REDCap Common User Issues and Solutions

Here you will find an ongoing list of issues or oddities in REDCap that have been reported by multiple users over time, along with solutions provided by users and administrators to document these items and give other users ways to navigate around them.

1) **Unexpected value results from calculations using fields that have missing data codes**

The scenarios in **green** indicate the calculation is correct.
The scenarios in **red** indicate the calculation is incorrect.
The scenario in **blue** indicate the scenario is not possible.

Suppose we have three data fields: 1) [MAP] (mean arterial pressure), 2) [SBP] (systolic blood pressure), and 3) [DBP] (diastolic blood pressure). [MAP] is a calculated field, rounded to one decimal place, with the calculation [MAP] = **round((([SBP]+2*[DBP])/3),1)**.

This calculation produces a correct [MAP] result when both the [SBP] and [DBP] fields have values, regardless of whether [SBP] and [DBP] are designed as text fields with or without numeric validations.

However, the accuracy of this calculation, in the presence of missing data, depends on how the [SBP] and [DBP] fields are designed and how missing data are handled.

Scenario 1 - [SBP] and [DBP] are designed as text fields with numeric validations.

Scenario 1.1: [SBP] and [DBP] fields are left <blank> when their values are missing.

The resulting value of [MAP] is correctly also left <blank>.

Scenario 1.2: [SBP] and [DBP] fields have numeric missing data codes entered when their values are missing.

The resulting value of [MAP] is incorrectly calculated to be a numeric value since the numeric missing data code is used as an actual value in the calculation.

Scenario 1.3: [SBP] and [DBP] fields have character missing data codes entered when their values are missing.

Not possible due to the numeric validations on the [SBP] and [DBP] fields.

Scenario 2 - [SBP] and [DBP] are designed as text fields without numeric validations.

Scenario 2.1: [SBP] and [DBP] fields are left <blank> when their values are missing.

The resulting value of [MAP] is incorrect. The calculation treats any <blank> value as a '0'.

Example 1: if [SBP]=120 and [DBP] is <blank>, the resulting [MAP] value is 40.

Example 2: if [SBP] is <blank> and [DBP]=80, the resulting [MAP] value is 53.3.

Example 3: if [SBP] is <blank> and [DBP] is <blank>, the resulting [MAP] value is 0.

Scenario 2.2: [SBP] and [DBP] fields have numeric missing data codes entered when their values are missing.

The resulting value of [MAP] is incorrectly calculated to be a numeric value since the numeric missing data code is used as an actual value in the calculation.

Scenario 2.3: [SBP] and [DBP] fields have character missing data codes entered when their values are missing.

The resulting value of [MAP] is correctly left <blank>.

The **red scenarios 1.2 and 2.2**, where a numeric missing data code is used, should be avoided when the values are to be used in calculations.

A solution to the **red scenario 2.1** is to use the calculation [MAP] = **round(((([SBP]+2*[DBP])/3)*([SBP]*[DBP])/([SBP]*[DBP]),1)**.

Note:

- when both [SBP] and [DBP] fields have values, the additional portion of the calculation is simply a multiplication by '1'
- when one or both of these two fields are left <blank>, the calculation treats the <blank> values as a '0' resulting in a division by '0'
- invalid calculations cannot result in numeric values, so the result in REDCap from invalid calculations is correctly a <blank>

2) SAS export data informats for Date/Datetime fields

The generated SAS code does not apply numeric informats to Time (h:m:s), Time (mm:ss), Datetime or Datetime w/ seconds. Instead these are all imported as $500. However, better informats may be available, but should be used with care as how SAS interprets dates may depend on the value of the DATESTYLE system option.

This excerpt from SAS code shows the different date/time validation types:

```
/* Dates */

when ('date_mdy') do;
       sasinformat    = 'yymmdd10';
       sasformat      = 'date9';
end;

when ('date_dmy') do;
       sasinformat    = 'yymmdd10';
       sasformat      = 'date9';
end;

when ('date_ymd') do;
       sasinformat    = 'yymmdd10';
       sasformat      = 'date9';
end;

/* Datetimes */

when ('datetime_dmy') do;
       sasinformat    = 'ymddttm16';
       sasformat      = 'datetime16';
end;

when ('datetime_mdy') do;
       sasinformat    = 'ymddttm16';
       sasformat      = 'datetime16';
end;
```

```
        when ('datetime_ymd') do;
                sasinformat     = 'ymddttm16';
                sasformat       = 'datetime16';
        end;

/* Datetimes with seconds */

        when ('datetime_seconds_dmy') do;
                sasinformat     = 'ymddttm19';
                sasformat       = 'datetime19';
        end;

        when ('datetime_seconds_mdy') do;
                sasinformat     = 'ymddttm19';
                sasformat       = 'datetime19';
        end;

        when ('datetime_seconds_ymd') do;
                sasinformat     = 'ymddttm19';
                sasformat       = 'datetime19';
        end;

/* Times */

        when ('time') do;
                sasinformat     = 'time8';
                sasformat       = 'time5';
        end;

        when ('time_hms') do;
                sasinformat     = 'time8';
                sasformat       = 'time8';
        end;

        when ('time_mm_ss') do;
                sasinformat     = 'stimer';
                sasformat       = 'time8';
        end;
```