netgenium

Edit form

Framework NET Genium

Content

1 List of tabs in the edit form settings dialog 4
1.1 "General" tab 5
1.1.1 Name 5
1.1.2 Dictionary 5
1.1.3 Options 5
1.2 "Other" tab 6
1.2.1 Primary key 6
1.2.2 Record locking 6
1.2.3 Options 7
1.3 "Help" tab 8
1.3.1 Width of floating help 8
1.3.2 Options 9
1.3.3 Help 9
1.3.4 Notes 9
1.4 "Editing rights" tab 10
1.4.1 Co-owners of the records 10
1.4.2 Editing rights – options 10
1.5 "Rights of child objects" tab 11
1.5.1 Options 11
1.6 "Administration" tab 12
1.6.1 Location 12
1.6.2 Identifier 12
1.6.3 Commands 14
1.6.4 Index 1 14
1.6.5 Index 2 14
1.6.6 Delete history older than 14
1.6.7 C#/JavaScript 14
1.6.8 Statistics 15
1.6.9 Logging 15
1.6.10 History 16
2 Sync settings 17

2.1 List of servers with which data is synchronized 17
2.2 Link to setting up a new server 18
2.3 Server settings 19
2.3.1 Name 19
2.3.2 URL 19
2.3.3 Synchronisation password 19
2.3.4 Intranet URL 20
2.3.5 Language 20
2.3.6 Options 20
2.4 Link to new sync settings 22
2.5 Sync settings 23
2.5.1 Server 23
2.5.2 Edit form 23
2.5.3 Condition for synchronistation 23
2.5.4 Columns 23
2.6 One-time synchronization of all data 24

1 List of tabs in the edit form settings dialog

General	Setting general properties
Other	Primary key settings, locks, change tracking and javascript
Help	Set up floating help text and group controls
Editing rights	Rights settings
Rights of child objects	Setting the rights of child controls
Administration	Setting other properties

1.1 "General" tab

1.1.1 Name

- The name of the edit form ideally determines the name of one database record, expressed in a single number, which is stored in the database table associated with the form.
- The name of the edit form is displayed in the title of the web browser tab.

1.1.2 Dictionary

- The dictionary displays a list of application group names, applications, categories, view pages, edit forms, and controls used throughout the application.
- The dictionary search is used to suggest the correct name for the edit form and to maintain the consistency of the nomenclature throughout the application.

1.1.3 Options

- **Tab General + Other**
  - Checking this box determines whether the "Tab" and "Other" tab controls should also be created when creating a new edit form.
- **Change tracking + Watcher**
  - Checking this box determines whether change tracking controls should also be created when creating a new edit form:
    - "History" control
    - "Watcher" control
      - Created by
      - Created on
      - Modified by
      - Modified on

1.2 "Other" tab

1.2.1 Primary key

- The primary key specifies the data type of the primary key – the database column "id", which is part of every database table in NET Genium.
- The data type of the primary key can be changed from Integer to Long at any time, but a reverse change from Long to Integer is not possible.
- Before changing the primary key to Long, it is important to ensure that external functions, services, or other applications that work with the database table function properly. These are mainly DataSaver or DataSaverSynchro objects that must be changed to DataSaver2 and DataSaverSynchro2.

