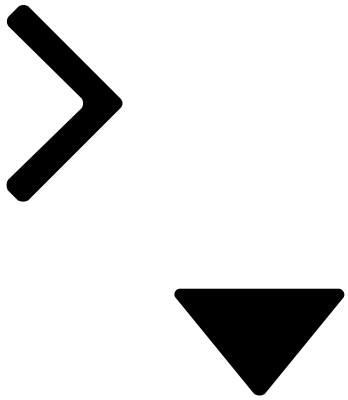
1. Formulas & Validations



2. Use Formula Fields

Use Formula Fields

Learning Objectives

After completing this unit, you'll be able to:

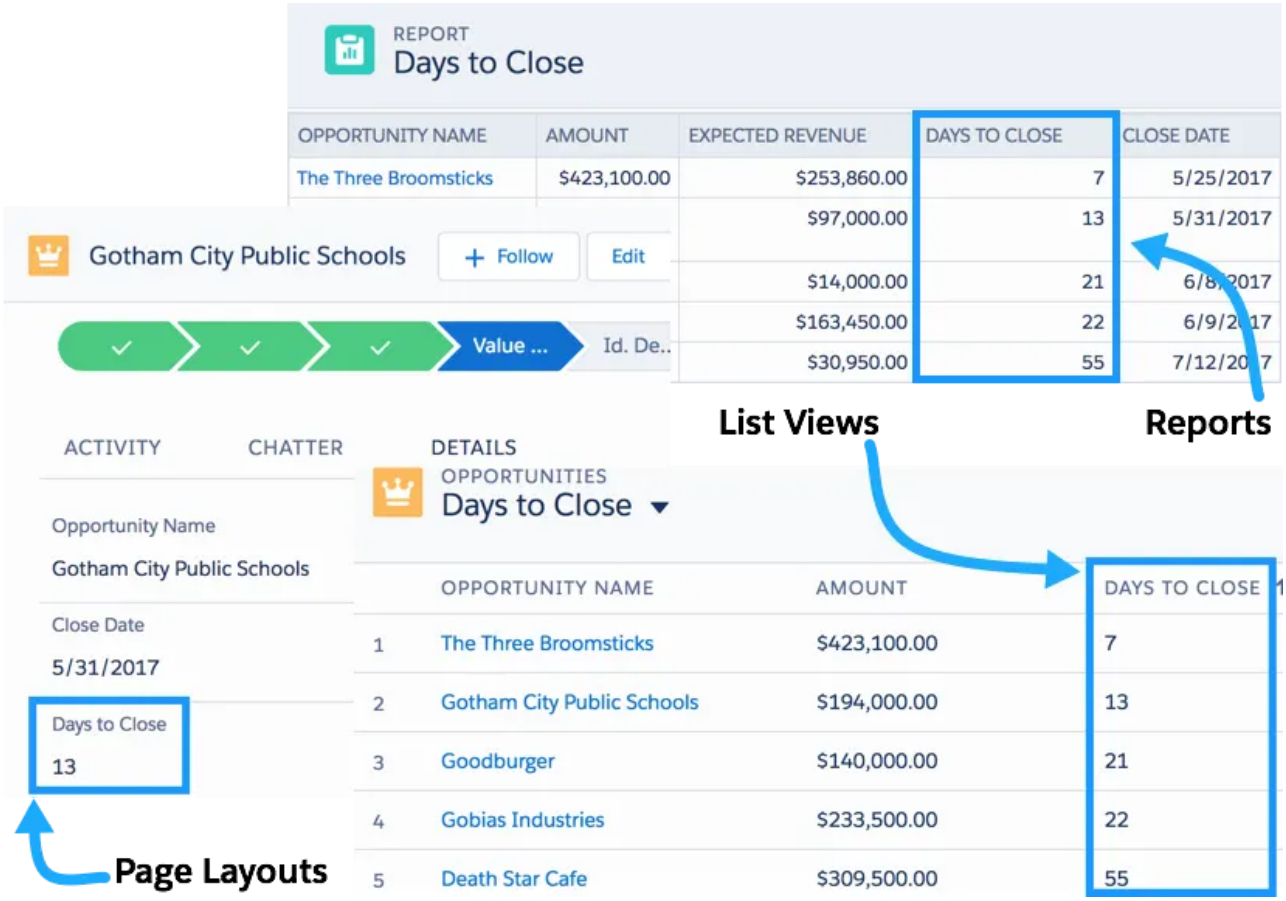
- Create a custom formula field and use the formula editor.
- Explain why formula fields are useful.
- Outline at least one use case for formula fields.
- Create simple formulas.

