

ActionBot Question Type Overview

Brief Overview

A WalkMe ActionBot Conversation can be made up of messages, questions, conditions, and actions.

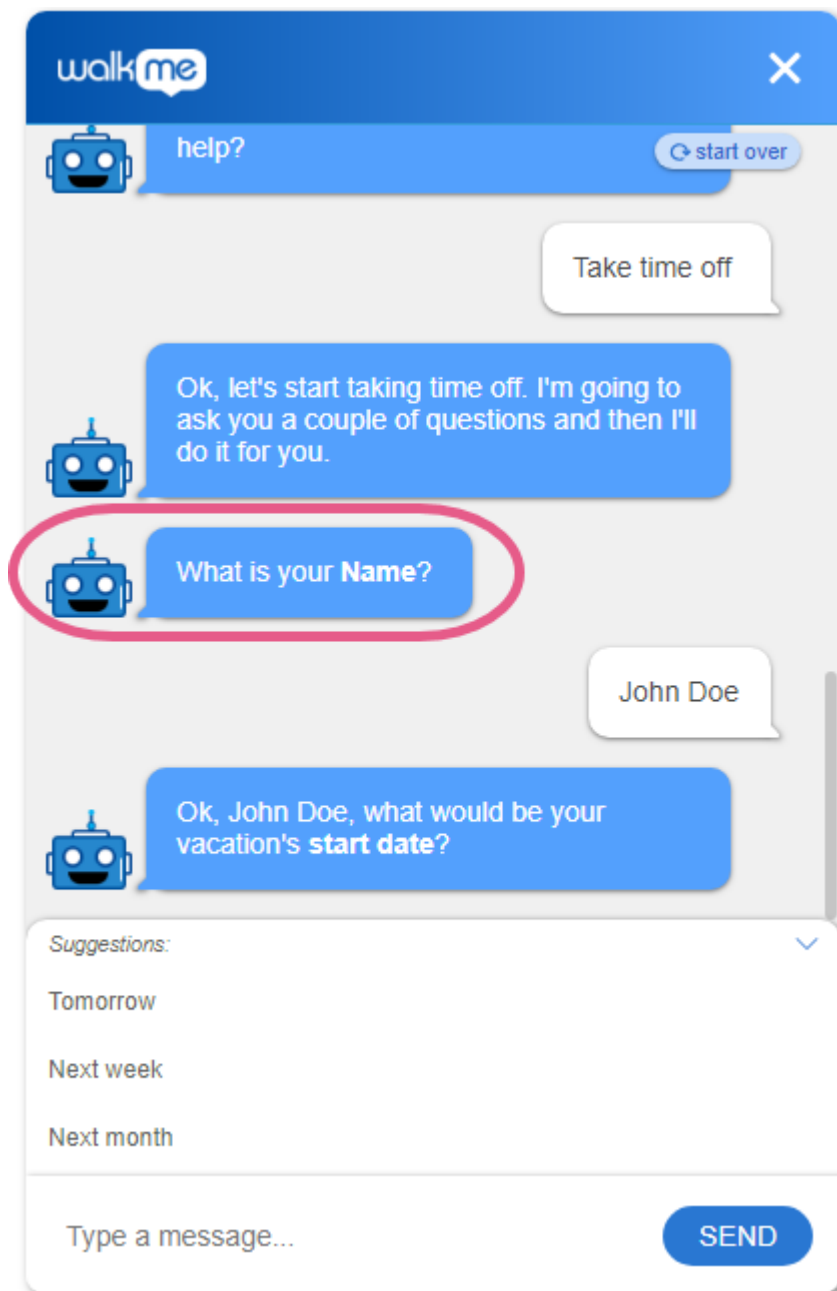
When you use a **question** in a Conversation, your Bot will wait to receive an answer from the end-user before moving to the next step in the Conversation. This is generally achieved using natural language processing (“NLP”).

