

Continue



Effectiveness matters in the fast world of business. Whether you're managing customer orders, processing data, or maintaining communications, automation can drastically reduce the time and effort required to complete routine tasks. Make offers a powerful, user-friendly platform to automate nearly any process, allowing you to connect thousands of different apps and services seamlessly. Best of all, it's completely free to start using. This guide will walk you through the basics of Make, from setting up your account to creating your first automation scenario. By the end of this tutorial, you'll have a solid understanding of how Make works and how to leverage it to streamline your workflow, saving valuable time daily.

- Getting Started with Make** The first step in your automation journey with Make.com is to sign up for an account. Here's how to get started: **Setting Up Your Account** Head to the Make.com website. Once there, you can easily sign up for a free account. With a free account, you'll receive 1,000 operations per month plenty to start exploring the platform's capabilities. You can subscribe to a Pro account, which offers 10,000 operations, extra features, and a free month if you find that you need more. To sign up, simply fill out the required information. After completing the registration process, you'll be directed to the main Make dashboard.
- Navigating the Make.com Dashboard** The Make dashboard is your central hub for managing automation, referred to as "scenarios." Here's how to navigate the dashboard and explore what it has to offer: **Exploring Templates** On the left-hand side of the dashboard, you'll find a menu with several options, including "Templates." Clicking on this brings you to a page filled with thousands of pre-built templates. These templates demonstrate various ways to connect different apps and services. For example, you might find a template that collects data via a webhook and then feeds that data into Google Sheets. Or perhaps you'll discover one that forwards new Telegram messages to Discord. Templates are a great way to get inspiration for your own automation. You can use them as they are or customize them to better suit your needs. If you're new to automation, templates can also serve as a helpful learning tool, showing you how different scenarios are built and how the various components interact. **Building on Existing Templates** If you find a template that closely matches what you need, you can click on it to learn more about how it works. From there, you can customize the template by adding additional steps or modifying existing ones. For example, if you find a template that forwards Telegram messages to Discord, you might want to add a step that also sends those messages to an email address. This flexibility allows you to tailor each automation to fit your specific workflow, ensuring that it meets all of your requirements.
- Creating Your First Automation Scenario** While templates are a fantastic starting point, creating your scenario from scratch will give you a deeper understanding of how Make works and what it can do. Let's walk through the process of building a basic automation scenario. **Accessing the Scenario Editor** To start building your scenario, go back to the dashboard. Click on "Scenarios" in the left-hand menu. You'll see that you don't have any scenarios yet, but that's about to change. In the top right-hand corner, click on "Create a New Scenario". This will take you to the scenario editor, where you can start designing your automation. **Understanding the Scenario Editor** At first glance, the scenario editor might seem overwhelming, with a lot of different options and icons. Don't worry we'll break it down step by step. In the center of the screen, you'll see a large plus icon. This is where you'll start building your scenario. The first action in your automation, referred to as the "trigger", is represented by the plus icon. An event that initiates automation is known as a trigger. For example, when a customer submits an order via Google Forms, this can trigger a series of automated actions.
- Selecting and Configuring a Trigger** To begin, click on the plus icon in the center of the screen. This will open a menu displaying all the different apps you can use to trigger your scenario. Make supports thousands of apps, so whether you're using Google Forms, Gmail, Slack, or something else, you're likely to find it here. Example: Using Google Forms as a Trigger Let's say you run a small online bakery, the Kevin Cookie Company. Customers place orders through a Google Form. You want to automate the process so that when a new order is submitted, it automatically triggers a series of actions, such as updating a Google Sheet and sending a confirmation email. To do this, start by searching for Google Forms in the app list. Once you find it, click on it to see the available actions. You'll see options like "Get a Form", "Search Forms" and "Create a New Form." For our purposes, we want to use "Watch Responses", which triggers the automation when a new response is received. Click on "Watch Responses." This opens a dialog where you'll need to connect your Google Forms account to Make. If this is your first time connecting Google Forms, you'll need to authenticate and grant Make.com access to your account. Once connected, you'll be able to select the specific form you want to monitor. Finding the Form ID Every Google Form is distinguished by a special ID. Open the desired form from your Google Forms page to obtain the Form ID. Examine the address bar URL in your browser. The string of letters in the URL that comes between "edit" and "D" is the Form ID. Select and duplicate this ID, then go back to Make and insert it into the relevant space. Setting Data Limits Next, you'll see an option to set a limit on how much data to pull from the form. For testing purposes, you might want to set this to a low number, like 2, to only pull in a couple of responses. Once you've configured everything to your liking, click "OK" to save the trigger.
