A decorative vertical bar on the left side of the slide, featuring a teal color with overlapping geometric shapes like diamonds and rounded rectangles.

# Level 2 Data Quality Academy

Data Quality in Tracker Programs

# Overview

- Transferring data from tracker to aggregate
- Common issues with data quality in patient records.
- Program rules for data quality checks
- Monitoring completeness through program indicators

# Common Issues

- Complex data entry with little or no workflow support
  - Incorrect patient data
- Calculating coverage baseded upon only tracker data
- Calculating completeness from individual patient data



## Program Rules

- Program rules provide a way to produce dynamic behaviors in response to user input in tracker and event capture
- Program rules allows you to create and control dynamic behavior of the user interface in the **Tracker Capture** and **Event Capture** apps
- During data entry, program rule expressions are evaluated each time the user interface is displayed and each time a data element is changed



## Program Rules

- Most types of actions will take effect immediately when the user enters values in the **Tracker Capture** and **Event Capture** apps
- Program rules consist of three components:
  - Variables
  - Expressions
  - Actions



# Program Rules - Variables

- Program rule variables represent attribute and data element values which will be evaluated as part of the expression -- *are the first step in making rules*
- They create a uniform way to include data values and attribute values in expressions and actions
- The program rule variable is defined as needed when creating expressions and actions
- When creating your first expression, you will typically have to define one or more program rule variables, as expressions usually contain at least one data element or attribute value to be meaningful
- The program rule variables are shared between all rules in your program, and by creating several rules, these will share the same library of program rule variables



# Program Rules - Variables

Example: You would like to use the attribute “Sex” in your expression. You would first define this as a program rule variable

← Program rule variable ?

Program (\*)  
Notifiable Medical Conditions Program

Name (\*)  
sex

☒ Use code for option set

Source type (\*)  
Tracked entity attribute

Tracked entity attribute  
Notifiable Medical Conditions Program Sex at Birth

SAVE

CANCEL

## Program Rule - Expressions

- During data entry, program rule **expressions** are evaluated each time the user interface is displayed, and each time a data element is changed
- Most types of actions will take effect immediately as the user enters values in tracker or event capture
- Expressions are built from **program rule variables, operators** and user-defined **static values**





# Program Rules - Operators

Operators are updated regularly in the [docs](#)

## Program Rules – Examples for data quality

- `{TrainedAdolescentHealthStaff} > {FacilityStaff}`
  - Assesses if there are more staff trained in adolescent health than staff within the entire facility
- `{vaccinedoses} >= 0 && {vaccinedoses} <= 4`
  - Assesses if the number of vaccines doses is greater than or equal to 0 **AND** less than or equal to 4
- `({SampleType} == 'Plasma' || {SampleType} == 'Sputum') && Sex == 'MALE'`
  - Assesses if the lab sample type is Plasma **OR** Sputum and the sex is male

# Program Rules - Custom Functions

- There are advanced functions that can be used in program rule expressions
- Examples include:

Function	Argument	Description
d2:ceil	(number)	Rounds the input argument <b>up</b> to the nearest whole number.
d2:monthsBetween	(date,date)	Produces the number of full months between the first and second argument.
d2:hasValue	(sourcefield)	Evaluates to true if the argument source field contains a value, false if no value is entered

## Program Rules - Custom Function Examples

- `d2:ceil({hemoglobinValue})`
  - If a value of 7.3 is entered for the hemoglobin value, this expression will round it up to 8
- `d2:monthsbetween({sampledate}, '2016-01-01')`
  - If the sample date is June 16, 2016, it will return a value of 5
- `d2:hasValue('vaccinedoses')`
  - If no value is entered this evaluates to false. If any value is entered, this will evaluate to true



## Program Rules - Actions

- Each program rule has one or more actions attached to it
- These are the behaviors that are triggered in the user interface when the expression is true
- Actions will be applied at once if the expression is true, and will be reverted if the expression is no longer true
- There are several types of [actions](#)



# Program Rules - Summary

← Program rule

1 Enter program rule details

Program (\*)

SARA RCH

Name (\*)