Introduction to Formula Fields

You've got a lot of data in your organization. Your users need to access and understand this data at-a-glance without doing a bunch of calculations in their heads. Enter formula fields, the powerful tool that gives you control of how your data is displayed.

Let's say you wanted to take two numeric fields on a record and divide them to create a percentage. Or perhaps you want to turn a field into a clickable hyperlink for easy access to important information from a record's page layout. Maybe you want to take two dates and calculate the number of days between them. All these things and more are possible using formula fields.

Let's look at a specific example. What if you wanted to calculate how many days are left until an opportunity's close date. You can create a simple formula field that automatically calculates that value. By adding the value to the Opportunity page layout, your users can quickly access this key information. You can also add this field to reports and list views for instant access.



When you're first learning formulas, it's best to start with simple calculations and build up to more complex scenarios. But, as you'll see, even simple formulas provide valuable information.

In this unit, we'll take you through the basics of using the formula editor and introduce you to formula syntax through several basic examples. We'll also touch on troubleshooting problems with your formula fields. Now let's have some fun!

Finding the Formula Editor

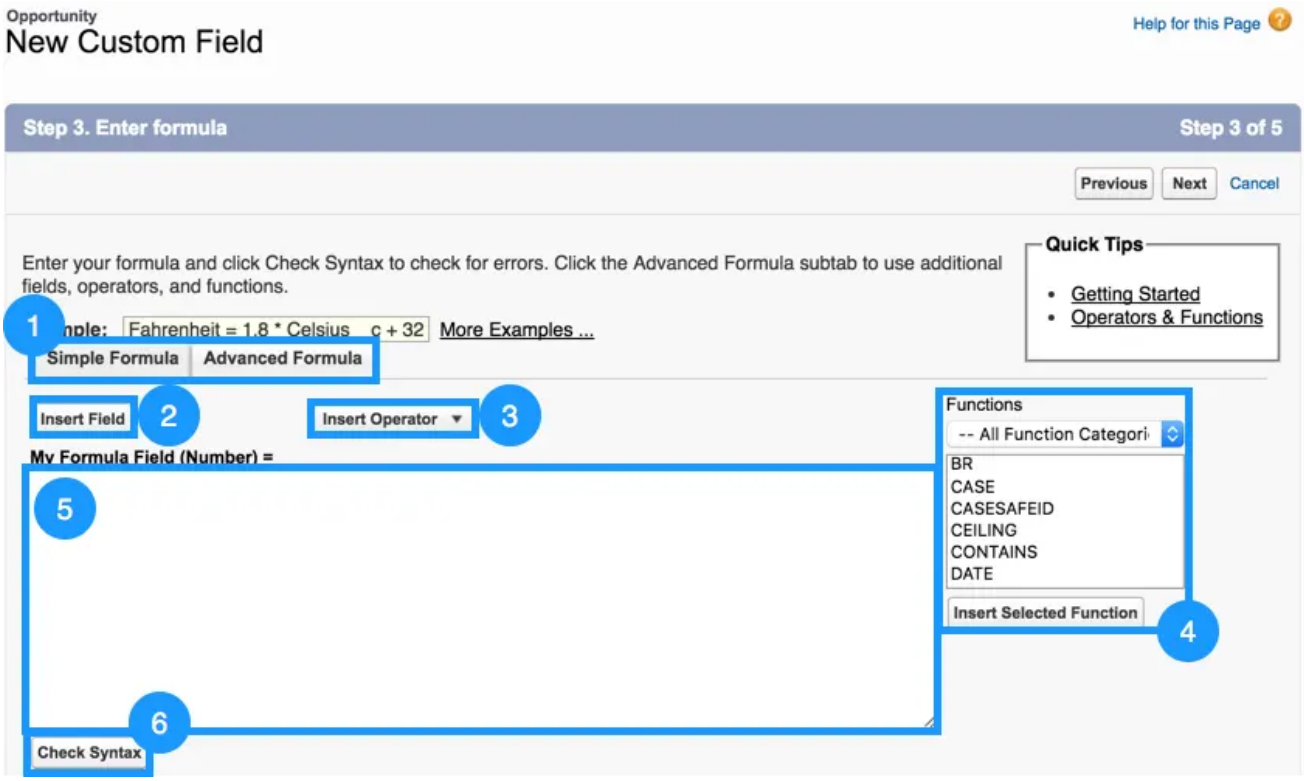
Before we dive into writing formulas, let's locate the formula editor and get to know its features.

