

# Guide-2

Techwars 2025-2026

## Overview

This series of guides will walk you through setting up a local data pipeline using Node-RED, Google Sheets, and Grafana.

By following the steps in this guide, you will install the required tools, connect them using APIs, and create a live data visualization dashboard.

By the end of this guide, you will have:

- A running Node-RED instance
- A Grafana dashboard connected to Google Sheets
- A basic understanding of how APIs connect these tools

## Prerequisites

Before starting, ensure the following software is installed:

- Node.js (LTS version): Required to install and run Node-RED
- Terminal / Command Prompt access: Used to install and run Node-RED and Grafana

Run the following command in your terminal:

`node -v`

If a version number appears, Node.js is installed correctly.

e.g. v22.20.0

Here is a video explaining how to install Node.js:

<https://www.youtube.com/watch?v=It5D2EWZMNO>

## Installing Node-RED

Official Website

<https://nodered.org/>

Run the following command in your terminal:

`sudo npm install -g --unsafe-perm node-red`

e.g.,

```
MacBook-Air ~ % sudo npm install -g --unsafe-perm node-red
```

Start node-RED by entering the following command:

“node-red”

e.g., MacBook-Air ~ % node-red

You will see lines similar to the image on the following page

Use this link to access your Node-RED application: <http://localhost:1880/>

```
$ node-red

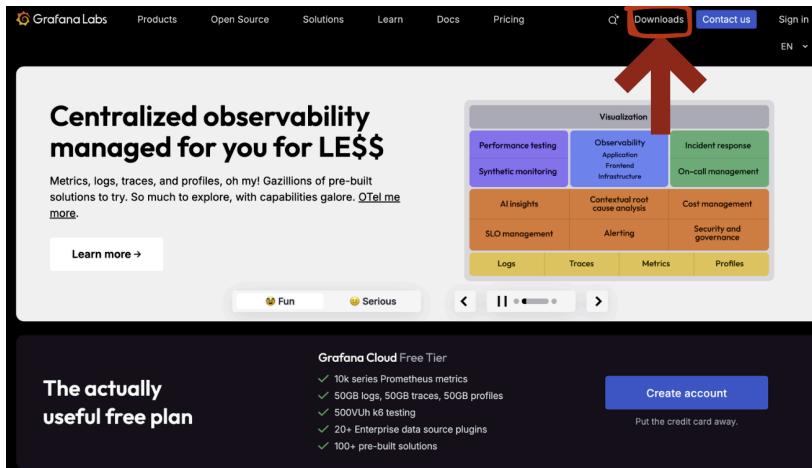
Welcome to Node-RED
=====
30 Jun 23:43:39 - [info] Node-RED version: v1.3.5
30 Jun 23:43:39 - [info] Node.js  version: v14.7.2
30 Jun 23:43:39 - [info] Darwin 19.6.0 x64 LE
30 Jun 23:43:39 - [info] Loading palette nodes
30 Jun 23:43:44 - [warn] rpi-gpio : Raspberry Pi specific node set inactive
30 Jun 23:43:44 - [info] Settings file  : /Users/nol/.node-red/settings.js
30 Jun 23:43:44 - [info] HTTP Static    : /Users/nol/node-red/web
30 Jun 23:43:44 - [info] Context store  : 'default' [module=localfilesystem]
30 Jun 23:43:44 - [info] User directory : /Users/nol/.node-red
30 Jun 23:43:44 - [warn] Projects disabled : set editorTheme.projects.enabled=true to enable
30 Jun 23:43:44 - [info] Creating new flows file : flows_noltop.json
30 Jun 23:43:44 - [info] Starting flows
30 Jun 23:43:44 - [info] Started flows
30 Jun 23:43:44 - [info] Server now running at http://127.0.0.1:1880/red/
```

