**Transform** and calculate data. In this module, we'll look deeper in different FreeMarker built-ins that can help us manipulate how data are printed into the PDF. This will also be the last module that revolves around the [? wizarding ?] mode.

By the end of this module, you should be able to perform arithmetic calculations using FreeMarker, similar to what we can do with the NetSuite Formula Builder, use FreeMarker built-ins to introduce number formatting as well as string manipulations like converting text to all caps, and finally, be able to efficiently test and debug your FreeMarker interpolations.

A quick overview of the FreeMarker interpolation expression. As a recap, the typical FreeMarker notation is composed of the datasource ID followed by a dot, then the field ID. The datasource can be a record, company information, or subsidiary.

Inside the boundaries of FreeMarker expressions, we can perform calculations like adding two or more values together, and also chain built-ins like the lowercase or uppercase built-ins to change how the text looks. There are also some special fields like the .now, which gives you the date and time when the PDF was printed, alongside with the .version and the .locale. FreeMarker offers a full online documentations of all available built-ins. We highly suggest going to their website to get yourself up to speed with everything FreeMarker.

Let's look at the right syntax informing our interpolations, and while we're at it, identify what are the common pitfalls in using built-ins. The first line shows us the proper way of doing calculations inside the expression. Within the dollar curly braces pair, we place the input fields, and in this case, record.amount1 followed by an addition sign, then the field accessor for record.amount2.

What's wrong on the second one is that we separated the two operants into two, and we used the division sign outside the expression. And if you're curious on what will be the outcome of the second line, it will be whatever the amount of amount1, a slash, whatever the amount of amount2 literally.

The word business trip is seen as Business Trip, capital B and capital T. When printed using the plain expression, it looks the same. Meanwhile in here, we see the uppercase built-in chained after the record.memo expression. And thus, we see the string transformed into all caps. Note to chain built-ins, we need to use the question mark before the function. In using built-ins, it is important to make sure that we are using them in the right letter casing. In here, substring should be used as question mark-- substring all in small caps, not just random casing for any letters.

Now, we are going to take a look at testing and debugging FreeMarker snippets using a validation tool. FreeMarker has this online tool found at try.freemarker.org. Note that this link might change in the future, so use your favorite search engine and look for the latest link. This page contains three major parts. First is the template section, where you write your own FreeMarker expressions, so all of your dollar curly braces codes come in here. Data model field, which stores your mock data set, and we'll discuss further later about this. And finally, result section, which displays the output from the template evaluation.

Let's take a closer look at the Data Model field. The sample object notation can be accessed by using the parent data record, then fetching the value via the property ID, which is memo. This is very similar to what we have been doing in FreeMarker inside of NetSuite. In writing FreeMarker templates, there are few unsupported characters. And these are single quotation, double quotation, greater-than sign, less-than sign, ampersand, or and sign.

And if, by mistake, you place one of these strings inside of an expression, it would be rendered as something else, i.e. double quote would be rendered as ampersand, Q-U-O-T, ended with a semicolon. In this walkthrough, I'll show how we can use FreeMarker built-ins and render the Memo field in all caps. Moreover, we'll be looking on how to perform calculations using FreeMarker built-ins in arithmetic operations.

And we can see that the Purpose field is currently empty. The Purpose field's ID is memo. So let's remember that. And for now, let's edit first expense report and populate that field. For example, let's put in a value of Went to a Business Conference. And click Save. Now the Purpose field is now populated.

Next up, go back to the template for an expense report. Open SuiteDreams Expense Report template. Scroll down. And we can see that the Purpose field is already printed into the PDF. So let's preview it.

And we see a placeholder of Lorem ipsum, which is good. And close the template for now. Go back to the expense report. And click Print. As of the moment, we see Went to a Business Conference under the Purpose column printed as is. So the capitalization is the same as what we see in the form.

Up next, what we're going to do is delete first this Interpolation field. Click it and delete. And type in record, not memo, the same interpolation this time followed by Capitalize built-in. This built-in capitalizes each letter of every word inside of that interpolation field.

Now we click Preview. And we see that every first letter of each word inside the Purpose field is automatically capitalized. Now let's save the template and go back to the expense report. Click Print again. There you go. The purpose of Went to a Business Conference is now capitalized for each first letter of the word.

Now let's quickly jump in on FreeMarker's string built-in documentation. In here, we see a couple of string built-ins ready for use. And one of them is capitalized, the one that we have used in our template. So let's view its documentation.

It says here the string with all words capitalized. And we see the example below, how it transforms the string value. Scroll up again. And view a couple more. And we have here the lowercase as well and the uppercase, so let's view that. The description says the uppercase version of the string. So it converts all letters into all caps.

Now let's use that inside of our template. Open up, again, the template. And go to the Purpose section. We cannot change the interpolation of this. We just have to highlight it and delete.

Type in the new interpolation code, record.memo, the same one. But this time, we use uppercase, upper\_case. So [? then close ?] the interpolation. Click Save. And click the Print button again inside of the expense report. And now the Went to a Business Conference purpose is all casted in all caps.

Now, next up, we'll create a new line on the Summary section of the PDF and compute for the percent reimbursable using FreeMarker interpolation. So let's go to Customization, Forms, and Advanced PDF Templates. And let's open, again, the SuiteDreams Expense Report.

Scroll down to the Summary section. And add a new line under the Percent Reimbursable field, which we have computed using NetSuite custom field formulas. And right-click on the Percent Reimbursable row, Insert Row After. And for the meantime, let's type in Percent Reimbursable -FM, which stands for FreeMarker.

Click Preview to see how it looks for the meantime. There you go, looking good. Go back again to the template. And let's add, now, our interpolation code. Type in record.reimbursable divided by record.total. Then close the interpolation. Then click Preview and see what happens.

Scroll down. And we see that Percent Reimbursable -FM is set to 1. Don't worry, that's just a placeholder. For now, let's save the template. Then go back again to the expense report. Then print.

And we see that the Percent Reimbursable -FM has a value of 0.675858. Let's see how we can represent that in percent. We're going back to the template. Scroll down. And let's go back to the interpolation code here. And we need to enclose this whole section inside of a parentheses.

So let's type in 1, opening parenthesis in front and one at the end, basically enclosing it into one single expression. Now, I follow it up by ?string.percent, which converts numerical value into string percentage value. Scroll up. Click Preview again. And now we see 100% instead of 1.

Next up, let's go back to the template and save it. Then go to the expense report again. And click Print button. And scroll down. And we can see that 68% is displayed as a value for Percent Reimbursable -FM. You may notice that we're losing some kind of precision when computing for values using FreeMarker interpolations. So as a conclusion, it's more wise to put in your calculations inside of NetSuite custom field formulas whenever possible.

Now it's your turn. Please complete your hands-on exercises for this module. Refer to your Student Guide to complete the required exercises. This concludes the Transform and Calculate Data module.