



**ActivityInfo**

Enhancing data entry efficiency

**Starting shortly, Please wait!**



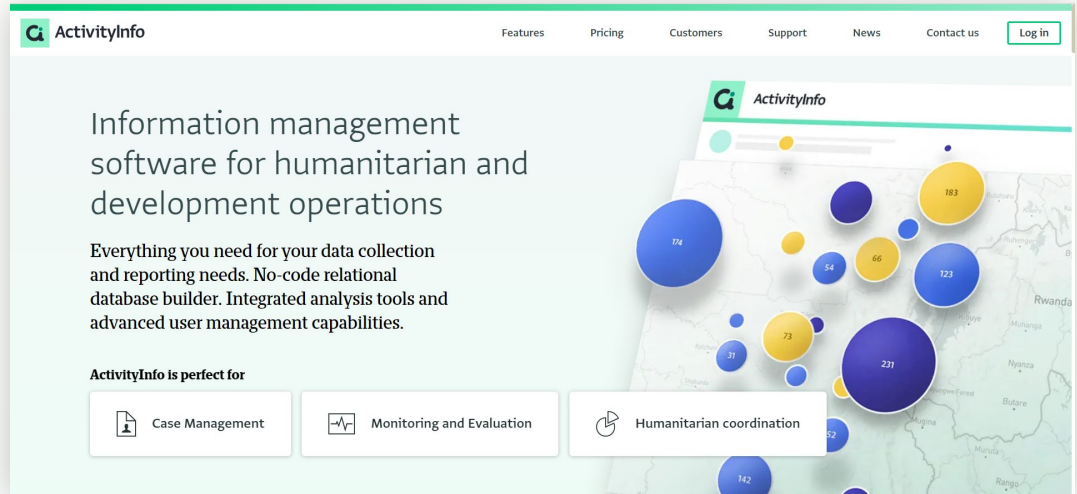
**ActivityInfo**

# Presented by the ActivityInfo Team

---

All in one information management software for humanitarian and development operations

- Track activities, outcomes
- Beneficiary management
- Surveys
- Work offline/online



The screenshot shows the ActivityInfo website homepage. At the top, there is a navigation bar with the ActivityInfo logo on the left and links for Features, Pricing, Customers, Support, News, Contact us, and a Log in button on the right. The main content area features a large heading: "Information management software for humanitarian and development operations". Below this heading is a sub-heading: "Everything you need for your data collection and reporting needs. No-code relational database builder. Integrated analysis tools and advanced user management capabilities." Underneath, it states "ActivityInfo is perfect for" followed by three icons and labels: "Case Management" (with a document icon), "Monitoring and Evaluation" (with a line graph icon), and "Humanitarian coordination" (with a globe icon). On the right side of the page, there is a map of Africa with several colored bubbles (blue, yellow, purple) overlaid, each containing a number representing data points. The bubbles are of varying sizes and colors, with numbers like 174, 183, 123, 73, 54, 66, 231, 71, 52, and 142.

# Meet your instructor

---



**Victoria Many**  
Customer Education Specialist  
BeDataDriven

# Outline

**01** Data entry efficiency

**02** Relevance rule and formulas

**03** Validation rules and formulas

**04** The use of Subsections for organizing data entry forms

**05** Unique field constraints versus Key fields

**06** Default values and the Read Only property

**07** NOW function and timestamps

# Data entry efficiency: Overview

---

Data entry efficiency refers to the ability to accurately and swiftly input data into a system or database while minimizing errors and maximizing productivity. It encompasses several factors, including the speed of data entry, the accuracy of entered data, the ease of use of data entry tools and interfaces, and the overall effectiveness of the data entry process.