How it Works

There are **eight types** of questions you can use as part of your Bot’s Conversation. These are as follows:

Text Type Question

- No NLP is used;
- Your Bot will consider the user’s whole input to be the exact answer and enter this data into a log;
- **Here is an example of a text type question:**

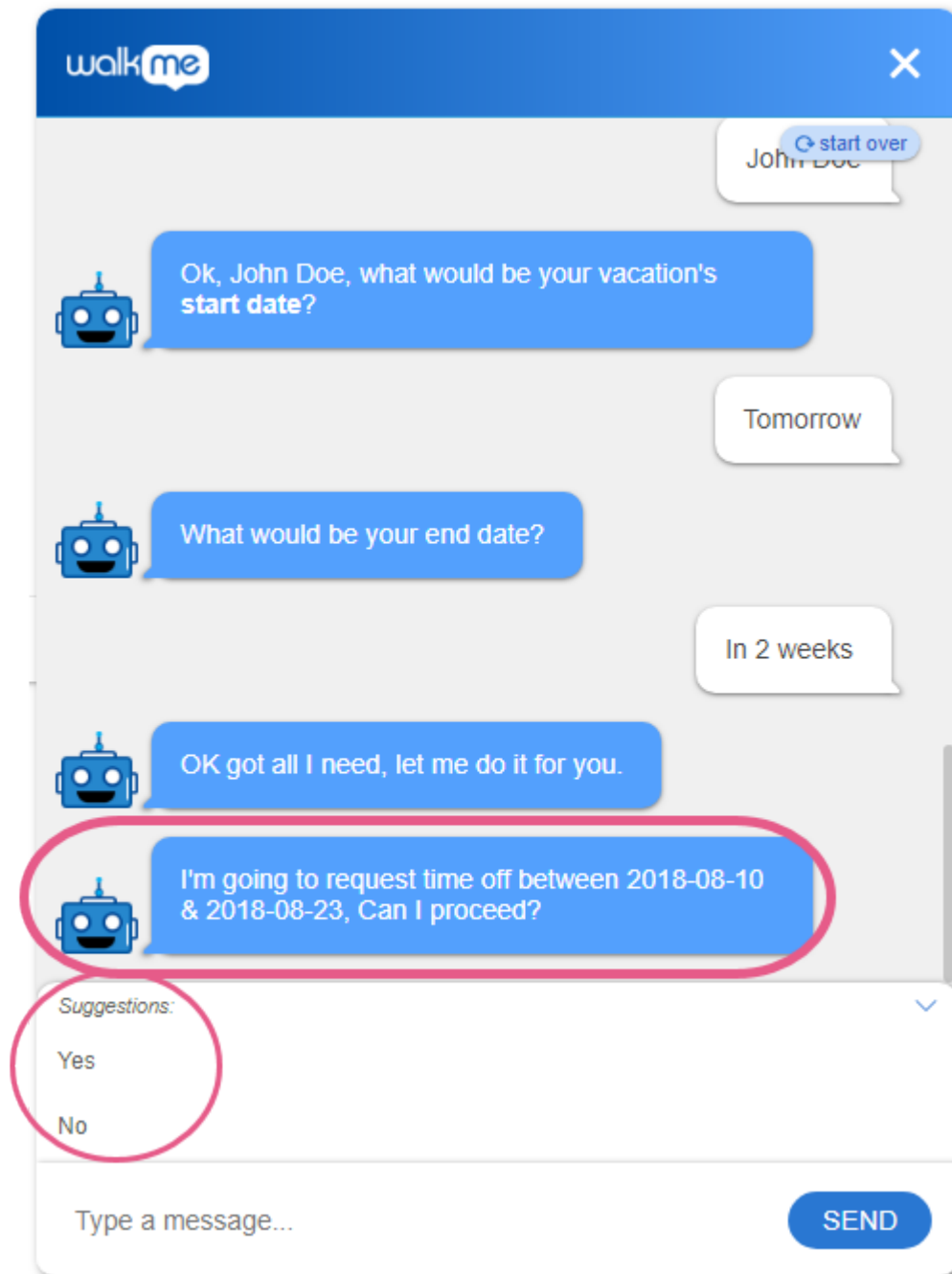


- The question: "What is your Name?" is the text type question in this Conversation;
- The answer extracted in this case is "John Doe."
 - This answer is then used by the Bot in the followup question.