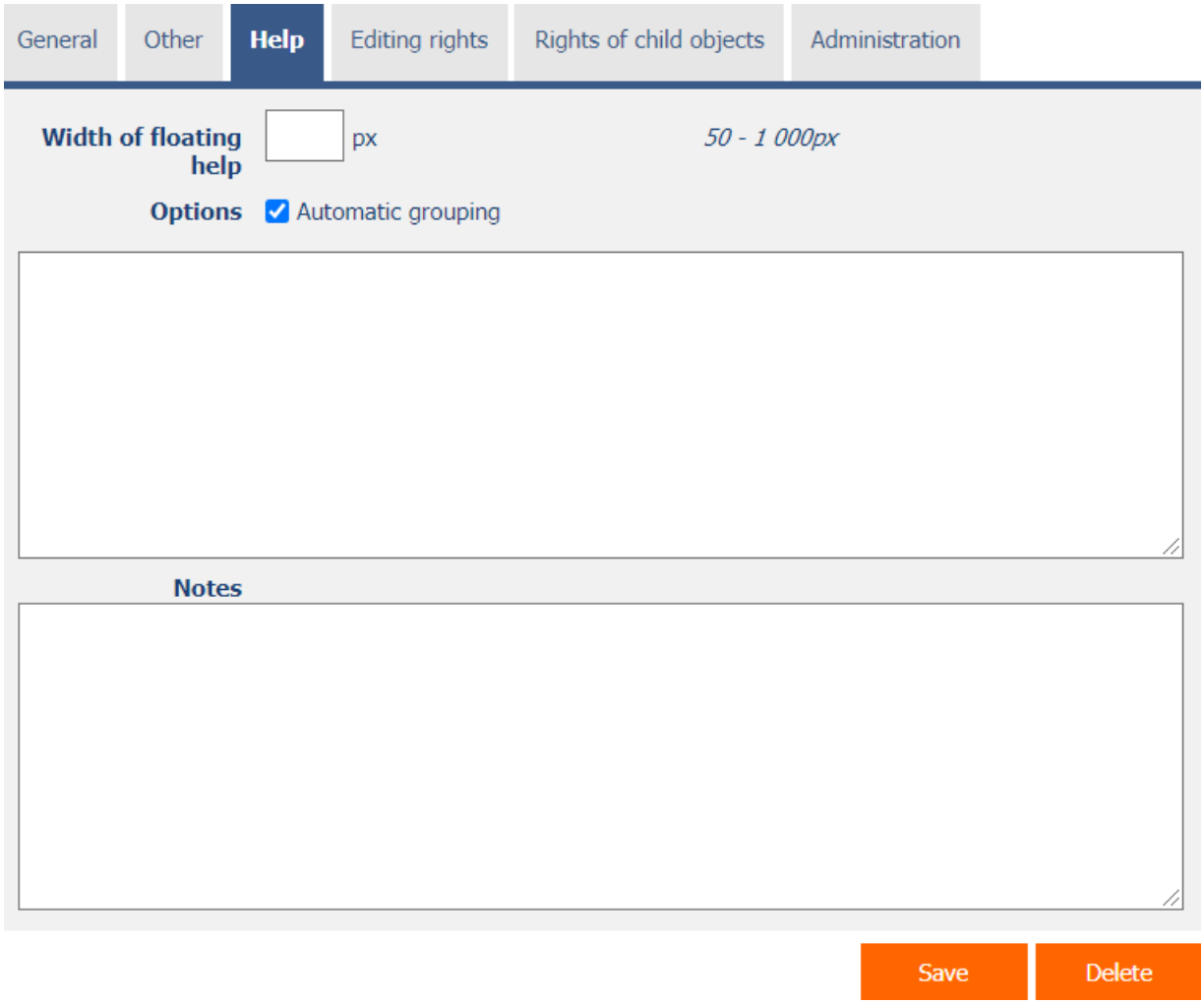
1.2.2 Record locking

- Database records opened in the edit form can be set so that only one user can edit them at a time.
- The user who opens the database record first automatically locks the record. While working in the edit form, the lock is automatically extended until the user exits the edit form. When the user leaves the edit form, the lock is automatically released immediately.
- Record locking is defined in the edit form settings by specifying a time interval in minutes during which the lock should be active. The length of the time interval plays a role only in case of technical problems of the user, due to which the lock cannot be extended automatically during the entire time the user works with the database record in the edit form. Typical issues include opening a different web browser tab, closing the current tab with a cross, launching a different application on a mobile phone, or internet outages. Therefore, in case of technical problems, the lock will not be released automatically until a selected time interval has elapsed since the last automatic lock extension. The length of the time interval therefore determines the period for which the lock will be active even in the event of a connection failure between the client and the server.
- Locked records have a lock icon in the view table instead of the magnifying glass icon that is used to open the database record. The lock icon contains a tooltip with information about who locked the record and until when the lock is active. Even so, the database record can still be opened using the lock icon; other users can only read the edit form with the locked record without editing it.

1.2.3 Options

- **Log opening records**
  - Checking this box determines whether each opening of a database record in the edit form should be recorded in the change tracking history.
- **Disable history of records**
  - Checking this box determines whether database record history logging should be disabled.
  - Completing the logging completely, or even temporarily, is recommended before bulk editing or deleting database records, where each change or deletion of a database record creates one record in the change tracking history.
- **Disable JavaScript**
  - Checking this box determines whether to disable the execution of control javascript events, as well as the generation of JavaScript control content.
  - This option is used in situations where the edit form is looped by inappropriate javascript. In some cases, it is not even possible to open the edit form settings in the usual way, and it is necessary to use reports to do so. A detailed description of the reports is given in a separate manual “Reports”.

1.3 "Help" tab



1.3.1 Width of floating help

- The width of the floating help determines the width of the help box for the edit form located in the upper right corner of the form. This box is displayed independently of the vertical scroll in the edit form.
- The floating box also includes the contents of all "RichText" and "HTML" controls with the "Floating help in edit form" check box selected. This content is located behind the help of the edit form in the order that these controls have in the edit form.
- If the width is not filled, the default width of the floating box of 200 pixels is used.

1.3.2 Options

- **Automatic grouping**
  - If “Automatic grouping” is checked, a bulleted list of all help controls located on the same row will be automatically generated in the help column of each edit form control, with the option to click the bulleted list item and display the help text.
  - If “Automatic grouping” is not checked, the help for controls to the left of snap-on controls is not displayed.
  - Help grouping is used for controls that have the “Snap to previous control” check box selected.