You can create custom formula fields on any standard or custom object. To start, we'll create a formula on the Opportunity object. Follow these steps to navigate to the formula editor:

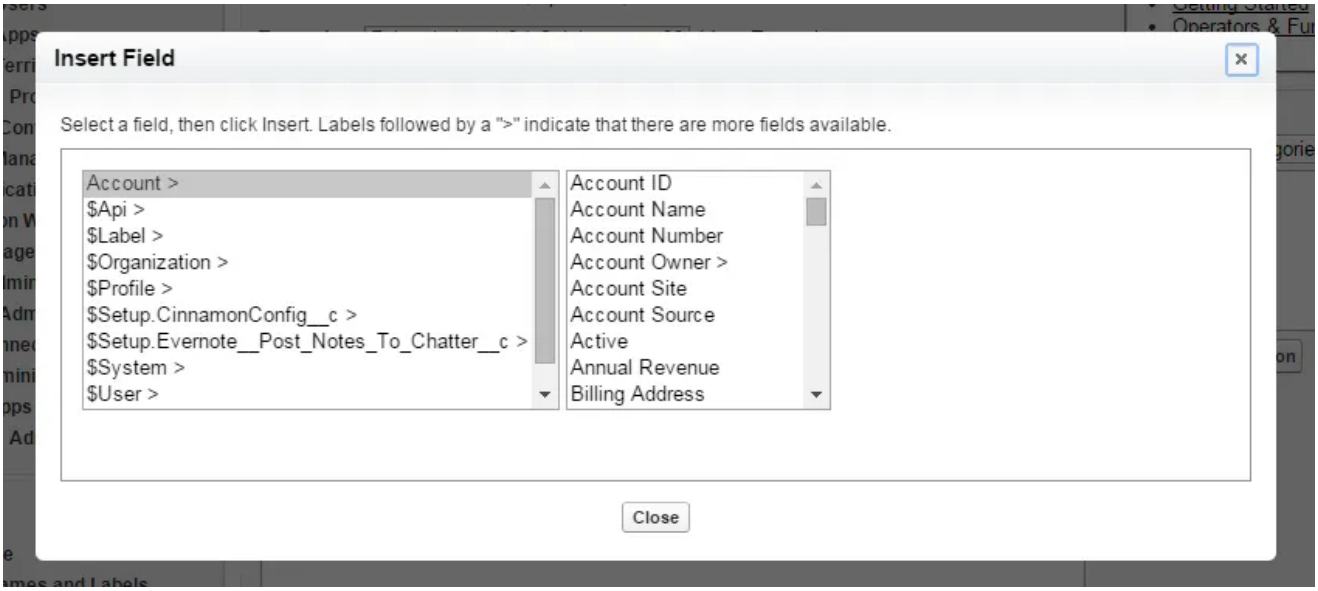
1. From Setup, open the Object Manager and click **Opportunity**.
2. In the left sidebar, click **Fields & Relationships**.
3. Click **New**.
4. Select **Formula** and click **Next**.
5. In **Field Label**, type `My Formula Field` . Notice that **Field Name** populates automatically.
6. Select the type of data you expect your formula to return. For example, if you want to write a formula that calculates the commission a salesperson receives on a sale, you select Currency. For now, pick **Text**.
7. Click **Next**. You've arrived at the formula editor! Time for our tour.

