






### **Upgrade of REDCap™ from V10.0.33 to V10.6.11**

This document contains information about the changes and new features, which have been implemented in this newer version of REDCap. Please review this document so you are aware of the upcoming changes.

Once the upgrade is complete, further information about the changes can be found in the “Help & FAQ” and “Training Videos” web pages. Additional information and education

can also be found by clicking on the:  (Tell me more),  VIDEO: (Video) icons and  [How to use Piping](#) (How to use) links found throughout the REDCap application.

For any additional assistance, as always, feel free to email your questions or problems to: [REDCapAdmin@mayo.edu](mailto:REDCapAdmin@mayo.edu)

## Table of Contents

Surveys.....	2
Automatic Survey Invitations.....	2
Online Designer.....	3
Alerts & Notifications .....	4
Data Import .....	5
Logging.....	5
Data Access Groups (DAGs) .....	5
Data Quality .....	6

## Surveys

- **New feature: New survey option “Save a PDF of completed survey response to a File Upload field”** - On the Survey Settings page in the Online Designer, users may select a File Upload field in the project where a static PDF file of a participant’s survey response will be stored immediately after they complete the survey. For longitudinal projects, if the target field exists on multiple events, users may set this feature so that it stores the PDF in the selected field in the current event (default) or else in a specific event in the project.
- **New feature:** Added "Language of text to be spoken" for the "Text-To-Speech" survey functionality, which is available on the Survey Settings page. For several years, REDCap has had a Text-to-Speech feature for surveys that, when enabled, allows questions and other text on survey pages to be converted into natural-sounding audio for the participant to hear. Up until now, it supported English only, but now REDCap users may utilize the Text-to-Speech feature in a variety of non-English languages and voices, assuming that the survey text is in a non-English language. This includes Arabic, Brazilian Portuguese, English (UK and US), French, German, Italian, Japanese, and Spanish (Castilian, Latin American, and North American).
- **Improvement:** The Survey Invitation Log has a new filter drop-down option to view “only deleted invitations” (i.e., permanently cancelled invitations).
- **Improvement:** When deleting an invitation from the Survey Invitation Log (either as a single invitation or using the multi-select option to delete many invitations at once), it now provides a new option in the dialog prompt to “Permanently cancel this invitation?”, in which the phrase “permanently cancel” implies that the invitation cannot be re-triggered/scheduled again in the future even if the ASI conditions are met again. If the user chooses to uncheck the option, then the scheduled invitation will be removed, but could possibly get re-triggered in the future if the ASI conditions are met again (assuming it was originally scheduled via an ASI).
- **Change:** When viewing the Participant List page, if the survey has the "survey-specific email invitation field" enabled, it will now display a small notice about this fact near the top of the Participant List table to inform the user and provide more clarity regarding the source of the email addresses being used for that particular survey.
- **Change:** A warning message is now displayed for users that attempt to use both the Survey Queue and Survey Auto-Continue features together in the same project. If one is already enabled while attempting to enable the other, the warning will inform the user that these two features can sometimes conflict with each other

## Automatic Survey Invitations

- **New feature:** Users may re-evaluate some or all Automated Survey Invitations for all records in a project. If an ASI has been modified after data has already been entered in the project, users may click the “Re-evaluate Auto Invitations” button in the Online Designer, which will re-evaluate selected ASIs for all records to ensure that invitations get properly sent or scheduled based on the new conditions of the ASI (otherwise they could only be triggered if each individual record had data modified). If a user modifies the conditional logic of an ASI, it will recommend that they utilize the “Re-evaluate Auto Invitations” functionality. If an ASI has the “Ensure logic is still true...” option checked, then it is possible during this process that some already-scheduled invitations might get removed (and thus would no longer be scheduled) based on the new conditions.
- **Improvement:** When setting up Automated Survey Invitations and Alerts & Notifications a new send-time option has been added. When defining when the ASI/Alert should be sent, the option “Send after a lapse of time” has a new setting added so that, if desired, the user may set the time lapse relative to the value of a date or datetime

field in the project. In previous versions, the time lapse setting could only be set relative to the time in which the ASI/Alert was triggered. That is still an option, but now users may also opt to send the ASI/Alert a certain amount of time either before or after the date/time of a specific field. This new setting will allow users to have greater control with regard to setting when ASIs/Alerts will be sent without getting too complicated in their setup, such as having to use complex logic (with datediff, etc.).

