# Aggregate Data Elements and Data Sets SOP

## Category Options, Categories, Category Combinations, Data Elements, Data Element Groups and Data Sets

This is to be done on the ***DEVELOPMENT*** server first! Never create these objects on the production server without the review having been completed on development.

## Category Options

* Check if the category option already exists to avoid duplicates. You must always search to see if the category option exists in all cases prior to proceeding.
  + If it exists, ***do not create a new one*** and stop at this step
  + If it does not exist, proceed to the next step
* You should never create category options which mix different categories (ie. Female (0-14 years)).
* Create the new category option
* Set the sharing settings:
  + Public : metadata can view, data can capture and view
  + NHMIS admin : metadata can edit and view, data can capture and view
* Users with the role **System Administrator** will be able to add and edit category options

## Categories

* Check if the category already exists to avoid duplicates. You must always search to see if the category exists in all cases prior to proceeding.
  + If it exists, ***do not create a new one*** and stop at this step
  + If it does not exist, proceed to the next step
* Similar to category options, categories should be independent of one another (ex. Age + Sex is not the correct way to create categories; create Age separately and Sex separately)
* Create the new category
* Use generic descriptors for the categories which are specific enough to identify what the category is, but which are not overly specific to any one use. See [Annex A – Category Best Practices](#_30j0zll) for some best practices regarding category naming.
* By default, “data dimension” is ticked when you make a new category. You must decide if this should be ticked or not. If you tick it, it will appear as a usable dimension in analysis apps. It will also result in increased analytics workload as DHIS2 will need to ensure this data is available in the analytics table. This, in turn, will result in increased disk space being used for the instance as well. ***This is not always needed and requires careful consideration***.
* DHIS2 provides an option for "Disaggregation" or "Attributes" for categories. Typically, you should choose "Disaggregation", unless you plan on using the category for a data set value attribute (such as a funding partner or project).
* Set the sharing settings:
  + Public : metadata can view
  + NHMIS admin : metadata can edit and view
* Users with the role **System Administrator** will be able to add and edit category options

## Category Combination

* Check if the category combination already exists to avoid duplicates. You must always search to see if the category combination exists in all cases prior to proceeding.
  + If it exists, ***do not create a new one*** and stop at this step
  + If it does not exist, proceed to the next step
* Create the new category combination
* Set the sharing settings:
  + Public : metadata can view
  + NHMIS admin : metadata can edit and view
* Users with the role **System Administrator** will be able to add and edit category combinations
* You should be EXTREMELY careful not to modify the category combination after you have created it. The reason for this is that each time you modify a category combination, new category option combinations may be created.
* Generate the new category option combinations by running the “Update category option combinations” option within the maintenance section of Data Administration
* Check that the new generated category option combinations are what you expect before you using it within the implementation
* Please see [Annex B – Category Combination Best Practices](#_1fob9te) for more information on the implications of modifying an existing category combination and how to check if your category option combinations have been generated correctly.

## Aggregate Data Elements

* Data elements will not be created with public access of any kind
* Data elements should have a proper name, short name and code
  + This is the responsibility of the health programs to define their given structure
  + The health program prefix must be in the data element name but should not be included in the short name or form name
* Data elements need to be assigned to a respective data set and data element group. If any data element is not present in a data set and data element group it should be reviewed either for removal or added to a group.
* Data elements need to be assigned to user groups so users can view and edit them. Use this table to make the assignment:

|  |  |
| --- | --- |
| Program | User Group Sharing Settings |
| Malaria | * Malaria access : metadata can view * Malaria data entry: metadata can view * Malaria admin : metadata can edit and view |
| TB | * TB access : metadata can view * TB data entry: metadata can view * TB admin : metadata can edit and view |
| Immunisation | * Immunisation access : metadata can view * Immunisation data entry : metadata can view * Immunisation admin : metadata can edit and view |
| HIV | * HIV access : metadata can view * HIV data entry : metadata can view * HIV admin : metadata can edit and view |
| NHMIS | * NHMIS2019 – Data Entry : metadata can view * NHMIS access : metadata can view * NHMIS admin : metadata can edit and view |

* The same data element cannot be present in two datasets with different period types. For example, BCG dose given cannot be present in both a monthly immunization data set and a quarterly immunization data set. In this scenario, you will create two different data elements - BCG dose given (monthly) and BCG dose given (quarterly).
* If a data element has two different sources, it should be treated as two separate data elements. For example: Population (HIS) and Population (Malaria). This should be treated as two different data elements with two different data sets and its own sharing settings. These data elements should be also assigned to different data element groups.
* Data elements that need to be updated with a **new disaggregation** will be re-used using the category combo over-ride function
* Users with the role **Administrator – Data Elements** or **System Administrator** will be able to create data elements