Adolescent Health Trained Staff > Facility Staff

Description

Adolescent Health Trained Staff > Facility Staff

Priority

SAVE

CANCEL

← Program rule

1 Enter program rule details

2 Enter program rule expression

3 Define program rule actions

Condition  
#{TrainedAdolescentHealthStaff} > #{FacilityStaff}

+ - \* / % > >= < <= == !=  
NOT AND OR

Variables

#{AdolescentHealthTraining}  
#{ANCServices}  
#{ANCTraining}  
#{ContraceptivesStocked}  
#{FacilityStaff}  
#{FamilyPlanningServices}  
#{FPTraining}  
#{IPTpTraining}

Functions

SAVE

CANCEL

← Program rule

1 Enter program rule details

2 Enter program rule expression

3 Define program rule actions

Action details

Show warning: "The amount of staff wh..." on "Staff - Adolescent Health Training"



SAVE

CANCEL

# Program Rules - Steps

1. Conceptualise the logic of the rule you want to create
2. Create a source field that points to the data element/attribute which will trigger your logic
3. Create a new program rule
  - a. Fill in the program rule details
    - i. Select the program
    - ii. Provide a name and description
    - iii. Assign a priority if applicable
4. Enter the program rule expression
  - a. Refer the the documentation to see many examples of the items that can be used within the expression, as well as examples of expressions
5. Define the program rules actions
  - a. Refer to the documentation for more detail on each individual action
6. Clear your cache and test the rule!

# Andriod

← Malaria simple cases

123 Files in Malaria simple cases

DD/MM/YYYY Org Unit Cat Combo

Date/Hour	Org Unit	Cat Combo	Age	Diagnosis	Completion	Lock	Sync
03/10/2017 11:12 AM	Org Unit	Cat combo	25	Diagnosis	44%	Lock	Error
03/10/2017 11:12 AM	Org Unit	Cat combo	25	Diagnosis	44%	Unlock	Correct
03/10/2017 11:12 AM	Org Unit	Cat combo	25	Diagnosis	44%	Unlock	Correct
03/10/2017 11:12 AM	Org Unit	Cat combo	25	Diagnosis	44%	Lock	Error
03/10/2017 11:12 AM	Org Unit	Cat combo	25	Diagnosis	44%	Lock	Warning
03/10/2017 11:12 AM	Org Unit	Cat combo	25	Diagnosis	44%	Unlock	Correct
03/10/2017 11:12 AM	Org Unit	Cat combo	25	Diagnosis	44%	Lock	