## Online Designer

- **New feature:** Select and modify multiple fields together on the Online Designer - Users may select multiple fields on the Online Designer by holding the Ctrl, Shift, or Cmd key on their keyboard while clicking on the field in the table, which will reveal the options to Move, Copy, or Delete all the selected fields. To make users aware of this feature, a floating note now appears near the right side of the page in the Online Designer with instructions on how to use this.
- **New Feature:** New logic editor for conditional logic, branching logic, calculations, report filters, etc. In every place where users might add/edit logic or calculations, the new logic editor will be displayed in a modal dialog to provide a better user experience for entering their logic. The logic editor provides much more space for entering large amounts of logic, including a full screen mode to take maximum advantage of their screen's real estate. It also provides bracket-matching and parentheses-matching where it will highlight a pair of matching brackets/parentheses to make it easier for users to gauge which brackets/parentheses belong together in the logic, thus reducing possible errors in the logic when typing
- **New action tag: @CALCDATE** - Performs a date calculation by adding or subtracting a specified amount of time from a specified date or datetime field and then provides the result as a date or datetime value - e.g., @CALCDATE([visit\_date], 7, 'd'). The first parameter inside the @CALCDATE() function should be a text field with date, datetime, or datetime\_seconds validation, in which you may specify (if needed) the event and repeating instance - e.g., @CALCDATE([baseline\_event][visit\_date], 7, 'd'). The second parameter represents the offset number amount that should be added or subtracted. It can be a decimal number or integer. Tip: To subtract (i.e., go backwards in time), use a negative number. The third parameter represents the units of the offset amount, which will be represented by the following options: 'y' (years, 1 year = 365.2425 days), 'M' (months, 1 month = 30.44 days), 'd' (days), 'h' (hours), 'm' (minutes), 's' (seconds). The unit option must be wrapped in quotes or apostrophes. **NOTE:** Both the source field and the result field must be a text field with date, datetime, or datetime\_seconds validation. It is important to realize that a field with @CALCDATE will not be editable on the survey page or data entry form, and the field will function almost exactly like a normal calculated field, in which its value may get updated via a data import, when running Data Quality rule H, or in real-time during normal data entry on a form or survey. This document contains some [examples of using @CALCDATE](#).
- **New action tag: @CALCTEXT** - Evaluates logic that is provided inside a @CALCTEXT() function and outputs the result as text, typically performed with an if(x,y,z) function - e.g., @CALCTEXT(if([gender]='1', 'male', 'female')). **NOTE:** It is important to realize that a field with @CALCTEXT will not be editable on the survey page or data entry form, and the field will function almost exactly like a normal calculated field, in which its value may get updated via a data import, when running Data Quality rule H, or in real-time during normal data entry on a form or survey. If desired, it is possible to return the value as a number - e.g., @CALCTEXT(if([age] >= 18, 'adult', 5\*[other\_field])). This document contains with some [examples of using @CALCTEXT](#).
- **New action tag: @PREFILL** - Sets a field's value to static text or dynamic/piped text whenever a data entry form or survey page is loaded, in which it will always overwrite an existing value of the field. The format must follow the pattern @PREFILL="????", in which the desired value should be inside single or double quotes. A field with @PREFILL will always be read-only, thus its value cannot be modified manually on the data entry form or survey page. For text fields, you may pipe and concatenate values from other fields in the project - e.g., @PREFILL='Name: [first\_name] [last\_name], DOB: [dob]'. For checkbox fields, simply separate multiple checkbox values with commas - e.g., @PREFILL='1,3,[other\_field:value]'. **NOTE:** The piped value does *not* get applied during any data imports (via API or Data Import Tool) but only operates when viewing survey pages and data entry forms. **NOTE:** A field with @PREFILL will have its value updated **ONLY** when the page loads, which means that its value will not be updated in real-time if you modify other fields on the same page that are piped into the