## Installing Grafana

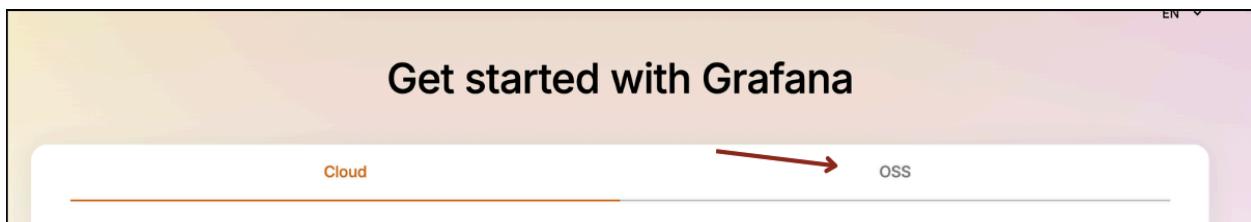
Official Website	<a href="https://grafana.com/">https://grafana.com/</a>
------------------	---

First, visit the Grafana website.

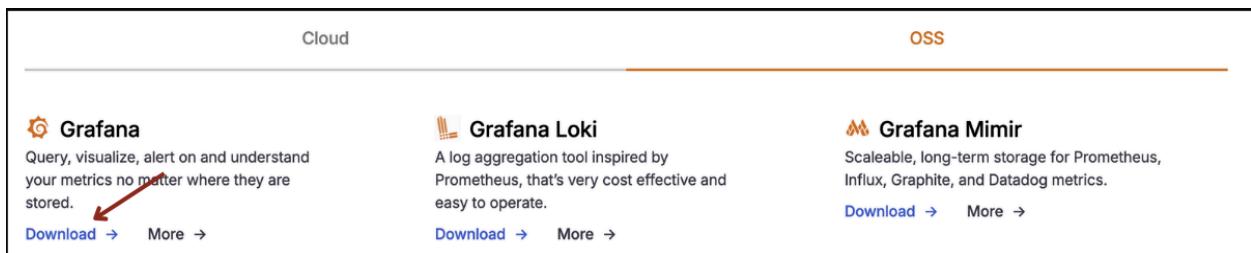
At the top bar click "Downloads"



Select the OSS menu



Click Download under "Grafana"



Select OS and follow the installation process for your OS



Standalone MacOS/Darwin Binaries (64 Bit) SHA256: e95017776bb52bc5c36e04ad2115262406acfd369827d53fd9316b38c4204aae  
Read the MacOS installation guide for more information.

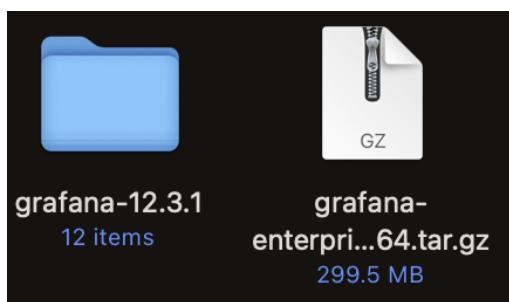
```
curl -O https://dl.grafana.com/grafana-enterprise/release/12.3.1/grafana-enterprise_12.3.1_20271043721_darwin_amd64.tar.gz
```



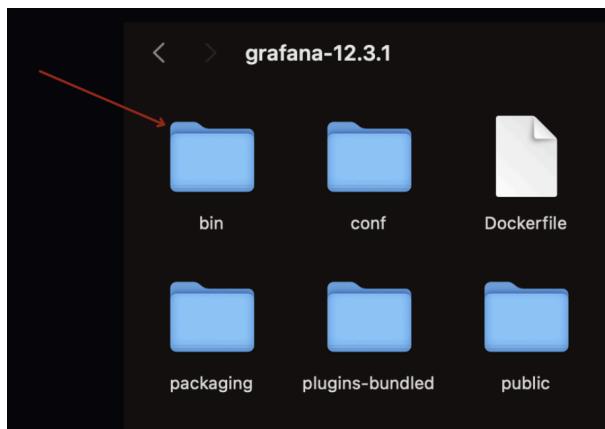
Standalone MacOS/Darwin Binaries (ARM64) SHA256: 1975f98abe8ff45b7fa732df6c29519d0d0c73ab91bd7bf4f76c37d9a0a03967  
Read the MacOS installation guide for more information.

```
curl -O https://dl.grafana.com/grafana-enterprise/release/12.3.1/grafana-enterprise_12.3.1_20271043721_darwin_arm64.tar.gz
```

