

# How-to: Integrate with Google Sheets

This document explains how to integrate the Civitech Verify API with Google Sheets.

*The Civitech Verify API makes it easy to check voter registration statuses against state voter files and track changes in status over time. If you have any questions about Verify or your Google Sheets integration, please don't hesitate to reach out to us at [support@civitech.io](mailto:support@civitech.io) - we're happy to help!*

---

**Even if you're new to APIs, with these steps and a little bit of tweaking, you can take a list of potential voters from your existing database, run it through the API, append voter registration statuses, and track those statuses over time.**

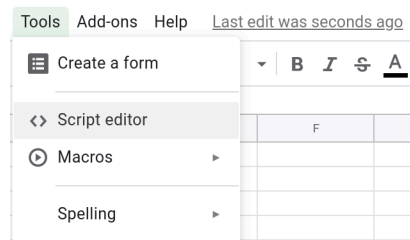
# Verify Voter Registration API

## Steps: Set up a Google Sheet to call the Verify API

1. Create a new [Google Sheet](#).
2. Fill the Google Sheet with your voter data.

	A	B	C	D	E	F	G
1	Ron	Tsosie	1299 Taylor St	Montgomery	36109	AL	
2	Brandie	Nguyen	9015 Fairview St	Wasilla	99654	AK	
3	Peggy	Lee	5737 Homestead Rd	Tucson	85718	AZ	
4	Russell	Kim	5079 E Pecan St	Little Rock	72209	AK	
5	Louella	Williams	568 Forest Ln	Santa Monica	90403	CA	
6	Irene	Johnson	8027 Hillcrest Rd	Pueblo	81001	CO	

- a. Make sure you include the required fields as found in the Verify API docs - they can be in any order.
  - b. Don't include any column headers on this sheet. You can add them into a formatted tab later, but for now, you really just want the raw data to send to the API.
  - c. Leave a column blank to hold voter registration status.
  - d. Note: if you're including birthdate, which is an optional field, make sure it is formatted as 'YYYY-MM-DD'. Google Sheets will often reformat your dates, so you may have to copy and paste this column without formatting, or use the formula =TEXT(J1,"yyyy-mm-dd") to convert a date in cell J1 (for example) to the correct string format and avoid an error.
3. Create a Google Script to call the API and populate registration status for each row.
    - a. Go to Tools > Script Editor to [create a new Google Script](#)



- b. Delete the "myFunction{}" text that will auto-populate, and replace it with the sample script linked in [this document](#).

# Verify Voter Registration API

```
1  var KEY = ""; // THE API KEY THAT CIVITECH PROVIDED;  
2  var SECRET = ""; // THE API SECRET THAT CIVITECH PROVIDED  
3  
4  function verifyVoters() {  
5      var rows = SpreadsheetApp.getActiveSheet().getDataRange().getValues();
```

- c. Add your Civitech Verify Key and Secret at the top of the script, inside the quotation marks.
- d. Update the column numbers, as indicated in the sample script, with the correct column numbers. Note that column A = 0, column B = 1, and so on.

```
16  var first_name = encodeURIComponent(current_row[0])  
17  var last_name = encodeURIComponent(current_row[1])  
18  var suffix = ""  
19  var address = encodeURIComponent(current_row[2])  
20  var city = encodeURIComponent(current_row[3])  
21  var zip = encodeURIComponent(current_row[4])  
22  var state = encodeURIComponent(current_row[5])
```

- e. If you'd like to include the optional fields as part of your matching (middle\_name, suffix, birth\_date, phone\_num), copy another line's code and designate the column number correctly, and then add the field into what's passed to the API as part of the URL.

## 4. Save and Run the script!

## 5. View your results in the Google Sheet

- a. The API returns exact registration statuses, giving you the info you need to take action before Election Day. You can find definitions for statuses returned through Civitech's products [here](#).

**Note for Large Lists:** The Verify API has a rate limit of 10 requests per second. If you have any questions about how the rate limit applies to your use case, please don't hesitate to reach out to us at [support@civitech.io](mailto:support@civitech.io).