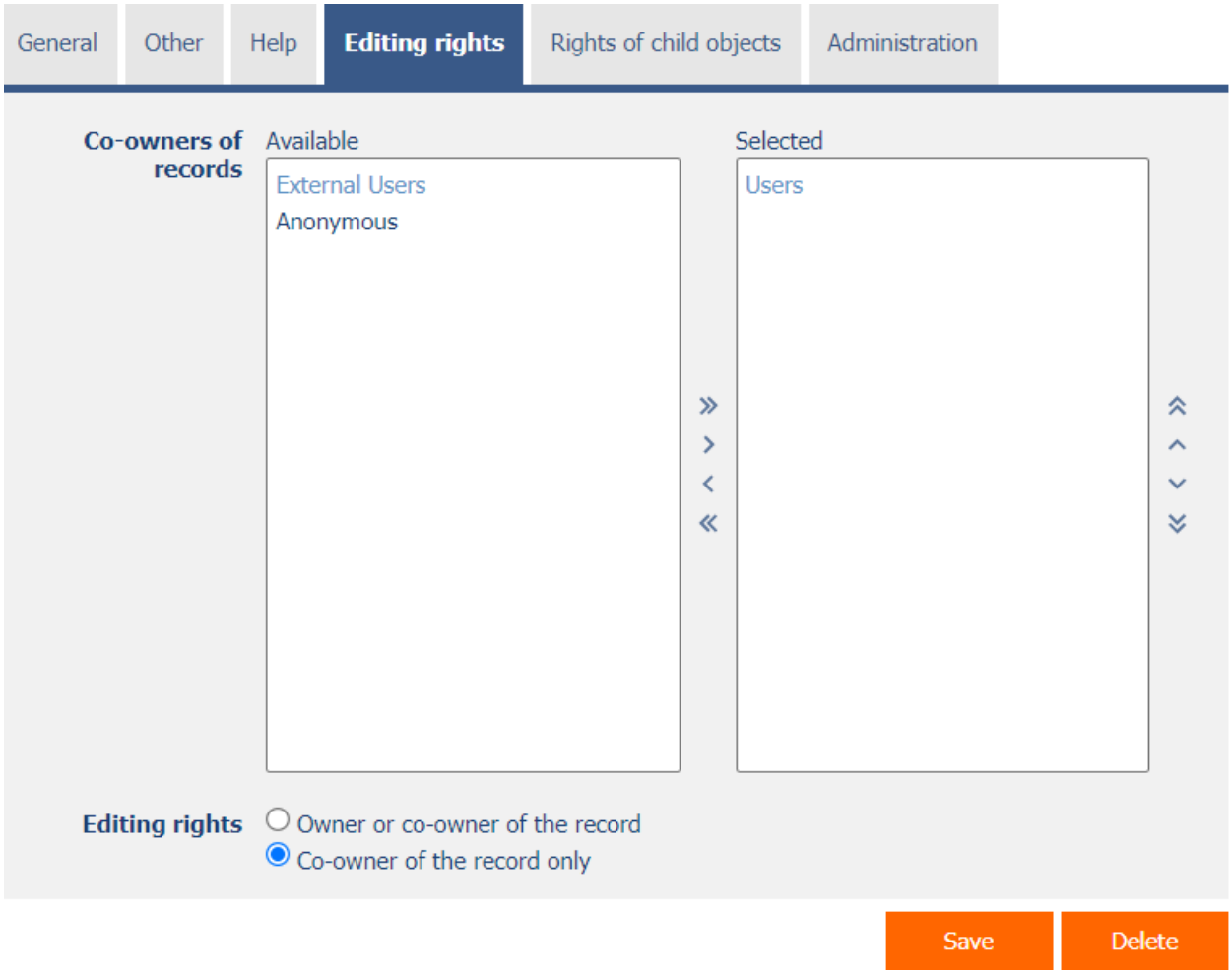
1.3.3 Help

- Help is the text displayed in the upper right corner of the edit form inside the floating box.
- The floating box also includes the contents of all “RichText” and “HTML” controls with the “Floating help in edit form” check box selected. This content is located behind the help of the edit form in the order that these controls have in the edit form.

1.3.4 Notes

- Notes are used to enter any text intended for the application administrator.
- Calls to server functions that control the behavior of the edit form are placed in the notes:
  - DISABLEAUTOFILL()
  - DISABLEPAGETOOLS()
  - LOWRESOLUTION()
- The definition of the database view is placed in the notes if the edit form is set to read data from the database view.

1.4 "Editing rights" tab



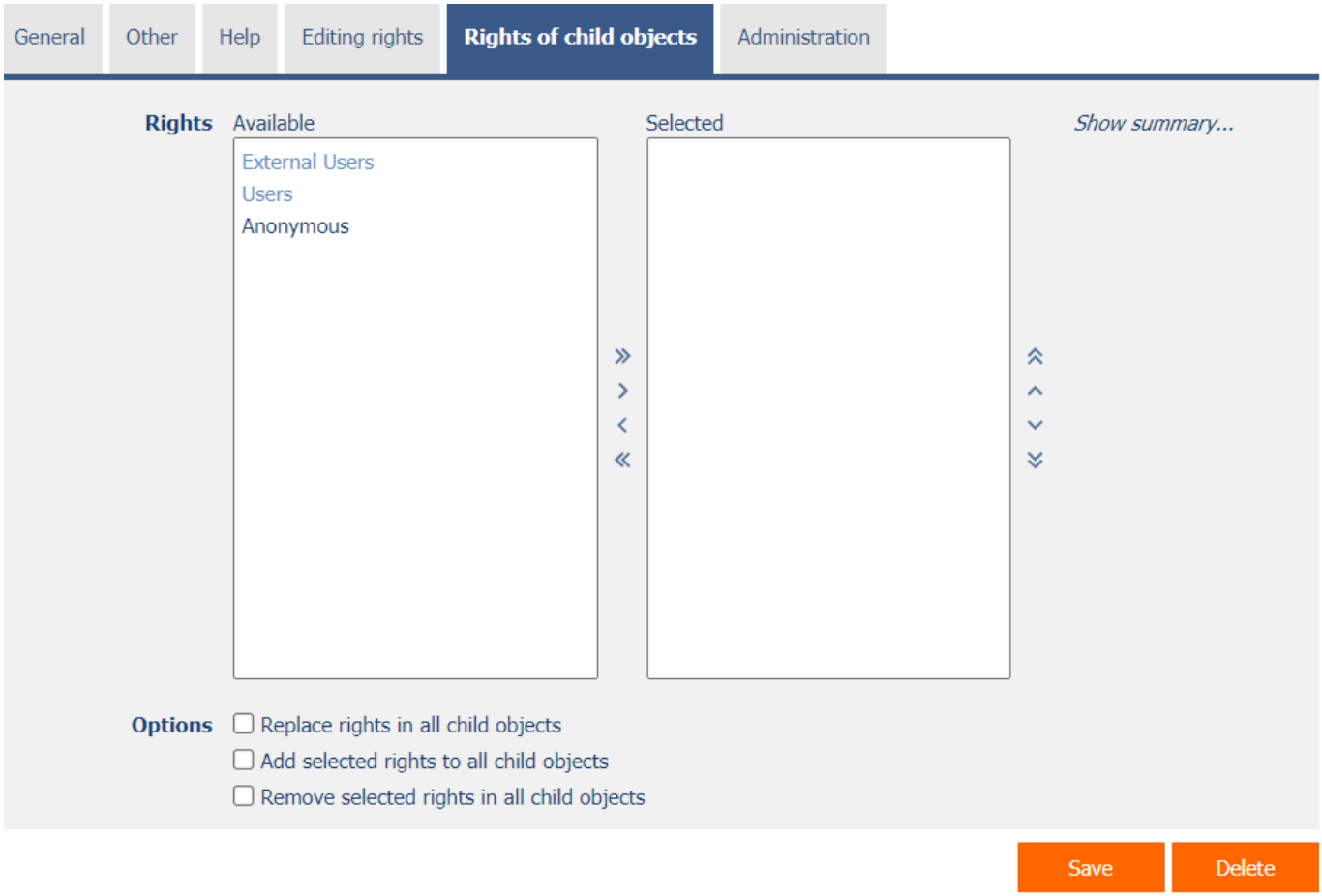
1.4.1 Co-owners of the records

- Record co-owners define a list of user groups and users who extend ownership of database records. In addition to the owner of the record, i.e. the user who created the database record, they extend ownership by other users who automatically become co-owners of all records in the database table. A detailed description of co-ownership of records is given in a separate manual "Administrator's Guide".

1.4.2 Editing rights – options

- **Owner or co-owner of the record** – database records in the edit form can be edited only by the owner or co-owner of the record.
- **Co-owner of the record only** – database records in the edit form can be edited only by the co-owner of the record.