## Data Element Groups

* Data Element Groups will not be created with public access of any kind
* Health Program users that are part of their respective admin group and have the Administrator – Data Element role can create Data Element groups.
* Data element groups need to be assigned to user groups so users can view and edit them and. Use this table to make the assignment:

|  |  |
| --- | --- |
| Program | User Group Sharing Settings |
| Malaria | * Malaria access : metadata can view * Malaria data entry: metadata can view * Malaria admin : metadata can edit and view |
| TB | * TB access : metadata can view * TB data entry: metadata can view * TB admin : metadata can edit and view |
| Immunisation | * Immunisation access : metadata can view * Immunisation Data Entry : metadata can view * Immunisation admin : metadata can edit and view |
| HIV | * HIV access : metadata can view * HIV data entry : metadata can view * HIV admin : metadata can edit and view |
| NHMIS | * NHMIS2019 – Data Entry : metadata can view * NHMIS access : metadata can view * NHMIS admin : metadata can edit and view |
|

## Data Sets

* Data Sets will not be created with public access of any kind
* Data sets need to be assigned to user groups so users can view and edit them and. Use this table to make the assignment for each health program:

|  |  |
| --- | --- |
| Program | User Group Sharing Settings |
| Malaria | * Malaria access : metadata can view, data can view * Malaria data entry: metadata can view, data can capture and view * Malaria admin : metadata can edit and view, data can view |
| TB | * TB access : metadata can view, data can view * TB data entry: metadata can view, data can capture and view * TB admin : metadata can edit and view, data can view |
| Immunisation | * Immunisation access : metadata can view, data can view * Immunisation Data Entry : metadata can view, data can capture and view * Immunisation admin : metadata can edit and view, data can view |
| HIV | * HIV access : metadata can view, data can view * HIV data entry : metadata can view, data can capture and view * HIV admin : metadata can edit and view, data can view |
| NHMIS | * NHMIS2019 – Data Entry : metadata can view, data can capture and view * NHMIS access : metadata can view, data can view * NHMIS admin : metadata can edit and view, data can view |
|

* Data sets should contain a proper name, short name and code
* Data sets should be assigned only to organization units that are expected to submit the data set
* Users with the role **Administrator – Data Sets** or **System Administrator** will be able to create data data sets

## Annex A – Category **Best Practices**

For complex categories, it may be useful to describe what the category contains.

Use generic descriptors for the categories which are specific enough to identify what the category is, but which are not overly specific to any one use.

Here are some poor examples of categories.

* Age + Gender : This is really a combination of two categories, not one.
* Age : This is too generic, and does not provide enough information to users as to what is contained in the category.
* Age (HIV Testing): The use of this name is too specific. Likely this category could be reused for other purposes, but it would appear from the name, that its sole purpose is for as specific indicator.
* Age (<1 ) : This would appear to contain a single option, and is not inclusive of all possible ages.
* Age (<1,>1,<5,5+): This category would appear to contain overlapping options, such as <1 and <5. Category options within a category should consist of mutually exclusive groups. There can be exceptions to this rule, but in general, its a good idea to ensure that all of the category options within a category are mutually exclusive.

Here is a good example of a category.

* Age (<1/1-5/<5/5+) : All possible ages are represented. From the name of the category, it is clear what different category options are available.

## Annex B – Category Combination Best Practices

Modifying underlying categories that are within already existing category option combinations can have unintended consequences. Let;s say you create a category combination called "Age + Gender". You then proceed to modify the "Age" category. In order for the new category option combinations to be created you would need to perform a maintenance task to "Update category option combinations". This maintenance operation will remove any obsolete category option combinations as well as create any new ones which are required. However, this may have a serious effect on existing data, validation rules and indicators (since they may be linked to specific category option combinations).

As a result, ALWAYS be sure to check the category option combos of the category combo before you start to use it.

Using an API call like

https://dhis2nigeria.org.ng/dhis/api/categoryCombos/V9ocoqbjimn.csv?fields=categoryOptionCombos[id,name]

where "V9ocoqbjimn" is the UID of the category combination, you can get a CSV file of the category combinations with all of the category option combinations which looks like this.

|  |  |
| --- | --- |
| name | id |
| Female, 20 - 24yrs | u8xsz4aqkXq |
| Female, 25-49 years | kU83OOnzqWh |
| Female, 15-19yrs | PDMDO7RUo5t |
| Female, 10 - 14yrs | lo7h8zjx7bG |
| Male, 15-19yrs | sykrphiqQBv |
| Male, 50+ years | EzaEXNrymiJ |
| Female, 1- 4 yrs | XruJp4NYvUE |
| Male, 25-49 years | ScXbC3T5ZZz |
| Male, 20 - 24yrs | FjdK9yv8DxP |
| Female, 5-9yrs | W6YkAOHM29W |
| Female, <1y | PO2hBhliK38 |
| Female, 50+ years | eVVPMNgEHgG |
| Male, 5-9yrs | WpLN52wvi2N |
| Male, <1y | KLeULrvPcSD |
| Male, 10 - 14yrs | YBB0zk5yhLK |
| Male, 1- 4 yrs | JNgOdj7O35z |

This category combination consists of two categories:

* Gender, which has 2 category options
* Age (ART), which has 8 category options

This means we should end up with 16 (2x8) category option combinations if we have created everything correctly.

Before creating any data elements, indicators or validation rules with the new category combination, be sure that you confirm that all of the category option combinations that have been generated are correct before proceeding, as it can become difficult to rectify problems with these after they have been implemented and used.