Boolean Type Question

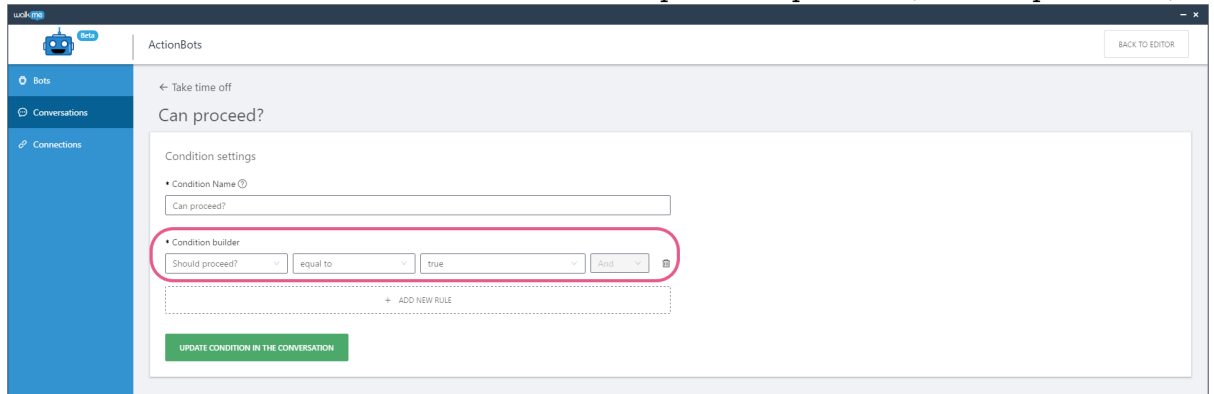
- Uses NLP;
- Your Bot will try to extract an affirmative or negative intent from the end-user and then assign a **true** or **false** value as the extracted answer;
- Example of an affirmative intent is "Sure;" the extracted value would be "true";

- Example of a “negative” intent is “Nope;” the extracted value would be “false”;
- Because there are only two possible extracted values, boolean type questions are often used in conditions, since it is easy to branch conversations by the value extracted;
- **Here is an example of a boolean type question:**

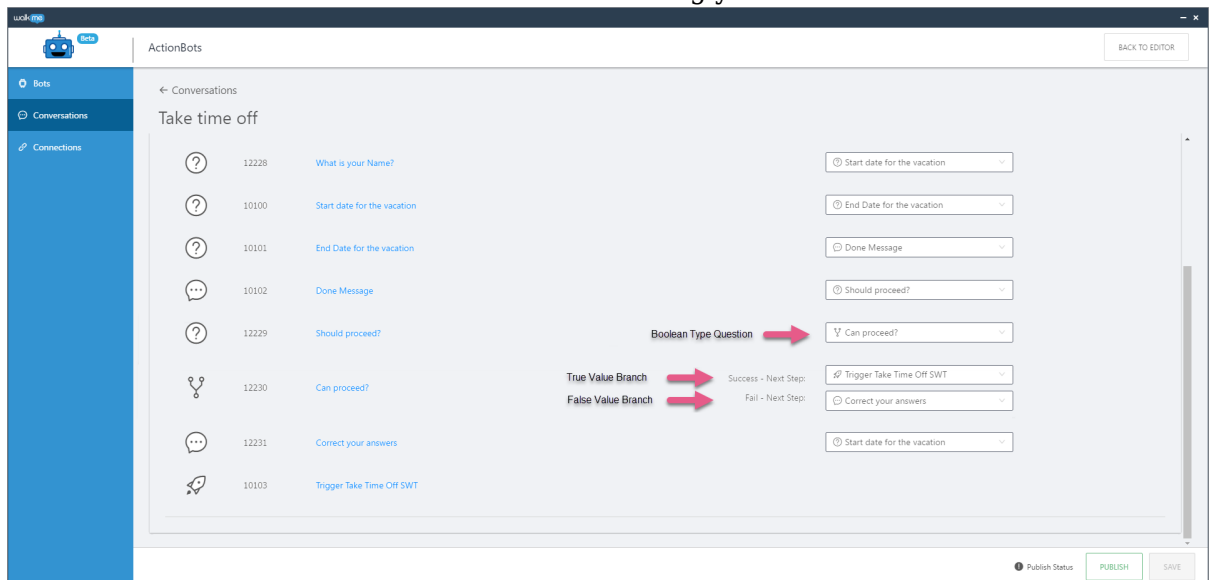


- The final question has only two possible values: “true,” if the Bot can move to the following step that’s associated with success, and “false” if the Bot made an error and should move to the following question associated with failure (to request the information again);