@PREFILL tag. NOTE: If being used on a date or datetime field, the date value inside the quotes must be in Y-M-D format - e.g., @PREFILL='2007-12-25' - regardless of the field's set date format. NOTE: The only difference between @PREFILL and @DEFAULT is that @DEFAULT is only applied when an instrument has no data yet, whereas @PREFILL will always be applied on an instrument, meaning that @PREFILL will ALWAYS overwrite the value if a field value already exists. TIP: To pipe the value of one multiple choice field into another multiple choice field, make sure you append ':value' to the variable being piped - e.g., @PREFILL=[my\_dropdown:value].

- **New special functions:** left(), right(), mid(), length(), find(), trim(), upper(), lower(), and concat(). These nine new functions can be specifically used when dealing with text values and may be especially useful when using them in conjunction with the @CALCTEXT action tag. To learn more and to see some practical examples of their usage, click the blue 'Special Functions' button in the Online Designer in any project.
  - **left** (text, number of characters) - Returns the leftmost characters from a text value. For example, left([last\_name], 3) would return 'Tay' if the value of [last\_name] is 'Taylor'.
  - **right** (text, number of characters) - Returns the rightmost characters from a text value. For example, right([last\_name], 4) would return 'ylor' if the value of [last\_name] is 'Taylor'.
  - **length** (text) - Returns the number of characters in a text string. For example, length([last\_name]) would return '6' if the value of [last\_name] is 'Taylor'.
  - **find** (needle, haystack) - Finds one text value within another. Is case insensitive. The "needle" may be one or more characters long. For example, find('y', [last\_name]) would return '3' if the value of [last\_name] is 'Taylor'. The value '0' will be returned if "needle" is not found within "haystack".
  - **mid** (text, start position, number of characters) - Returns a specific number of characters from a text string starting at the position you specify. The second parameter denotes the starting position, in which the beginning of the text value would be '1'. The third parameter represents how many characters to return. For example, mid([last\_name], 2, 3) would return 'AYL' if the value of [last\_name] is 'TAYLOR'.
  - **concat** (text,text,...) - Combines/concatenates the text from multiple text strings into a single text value. For example, concat([first\_name], ' ', [last\_name]) would return something like 'Rob Taylor'. Each item inside the function must be separated by commas. Each item might be static text (wrapped in single quotes or double quotes), a field variable, or a Smart Variable.
  - **upper** (text) - Converts text to uppercase. For example, upper('John Doe') will return 'JOHN DOE'.
  - **lower** (text) - Converts text to lowercase. For example, lower('John Doe') will return 'john doe'.
  - **trim** (text) - Removes any spaces from both the beginning and end of a text value. For example, trim(' Sentence with spaces on end. ') will return 'Sentence with spaces on end.'
- **Improvement:** Field variables and Smart Variables can now be piped into the "src" attribute of HTML image tags ("img"). Previous versions did not allow this but would allow the piping of values into the "href" attribute of hyperlinks. Now both are possible. Note: The value being piped into the "src" attribute must have already been saved prior to the page loading, which is how piping for the "href" attribute has always worked.
- **Change:** In the field-view display of the Online Designer, each field will no longer have a link to open the instructional popup for Field Embedding (since this is a bit repetitive), but instead there will be a single floating box on the right side of the page for informing users on how to utilize Field Embedding and for opening the popup