# Data entry efficiency: Principles

---

Some principles for efficient data entry

- **Automate:**
  - Simplify the interface
  - Use default values
  - Calculated fields
  - Allow bulk data entry
- **Reduce Choice:**
  - Provide validation and relevance checks
  - Use default values
  - Offer dropdowns and selection lists
- **Standardize:**
  - Use standardized formats
  - Provide data entry validation and feedback

# Data entry efficiency in ActivityInfo

---

- Collect data online, offline, on desktop, or mobile devices
  - Simplify consistent data entry with web-based forms on desktop, tablet, or mobile
  - Easy data collection interface for all users
  - Mobile app for efficient data collection
  - User-friendly data entry page with descriptions, section headers, date, and selection fields
  - Easy navigation between sections and questions
  - Use collection links to add data without logging into the platform
  - Offline data collection in remote areas with limited internet connectivity
    - Sync data when internet connection is available again

# Our Intervention

## Use Case

---

we have a humanitarian intervention where we are administering health related support, food, water, shelter and other support resources are distributed to the affected population. We are attending to both adults and children. Also, we are collecting both primary and secondary data and administering post distribution surveys to gauge our impact.

We will be adopting this storyline to illustrate the use of all the features in this session.



| Automate

# Calculated Fields and Formula

## Overview of Calculated fields

---

- Adding Calculated fields automates calculations and allows for direct results based on other fields.

With the formula:

- Formulas can be written for Calculated fields using Codes from other fields in the Formula editor.
- Calculated fields can be used for various purposes, including: Performing mathematical operations such as addition, subtraction, multiplication, and division on indicators. And many more possibilities! -
- Calculated fields could be hidden and not appear in the Data Entry page to avoid confusion for users when adding records

# EXAMPLE: Calculated field

---

**Scenario:** we collect primary data on beneficiaries. During project implementation some beneficiaries might be more vulnerable than others and we want to calculate a vulnerability score to alert us to each beneficiaries case. This is calculated based on other fields in the forms or subforms.

**Formula:**

```
IF(Totalvulnerabilityscore < 35, "NotVulnerable",  
IF(Totalvulnerabilityscore <= 50, "Moderately Vulnerable",  
IF(Totalvulnerabilityscore > 50, "HighlyVulnerable")))
```

# Default Value

## Overview of Default Value

---

- **Default**

- Default values are preset values that automatically appear in a field when creating a new record.

### Default values:

- Streamline data entry and automate the process, increasing consistency and saving time for users.
- Are only generated when adding a new record, not when editing an existing record.
- If a field with a default value is left blank when using the importer to add a new record, the record will be added with that field left blank.
- There are two ways to define a default value for a field:
  - Assigning an explicit value, which is a single value specified by the user.
  - Using a formula to automatically generate a calculated value, which is useful when the expected value can vary based on other field values or context.

# NOW and Timestamps

## Overview of NOW Function

---

**NOW Function** provides the current date and time information for records in a form.

NOW function is :

- Useful in situations where real-time information is needed, such as tracking record entry events, or timestamping data.
- The format for the date and time that would be returned is :YYYY-MM-DD HH:mm:ss +0100
- Syntax is :NOW()

# EXAMPLES: Default, NOW and Read-only

---

**Scenario:** we collect primary data on beneficiaries. We want to maintain for monitoring purposes the date and time when data collection started. We do not wish the enumerators to be able to modify this field. Thus we place a text field, with the default value formula:

```
TEXT(NOW())
```

We then set the field as “**Read-only**”

| Standardize and  
Reduce Choice

# Relevance rules and Formula

## Overview of relevance rules

---

### Relevance rules and formulas

- You can define various Relevance rules to customise when a field will appear by writing a Formula to define a more advanced rule.