- In this example Conversation, we've named the boolean question "Should proceed?" and set a condition called "Can proceed" as the following step in the flow;
- we've then used the *Condition builder* in the Conversations tab to assign value to the condition based on the value extracted from the previous question ("Should proceed?"):

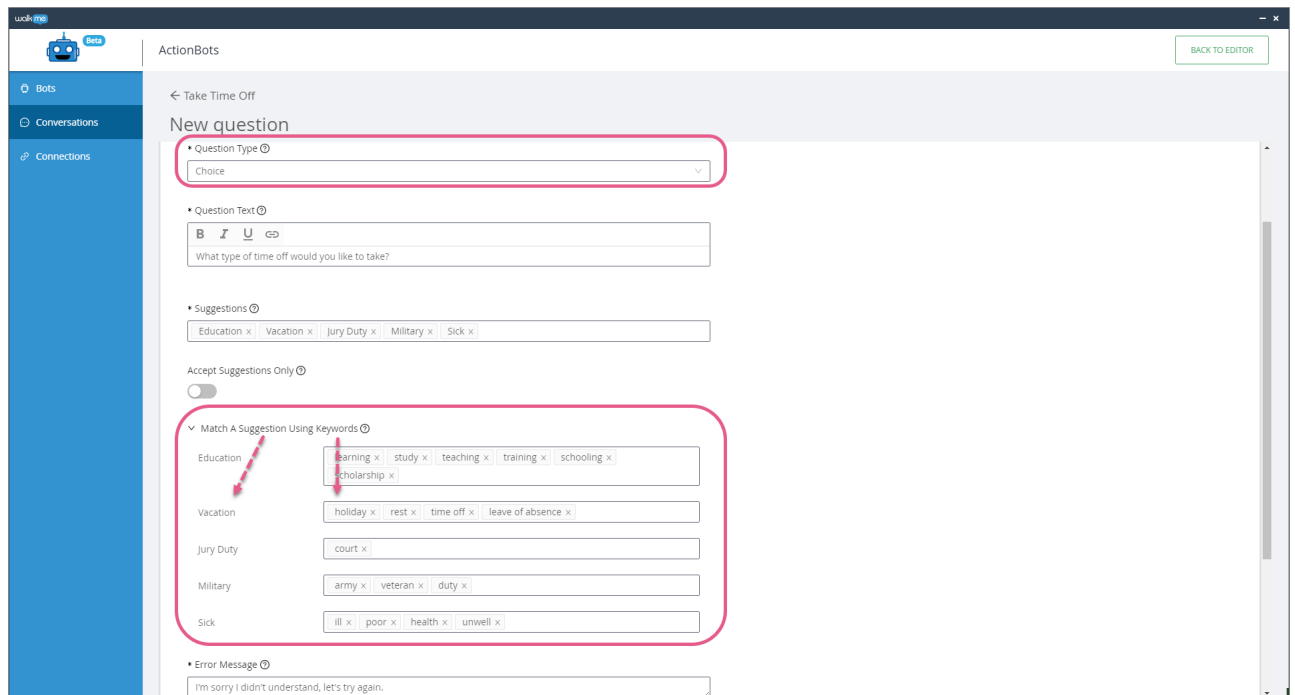


- Once the Bot has extracted a value from the end-user's answer, this value will be used in this condition to branch the Conversation accordingly:



Choice Type Question

- Doesn't use NLP. Uses keyword search instead;
- You can match numerous keywords with each suggestion;
- End-users may only select one of the suggested values per answer, but can select the value either by clicking on or typing the suggestion itself, or by using any of the associated keywords in their response;
- The Bot will extract a value in either of the following cases:
 1. When the end-user picks it from among the suggestions; or
 2. If there's a match between of a keyword the user types and a suggestion.
- **Here's a behind-the-scenes peek at a choice type question:**