- Adding Actions to Your Scenario** Now that you've set up your trigger, it's time to decide what happens next. The steps that come after the trigger are called actions. For instance, after receiving a new order, you might want to log the details in a Google Sheet, send a confirmation email to the customer, and notify your fulfillment team. Example: Logging Orders in Google Sheets To add an action, click on the plus icon that appears after your trigger. This opens a menu similar to the one you saw earlier, where you can choose which app to connect. Let's start by selecting Google Sheets. Once you've selected Google Sheets, you'll see a list of possible actions, such as "Add a Row", "Update a Row", or "Get a Spreadsheet." Since we want to log new orders, choose "Add a Row." This will open a dialog where you can configure the action. **Mapping Data from Google Forms to Google Sheets** Now comes the critical step of mapping the data. This is where you tell Make how to transfer information from the Google Form to the correct columns in your Google Sheet. For example, the first column in your spreadsheet might be "Customer Name." Click on the corresponding text field in Make to open a list of data fields available from the Google Form. Navigate through the options until you find the customer's name, then select it. This tells Make to place the customer's name from the form into the "Customer Name" column in the spreadsheet. Repeat this process for each column in your spreadsheet, mapping data from the Google Form to the appropriate fields in the Google Sheet. **Connecting Google Sheets** Just like with Google Forms, you'll need to connect your Google Sheets account to Make.com automation. Once connected, select the specific spreadsheet where you want to log the data. In this example, we'll use a spreadsheet called "Kevin Cookie Company Order Tracking Sheet." Next, you'll need to select the sheet within the spreadsheet (if there's more than one) and confirm whether it contains headers. Headers are the labels at the top of each column, like "Customer Name" "Email", "Shipping Address", etc. If your sheet has headers, select "Yes." 6. **Testing and Running Your Scenario** Before you go live with your automation, it's important to test it to make sure everything is working correctly. Make makes it easy to run tests and debug your scenarios. Running a Test Order To test your scenario, go back to Google Forms and submit a sample order. For example, fill out the order form as if you were a customer, then submit it. Running the Scenario in **Make Back in Make, click the "Run Once" button in the bottom left-hand corner of the scenario editor. This will execute the scenario, processing the test order you just submitted. As the scenario runs, you'll see it progress through each step. If everything is set up correctly, you should see the new order logged in your Google Sheet. You can also click on the completed steps in Make to view the data that was processed at each stage. This is a great way to verify that the data mapping is working as expected.