← Sarah Thompson

Sara Thompson  
21 yrs  
Registration OU 06/09/2016

Open

Following up

EDIT VIEW MORE

ANC Program  
Last visit: 16/09/2016  
Date of enrollment: 14/09/2016

Malaria Cases  
ADD TO PROGRAM

← 1 of 7 Diagnosis Method

44%

Microscope

RDT

No tested

PREVIOUS NEXT

← 6 of 7 Treatment

44%

3 ASMQ Tablets 25/50 mg

6 ASMQ Tablets 25/50 mg

3 ASMQ Tablets 100/200 mg

6 ASMQ Tablets 100/200 mg

6 ASMQ Tablets 100/200 mg + one 15mg PQ Tablet Referred

Referred

PREVIOUS NEXT



# Program Indicators

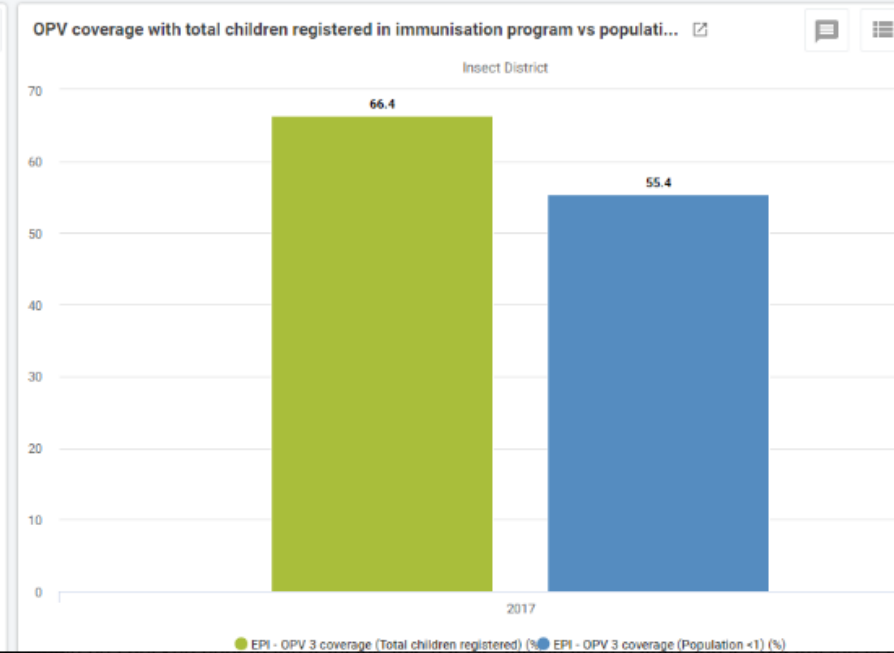
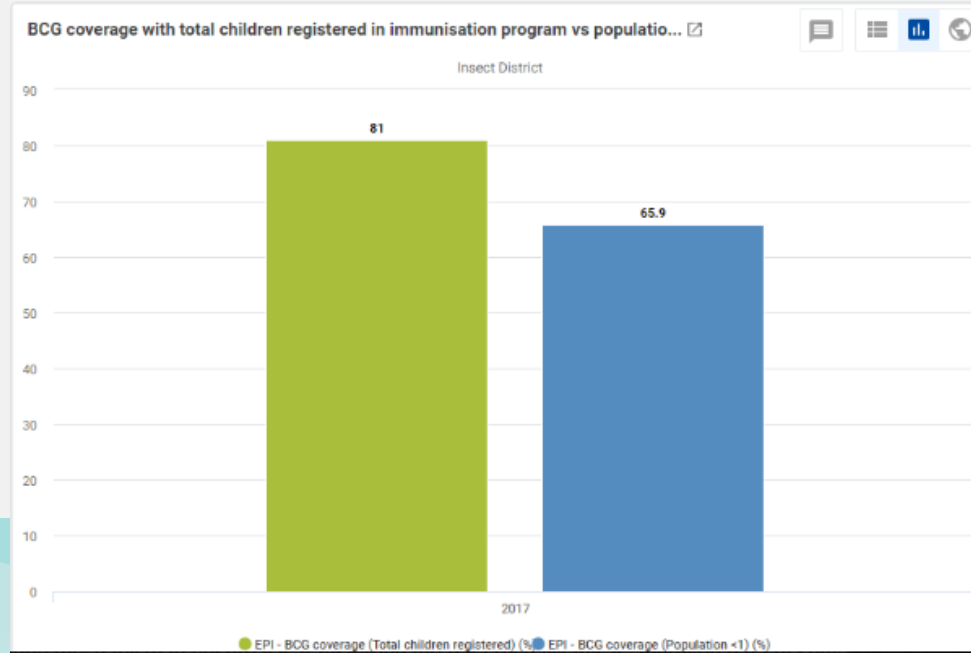
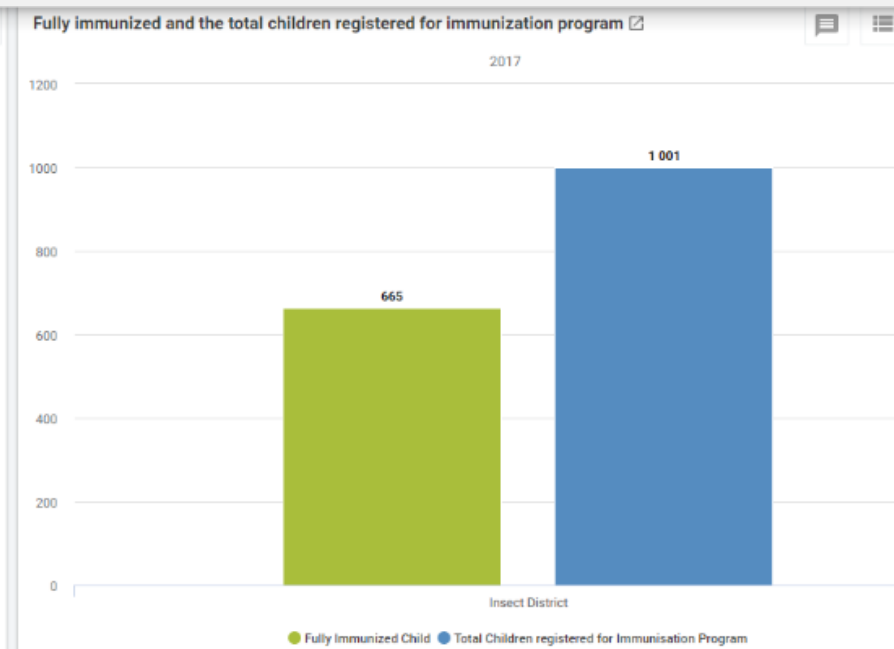
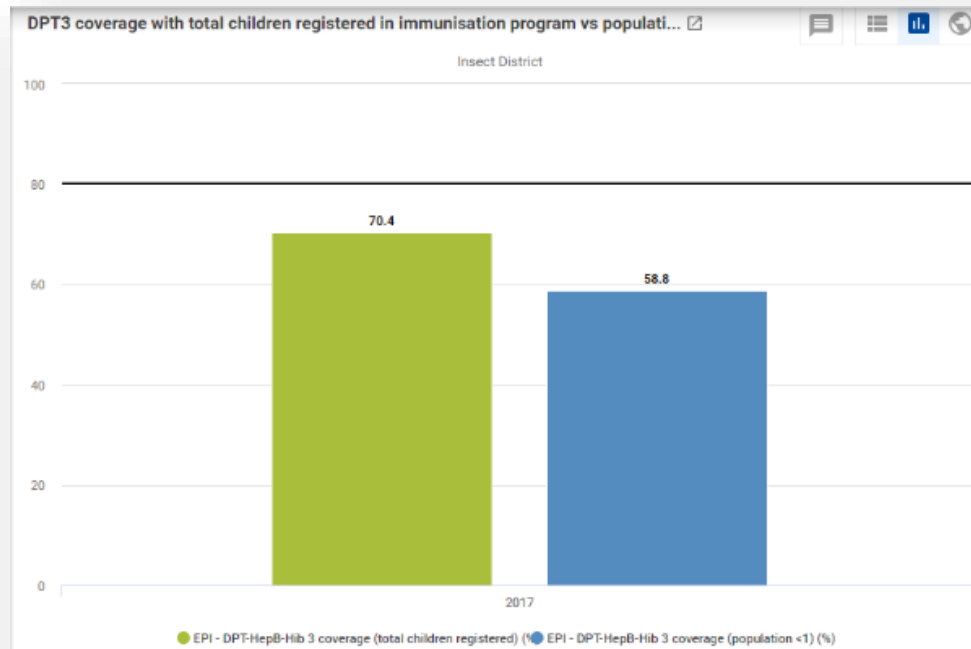
- Program indicators allow for the creation of values based on **data elements** and/or **attributes** belonging to event or tracker programs
- Program indicators use a combination of **expression**, **aggregation type** and **filters** in order to determine how the indicator will be evaluated

# Example Program Indicators

- Completeness of data elements in tracker programs.
  - $\text{Count of BCG Doses given} / \text{Total number of tracked children}$
  - Return the percentage of tracked children who have a recoded a BCG Dose

# Coverage Indicators Using Tracker Data

- Indicators using both denominators (registrations and population) rather than prescribing one over the other;
  - we generally wouldn't recommend using only registrations alone.
- Have a series of charts to then highlight the differences in coverage between the indicators calculated using both methodologies.



# Monitoring aggregate data compiled from tracker to assure that errors have not occurred in data transfer

- The exchange mechanism to ensure it is transferring the data
  - usually this is approved data is aggregated via program indicators.
- This would generally be transferred to data elements in the aggregate warehouse and should match the input 1-1 from the tracker system; this can be checked after the transfer is complete

# Required for transferring data:

1. The program indicators (or system specific process if DHIS2 is not being used) necessary to aggregate and anonymize your data
2. The data approval/review process in place to check everything before the transfer
3. The mechanism to move the data from one system to another
4. The review process in place to check after the transfer
5. Any other data quality checks you would normally apply to this data