QID} tr[id^="jvatbd"]:eq(6) .radio-item:eq(3) *').remove();
    $('#question{QID} tr[id^="jvatbd"]:eq(6) .radio-item:eq(4) *').remove();
  });
</script>
```

Wichtig sind die Angaben in „eq(x)“.

Das erste gibt die Zeile an (Start bei 0), das zweite die Spalte (ebenfalls bei 0 startend)

Der Platzhalter {QID} fügt immer die aktuelle FragenID ein, darf also keinesfalls geändert werden.

13.2. Matrix(Zahlen) mit Checkbox-Layout

13.2.1. Anwendungsbeispiel

Analog wie vor, nur dass hier z.B. mehrere Vorstellungen ausgewählt werden können.

Matrix(Zahlen) mit Checkbox-Layout					
	10:00 Uhr	13:00 Uhr	15:00 Uhr	18:00 Uhr	21:00 Uhr
Montag			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Dienstag	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Mittwoch	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Donnerstag	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Freitag	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Samstag	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Sonntag			<input type="checkbox"/>		

13.2.2. Implementierung

Ebenfalls ziemlich ähnlich der vorigen.

```
<script type="text/javascript" charset="utf-8">
$(document).ready(function() {
    // Remove in row 1
    $('#question{QID} tr[id^="javatbd"]:eq(0) .checkbox-item:eq(0) *').remove();
    $('#question{QID} tr[id^="javatbd"]:eq(0) .checkbox-item:eq(1) *').remove();
    // Remove in row 6
    $('#question{QID} tr[id^="javatbd"]:eq(5) .checkbox-item:eq(3) *').remove();
    $('#question{QID} tr[id^="javatbd"]:eq(5) .checkbox-item:eq(4) *').remove();
    // Remove in row 7
    $('#question{QID} tr[id^="javatbd"]:eq(6) .checkbox-item:eq(0) *').remove();
    $('#question{QID} tr[id^="javatbd"]:eq(6) .checkbox-item:eq(1) *').remove();
    $('#question{QID} tr[id^="javatbd"]:eq(6) .checkbox-item:eq(3) *').remove();
    $('#question{QID} tr[id^="javatbd"]:eq(6) .checkbox-item:eq(4) *').remove();
});
</script>
```

Der einzige Unterschied ist – logischerweise – dass hier „checkbox-items“ angesprochen werden im Gegensatz zu den vorherigen „radio-items“.

13.3. Matrix(Texte)

13.3.1. Anwendungsbeispiel

Auch bei einer Matrix(Texte) kann es Situationen geben, bei denen Eingaben in bestimmte Kombinationen sinnlos sind. Hier ein etwas gewolltes Beispiel:

Matrix(Texte)					
	Farbe	Länge	Gewicht	Haarlänge	Flügelzahl
Ameise	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Biene	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Condor	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Dachs	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Eidechse	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Flunder	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

In diesem Fall werden die Eingabefelder nicht entfernt (es ist mir nicht gelungen), sondern „disabled“. Dadurch wird automatisch die Vergrauung erzeugt, sowie der „Sperrcursor“.

13.3.2. Implementierung

Im Grunde sieht es ähnlich aus wie zuvor.