- Enhancing Your Scenario with Additional Steps** Once you have the basic scenario working, you can enhance it by adding more steps. For example, you might want to automatically send a confirmation email to the customer or notify your team via Microsoft Teams. Sending a Confirmation Email After setting up the basic automation scenario, you can further enhance it by adding a step to automatically send a confirmation email to your customers once their order is received. This is a crucial part of customer service, ensuring that your customers feel acknowledged and informed about their order status. In Make, you can easily add this step by selecting the email service you use, such as Gmail or Outlook, from the list of apps. Once selected, you can configure the email template, including dynamic fields like the customer's name, order details, and any other personalized information. This not only automates the process but also adds a personal touch to your communications, enhancing customer satisfaction and trust in your business. **Conclusion** By following this step-by-step guide, you've taken your first steps into the world of automation with Make. You've learned how to set up your account, navigate the dashboard, create your own automation scenario, and even enhance it with additional steps. With this powerful tool at your disposal, you can streamline your workflow, save time, and ensure that routine tasks are handled efficiently and accurately. As you continue to explore and experiment with Make, you'll discover countless ways to connect different apps and services, creating automations that suit your specific needs. Whether you're managing a small business, organizing personal projects, or handling complex workflows, Make provides a versatile and accessible platform to help you work smarter. Remember, automation is all about freeing up your time to focus on what truly matters. With Make, you have the ability to simplify and enhance your processes, making your work life easier and more productive. So, dive in, start building, and watch how automation transforms the way you work. Get in Touch with an Expert: www.growwstacks.com Join our Skool Community Fore Learn More Automation: Automation DIY GNU Make is a tool which controls the generation of executables and other non-source files of a program from the program's source files. Make gets its knowledge of how to build your program from a file called the makefile, which lists each of the non-source files and how to compute it from other files. When you write a program, you should write a makefile for it, so that it is possible to use Make to build and install the program. Capabilities of Make Make enables the end user to build and install your package without knowing the details of how that is done – because these details are recorded in the makefile that you supply. Make figures out automatically which files it needs to update, based on which source files have changed. It also automatically determines the proper order for updating files, in case one non-source file depends on another non-source file. As a result, if you change a few source files and then run Make, it does not need to recompile all of your program. It updates only those non-source files that depend directly or indirectly on the source files that you changed. Make is not limited to any particular language. For each non-source file in the program, the makefile specifies the shell commands to compute it. These shell commands can run a compiler to produce an object file, the linker to produce an executable, ar to update a library, or TeX or Makeinfo to format documentation. Make is not limited to building a package. You can also use Make to control installing or deinstalling a package, generate tags tables for it, or anything else you want to do often enough to make it worth while writing down how to do it. Table of contents Download Documentation Mailing lists Getting involved Licensing A rule in the makefile tells Make how to execute a series of commands in order to build a target file from source files. It also specifies a list of dependencies of the target file. This list should include all files (whether source files or other targets) which are used as inputs to the commands in the rule. Here is what a simple rule looks like: target: dependencies ... commands ... When you run Make, you can specify particular targets to update; otherwise, Make updates the first target listed in the makefile. Of course, any other target files needed as input for generating these targets must be updated first. Make uses the makefile to figure out which target files ought to be brought up to date, and then determines which of them actually need to be updated. If a target file is newer than all of its dependencies, then it is already up to date, and it does not need to be regenerated. The other target files do need to be updated, but in the right order: each target file must be regenerated before it is used in regenerating other targets. Advantages of GNU Make GNU Make has many powerful features for use in makefiles, beyond what other Make versions have. It can also regenerate, use, and then delete intermediate files which need not be saved. GNU Make also has a few simple features that are very convenient. For example, the -o file option which says "pretend that source file file has not changed, even though it has changed." This is extremely useful when you add a new macro to a header file. Most versions of Make will assume they must therefore recompile all the source files that use the header file, but GNU Make gives you a way to avoid the recompilation, in the case where you know your change to the header file does not require it. However, the most important difference between GNU Make and most versions of GNU Make is that GNU Make is free software. Makefiles And Conventions