1.5 "Rights of child objects" tab

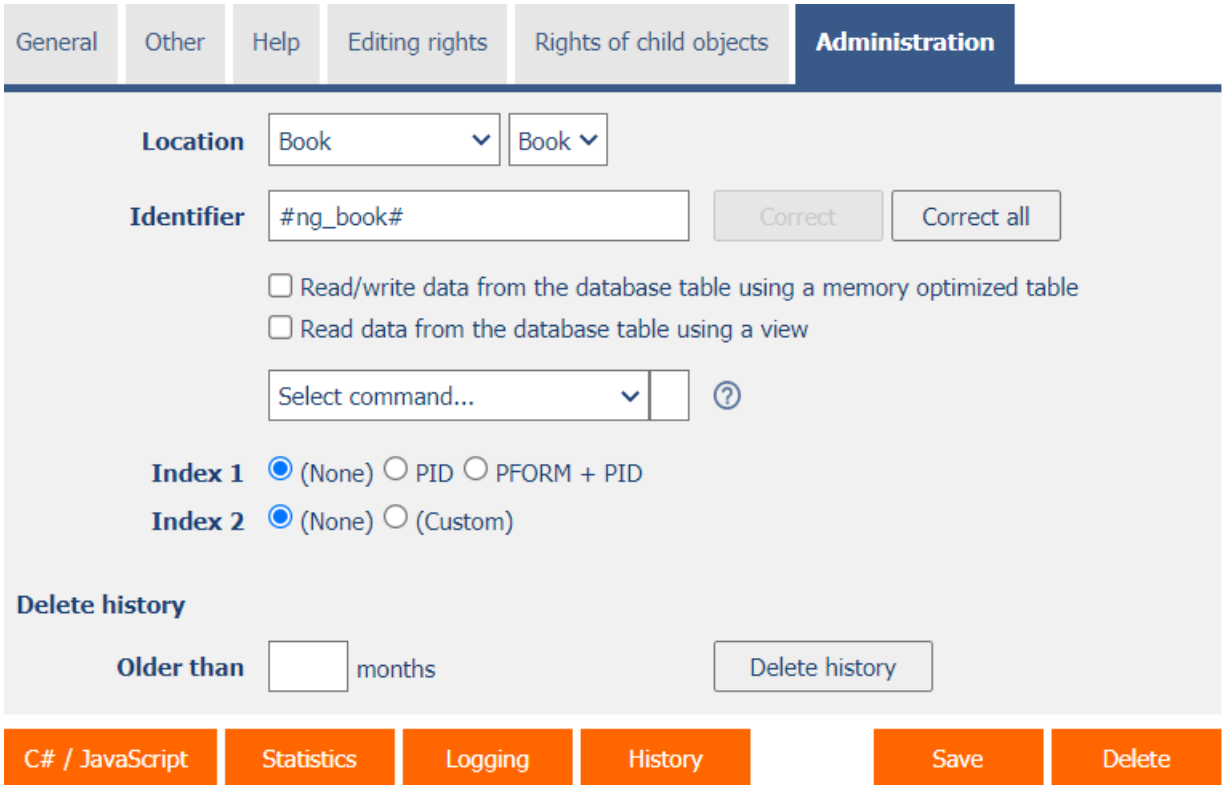


- Rights define a list of user groups and users who have permission to display controls on an edit form and to display the values of those controls elsewhere in the application.
- The properties of the edit form are only the settings of the co-owners of the records, anyone has the right to enter the edit form. It is up to setting the rights of individual form controls, what everything is displayed to the user in the edit form.

1.5.1 Options

- **Replace rights in all child objects** - Checking this box determines whether the rights settings in all child controls should be overwritten to the same permissions as those set in the edit form itself when the edit form is saved.
- **Add selected rights to all child objects** - Checking this box determines whether all selected application groups and users should be added to all child controls when saving the edit form.
- **Remove selected rights in all child objects** - Checking this box determines whether all selected application groups and users should be removed from all child controls when saving the edit form.

1.6 "Administration" tab



1.6.1 Location

- Select the application in which the edit form is located.
- The drop-down list contains all applications.
- Edit forms located in the "Settings" system application may not be moved or deleted.

1.6.2 Identifier

- The identifier specifies the name of the database table in the database in which the user data filled in by the user in the edit form is stored.
- The identifier is derived from the name of the edit form. It starts with the prefix "ng_", and also contains the alphanumeric characters used in the edit form name (system edit forms do not contain "ng_"). When changing the name of the edit form, the "Correct" button is available, which is used to correct the identifier and thus rename the database table in the database. For clarity and elimination of possible misunderstandings, it is recommended to keep the identifier in accordance with the name of the edit form. If external applications also access the database and, for example, read or write data to the database table of this edit form, it is necessary to schedule the change of the identifier until a suitable time.
- The "Correct all" button is available to automatically correct the identifiers of all database controls of the edit form.

- **Read/write data from the database table using a memory optimized table**

🔨 *For SQL Server only*

  - In-Memory tables are database tables located in the operational memory, which are suitable for frequent writing of data – especially for creating new records (INSERT) or updating existing ones (UPDATE).
  - In-Memory tables are not suitable for troubleshooting database query processing speed while reading data (SELECT). The solution to these problems is always to set the indexes correctly.
  - Checking this box determines whether a database table with the same identifier as the edit form should be used to read and write data, with only the word "_memory" at the end of the name.
  - In-Memory tables must be created manually. For this purpose, it is possible to use the command "CREATE ng_book_memory" from the drop-down list located below the check box. This command displays a database query to create a new In-Memory table with the same structure as the edit form, and attempts to create the database table immediately.
  - In-Memory tables do not allow additional changes to the database structure. You must first delete the old table, and then create a new one with an updated structure. For this purpose, it is possible to use the command "DROP ng_book_memory" from the drop-down list located below the check box. This command displays a database query to delete an In-Memory table, and attempts to delete the database table immediately.
  - Before deleting In-Memory tables, it is important to back up the data stored in these tables, and then restore that data. To do this, you can use the command "EXPORT DATA FROM ng_book_memory TO 'Backup\ng_book_memory.xml'" from the drop-down list located below the check box, followed by "IMPORT DATA TO ng_book_memory FROM 'Backup\ng_book_memory.xml'" or "IMPORT DATA TO ng_book_memory FROM ng_book ".

