

Use Case: Send a Custom Date to a Salesforce Date Field

Download the PDF of this article.

In this Article

Related Articles

Overview

When sending a custom date from FormAssembly to Salesforce, it is important to consider the type of Salesforce field receiving the value.

The commonly used **@YMDNOW()** formula returns an **ISO 8601 Date/Time string**. While this format can be used with a Salesforce Date/Time field, it can cause a data type validation error when the value is sent to a Salesforce Date field.

If you need to send a calculated or custom date to a Salesforce Date field, you can use a formula that returns the date in a format Salesforce can interpret as a Date.

Sending a Custom Date to a Salesforce Date Field

The following formula can be used to create a custom date and format the result as a date:

```
@CONCATENATE(@MONTH(@YMDNOW("+30 days")),"/",@DAY(@YMDNOW("+30 days")),"/",@YEAR(@YMDNOW("+30 days"))))
```

In this example, the **+30 days** calculates a date 30 days from the current date.

The formula then uses **@MONTH()**, **@DAY()**, and **@YEAR()** to return the date in an **MM/DD/YYYY** format.

Why Not Use @YMDNOW() Directly

A formula such as:

```
@YMDNOW("+30 day")
```

returns an **ISO 8601 Date/Time string**, rather than a Date value.

For example, the result may look similar to:

```
2026-10-30T00:00:00.000Z
```

This value contains both a date and a time component. It is appropriate for a Salesforce **Date/Time field**, but a Salesforce **Date field** expects a date value without the time component.

Because the data types do not match, sending an ISO 8601 Date/Time string to a Salesforce Date field can result in a Salesforce **data type validation error**.

By separating the month, day, and year and concatenating them into a date format, the value can be sent to a Salesforce Date field without including the Date/Time component.

Adjusting the Number of Days

The number of days can be changed depending on the desired outcome.

For example, to send a date **30 days before** the current date:

```
@CONCATENATE(@MONTH(@YMDNOW("-30 days")),"/",@DAY(@YMDNOW("-30 days")),"/",@YEAR(@YMDNOW("-30 days")))
```

To send a date **2 months after** the current date:

```
@CONCATENATE(@MONTH(@YMDNOW("+2 months")),"/",@DAY(@YMDNOW("+2 months")),"/",@YEAR(@YMDNOW("+2 months")))
```

The same approach can be used with other day, month, or year offsets as needed.

Sending a Calculated Date Based on a Respondent-Entered Date

If the custom date is based on a date entered by the respondent rather than today's date, the date can also be calculated using the respondent's date field.

For example:

```
@COMPUTE(@DATEVALUE(%%tfa_XXXX%%)-30)
```

In this example, `%%tfa_XXXX%%` represents the FormAssembly date field containing the respondent's date, and `"-30"` represents the number of days before the respondent's entered date.

The FormAssembly field **should** have date validation enabled, and the resulting value **should** be mapped to a Salesforce Date field.

This formula returns the calculated date as a serial number representing the date. Salesforce can interpret the serial date value and convert it to a date when it is sent to a Salesforce Date field.