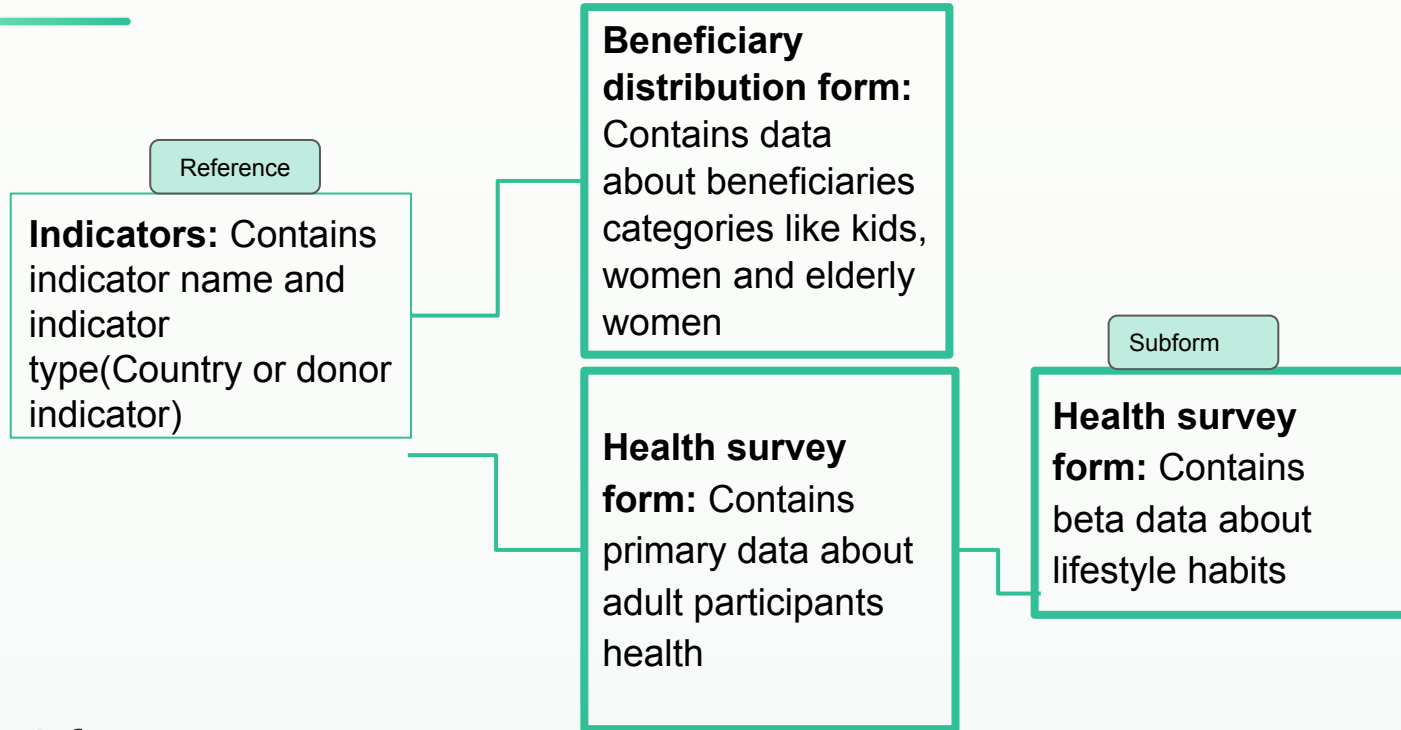
With the formula:

- You can customise when a field will appear in the Data Entry page.
- You can define under which conditions (rules) the field should appear to the user.
- This way you can save time and avoid confusion for users adding Records who don't need to fill in specific fields.



# EXAMPLES: Field based Relevance rule

## Data Model



# EXAMPLES: Field based Relevance rule

**Scenario:** We have a scenario where we collect secondary data around kids, women and elderly women for different indicators. Depending on the indicator, we require different fields per women, kids and elderly women. We wish to mainstream the data collection via one data collection form, instead of creating different collection forms per indicator. So, we use the following relevance rules based on indicator ID stored in our reference data and referenced in our data collection form:

**Formula for Kids field:** `Indicatorid.Indicators == "Indicator 1" || Indicatorid.Indicators == "Indicator 2"`

**Formula for women's field:** `Indicatorid.Indicators == "Indicator 2" || Indicatorid.Indicators == "Indicator 3"`

**Formula for elderly women's field:** `Indicatorid.Indicators == "Indicator 1" || Indicatorid.Indicators == "Indicator 3"`

# EXAMPLES: Relevance rule for subform

**Scenario:** For the adults in this intervention, in the second form, we are managing a health survey to collect primary data on their medical conditions and risk factors. We have a subform that collects further information about adult patients' lifestyle habits but we only want to display the subform asking detailed questions about lifestyle habits if any of the following conditions is met:

- The adult is a smoker (Field1 = "Yes")
- their cholesterol level is higher than 10 (Field2 > 10).
- The adult is not physically active (Field3 = "No")
- their blood pressure is high (Field4 = "High").