```
<script type="text/javascript" charset="utf-8">
$(document).on('ready pjax:scriptcomplete',function(){
  // Reihe 1
  $("#answer{SGQ}Y001_X004").prop('disabled', true);
  $("#answer{SGQ}Y001_X005").prop('disabled', true);
  // Reihe 3
  $("#answer{SGQ}Y003_X004").prop('disabled', true);
  // Reihe 4
  $("#answer{SGQ}Y004_X005").prop('disabled', true);
  // Reihe 5
  $("#answer{SGQ}Y005_X005").prop('disabled', true);
  // Reihe 6
  $("#answer{SGQ}Y006_X004").prop('disabled', true);
  $("#answer{SGQ}Y006_X005").prop('disabled', true);
  // Reihe 7
  $("#answer{SGQ}Y007_X001").prop('disabled', true);
  $("#answer{SGQ}Y007_X002").prop('disabled', true);
  $("#answer{SGQ}Y007_X004").prop('disabled', true);
  $("#answer{SGQ}Y007_X005").prop('disabled', true);
});
</script>
```

Allerdings wird hier direkt auf die Codes der Zeilen und Spalten referiert (ich hatte bereits einmal gesagt, dass ich es vorziehe, die Spalten (y-Achse) mit „Y001, Y002,...“ zu kodieren, die Zeilen (x-Achse) mit „X001, X002, ...“).

Der Platzhalter {SGQ} fügt immer die aktuelle Studien-/Gruppen-/FragenID ein, darf also keinesfalls geändert werden.

14. Matrix(Zahlen/Texte) mit Filterung der Spalten

(nicht kompatibel mit Version 5.3.27.)

14.1.1. Anwendungsbeispiel

Es geht darum, in einer Matrix nur die Zeilen / Spalten anzuzeigen, die in irgendeiner Vorfrage als relevant benannt wurden.

Beispiele:

1. Welche dieser Automarken haben Sie schon besessen?
 2. Welche dieser hatten die Farbe „Rot“?
-
1. An welchen Wochentagen steht Ihnen das Pferd laut Vertrag zur Verfügung?
 2. An welchen dieser Tage sind Sie in der vergangenen Woche ausgeritten?

Man sieht, in der zweiten Frage sollten nur die Antwortoptionen gezeigt werden, die in der ersten Frage genannt wurden.

Bei den Zeilen einer Matrix ist dies leicht zu bewerkstelligen mithilfe der Teilfragenrelevanz. Also nur in die „Relevanz-Gleichung der Teilfrage die Bedingung eintragen, unter welcher diese Teilfrage angezeigt werden soll.

Code	Teilfrage	Relevanz-Gleichung
Y001	Adventskranz basteln	Q1_SQ001=="Y"

Ist klar; „Adventskranz basteln“ wird nur angezeigt, wenn die Teilfrage „SQ001“ der Frage „Q1“ angewählt wurde – wohl eine Mehrfachnennungsfrage, da hier das Anwählen als „Y“ gespeichert wird.

Für die Spalten einer Matrix gibt es aber keine Relevanzgleichung. Daher muss man anderweitig herangehen, also mit etwas javascript.

Es gibt also irgendeine Frage, wie

Welche dieser Personen kennen Sie?

Bitte wählen Sie die zutreffenden Antworten aus:

<input checked="" type="checkbox"/> Adalbert Ameise	<input checked="" type="checkbox"/> Florian Flamingo
<input checked="" type="checkbox"/> Berta Biber	<input type="checkbox"/> Gabriele Gnu
<input type="checkbox"/> Carlo Chinchilla	<input type="checkbox"/> Horst Habicht
<input type="checkbox"/> Doris Dachs	<input checked="" type="checkbox"/> Joachim Jaguar
<input checked="" type="checkbox"/> Emil Eidechse	<input checked="" type="checkbox"/> Karin Kröte

Dann muss die nächste Frage eben nur für die bekannten Personen gestellt werden.

Mit welchen dieser Personen würden Sie die folgenden Freizeitgestaltungen gerne gemeinsam unternehmen?

	Adalbert Ameise	Berta Biber	Emil Eidechse	Florian Flamingo	Joachim Jaguar	Karin Kröte
Adventskranz basteln	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bäume fällen	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Champagner schlürfen	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Domino spielen	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Eislaufen	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Dies ist besonders interessant, wenn beide Achsen gefiltert werden sollen; sonst könnte man sich auch überlegen, die Matrix einfach zu drehen.

14.1.2. Implementierung

Um dieses Problem zu lösen, gibt es ein plugin namens „hideEmptyColumn“ von Denis Chenu.

Hier zum Download:

[Plugin "hideEmptyColumn"](#)

Da dieses Tutorial aber alle diejenigen ansprechen soll, die keine Rechte haben, plugins zu installieren (seien es Studenten oder Nutzer der bei der LimeSurvey GmbH gehosteten Version), hier die Lösung mittels javascript und css.

Nebenbei bemerkt: Das plugin macht nichts anderes als dieses javascript aufzurufen.

Also – wie immer – kommt das folgende script in den Fragetext (im Quellcode-Modus)

```

<script>
/*
 * JavaScript functions to hide empty answers column in LimeSurvey
 * This allow using Expression Manager to hide answers in array question type
 *
 * @author Denis Chenu <denis@sondages.pro>
 * @copyright 2013-2015 Denis Chenu <http://sondages.pro>
 * @license magnet:?xt=urn:btih:1f739d935676111cfff4b4693e3816e664797050&dn=gpl-3.0.txt GPL-v3-or-Later
 * @version 1.1
 *
 */
$("table.question,table.ls-answers").each(function(){
    var basetable=$(this);
    basetable.find('col.odd,col.even,col.ls-col-odd,col.ls-col-even').addClass('havewidth');
    $(this).find("thead th").each(function(){
        if($.trim($(this).html())==""){
            if(basetable.find('col.havewidth').length){
                basetable.find('col.havewidth').removeAttr('width').removeClass('havewidth');
            }
            var colindex=$(this).parent('tr').find('td,th').index($(this));
            basetable.find('col').eq(colindex).css('width',0);
            basetable.find('col').eq(colindex).width(0);
            basetable.find('tr').each(function(){
                $(this).find('td,th').eq(colindex).addClass('hideEmptyColumn-hiddencolumn').children().hide();
            });
            basetable.addClass('hideEmptyColumn-widthhiddencolumn');
        }
    });
});
</script>

```

Es folgt ein kleines css script