We have developed conventions for how to write Makefiles, which all GNU packages ought to follow. It is a good idea to follow these conventions in your program even if you don't intend it to be GNU software, so that users will be able to build your package just like many other packages, and will not need to learn anything special before doing so. These conventions are found in the chapter "Makefile conventions" (147 k characters) of the GNU Coding Standards (147 k characters). Download Make can be found on the main GNU ftp server (download Make via HTTPS, download Make via HTTP or download Make via FTP), and its mirrors; please use a mirror if possible. Documentation Documentation for Make is online, as is documentation for most GNU software. It can also be accessed by running info make or man make, or by looking at /usr/share/doc/make/, or similar directories on your system. A brief summary is available by running make --help. Mailing lists Make has the following mailing lists: bug-make for discussing most aspects of Make, including development and enhancement requests, as well as bug reports. help-make for general user help and discussion. Announcements about Make and most other GNU software are made on the info-gnu mailing list (archives). Security reports that should not be made immediately public can be sent directly to the maintainer. If there is no response to an urgent issue, you can escalate to the general security mailing list for advice. Getting involved Development of Make, and GNU in general, is a volunteer effort, and you can contribute. For information, please read How to help GNU. If you'd like to get involved, it's a good idea to join the discussion mailing list (see above). Development tools For development sources, issue trackers, and other information, please see the Make project page at savannah.gnu.org. Testing Trying the latest test release (when available) is always appreciated. Test releases can be found on the GNU "alpha" server (HTTPS, HTTP, FTP) and its mirrors. Translating To translate the program messages into other languages, please refer to the Translation Project page for Make. New translations or updates to the existing strings will not be incorporated into Make if they are sent elsewhere. For more information, see the Translation Project home page. Maintainer Make is currently maintained by Paul Smith. Please use the mailing lists for contact. Licensing Make is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version. "The Free Software Foundation (FSF) is a nonprofit with a worldwide mission to promote computer user freedom. We defend the rights of all software users." transitive verb 1 a : to bring into being by forming, shaping, or altering material : fashion b : to lay out and construct c : compose, write 2 : to put together from components : constitute 3 a : to cause to happen to or be experienced by someone b : to cause to exist, occur, or appear : create c : to favor the growth or occurrence of d : to fit, intend, or destine by or as if by creating 4 a : enact, establish b : to execute (see execute sense 2) in an appropriate manner c : set, name 5 a : to cause to be or become b : appoint 6 : to cause to act in a certain way : compel 7 a : to carry out (an action indicated or implied by the object) b : to perform with a bodily movement 8 : to frame or formulate in the mind 9 a : to set in order b : prepare, fix c : to assemble and set alight the materials for (a fire) d : to shuffle (a deck of cards) in preparation for dealing 10 a : to amount to in significance b : to count as that makes the third time you've said it c : to form the essential being of d : to form by an assembling of individuals 11 a : to be or be capable of being changed or fashioned into b : to develop into she will make a fine judge 12 a : reach, attain made port before the storm –often used with ityou'll never make it that far b : to gain a place on or in the story made the papers c : to gain the rank of d : to succeed in providing or obtaining 13 : to gain (something, such as money) by working, trading, or dealing 14 a : to act so as to earn or acquire b : to score in a game or sport c : to convert (a split) into a spare in bowling d : to succeed in holding 15 a : to conclude as to the nature or meaning of something what do you make of this development? b : to regard as being not the fool some make him 16 a : to produce as a result of action, effort, or behavior with respect to something tried to make a thorough job of it b archaic : to turn into another language by translation 17 a : to include in a route or itinerary make New York on the return trip –often used with it b : catch sense 6b made the bus just in time 18 : to cause or assure the success or prosperity of can either make you or break you 19 : to provide the most enjoyable or satisfying experience of meeting the star of the show really made our day 20 a : to compute or estimate to be : make it 23 miles to the border from here. b : to form and hold in the mind 21 a : to begin or seem to begin (an action) 22 23 : to prepare (hay) by cutting, drying, and storing 24 a : to cause (an electric circuit) to be completed b chiefly dialectal : shut the doors are made against you—William Shakespeare 25 : to persuade to consent to sexual intercourse : seduce intransitive verb 1 : to have considerable effect courtesy makes for safer driving 2 : set out, head 3 a : to begin or seem to begin a certain action made as though to hand it to me b : to act so as to be or to seem to be c : behave, act Don't make as if you didn't hear me. d slang : to play a part —usually used with like 4 : to increase in height or size 5 : to reach or extend in a certain direction The forest makes up nearly to the snow line. 6 : to undergo manufacture or processing the silk makes up beautifully 7 archaic : to compose poetry