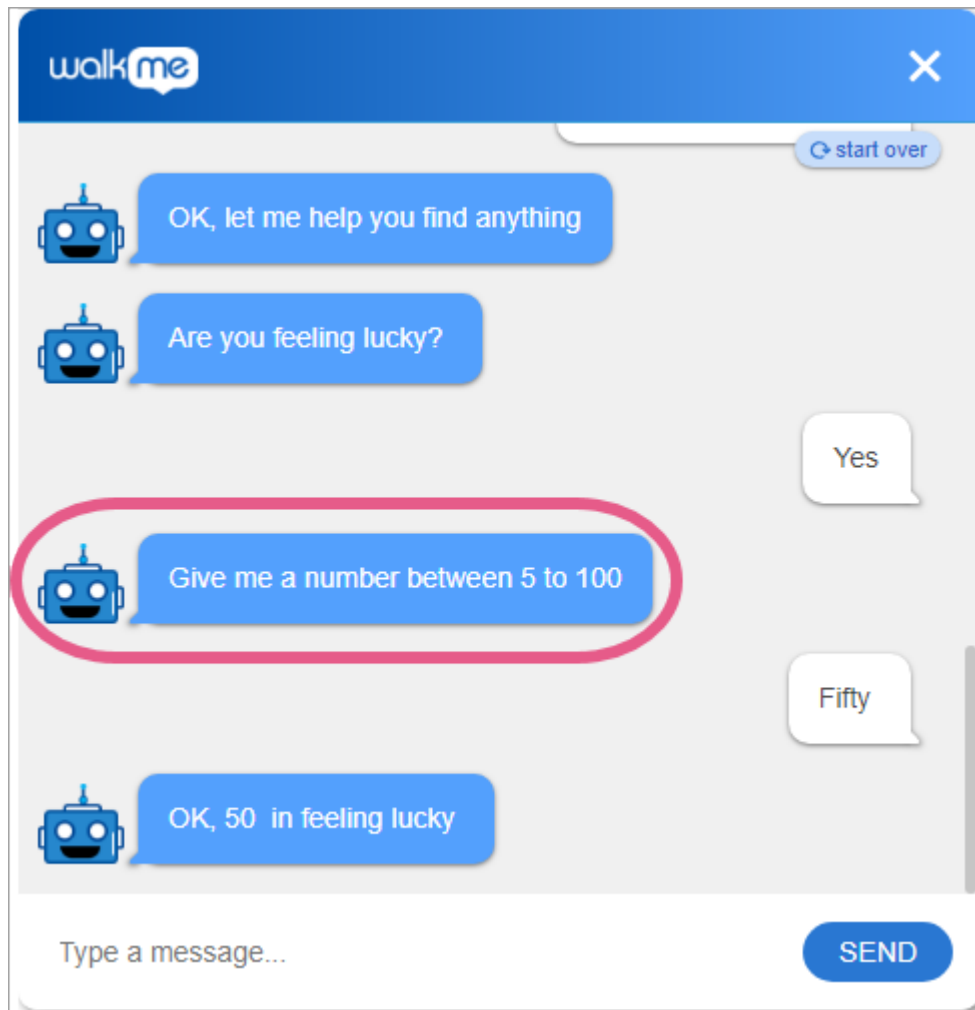


In this example, there are five suggestions (to the left): Education, Vacation, Jury Duty, etc., and each suggestion has one or more keywords (to the right);

- If the end-user types in any of the keywords, the value extracted will be the matching suggestion.
 - e.g., if the end-user responds to the question with: “I want to study,” or “I got a scholarship,” the Bot will extract the value *Education*.

Number Type Question

- Uses NLP;
- The Bot will try to find a number in the end-user’s answer and use the number as the extracted answer;
- For example, if the end-user’s answer is “ Five hundred dollars “ the extracted value would be 500;
- **Here’s an example of a number type question:**



- Note that even though the user typed the word "Fifty," the Bot was able to use NLP to extract the value "50."

Date Type Question

Date type questions allow the ActionBot to return the formatted date string in the given format assigned by the ActionBot builder.


Supports ActionBot NLP

The bot will try to extract a date from the end-user's input and then assign the date in the same format as defined in the Format settings option.

The date can vary between different input types such as "Tomorrow," "Next week," and 1.2, and or 1-9-1990.

Example: Vacation Start date


walkme




Hi! I'm WalkMe ActionBot. How can I help?

↻ start over

Take time off




Ok, let's start taking time off. I'm going to ask you a couple of questions and then I'll do it for you.