- **Read data from the database table using a view**

  - Checking this box determines whether a database view with the same identifier as the edit form should be used to read the data, with only the word "_view" at the end of the name. It is not possible to write data to views, so it is recommended to hide the "Save" button in these edit forms. An error occurs when trying to save data to a view.
  - Views must be created manually. For this purpose, it is possible to use the command "CREATE ng_book_view" from the drop-down list located below the check box.
    - This command displays a database query to create a new view with the same structure as the edit form, and attempts to create the database view immediately.
    - The view created by this command has the same structure as the edit form, and therefore it is necessary to modify its definition (CREATE VIEW) additionally.
    - It is advisable to save the resulting definition in a note of the edit form, where NET Genium will search for it the next time the "CREATE ng_book_view" command is run, and run it in this wording.
  - Views do not allow additional changes to the database structure. You must first delete the old view, and then create a new one with an updated structure. For this purpose, it is possible to use the command "DROP ng_book_view" from the drop-down list located below the check box. This command displays a database query to delete the view, and attempts to delete the database view immediately.

1.6.3 Commands

- **CREATE REPORT FROM ng_book** – Creating a report on the length of strings stored in the edit form
- **CREATE ng_book_memory** – Create In-Memory table
- **IMPORT DATA TO ng_book_memory FROM 'Backup\ng_book_memory.xml'** – Import data into In-Memory table from XML file located in the "Backup" directory
- **IMPORT DATA TO ng_book_memory FROM ng_book** – Import data into the In-Memory table from the main database table of the edit form
- **EXPORT DATA FROM ng_book_memory TO 'Backup\ng_book_memory.xml'** – Export of data from the In-Memory table to an XML file located in the "Backup" directory
- **DROP ng_book_memory** – Delete In-Memory table
- **CREATE ng_book_view** – Creating a view
- **DROP ng_book_view** – Delete view
- **TRUNCATE TABLE ng_book + ng_book_history** – Delete all data in the database table, including history records, and set the primary key counter to an initial value of 1
- **TRUNCATE TABLE ng_book_history** – Delete all history records, and set the primary key counter to an initial value of 1
- **RECREATE TABLE ng_book + ng_book_history** – Data backup and re-creation of database tables, including subsequent data import

1.6.4 Index 1

- Setting the index to the "pid" database column or the associated index to the "pform" database column along with the "pid" column.

1.6.5 Index 2

- Setting the index to any list of database columns of the edit form.
- In the text box located below the check box, it is possible to define the names (identifiers) of all columns that are to be in the federated index – they should be part of the index key.

1.6.6 Delete history older than

- Setting the number of months during which it is necessary to keep the history of database records of this edit form.
- If the number of months is not filled in, the default setting of 6 months is used.
- History records older than the time period set in this way can be deleted once using:
    - "Clear history" buttons located to the right of the text box.
    - Smaller icons for deleting record history from reports. A detailed description of the reports is given in a separate manual "Administrator's Guide".
    - "DELETEHISTORY" server functions.

1.6.7 C#/JavaScript

- Using the "C#/JavaScript" button, a text report will be displayed with an example in C# for saving the record to the database table associated with this edit form.
- The report includes a list of all database controls in the edit form.

1.6.8 Statistics

- The statistics display aggregated information on the use of the edit form across the entire application.
- Using the "Statistics" button, a detailed report of all places where the edit form is used or referenced is displayed, including the option to open the settings of such a place or control.
- Before each deletion of the edit form, it is recommended to thoroughly go through the statistics of the editing element, and to eliminate possible misunderstandings.
- Before each deletion of the edit form, the evaluation of the statistics starts automatically. If the edit form is used or referenced somewhere in the application, it is necessary to confirm the deletion of the edit form again.

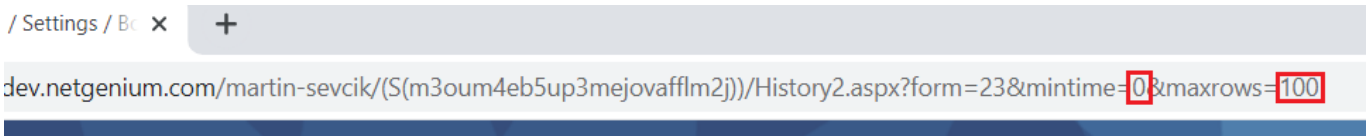
1.6.9 Logging

- Using the "Logging" button, a detailed report is displayed with records of the display of the edit form and data about
    - the date and time of displaying the edit form,
    - the user who viewed the edit form,
    - edit form processing time in milliseconds; and
    - information on whether this is the first loading of the edit form or its updating (postback).

Logging 9 / 9

#	Date	User	Time	Time [ms]	PostBack
1	09/06/2021 21:03:55	Administrator	0.016 seconds	16	
2	09/06/2021 21:03:45	Administrator	0.016 seconds	16	
3	09/06/2021 21:03:28	Administrator	0.125 seconds	125	
4	09/06/2021 21:02:26	Administrator	0.031 seconds	31	
5	09/06/2021 21:01:57	Administrator	0.031 seconds	31	
6	09/06/2021 21:01:36	Administrator	0.031 seconds	31	
7	09/06/2021 21:00:45	Administrator	0.016 seconds	16	
8	09/06/2021 21:00:00	Administrator	0.031 seconds	31	
9	09/06/2021 20:57:32	Administrator	0.047 seconds	47	

- The minimum display time for processing an edit form is set to 0 milliseconds by default, so all display records of edit forms are displayed, including those whose processing took 0 milliseconds. This minimum time can be increased or decreased by changing the "mintime" parameter in the report URL.
- The number of records is limited to 100 by default. This number can be manually increased or decreased by changing the "maxrows" parameter in the report URL.