## Tracking changes over time

Users of the Verify API often want to track changes in registration statuses for voters on their voter list as new registrations are processed. Verify's underlying data is updated as often as the Secretary of State files themselves, at a monthly or even weekly cadence where possible, making it easier to catch problems and follow up with registrants whose voter registration forms haven't made it onto the rolls. As soon as Civitech has updated data, you can access that data through the API and take action.

# Verify Voter Registration API

The steps below show two routes to track changes over time: simply copying the registration status column and keeping a historical record of your results, and using the Get Voters endpoint.

## Option 1: Set up a Google Sheet to use the Verify endpoint to track changing statuses

- 1. Create a historical record of statuses.**
  - a. Make a copy of the current tab, or copy the registration\_status column and any other relevant columns into a new tab. Label it with the date you ran the Verify API script.
  
- 2. Run the script again!**
  - a. When you want an update, run the Verify API script again by going to your Script Editor and pressing run! The script will write over the previous results with the most recent statuses in our database.
  
- 3. Copy the results into a second blank tab & analyze the results.**
  - a. Once you have two tabs that show registration statuses for your list, you can compare and see if any have changed.
  - b. Here are a few favorite tricks for analyzing data in Sheets: =vlookup(), =countif(F:F, ="Active"), =if(F1<>G1,"Status Changed") or other formulas; Create a Pivot Table; or Add filters.

# Verify Voter Registration API

## Option 2: Set up a Google Sheet to use the Get Voters endpoint to track changing status

When you check a potential voter's status, their record is stored securely in Alloy's database and is accessible only using your API key and secret. You can use the Get Voters endpoint to check for updates or filter on specific parameters. This is especially useful for large lists or if you haven't stored a historical record of your previously checked statuses -- Alloy does it for you!

1. **Create a new empty Google Sheet, in a new document.**
2. **Determine the parameters you want to pull.**
  - a. The Get Voters API endpoint documentation is [here](#), and the specific parameter options are listed below. Step #3 will show you how to edit a Google Script to pull a list based on your selected parameters:
    - i. **All voters:** If you want the most recent status for all of your voters, simply call the Get Voters API with no parameters.
    - ii. **Recently changed statuses:** If you initially checked a list of voter registration forms 27 days ago and want to see if any statuses have changed since then, you would want the parameter: *within\_days = 26*
    - iii. **Statuses that haven't recently changed:** On the flip side, if you want to pull the opposite list of statuses that haven't changed since you checked a list 27 days ago, you would use the parameter: *exceeding\_days = 27*
    - iv. **Specific statuses:** If you want to pull only specific statuses (a list of voters accepted onto the rolls and marked as Active, for example), use the parameter *registration\_status = Active*
    - v. **Zip Code:** Narrow to a specific zip code using *zip = 30909*
3. **Create a Google Script to call the Get Voters API endpoint**
  - a. Go to Tools > Script Editor to write a new script.
  - b. Add your parameters to the API URL that is called in the script, in this line:
    - i. `var url = Utilities.formatString('https://regverify.campaignos.app/v1/voters?')`
  - c. For example, to add *within\_days = 26*, you would change the API URL to:
    - i. `var url = Utilities.formatString('https://regverify.campaignos.app/v1/voters?within_days=26')`
  - d. To add in multiple parameters, add them together with a "&". For example, to pull Active voters whose statuses have changed in the last 30 days, you would change the API URL to:

## Verify Voter Registration API

- i. `var url = Utilities.formatString('https://regverify.campaignos.app/v1/voters?registration_status=Active&within_days=30')`
- e. Note: if you are trying to return a list of more than 1000 tracked voters, you'll need to use the `page_size` and `offset` query parameters in your URL. For example, if you have a list of 2,000 voters, you could run the query once, copy the results into a different tab, and run it again with the parameters `“?page_size=1000&offset=1001”` appended to the URL.

### **4. Run the script and view your results in the Google Sheet!**