

# Developer properties

## Overview

Developer properties is a best practice method for providing the developer with more information about the current state of the current working environment. This best practice does not limit or enforce the values which are displayed by the developer.

## Implementation

Using a single text label on any desired layout, you specify the following calculation within a single conditional formatting directive. Accessed via **Format > Conditional...** while the text label is selected in Layout mode.

```
// Current view properties

Let ( [
    $$ CURRENT.TABLE = Get ( LayoutTableName );
    $$ CURRENT.LAYOUT = Get ( LayoutName );
    $$ CURRENT.FIELD = Get ( ActiveFieldName );
    $$ CURRENT.FIELDTABLE = Get ( ActiveFieldTableName )
];
    If ( Developer ; False ; True )
)
```



### Developer function dependency

Using the above calculation, you will need to add a [Developer function](#) to your file's custom functions.

The primary goal is to only show the specified information to authorized developers. This is accomplished by setting either or both the text and fill colors to those which will match the background of the layout or elements the Developer properties is above. The properties will be hidden to non developers in normal solution use.



### Note the spaces

The example calculation above uses a space character between the \$\$ (double dollar signs) and the VARIABLE.NAME. The effect is sorting the \$\$ CURRENT.SOMETHING variables to the top of the Data Viewer list when viewed in name sorted order.

The display of the properties, when in modes other than Layout is accomplished using merge variables (only available in FileMaker 11 and higher). The following text would be used within the text label itself.

```
Layout:      <<$$ CURRENT.LAYOUT>>
Table:       <<$$ CURRENT.TABLE>>
Field:       <<$$ CURRENT.FIELD>>
FieldTable: <<$$ CURRENT.FIELDTABLE>>
```

## Custom function suggestion

It is suggested you actually create a custom function for the display of these Developer properties. Doing so ensures that you can copy and paste the same text label object to any layout and be able to universally make changes to the resulting global variables. The one shortcoming is that the text within each label on each layout may need to be modified to show or remove any properties.



### Refresh will be needed

Because of how FileMaker renders layout contents, the global merge variables may not always be correct. You may need to create the habit of quickly going into Layout mode then back to Browse to view updates to the global variables - or creating a developer specific script which uses a simple **Refresh window** script step.

## Reasoning

- **Information accessible** when the Status bar is hidden and locked.
- **Useful information in Data Viewer** is presented while developing.