```

<style type="text/css">
td.hideEmptyColumn-hiddencolumn,th.hideEmptyColumn-hiddencolumn{
    overflow:hidden !important;
    border-spacing:0 !important;
    border-collapse:collapse !important;
    padding:0 !important;
    width: 0 !important;
    height: 0 !important;
    min-width: 0 !important;
}
td.hideEmptyColumn-hiddencolumn *,th.hideEmptyColumn-hiddencolumn * {
    display:none;
}
table.ls-answers > tbody > tr > td.hideEmptyColumn-hiddencolumn{border-collapse:collapse !important;}
table.hideEmptyColumn-widthhiddencolumn{
    border-collapse:collapse !important;
}
</style>

```

Und nun kommt noch der Teil, in welchem man bestimmt, welche Spalten denn „leer“ sind und welche nicht. Dazu wird in den Teilfragen der x-Achse auf die Antworten der Vorfrage referiert.

X-Skala

Position	Code	Teilfrage
☰	1	{Q14a_1.shown}
☰	2	{Q14a_2.shown}
☰	3	{Q14a_3.shown}

In {Q14a_1.shown} steht also entweder der Text der Teilfrage von Q14a – falls ausgewählt, oder nichts. In dem Fall wäre die Spalte leer – wie der Name des plugins sagt.

Und: Wie man sich leicht vorstellen kann, darf die Vorfrage zur Auswahl nicht auf derselben Seite sein wie die Matrix. Also entweder „Frage für Frage“ anzeigen, oder die Vorfrage in eine separate Gruppe (wie ich es im der Beispiel-Datei auf gemacht habe)

15. Dual Matrix – Zweite Spalte gefiltert nach erster Spalte

15.1.1. Anwendungsbeispiel

Häufig kommt es vor, dass mehrere Stufen anbieten möchte um ein Element aus einer sehr großen Menge auszuwählen.

Sei es:

- Kontinent → Staat (statt alle ca. 200 Staaten der Erde in einer Liste anzubieten)
- Automarke → Modell
- Tier-Ordnung → Tier-Familie
- Bundesland → Landkreis

Das Problem hierbei ist, dass weder die Einfachnennung mittels Liste(Optionsfelder) noch die Liste(Klappbox) über eine Antwortrelevanz verfügt, ähnlich der Teilfragen-Relevanz bei Matrizen und Mehrfachnennungen.

Es gibt natürlich Lösungsmöglichkeiten.

Zum Beispiel den bekannten konventionelle Weg:

Zwischen die beiden Fragen wird eine (versteckte) Mehrfachnennungsfrage und eine (ebenfalls versteckte) Gleichungsfrage gesetzt. Die Gleichungsfrage setzt das gewählte Element in der Mehrfachfrage; diese dient als Matrixfilter für die zweite Frage.

Diese Lösung hat aber den Nachteil, dass die beiden Klappboxen in zwei verschiedenen Fragen stehen, dadurch von der Optik her nicht optimal. Gut, man kann auch hier mit einem bisschen Styling nachhelfen und die beiden Fragen aneinanderrücken.

