

QFQ - Feature #11534

Report: Action on selected rows - Table batchprocessing feature

17.11.2020 12:06 - Simon Grüning

Status:	New	Start date:	17.11.2020
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	22.4.0	Spent time:	0.00 hour
Discuss:			
Description			
See attached picture. A feature to add an additional column of check-boxes at the beginning of a table, and the ability to have buttons ex. "with selected: move all to bla". So batch processing of table rows / records.			
Implementation idea: via javascript? Buttons passing list of id's (or respective sip's) of records and action onto backend.			

History

#1 - 18.11.2020 08:15 - Carsten Rose

- File selectRow.png added

- Subject changed from Table batchprocessing feature to Report: Action on selected rows - Table batchprocessing feature

- Hat Philipp bereits via JS implementiert (selectRow.png / webwork18/semLive / man muss eingeloggt sein)
- Wir sollten ein allgemeines QFQ Feature implementieren:
 - `Select All` Checkbox:
 - Als `... AS_link`
 - Neuer Qualifier fuer SelectRow: Bsp: `rowCheckAll:myRowSelect`
 - `myRowSelect` ist dann die Class ueber die alle Row-Checkboxes eingesammelt werden.
 - Row-Checkboxes
 - Als `... AS_link`
 - Neuer Qualifier `rowCheck:myRowSelect`.
 - Mittels `U:var=value` werden die per Row Values gesetzt.
 - Wird der Qualifier `s` (SIP) gesetzt, wird nicht der Value, sondern die generierte SIP uebergeben.
 - Row-Action
 - Als `... AS_link`
 - Neuer Qualifier `rowAction:myRowSelect`.
 - Sobald min. eine Checkbox selektiert ist, sind die rowAction Elemente aktiv.
 - Wenn keine Checkbox mehr aktiv ist, werden die Elemente inaktiv.
 - Action Elemente sind alle Button / Links mit der Custom CSS Class `myRowSelect` (evtl. einfach alle die keine Checkbox sind).
 - Auf dem Action Element gibt es ein HTML Attribut `data-name=s` (bei SIP Betrieb) und `data-name=value` falls kein SIP verwendet wird. Der JS Code setzt in Window.location den GET Parameter entsprechend `s` oder `value`.
 - SIP Auswertung: enthaelt der `s`-GET Parameter kommagetrennte SIPs, dann fuehrt diese zu einer entsprechenden Kommaliste mit den angegebenen Variablen. Beispiel:
 - Drei Beispiel Checkbox Values (rowCheck): s=10000001 (a=10,b=199) / s=10000002 (a=11,b=200) / s=10000003 (a=12,b=201)
 - Ein Action Link (rowAction): window.location(`...id=moveAction&s=10000001,10000002,10000003`)
 - Ziel Page (moveAction): {{a:S}}=10,11,12 / {{b:S}}=199,200,201

Features:

- Pro Seite kann es mehrere Checkbox Columns geben - daher ist es sinnvoll eine Custom CCS-Class (als Identifier) a la `myRowSelect` zu definieren.
- Es kann mehr als ein Action Element geben.
- Es kann mit und ohne SIP gearbeitet werden.
- Falls ohne SIP
- Der Javascript Code sammelt die SIPs/Values erst bei onClick() auf einem der Action Elemente ein.
- Wir koennten Probleme mit der maximal Laenge der GET Variablen bekommen. In QFQ by default 50. Es gibt ein QFQ Namensscheme <GET Variable Name>_<length> - ich bin nicht sicher ob uns das helfen koennte.

Files

table batch processing sample qfq request.png
selectRow.png

57.1 KB
115 KB

17.11.2020
17.11.2020

Simon Grüning
Carsten Rose