## Alerts & Notifications

- **New feature:** Users may re-evaluate some or all Alerts & Notifications for all records in a project. If an alert has been modified after data has already been entered in the project, users may click the "Re-evaluate Alerts" button on the Alerts & Notifications page, which will re-evaluate selected alerts for all records to ensure that notifications get properly sent or scheduled based on the new conditions of the alert (otherwise they could only be triggered if each individual record had data modified). If a user modifies the conditional logic of an alert, it will recommend that they utilize the "Re-evaluate Alerts" functionality. If an alert has the "Ensure logic is still true..." option checked, then it is possible during this process that some already-scheduled notifications might get removed (and thus would no longer be scheduled) based on the new conditions.
- **Change/improvement:** In Alerts & Notifications, the email "To" field that is used for an alert now allows fields having custom email validation (i.e., added manually in the redcap\_validation\_types database table) to be

utilized as the recipient. Previous versions only allowed fields to be used that had the explicit "email" field validation.

## Data Import

- **New feature: Auto-numbering of repeating instances for data imports** - When using repeating events or repeating instruments, it may be difficult when performing dynamic imports of data for these because it is not easily known how many repeating instances already exist in a project for a given repeating event/instrument, thus often forcing users to invent clever ways to determine this, such as performing data exports beforehand and then dynamically determining what the next repeating instance number should be. However, that is no longer necessary. When performing a data import now for a repeating event/instrument, users may use the literal value "new" as the value for the "redcap\_repeat\_instance" field in their data import. By doing so, REDCap will perform the instance auto-numbering on its own to increment the repeating instances properly based on the highest numbered instance that already exists in the saved data in the project.
- **Improvement:** Import Records API method has a new parameter "csvDelimiter" to specify the delimiter character when sending data in CSV format. Options include: comma ',' (default), 'tab', semi-colon ';', pipe '|', or caret '^'.
- **Improvement:** Large performance improvement when importing lots of records (via Data Import Tool, Mobile App, API, or REDCap::saveData) when record auto-numbering is enabled in a project and the import process is forcing the new records to be auto-numbered on the fly
- **Improvement:** When importing a CSV file of data on the Data Import Tool or when importing a CSV Data Dictionary, users may now specify the delimiter of the CSV file as a Comma (default), Tab, or Semicolon.

## Logging

- **Improvement:** The Logging page in a project now defaults to only displaying the logged events from the past week, although users can always adjust the filter settings to expand the time range values after initially loading the page. This change makes the Logging page load much faster when initially loaded by a user.
- **Improvement:** The Logging page in a project now has helpful buttons to quickly adjust the time range filters to Past Day, Past Week, Past Month, Past Year, and all time ("no limit") to make it easier for users to view the logging from various time ranges.
- **Improvement:** The Logging page in a project now has more download options for exporting the logging. In addition to exporting all logged events, there are now the following new buttons: 1) Export all pages using current filters and 2) Export current page.

## Data Access Groups (DAGs)

- **New feature: Data Access Group import/export and DAG-User assignment import/export** - The Data Access Groups page in a project now displays a drop-down list of options for users to import/export Data Access Groups, which allows users to bulk create or rename DAGs via a CSV file. It also allows for the import/export of DAG-user assignments via CSV file to bulk assign/reassign/unassign users from DAGs in a project. Note: The DAG-user assignment import affects only a user's \*current\* DAG assignment; thus, it has no effect on the DAG Switcher assignments for the user.
- **Improvement:** In previous versions, the DAG Switcher would not be very performant if there existed very many users and/or Data Access Groups within a project, thus the DAG Switcher would (by design) be automatically disabled for projects in those situations, which was not ideal with regard to user experience. The DAG Switcher is now no longer limited in that way and will now function fully regardless of there being many users and DAGs in a given project.

## Data Quality

- **New feature: Data Quality Rule import/export** - The Data Quality page in a project now displays a drop-down list of options for users to import/export custom Data Quality rules via a CSV file. Note: This does not apply to the pre-defined DQ rules (rules A-I). Also, when DQ rules are imported, the process is additive only, meaning that the CSV upload cannot replace or edit existing DQ rules but will only add new ones to the project.