Using the Formula Editor

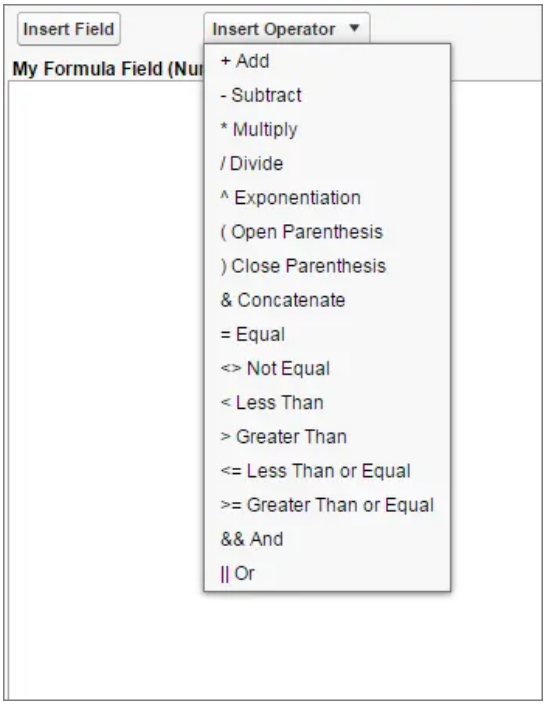
This image highlights the most important parts of the formula editor:



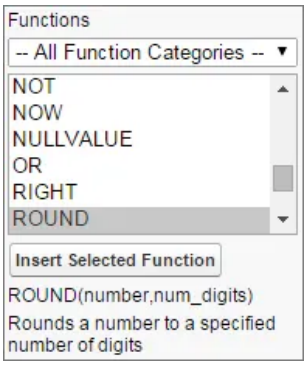
1. The formula editor comes in two flavors: Simple and Advanced. It's tempting to use the Simple editor, but we always recommend using the Advanced editor. Advanced doesn't mean more complicated. It means more tools for you to create powerful formulas.
2. The **Insert Field** button opens a menu that allows you to select fields to use in your formula. Inserting from this menu automatically generates the correct syntax for accessing fields.



3. The **Insert Operator** button opens a drop-down list of the available mathematical and logical operators.



4. The Functions menu is where you view and insert formula functions. Functions are more complicated operations that are pre-implemented by Salesforce. Some functions can be used as-is (for example, the `TODAY()` function returns the current date), while others require extra pieces of information, called parameters. The `LEN(text)` function, for instance, finds the length of the text you input as a parameter. The formula `LEN("Hello")` returns a value of 5.



5. The text area is where you enter your formula. When writing formulas, keep in mind that:

    - Whitespace doesn't matter. You can insert as many spaces and line breaks as you want without affecting the formula's execution.
    - Formulas are case sensitive. Pay attention to capitalization of field and object names.
    - When working with numbers, the standard order of operations applies.

6. Once you've written a formula, you can use the **Check Syntax** button to ensure that everything is in working order before saving. If your formula has issues, the syntax checker alerts you to specific problems.

We don't need to continue creating this formula field, so click **Cancel**. Now that you know your way around, let's put the editor to use with some simple examples.

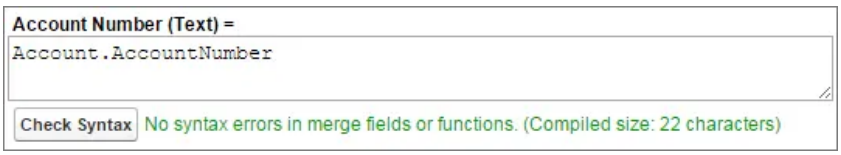
The Formula Editor in Action

Example 1: Displaying an Account Field on the Contact Detail Page

Record detail pages contain a ton of information, but sometimes it's not enough. Sometimes you need more! For your first formula, let's do something simple. Let's take a single field from an Account and show it on a Contact using what's called a *cross-object formula*. Let's take a look.

First create a Contact. If you've never created a Contact before, go to the Contacts tab and click **New**. Enter any value for Last Name and make sure that you fill in a value for the Account Name field by clicking the lookup icon. Next we'll create a formula to display the account number on the Contact page:

1. From Setup, open the Object Manager and click **Contact**.
2. In the left sidebar click **Fields & Relationships**.
3. Click **New**.
4. For the field type, select **Formula** and click **Next**.
5. Call your field `Account Number` and select **Text** for the formula return type. Click **Next**.
6. Click **Insert Field** on the Advanced Formula Editor. Select **Contact** | **Account** | **Account Number** and then click **Insert**.