What is your Name?

Joe



Ok, Joe, what would be your vacation's start date?

Suggestions:

▼

Tomorrow

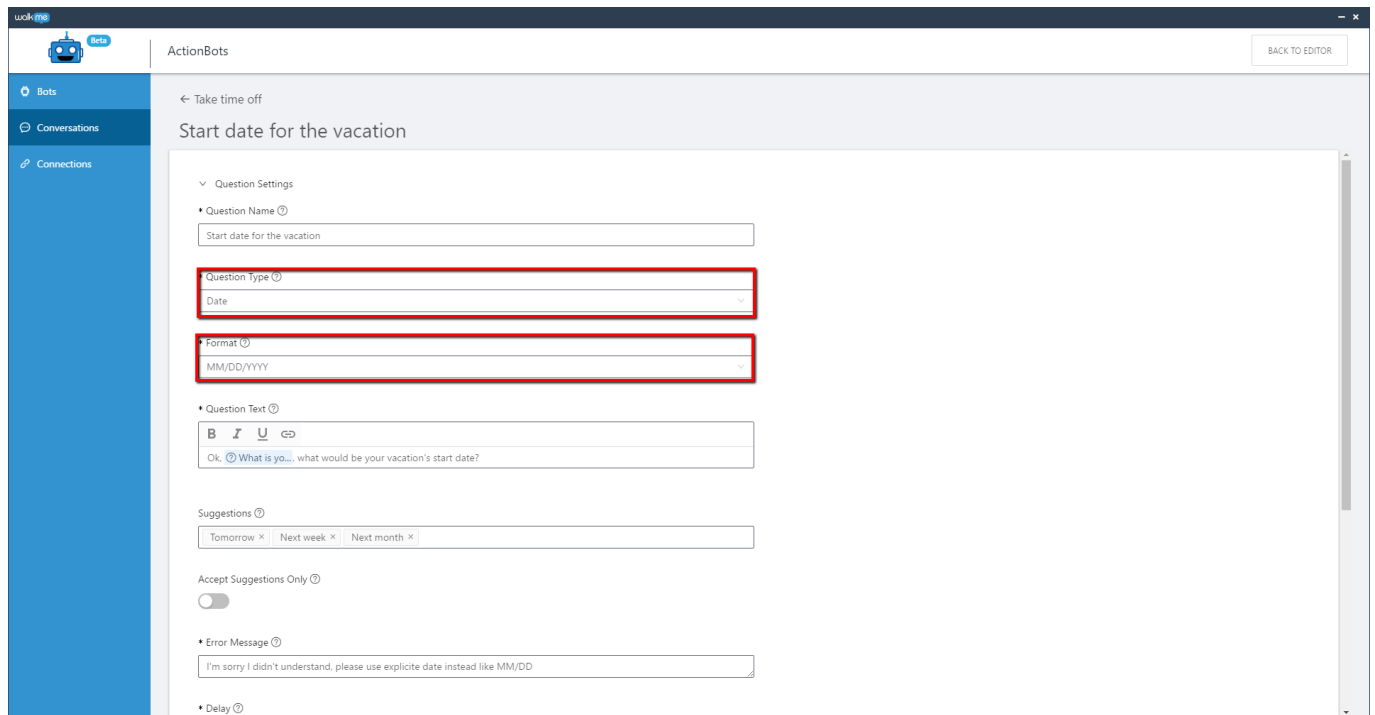
Next week

Next month

Type a message...

SEND

71 Stevenson Street, Floor 20 San Francisco, CA 94105 | 245 Fifth Avenue, STE 1501 New York, NY, 10016 | 421 Fayetteville St STE 215 Raleigh, NC 27601 www.walkme.com



walkme ActionBots

← Take time off

Start date for the vacation

Question Settings

Question Name [?]

Start date for the vacation

Question Type [?]

Date

Format [?]

MM/DD/YYYY

Question Text [?]

B *I* U [?]

Ok. What is yo..., what would be your vacation's start date?

Suggestions [?]

Tomorrow × Next week × Next month ×

Accept Suggestions Only [?]

☐

Error Message [?]

I'm sorry I didn't understand, please use explicit date instead like MM/DD

Delay [?]