Oder man stellt nur die zweite Ebene als Drop-Down dar, was dann in einer langwierigen Suche mit Scrollen endet.

Oder man benutzt ein „autocomplete“ wie es in Kapitel 3.2. des Tutorials „Mehrfachnennung, mehrfache kurze Texte“ gezeigt wird.

Hier ist eine – wie ich denke – elegante Lösung mittels einer „Dual Matrix“ mit sogenannter „Listenauswahl-Präsentation“. Das heißt ja nichts anderes, als dass keine Klick-Kästchen angezeigt werden, sondern zwei Drop-Downs.

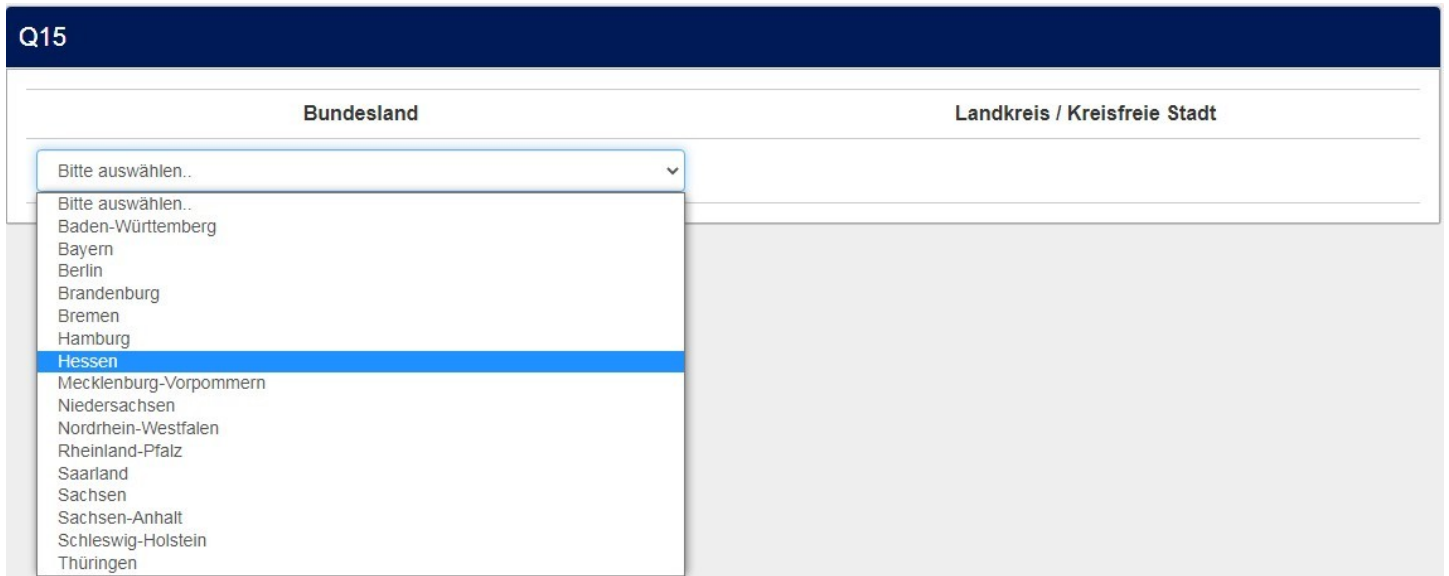
Beim ersten Auftauchen der Frage sieht diese so aus (Teilfragenbreite auf „0“ gesetzt):

The screenshot shows a form titled "Q15" with a dark blue header. Below the header, there are two columns. The left column is labeled "Bundesland" and contains a dropdown menu with the text "Bitte auswählen.." and a small downward arrow. The right column is labeled "Landkreis / Kreisfreie Stadt".

Erst nach Auswahl eines Bundeslandes

Q15

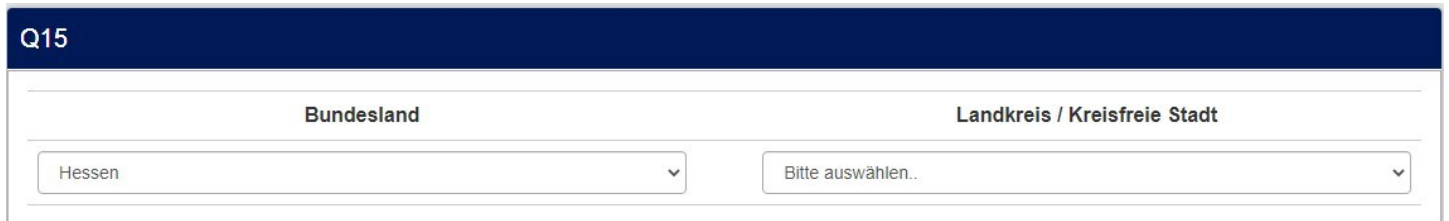
Bundesland	Landkreis / Kreisfreie Stadt
Bitte auswählen..	



wird die zweite Klappbox sichtbar,

Q15

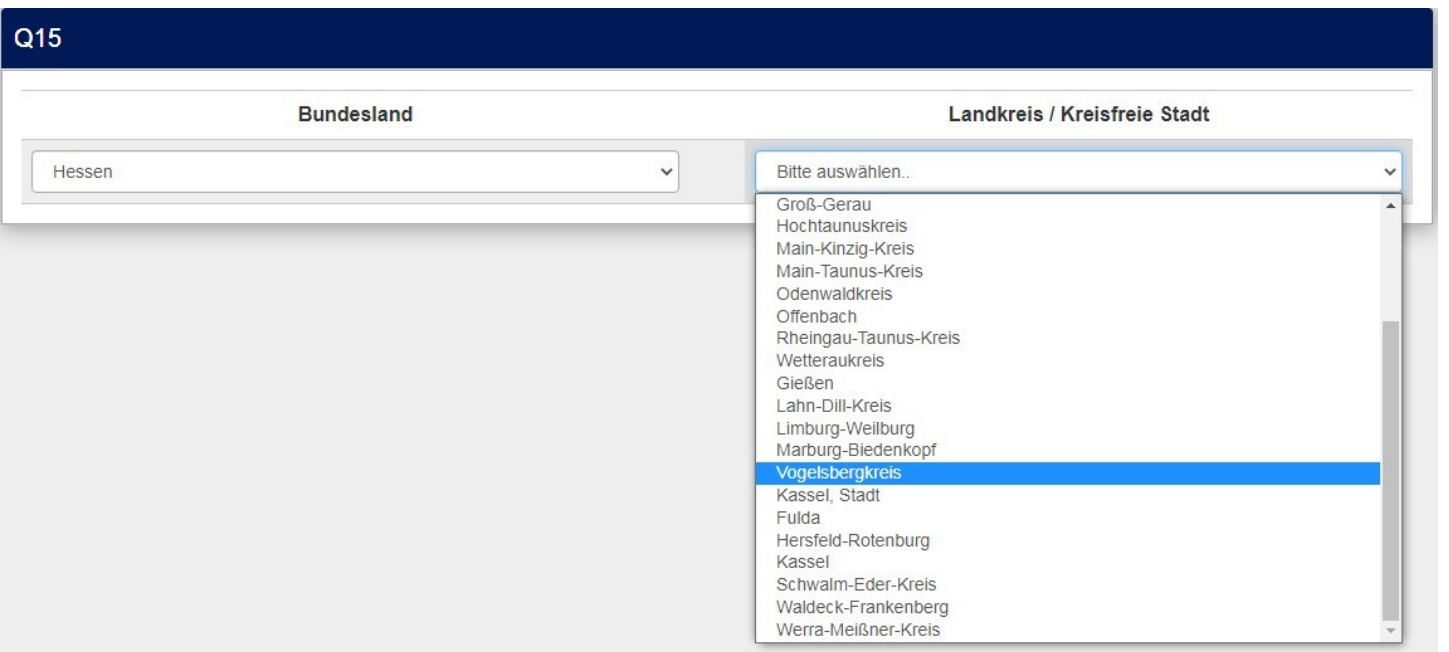
Bundesland	Landkreis / Kreisfreie Stadt
Hessen	Bitte auswählen..



in welcher nur noch die Landkreise dieses Bundeslandes angezeigt werden.

Q15

Bundesland	Landkreis / Kreisfreie Stadt
Hessen	Bitte auswählen..



15.1.2. Implementierung

Das hierfür benutzte script hat mehrere Autoren – zunächst entwickelt von Denis Chenu (im Forum: DenisChenu) und angepasst für die neueren Versionen von Luciano Fanfoni (hier im Forum: IFanfoni).

Doch zunächst zur Codierung der Antwortoptionen. Es ist klar, dass hier eine gewisse Systematik vorliegen muss, damit das script zusammengehörige Elemente erkennt.

Zunächst: Jeder Antwortcode muss 5 Zeichen lang sein. (Müsste er theoretisch nicht, aber die Entwickler haben es nun einmal so gemacht.)

Daher sind die Bundesländer alphabetisch sortiert und codiert mit ihrem amtlichen Länderschlüssel, aufgefüllt mit „000“. Damit ist auch schon klar, wie die Landkreise codiert sind. Sie bekommen auch ihre amtliche Kreisnummer (dort ist der Länderschlüssel ja als die ersten zwei Stellen enthalten)

Code	Antwortoptionen
08000	Baden-Württemberg
09000	Bayern
11000	Berlin
12000	Brandenburg
04000	Bremen
02000	Hamburg
06000	Hessen

05974	Soest
05978	Unna
06411	Darmstadt, Stadt
06412	Frankfurt, Stadt
06413	Offenbach, Stadt
06414	Wiesbaden, Stadt
06431	Bergstraße

Dann wird das script wie immer in den Fragetext (im Quellcode-Modus) eingefügt.