Congratulations, you've written your first formula!

Let's see this formula in action. The next page lets you set field-level security. For now, click **Next** so we can add our formula field to the page layout. For the time being, make sure that all the checkboxes are selected. Click **Next** and then click **Save**.

Now it's time to see what you've done. Open the detail page for the Contact object you just created and find your new Account Number formula field. Cool!

Example 2: Displaying the Number of Days Until an Opportunity Closes on a Report

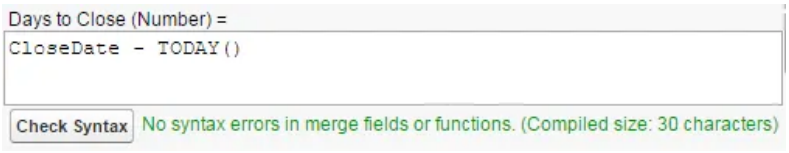
You can also use formula fields in reports to increase the visibility of important information. Say, for example, you wanted a report column that displays the number of days until an opportunity is closed. First, create an Opportunity to test our formula.

If you've never created an Opportunity before, go to the Opportunities tab and click **New**. Fill in any value for the Name, select any Stage, and set a close date that's at least three days in the future. Then create a custom formula field called `Days to Close` on the Opportunities object with a Number return type:

1. From Setup, open the Object Manager and click **Opportunity**.
2. In the left sidebar click **Fields & Relationships**.
3. Click **New**.
4. Select the **Formula** and then click **Next**.
5. In the **Field Label** text area, type `Days to Close` .
6. Select the **Number** radio button.
7. Click **Next** to open the formula editor.

We need to find the difference between the opportunity close date and today's date. Let's start by inserting the Close Date field in the editor. Since we're finding a difference, use subtraction. Select **- Subtract** from the **Insert Operator** menu.

But how do we tell our formula that we need today's date? Luckily, there's a function called `TODAY()` that updates to match the current date. Find it in the Functions menu on the right side of the editor and click **Insert Selected Function**.



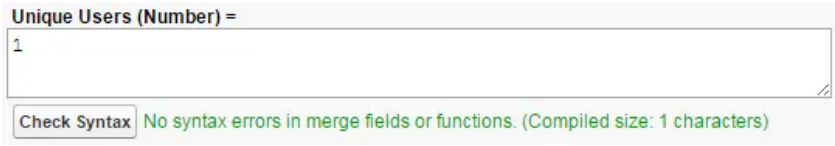
After you click through the save screens, it's time to put your new formula field in a report. From the Reports tab, click **New Report**. Then select **Opportunities** and click **Create**. Your opportunity appears in the Preview panel. Search for `Days to Close` in the Fields menu on the left side of the page. This field is the formula field you just created. Drag it to the last column in your report. The column populates automatically with the calculated value.

We won't return to this report, so you can either save it or move directly to the next example.

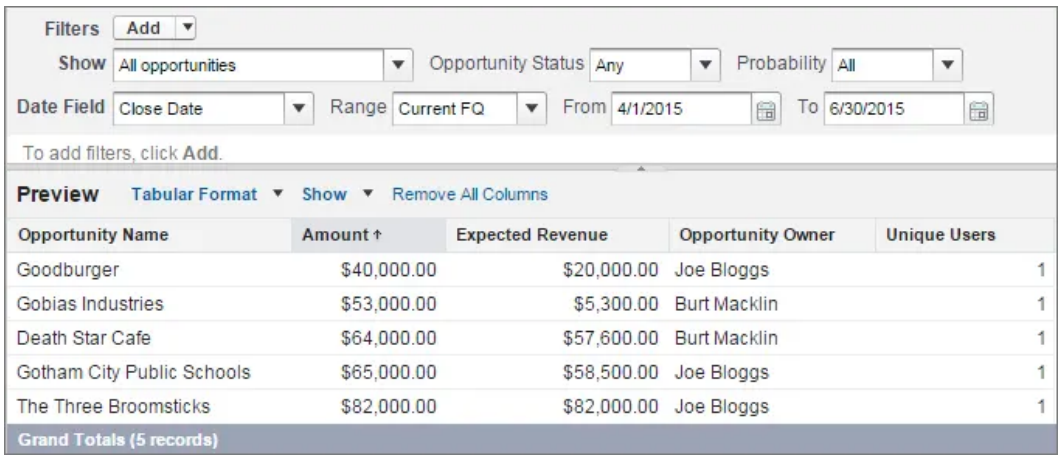
Example 3: Finding Distinct Objects Using the Power of One