Date Format Types

Type	Expected Output Example
DD/MM/YYYY	20/10/2021
MM/DD/YYYY	10/20/2021
YYYY/MM/DD	2021/20/10
Users Locale Format	The computer's date format using user locale. ex: 10/21/2021
Custom Date Format	A custom format created by the builder. The full list of options can be found below.

Accepted Custom Date Formats

Custom date formats can be built using a combination of the following recommended/common patterns. Please note that these patterns are **case-sensitive**.

See <https://date-fns.org/v2.7.0/docs/format> for full list of patterns.

Unit	Pattern	Result examples
Calendar year	y	44, 1, 1900, 2017
	yo	44th, 1st, 0th, 17th
	yy	44, 01, 00, 17
	yyy	044, 001, 1900, 2017
	yyyy	0044, 0001, 1900, 2017
Month (formatting)	M	1, 2, ..., 12
	Mo	1st, 2nd, ..., 12th
	MM	01, 02, ..., 12
	MMM	Jan, Feb, ..., Dec
	MMMM	January, February, ..., December
	MMMMM	J, F, ..., D
Day of month	d	1, 2, ..., 31
	do	1st, 2nd, ..., 31st
	dd	01, 02, ..., 31

Example custom formats for date July 4th, 2021:

Example Pattern	Example Output
MMM do, yyyy	Jul 4th, 2021
MM/dd/yyyy	07/04/2021
M-d-yy	7-4-21

☐ Tips:

- The ActionBot can also extract and return **date+time** for questions like “What date and time is your flight?”
- **Advanced level:** You can configure the extracted date to a different format in the Smart Walk-

Thru (for example, instead of MM/DD/YYYY by adding `${theNameOfTheVale@requestedFormat}`)

- **Example:** For this example, we extracted a start date and called it `${startDate}` in the Action. In the Smart Walk-Thru, we can adjust it to the desired format as follows:

Current Format	Requested Format	Value to use in the Smart WalkThru
MM/DD/YYYY (12/03/2018)	DD-MMM (03-Dec)	<code>\${startDate@DD-MMM}</code>
MM/DD/YYYY (12/03/2018)	Quarter (4)	<code>\${startDate@Q}</code>
MM/DD/YYYY (12/03/2018)	D/M/YY (3/12/18)	<code>\${startDate@D/M/YY}</code>

Phone Type Question

- Uses NLP;
- The Bot will try to extract anything in the end-user's answer that could match a phone number (using NLP models).

Regex Type Question

- Doesn't use NLP;
- This question type allows you to configure any question type you can build using regular expressions;
 - Insert the regular expression into the *Pattern* field in the Editor.
- When using this question type you will see the option to configure RegEx **flags**, which can have any combination of the following values:
 - **g** - global match; find all matches rather than stopping after the first match;
 - **i** - ignore case; if **u** flag is also enabled, use Unicode case folding;
 - **m** - multi-line; treat beginning and end characters (^ and \$) as working over multiple lines (i.e., match the beginning or end of each line (delimited by n or r), not only the very beginning or end of the whole input string);
 - **u** - Unicode; treat pattern as a sequence of Unicode code points;
 - **y** - sticky; matches only from the index indicated by the lastIndex property of this regular expression in the target string (and does not attempt to match from any later indexes).
- **Here's an example scenario, solution and regex value for a regex type question:**
 - **Scenario:** You want to build a question that validates that an end-user's input is a valid email;
 - **Solution:** use a regex type question to validate the input is a valid email pattern;
 - **Regex**

value: ^((([<>()[].,;:s@"]+(.[<>()[].,;:s@"]+)*)(["'+"])|@((((0-9){1,3}.(0-9){1,3}.(0-9){1,3}.(0-9){1,3})|([a-zA-Z0-9]+.)+[a-zA-Z]{2,})))\$

Choice – Salesforce Type Question

- This question type only appears once you've configured at least one Salesforce Connection (added in the *Connections* tab);
- This question type has many facets, and has its own article. Read the [ActionBot Connections — Salesforce Connection Type Question](#) article.

Tip: Simplify the conversation with form mode

For conversations that do not require splits in the Bot's dialogue, the [ActionBot Form Mode](#) allows you to switch the conversation from question-by-question to a single form. This simplifies the interaction with ActionBot.