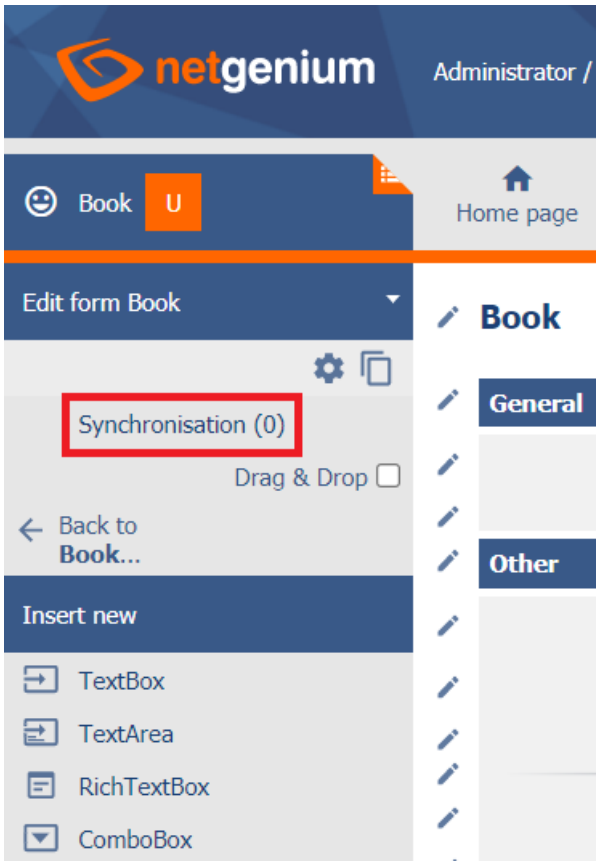


1.6.10History

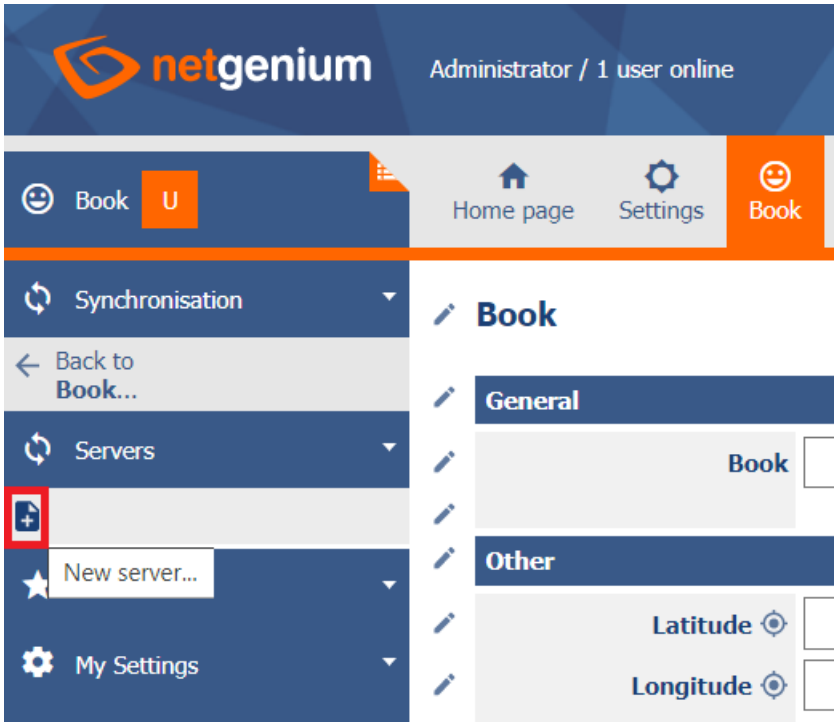
- Using the “History” button, a detailed report with individual change tracking records is displayed.

2 Sync settings

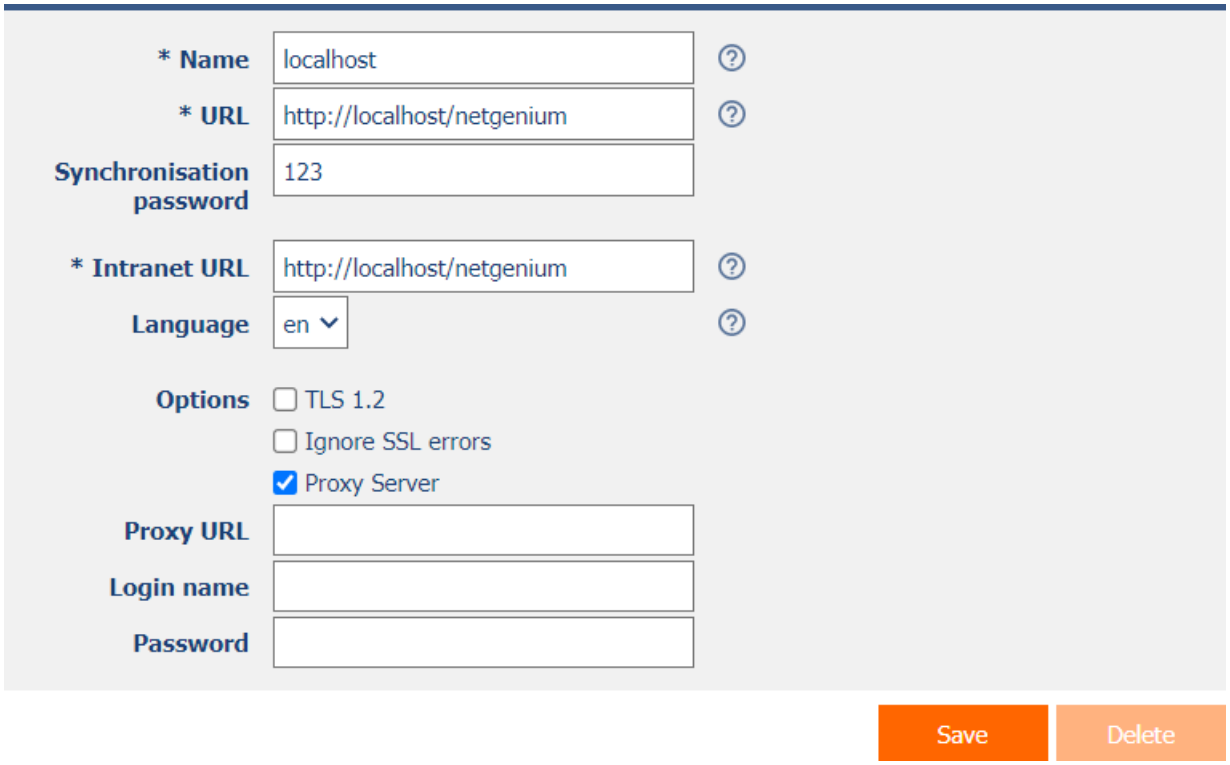
2.1 List of servers with which data is synchronized