Organizations often want to count the number of unique objects in a report with hundreds of records. Say, for example, you have a hundred opportunities listed in a report, but only a handful of users own all these opportunities. How do you find the number of distinct users? This task sounds difficult, but it's one of the easiest formulas you can write. It's called the Power of One.

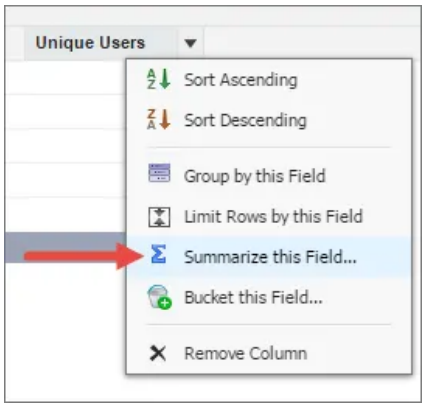
To write this formula, create a custom formula field on the User object. Name it `Unique Users` , give it a Number return type, and select 0 from the Decimal Places drop-down list. Click **Next** to open the formula editor. For this formula, you don't need to insert any fields, operators, or functions. Instead, enter the number 1.



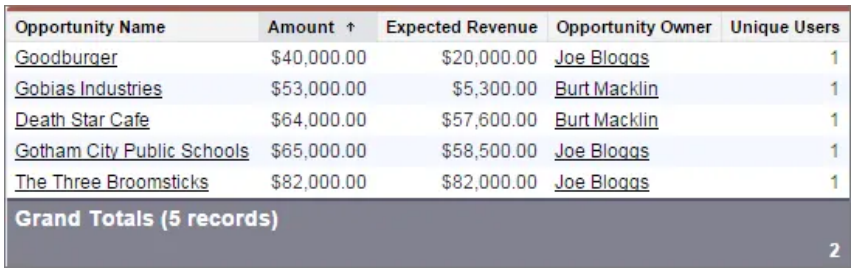
Yes, that's really it! Save your formula as usual, and then click the Reports tab to see it in action. Create an Opportunities report and drag your Unique Users field onto the report from the left panel. You should see something similar to the following:



Here, we have five opportunities between two distinct users. Where does our formula come in? On the Unique Users column, click the drop-down menu and select **Summarize this Field**.



In the popup dialog, select **Sum** and click **Apply**. The number of distinct users appears at the bottom of the column. As you can see, even simple formulas provide powerful insights into your organization.



Note

You can use the Power of One on any object. For example, if you had a report with 10 accounts, each with three opportunities, your Opportunities report returns 30 records. Adding the Power of One formula field to Account allows you to see the number of distinct accounts represented in the records. Some pros say they add a Power of One field to every object in their org!

Debugging Formulas

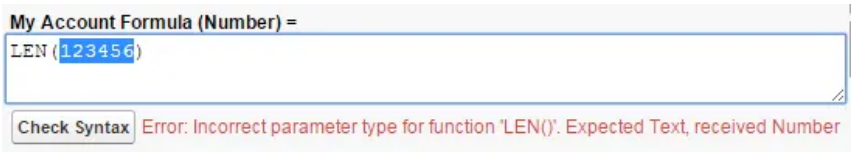
Syntax errors are an inevitable part of working with formulas. The **Check Syntax** button in the editor is an important tool for debugging your formulas. The syntax checker tells you what error it encountered and where it's located in your formula. Here are some common syntax issues:

1. **Missing parentheses**: This error most often occurs when the number of opening parentheses doesn't match the number of closing parentheses. It can be particularly difficult to avoid this error if you're using several functions at once. Try breaking your function

into multiple lines so it's easier to tell which sets of parentheses belong together.