```
<script type="text/javascript" charset="utf-8">

$(document).ready(function() {

    $('.array-flexible-dual-scale .dropdown-item select').each(function(i) {
        if($('option[selected="selected"]', this).length == 0) {
            $(this).prepend('<option value="">Please choose...</option>');
            $(this).val('');
        }
    });
    selectFilterDualScale({QID});
});

/*
Copyright 2012 Denis Chenu for <http://www.sondages.pro>
Distributed under MIT licence

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Luciano Fanfoni - lfanfoni@gmail.com
Modifiche per compatibilità versioni 2.7X

*/

/* Function to filter the second part of an Array dual scale (with dropdown attribute) by the first part
var qID : the number of question to filter
*/
function selectFilterDualScale(qID){
    if($("#question"+qID).hasClass('array-flexible-dual-scale')){

        //modifiche Steve
        //RIPRENDE IL VECCHIO
        //$("#question"+qID+" table.question tbody tr").each(function(index){
```

```

$("#question"+qID+" table.questions-list tbody tr").each(function(index){

    var idSelectFiltering = jqSelector($(this).find("select").eq(0).attr('id'));
    var idSelectFilter = jqSelector($(this).find("select").eq(1).attr('id'));
    var idNewSelectFilter = jqSelector('select'+qID+'_'+index);

        // if(index==0) $('select',this).attr('style','font-weight: bold');

    //NUOVO DI STEVE
        var NewSelectElement = "<select id='"+idNewSelectFilter+"' class='form-control'><option
value=' '>"+$("#idSelectFilter+" option[value='']:first").text()+"</option></select>";

    /* VECCHIO SCRIPT DI DENIS
        var NewSelectElement = "<select id='"+idNewSelectFilter+"'><option value=' '>"+$
("#idSelectFilter+" option[value='']:first").text()+"</option></select>";*/
    //RIPRENDE DENIS
    $("#idSelectFilter).hide();
    $("#idSelectFilter").after(NewSelectElement);

