

```

/*
Herzlich willkommen,
im Folgenden werden wir Sie bei Ihren ersten Schritten
mit der Q. Skriptsprache begleiten.

Nachdem Sie diese Datei zusammen mit unserem Tutorial
durchgearbeitet haben, können Sie...
- unsere Standardfragetypen hinzufügen, bearbeiten, entfernen
- Fragen bedingt einblenden (Filterung)

Und nebenbei werden Sie sich ein wenig an die Syntax der
Sprache gewöhnt haben.

Also fangen wir gleich an!

Dieser Text befindet sich in einem Kommentarblock. Er hat
keine Auswirkungen auf den Programmablauf und dient lediglich
der Übersicht und beinhaltet z.B. Anmerkungen und Erinnerungen.
*/

// Kommentarblock /* ... */
// Außerdem gibt es zeilenweise Kommentare ab // ...

/*
Bevor es nun zu den eigentlichen Umfrageinhalten geht,
importieren wir 2 externe Dateien, welche Einstellungen
zum grundlegenden Erscheinungsbild der Umfrage beinhalten.
Diese haben für dieses Tutorial keine weitere Bewandnis.
*/
// - Grundeinstellungen importieren -
#include "formats.q"
// - Grafische Buttons verwenden - EINKOMMENTIEREN und AUSPROBIEREN
#include "gbuttons.q"

// -----
// ----- Inhalte der Umfrage -----
// -----
/*
Grundsätzlich ist Groß- und Kleinschreibung von Schlüsselwörtern unerheblich.

Theoretisch können Sie die Fragen im Skript auch wahllos aneinander reihen.
Wie sie Reihenfolge tatsächlich in der Onlineumfrage, wird am Skriptende im
sogenannten main-Block definiert.

Aber der Reihe nach...

Beginnen wir mit der ersten Frage. Oder vielmehr mit einem Begrüßungstext
ohne Antwortmöglichkeiten. Dies lässt sich mit einer TEXTQ realisieren.
Wir nennen Sie 'fragel' (das ist beliebig) und weisen ihr einen Text zu:
*/

textq fragel;
text="Herzlich Willkommen zu meiner ersten Onlinebefragung!";

/*
Damit diese Frage nun als erster Bildschirm erscheint, fügen wir ihn dem
bereits erwähnten main-Block als erstes zu. Bitte scrollen Sie nach unten
und sehen Sie sich 'block main' an...

Weiter mit der ersten richtigen Frage. Diese soll 1-aus-N Antworten erlauben.
Hierfür ist der Typ SINGLEQ vorgesehen. Nennen wir sie kreativ 'frage2'.
*/

singleq frage2;
text="Hat bisher alles gut geklappt?";
labels=
1 "Ja"
2 "Nein"
;

/*
Zusätzlich zum Text gibt es hier nun 'labels', welche die Antwortoptionen
repräsentieren. Zwischen = und ; können beliebig viele Antworten eingefügt werden.
Sie beginnen mit einer (beliebigen) Zahl und dem Antworttext in Anführungszeichen.
Später im Datensatz werden die Antworten entsprechend dieser Zahlen codiert.

```

```

Weiter...
Gleiches Spiel mit einer M-aus-N Frage. Die MULTIQ.
*/

multiq frage3;
text="Welche Filme haben Sie 2010 im Kino gesehen?";
title="Mehrfachnennung möglich.";
labels=
1 "Avatar"
2 "Alice im Wunderland"
3 "Shrek 3D"
4 "Das Ende ist mein Anfang" random
5 "Frozen" random
6 "Der Kautions-Cop" random
7 "Inception" random
99 "Keine davon" single
;

/*
Neben dem Text kann auch ein Title angegeben werden. Dieser kann z.B. anders
formatiert sein und Intervieweranweisungen enthalten.
Außerdem sind die labels 4,5,6 und 7 mit dem Schlüsselwort random versehen.
Die Positionen dieser 4 Antworten werden von Fall zu Fall zufällig ermittelt.
Das geht beides natürlich auch in der SingleQ und anderen Fragetypen.
Nur in der MultiQ ist das Schlüsselwort 'single' verfügbar. Es annotiert
in diesem Fall Antwort 99 -Keine davon- als exklusiv.

Weiter...
Nun möchten wir eine offene Frage stellen. Aber nur, wenn in 'frage2' mit
-Nein- geantwortet wurde. Für offene Fragen gibt es die OPENQ.
*/

openq frage4;
text="Was hat denn bisher nicht geklappt?";
flt = (frage2 eq 2);

/*
Die OpenQ ist grundsätzlich mit der TextQ identisch. Es wird lediglich ein
Feld für textuelle Antworten eingefügt.
Mit dem Filter 'flt' wird eine Bedingung an diese Frage geknüpft. Sie wird
nur dann angezeigt, wenn 'frage2' gleich 2 ist. 2 ist die Labelnummer für die
Antwort -Nein- und 'eq' steht für 'equals' (Gleichheit). Neben 'eq' stehen die
Operatoren 'lt' (kleiner), 'gt' (größer), 'not' (Negation), 'or' (oder) und
'and' (und) zur Verfügung.
Filter dieser Form können an jeden Fragetyp geknüpft werden.

Weiter...
Nun möchten wir den Befragten auffordern den Film Avatar von 1-10 zu bewerten.
Natürlich nur dann, wenn er ihn auch gesehen hat.
Numerische Antworten bekommt man mit der NUMQ.
*/

opennumformat bereich = 1 2 0 0 1 10 "Falsche Eingabe!";
numq frage5;
text="Und wie fanden Sie Avatar?";
title="Bitte bewerten Sie von 1(-sehr schlecht) bis 10(-sehr gut).";
labels=
1 "Avatar" format bereich
;
flt = (1 in frage3);

/*
Um den erlaubten Wertebereich einzugrenzen ist ein OPENNUMFORMAT notwendig.
Es erhält, wie die Fragen, einen Namen 'bereich' und wird in den labels der NumQ
wieder darüber referenziert. Die Zahlen im OpenNumFormat bedeuten hier Folgendes:
1-2 Vorkommastellen, 0-0 Nachkommastellen, 1-10 insgesamt erlaubte Werte.

Der Filter wird hier mit 'in' definiert, weil 'frage3' eine MultiQ ist und mehrere
Antworten vorhanden sein könnten. Die Gleichheit ist nicht mehr eindeutig. 'in' prüft
das Vorhandensein einer bestimmten Antwort.
*/

block main = ( frage1 frage2 frage3 frage4 frage5 );

```