2.2 Link to setting up a new server



2.3 Server settings



2.3.1 Name

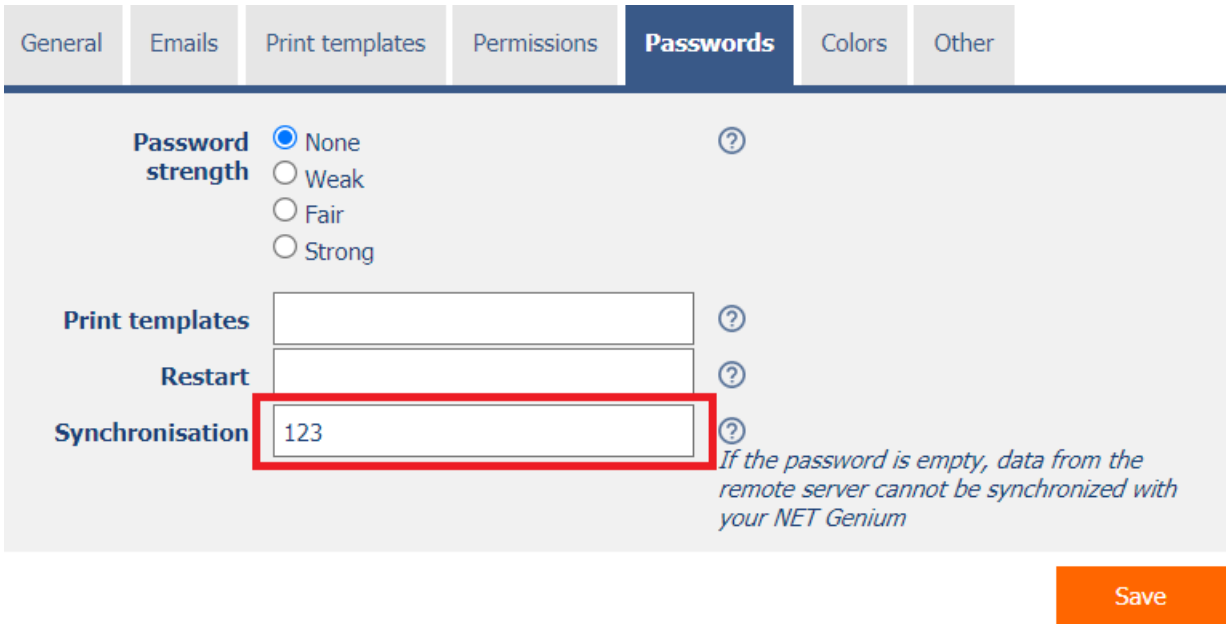
- The name of the remote server, resp. NET Genium.

2.3.2 URL

- Internet address of the remote NET Genium (SSL communication via the “https” protocol can be used).

2.3.3 Synchronization password

- Synchronization password set on the remote NET Genium in the NET Genium settings.



2.3.4 Intranet URL

- Intranet address of the remote NET Genium, to whose records users will be routed from e-mail messages sent by the script.

2.3.5 Language

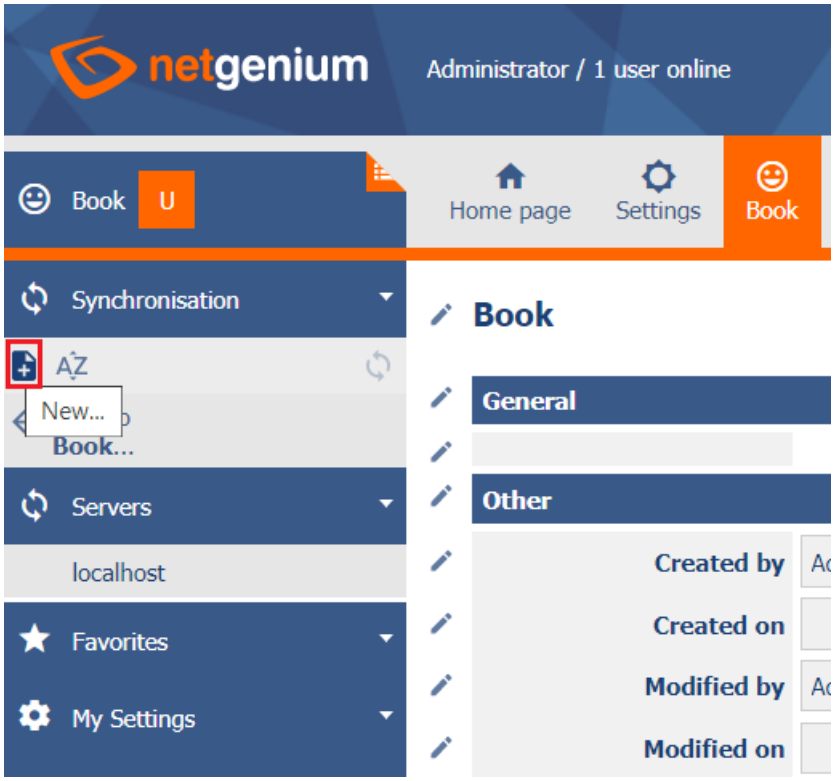
- The language in which links will be created in e-mail messages, directing users to NET Genium entries.