    $("#idSelectFiltering).change(function(){
        $("#idSelectFilter).val('');
        $('#idNewSelectFilter).val('');
        var valuefilter=$(this).val().substring(0, $(this).val().length - 3);
        $('#idNewSelectFilter+' option').not(':first').remove();
        if($(this).val()==""){
            $('#idNewSelectFilter).hide();
        }else{
            $('#idNewSelectFilter).show();
            $("#idSelectFilter).find('option').each(function(){
                if($(this).attr('value').substring(0, $(this).attr('value').length - 3)==valuefilter){
                    $(this).clone().appendTo('#idNewSelectFilter);
                }
            });
        }
    });

    $("#idNewSelectFilter).change(function(){
        $('#idSelectFilter).val($(this).val());
        saveval=$('#idSelectFiltering).val();
        $('#idSelectFilter).trigger('change');
        if($(this).val()==""){
            $('#idSelectFiltering).val(saveval);
            $('#idSelectFiltering).trigger('change');
            $('#idSelectFilter).val($(this).val(""));
        }
    });

    if($("#idSelectFiltering).val()!=''){
        //if($("#idSelectFiltering+' option[selected="selected"]').length != 0){
            var valuefilter=$("#idSelectFiltering).val().substring(0, $
$("#idSelectFiltering).val().length - 3);

```



```

    $('#'+idSelectFilter+' option').each(function(){
        if($(this).attr('value').substring(0, $(this).attr('value').length - 3)==valuefilter){
            $(this).clone().appendTo('#'+idNewSelectFilter);
        }
    });

    if($("#"+idSelectFilter).val()!=''){
        $('#'+idNewSelectFilter).val($("#"+idSelectFilter).val());
    }
}
}

$('#'+idNewSelectFilter).hide();
}

$('#td:last',this).attr('style','text-align:left');

}); // each iniziale

$("#question"+qID+" table.question .ddarrayseparator").attr('style','padding: 0px 0px; width: 1%');
$("#question"+qID+" table.question thead tr th:last").attr('style','text-align:left');

} // chiude if
} //chiude funzione
function jqSelector(str) {
    return str.replace(/([\;&.\.+*\~':"!\^#%@\[\]\(\)\>=\|])/g, '\\$1');
}
</script>

```

16. Textausrichtung

16.1.1. Anwendungsbeispiel

Da im normalerweise zur Demonstration verwendeten Theme die hier gezeigten Änderungen bereits eingebaut sind, wird hier das Standardtheme „fruity“ benutzt.

Was für Matrizen, die als „semantisches Differential“ benutzt werden, ganz gut und intuitiv aussieht,

Q16

	3	2	1	0	1	2	3	
Adalbert Ameise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Peter Pfau
Berta Biber	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Quasimodo Qualle
Carlo Chinchilla	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Renate Reiher
Doris Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Sabine Stieglitz
Emil Elefant	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Theobald Tapir
Florian Flamingo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Ulrike Uhu
Gabriele Gnu	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Valentina Viper

nämlich die rechtsbündige Anordnung der „linken Seite“, sieht in einer normalen Matrix doch etwas merkwürdig aus.

Q16

	riesig	sehr groß	groß	nicht so groß	eher klein	klein	sehr klein	winzig
Adalbert Ameise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Berta Biber	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Carlo Chinchilla	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Doris Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Emil Elefant. Das ist dieser Typ mit den großen Ohren (aber nur, wenn er aus Afrika kommt) und der langen Nase, im Volksmund auch Rüssel genannt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Florian Flamingo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Gabriele Gnu	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Außerdem stört mich, dass die Texte der Antwortoptionen von vornherein „fett“ sind, und bei mehrzeiligen Items die Radio-Buttons am oberen Rand der Zelle kleben.

Dies soll nun behoben werden.

16.1.2. Implementierung

Es werden nur drei winzige css-Codes benutzt, die - wie immer - in den Fragetext (im Quellcode-Modus) einzufügen sind.

Da dies sicher Änderungen sind, die umfrageweit gelten sollen, ist es besser, diese in die „custom.css“ zu schreiben (dann aber ohne die <style>...</style> tags.

```
<style type="text/css">
/* Teilfragentexte linksbündig */
.ls-answers tbody .antworttext {
  text-align: left;
}

/* Überschrift nicht „fett“, sondern „normal“ */
.ls-answers .ls-heading th {
  font-weight: normal;
}