You'll also see this error if you forget a comma between two function parameters. This error is confusing because the actual problem doesn't match up with the syntax checker. If you're certain your parentheses are correct, double check that the commas in your function are correct as well.

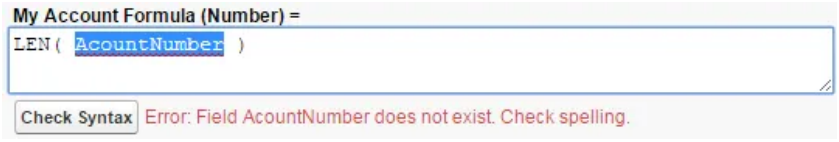
2. **Incorrect parameter type**: If you give a function a number parameter when it expects text (or any other combination of data types), this is the error you'll see. Always check the help text or the documentation so you know what kind of parameters a function accepts.



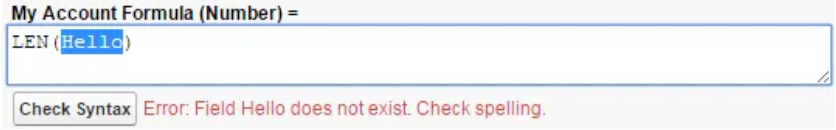
3. **Incorrect number of parameters for function**: If you input too many or too few parameters into a function, the syntax checker alerts you. Again, check the help text or documentation for guidelines on inputting parameters to specific functions.

4. **Formula result is incompatible with formula return type**: You'll see this error if you select one data type when creating the formula field but write a formula that returns a different data type. In the example below, you can see that My Account Formula expects to return a number (shown in parentheses next to the formula name), but the `TODAY()` function returns a date. The error tells you what the expected data type is, but you can always reference the documentation beforehand to avoid the error.

5. **Field does not exist**: This error indicates that you've included a field in your formula that your object doesn't support. In this case, check your spelling and capitalization. If you can't find any mistakes, try inserting the field from the **Insert Field** menu again to make sure you're referencing it correctly.



Another reason you see this error is if you forget to put quotation marks around a text literal or a hyperlink.



6. **Unknown function**: In this case, check that Salesforce supports the functions you're using. You'll also get this error for misspelled functions.

Further Examples

Let's look at a few more examples. You can create these formulas yourself or simply read through.

1. This formula creates a hyperlink to an external website using the `HYPERLINK()` function. Adding hyperlinks to page layouts helps your users access important information quickly from the detail pages.

   Account Website (Text) =

   ```
   HYPERLINK("http://www.VeryImportantWebsite.com", "Very Important Website")
   ```

   Check Syntax — No syntax errors in merge fields or functions. (Compiled size: 145 characters)

2. If you want to apply a discount to an opportunity amount, you can use the following formula. In this case, we're applying a 12% discount and then rounding the result to two decimal places using the `ROUND()` function.

   Discounted Amount (Number) =

   ```
   ROUND( Amount - (Amount * 0.12), 2 )
   ```

   Check Syntax — No syntax errors in merge fields or functions. (Compiled size: 66 characters)

3. This formula is a *checkbox* formula that determines whether a particular opportunity is a "big" opportunity. It checks whether the number of employees at the opportunity account's associated company is greater than 1,000 AND whether the opportunity amount is greater than $10,000. If both statements are true, the field appears as a checked box on the Opportunity page layout. Otherwise, it appears as a blank box.

   Big Opportunity? (Checkbox) =

   ```
   AND( Account.NumberOfEmployees > 1000,  Amount > 10000 )
   ```

   Check Syntax — No syntax errors in merge fields or functions. (Compiled size: 69 characters)

The formulas documentation contains numerous examples for many different use cases. While you're browsing these examples, keep in mind that many of them contain advanced concepts that weren't covered in this unit. Make sure you're comfortable with the information presented here before tackling these formulas.

Resources

- Formulas Help & Training
- Formulas Quick Reference
- Formula Help on the Developer Forums
- Formula Ninjas Dreamforce Presentation

Assessment Complete!

+500 points

Formulas & Validations
100%
Progress: 100%
Retake this Challenge
View more modules