2.3.6 Options

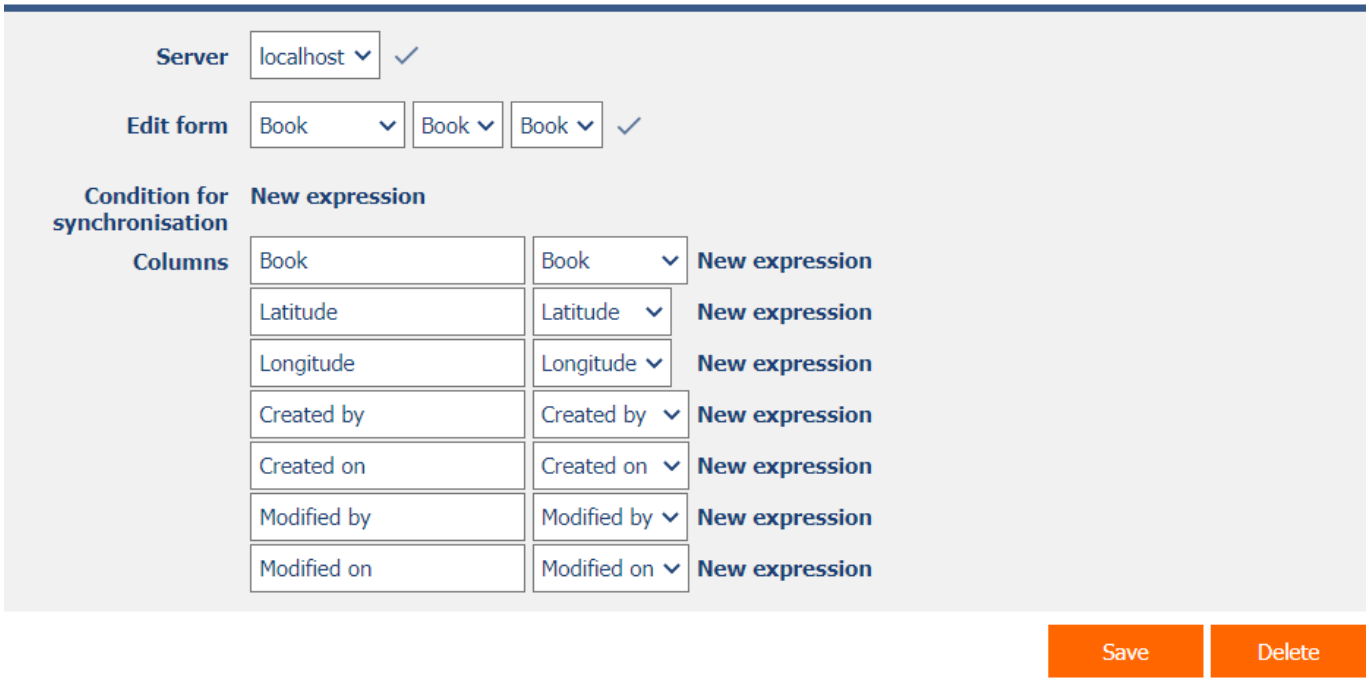
- **TLS 1.2**
  - Checking this box determines whether legacy TLS 1.2 protocol will be used on older servers to communicate with the remote NET Genium through Web services.
  - On servers running Windows 2012 and higher, TLS 1.2 is automatically selected without having to check this box.
- **Ignore SSL errors**
  - Checking this box determines whether errors in SSL certificates will be ignored when communicating with remote NET Genium via the “https” protocol.
- **Proxy Server**
  - Checking this box determines whether a Proxy Server will be used to communicate with the remote NET Genium.
- **Proxy URL**
  - Proxy Server Address.

- **Login name**
  - Login name for communication with the Proxy Server.
- **Password**
  - Password for communication with the Proxy Server.

2.4 Link to new sync settings



2.5 Sync settings



2.5.1 Server

- The name of the server with which the data from the database table of the currently open edit form will be synchronized.

2.5.2 Edit form

- Edit form in the remote NET Genium, with which the data from the database table of the currently open edit form will be synchronized.

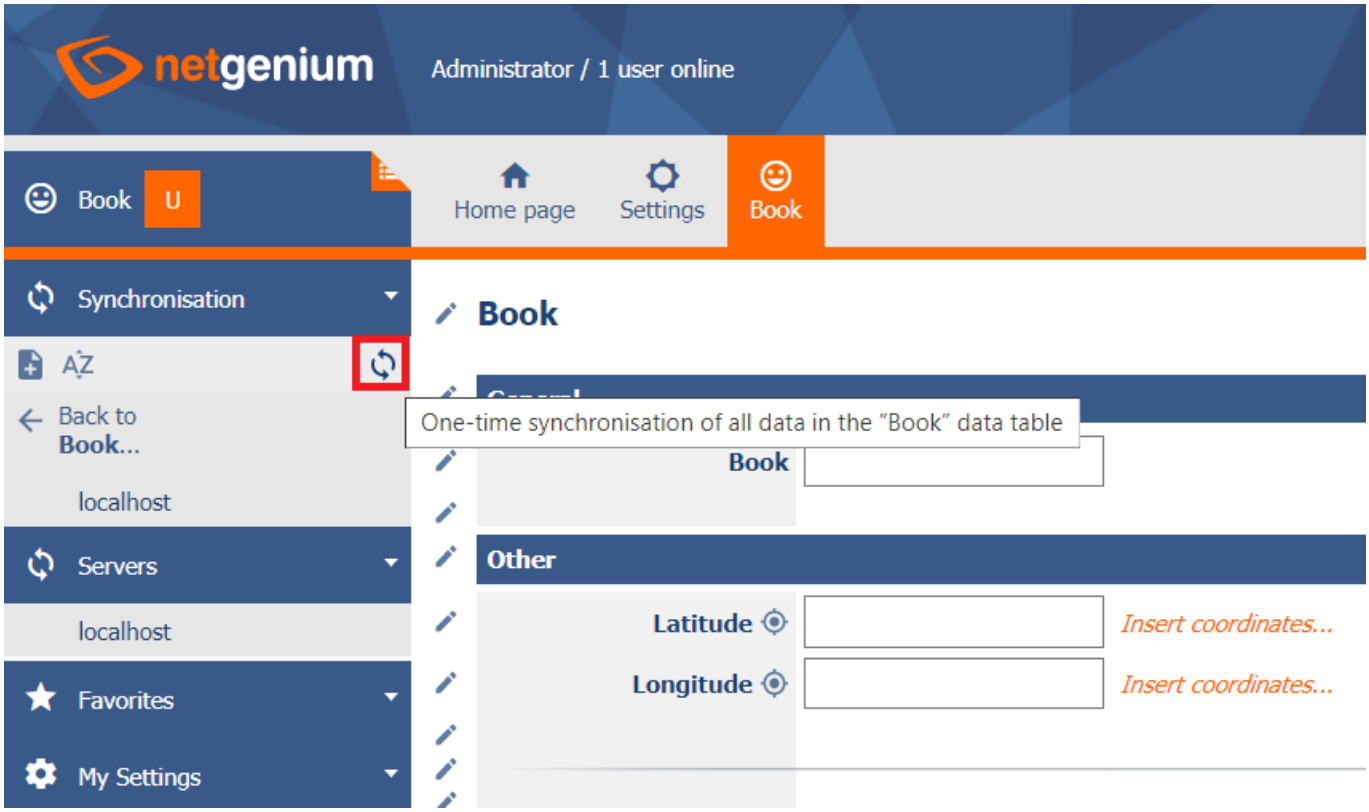
2.5.3 Condition for synchronization

- A condition that limits the scope of synchronized data to a specific set only.

2.5.4 Columns

- The column settings determine which column (left) of the remote edit form will be synchronized, and which column (right) in the currently open edit form.
- Each column can contain a condition only when the value of the column will be synchronized.

2.6 One-time synchronization of all data



A maximum limit of 10,000 records is set for one-time synchronization.