If neither of these conditions is met, the subform will be hidden, as it is not relevant to the adults current lifestyle habits.

## Formula:

```
[Smoking Status].Yes || [Cholesterol Level] > 10) || [Physical Activity].No) || [Blood Pressure].Low
```

# Validation rules and Formula

## Overview of Validation rules

---

### Validation rules and formulas

- Writing a formula for a validation rule allows for more advanced and customized validation rules.

With the formula:

- You can define validation rules for specific types of fields to determine the validity of added records.
- Validation rules prevent users from saving records that do not comply with the defined rules.
- Validation rules can be added to reference fields to restrict the available answer options for users.
- Validation rules can be based on the fields within the form or on specific formulas.

# EXAMPLES: Validation rule - range

---

**Scenario:** we collect primary data for the post distribution survey and we have a question that we ask “ During the last 7 days, how many days did you consume the following food type?”. We expect an input between 0 and 7 thus, the validation formula is as follows


**Formula:** `question>=0 && question<=7`

# EXAMPLES: Validation rule - filtering case

---

**Scenario:** we collect primary data on beneficiaries. During project implementation some beneficiaries will leave the project. Thus those will be considered as “inactive”. We collect post distribution surveys and we want to ensure that enumerators will not choose ID of inactive users. Note that the post distribution form has a reference field to the registration form.

**Formula:** `[Serial Number].[Beneficiary status].Active`



Demo

# Unique field constraints versus Key fields

## Overview of Unique and Key fields

---

- **Unique fields**

- The Unique property in ActivityInfo is used to prevent duplication in form design

By marking a field as "unique":

- it ensures that no two records in the form can have the same value in that specific field.
  - Within the form where the field is located, and
  - Among the records visible based on the user's permissions.
- It helps keep data clean and avoids confusion or errors caused by repeated information in different parts of the form.



# EXAMPLES: Unique field constraints

---

**Scenario:** we collect primary data on beneficiaries. The beneficiary ID Includes an identifier, thus this is the only and unique of the form. We wish to identify uniqueness based on : first name, last name, document ID, phone number. Thus we create a calculated field which we consider as “unique”

**Formula:**

```
CONCAT([First Name], [Last name], [Document ID], [Phone number])
```

# Locking data entry

## Overview of locking data entry

---

**A Lock is** a restriction that prevents users from adding or editing data in a resource (Folder, Form, or Subform)

- Locks preserve validated versions of data
- Locks can be based on a date range or rule.
- Date range Locks can be set on Forms or Subforms with a Date/Month/Week field as a Key field, or on Databases or Folders containing such Forms.
  - Users receive an error message when trying to add a Record for the locked period and cannot save the Record. -
- Rule-based Locks are applicable only to Forms and Subforms.
  - They use a Formula to define a rule based on criteria like region or response type.
  - Locks have an 'active' Property that allows temporary suspension of the Lock.

# EXAMPLES: Locking data entry

---

**Scenario:** we collect secondary data and the reporting for last year(2023) has been finalized. We wish to lock the records for 2023. We can employ a date based lock as based on the month reference date that we use inside our form

# Sections and Subsection fields

## Overview of sections and subsections

---

**Section and subsection headers:** Allow creation of sections and subsections in Forms and Subforms.

Section and subsection fields are:

- Facilitates navigation in long Forms/Subforms.
- Enhanced user navigation on the Data Entry page.
- Improved organization and clarity in form structure.
- Flexibility in designing user-friendly interfaces.



Demo

# Best Practices in Data entry

---

- Pilot data entry for feedback and Improvement prior to deployment
- Clear and consistent data entry guidelines
- User-friendly interface
- Data entry training and support
- Data entry controls and permissions
- Ongoing database maintenance

# Questions?

Follow us:

LinkedIn page: <https://www.linkedin.com/showcase/activityinfo/>

LinkedIn group: <https://www.linkedin.com/groups/5098257/>

Twitter: <https://twitter.com/activityinfo>