/* Radio-Buttons vertikal zentriert */
.ls-answers td.radio-item,
.ls-answers td.checkbox-item {
  vertical-align: middle;
}
</style>
```

Mit „.ls-answers td.checkbox-item“ nehmen wir gleich die Zentrierung für Matrizen(Zahlen) im Checkbox-Layout mit.

Q16

	riesig	sehr groß	groß	nicht so groß	eher klein	klein	sehr klein	winzig
Adalbert Ameise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Berta Biber	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Carlo Chinchilla	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Doris Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Emil Elefant. Das ist dieser Typ mit den großen Ohren (aber nur, wenn er aus Afrika kommt) und der langen Nase, im Volksmund auch Rüssel genannt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Florian Flamingo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Gabriele Gnu	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Ich persönlich sehe einen Vorteil der „nicht-fetten“ Überschrift darin, dass man nun eigene Auszeichnungen (einzelne Wörter zur Verdeutlichung fett) besser bewerkstelligen kann.

17. Reset-Button in Matrix

17.1.1. Anwendungsbeispiel

Ehrlich gesagt, weiß ich nicht, warum man dieses anwenden sollte.

In Matrizen kann man aus einer Zeile die Auswahl nicht mehr löschen, sondern nur ändern (daher stammt ja auch der Begriff „Radio-Button“; ein gedrückter Knopf springt nur heraus, wenn man einen anderen drückt)

Wenn nun eine Matrix als Pflichtfrage angelegt ist, muss in allen Zeilen eine Antwortoption ausgewählt werden. Daher muss hier – falls es erlaubt sein soll – eine eigene Antwortoption „Weiß nicht / keine Angabe“ zusätzlich vorhanden sein.

Bei Matrizen, die nicht obligatorisch sind, wird, falls es entsprechend im Menue „Präsentation“ ausgewählt wurde, eine bereits vorbesetzte Spalte „Keine Angabe“ zusätzlich angezeigt.

Zeige "Keine Antwort"

 An

Q17						
	1	2	3	4	5	Keine Antwort
Adalbert Ameise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Berta Biber	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Carlo Chinchilla	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Doris Dachs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

Das ist also ein „Notausstieg, wenn jemand z.B. bereits einen Wert ausgewählt hat, und sich dann denkt „Ach nee, doch lieber nicht“.

Wenn man nun diese „Keine Antwort“-Spalte nicht möchte, sei es, dass deren Aufforderungscharakter, alles so zu belassen und einfach auf „Weiter“ zu klicken, zu groß zu sein scheint, sei es aus Furcht, Teilnehmer könnten glauben, dies sei bereits vorausgefüllt und dürfe nicht geändert werden, so setzt man in „Präsentation“

Zeige "Keine Antwort"

 Aus

Um nun doch eine „verclickte“ Eingabe wieder löschen zu können gibt es diese Lösung.

Q17

	1	2	3	4	5
Adalbert Ameise Zeile zurücksetzen	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Berta Biber Zeile zurücksetzen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
Carlo Chinchilla Zeile zurücksetzen	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Doris Dachs Zeile zurücksetzen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Und, wenn man den Klick bei „Adalbert Ameise“ wieder löschen möchte...

Q17

	1	2	3	4	5
Adalbert Ameise Zeile zurücksetzen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Berta Biber Zeile zurücksetzen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
Carlo Chinchilla Zeile zurücksetzen	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Doris Dachs Zeile zurücksetzen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Wie anfangs bereits gesagt, ich habe keine Ahnung, warum man so vorgehen sollte.

Matrizen sind m.E. immer beantwortbar, da es dort meist nicht um Wissen, sondern um Einschätzungen geht. Ich kann auch das Alter und die Größe einer Brauerei schätzen, deren Namen ich noch nie gehört habe. Das gibt dem Forscher meist sogar mehr Erkenntnis als reines Wissen (Aber da spricht der alte Markt-, Meinungs- und Sozialforscher aus mir).

Sollte es in einer Matrix Teilfragen geben, die für den Teilnehmer nicht zutreffend sind, hätten sie bereits in einer Vorfrage eliminiert werden sollen.

Genug davon.

17.1.2. Implementierung

Ein kleines javascript und ebenso ein winziges Stückchen css werden in den Fragetext (im Quellcode-Modus) eingefügt.

```
<script type="text/javascript" charset="utf-8">

  $(document).on('ready ajax:scriptcomplete',function(){

    var thisQuestion = $('#question{QID}');

    // Insert the reset buttons
    $('#table.subquestion-list .answertext', thisQuestion).append('<div><button type="button" class="btn btn-sm btn-info inserted-reset">Zeile zurücksetzen</button></div>');

    // Listener on the reset buttons
    $('.inserted-reset', thisQuestion).on('click', function(e) {
      var thisRow = $(this).closest('tr');
      $('input:radio', thisRow).prop('checked', false);
      checkconditions('', $('input:radio:eq(0)', thisRow).attr('name'), 'radio');
    });
  });
</script>
```

Und das css

```
<style type="text/css">
  #question{QID} table.subquestion-list th,
  #question{QID} table.subquestion-list td{
    vertical-align: middle;
  }

  .inserted-reset {
    margin-top: 5px;
    padding: 2px 12px;
    font-style: italic;
  }
</style>
```

Im Code befindet sich die Zeilenbreite **class="btn btn-sm btn-info inserted-reset"**

„btn-info“ gibt die Farbe des Buttons. Dies sind die Standard bootstrap-Klassen. Kann man ändern.



Außerdem kann man in der css-Klasse „inserted-reset“ Änderungen und Zusätze vornehmen. Hier z.B. die Kursivschreibweise des Textes.

18. Art „semantisches Differential“, aber Items in der Mitte

18.1.1. Anwendungsbeispiel

Wei der Titel schon sagt, ist dies nichts anderes als das normale semantische Differential, allerdings mit einer anderen Darstellung.

Matrix								
sehr	ein wenig	ein bisschen			ein bisschen	ein wenig	sehr	
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Manche denken, dass "Adalbert Ameise" die geeignetste Person für das Amt des Klassen-sprechers ist	ABER	Andere dagegen halten "Berta Biber" für viel besser geeignet, dieses Amt auszuüben	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Manche denken, dass "Adalbert Ameise" die geeignetste Person für das Amt des Klassen-sprechers ist	ABER	Andere dagegen halten "Berta Biber" für viel besser geeignet, dieses Amt auszuüben	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Hier sind zwei Teilfragen dargestellt, die zunächst ähnlich aussehen, im Hintergrund aber verschieden sind.

Die obere Zeile wird mithilfe einer Tabelle „gebaut“, die untere nur mit „bootstrap“-Klassen.

Im Einzelnen:

Zeile 1 (als „table“):

```
<table align="center" border="0" cellpadding="0" cellspacing="0" style="width:100%;">
  <tbody>
    <tr>
      <td style="width:20%;height:85px;text-align:center;font-size:16px;"><strong>MITTE</strong></td>
      <td style="width:40%;height:85px;"><span style="font-size:16px;">Rechter Text</span></td>
      <td style="width:40%;height:85px;"><span style="font-size:16px;">Linker Text</span></td>
    </tr>
  </tbody>
</table>
```

Zeile 2 (mit „bootstrap“):

```
<div class="row row-no-gutters" style="font-size:16px;">
  <div class="col-sm-5">Linke Seite</div>
  <div class="col-sm-2" style="font-weight:bold">MITTE</div>
  <div class="col-sm-5">Rechte Seite</div>
</div>
```

Wichtig: Vor den Text muss das pipe-Symbol „|“ eingefügt werden, da es sich um die rechte Seite des „normalen“ semantischen Differential handelt.

18.1.2. Implementierung

Wieder werden einige scripte in den Fragentext (im Quellcode-Modus) eingefügt.

Zunächst wird die letzte Spalte in die Mitte gerückt.

```
<script type="text/javascript" charset="utf-8">
  $(document).on('ready pjax:scriptcomplete',function(){
    // Define this question
    var thisQuestion = $('#question{QID}');

    // Move the last column
    $('col:nth-last-child(2)', thisQuestion).before($('col:last-child', thisQuestion));
    $('col:nth-last-child(3)', thisQuestion).before($('col:nth-last-child(2)', thisQuestion));
    $('col:nth-last-child(4)', thisQuestion).before($('col:nth-last-child(3)', thisQuestion));

    $('table.subquestion-list tr', thisQuestion).each(function(i) {
      $('> *:nth-last-child(2)', this).before($('> *:last-child', this));
      $('> *:nth-last-child(3)', this).before($('> *:nth-last-child(2)', this));
      $('> *:nth-last-child(4)', this).before($('> *:nth-last-child(3)', this));
    });
  });
</script>
```

Nun folgt das schon bekannte script, zur individuellen Einstellung der Spaltenbreite

```
<script type="text/javascript" charset="utf-8">
  $(document).on('ready pjax:scriptcomplete',function(){
    var thisQuestion = $('#question{QID}');
    // Add a question class
    thisQuestion.addClass('custom-array');

    // Column-specific classes
    $('table.subquestion-list tr', thisQuestion).each(function(i) {
      $('th, td', this).each(function(i) {
        $(this).addClass('column-'+i);
      });
    });
  });
</script>
```


Und das zugehörige css.

```
<style type="text/css">.custom-array table.subquestion-list col {  
    width: auto !important;  
}  
.custom-array table.subquestion-list thead .column-0 { width: 0%; }  
.custom-array table.subquestion-list thead .column-1 { width: 10%; }  
.custom-array table.subquestion-list thead .column-2 { width: 10%; }  
.custom-array table.subquestion-list thead .column-3 { width: 10%; }  
.custom-array table.subquestion-list thead .column-4 { width: 40%; }  
.custom-array table.subquestion-list thead .column-5 { width: 10%; }  
.custom-array table.subquestion-list thead .column-6 { width: 10%; }  
.custom-array table.subquestion-list thead .column-7 { width: 10%; }  
  
.ls-answers .answertextright {  
    text-align: center;  
}  
</style>
```

19. Anhang

19.1.1. Iss-Export einer Beispiel-Datei

Hier ist der Link zu einer gezippten Iss-Datei, die die hier beschriebenen Beispiele enthält.

www.Mafosurvey.de/lime/Tutorial_Matrizen.zip