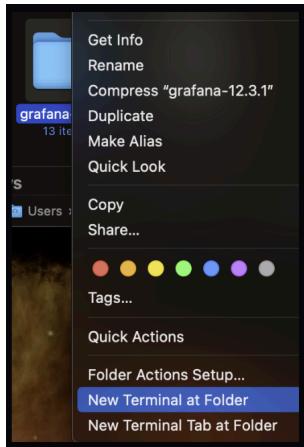
Extract the zip file to access the binaries



Open cmd in the grafana/bin directory



use `grafana-cli.exe` `plugins install grafana-googlesheets-datasource`. (Mac users: command click the unzipped folder, select “new terminal at folder and run the following command in the terminal: `./bin/grafana-server`) (If you are having trouble with the plugins command, skip it and refer to the accessing sheets section later.)



Run grafana and login (username and password are set to “admin” by default) (Grafana Usage explained later)



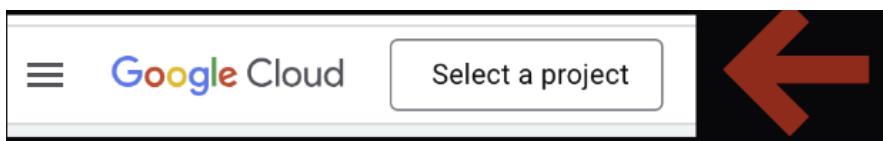
## Setting Up Google Cloud Service Account

Official Website

<https://console.cloud.google.com>

To get started, visit the website.

Create a Project - Click Select Project, then click new project, name your project, click create, click the bell icon (home page), then click select project.



Select a project

New project

Project name \* MyNewProject

Project ID: mynewproject-482210. It cannot be changed later. [Edit](#)

Location \* No organization [Browse](#)

Parent organization or folder

[Create](#) [Cancel](#)



Search

Notifications

Create Project: MyNewProject Just now

Select Project

See all activities



Go to APIs & Service → Library

Google Cloud MyNewProject Search (/) for resources, docs, products, and more

Dashboard Activity Recommendations

Project info

Project name: MyNewProject

Project number: 601890034940

Project ID: mynewproject-482210

Add people to this project

Go to project settings

Resources

BigQuery Data warehouse/analytics

API APIs Requests (requests/sec)

No data is available for the selected time frame.

3:15 3:30 3:45

Go to APIs overview



API APIs & Services ☆

**Enabled APIs & services**

- Library ←
- Credentials
- OAuth consent screen
- Page usage agreements

Traffic

Search and enable "Google Sheets API"  
Search and enable "Google Drive API"

Google Workspace View all (31)

 <p>Google Drive API Google Enterprise API <span>②</span></p> <p>Create and manage resources in Google Drive.</p>	 <p>Google Calendar API Google Enterprise API <span>②</span></p> <p>Manage calendars and events in Google Calendar.</p>	 <p>Gmail API Google Enterprise API <span>②</span></p> <p>View and manage Gmail mailbox data.</p>	 <p>Google Sheets API Google Enterprise API <span>②</span></p> <p>Read and write Google Sheets data</p>
--	--	--	--



### Google Sheets API

Google Enterprise API

Read and write Google Sheets data

[Enable](#) [Try this API ↗](#)



### Google Drive API

Google Enterprise API

Create and manage resources in Google Drive.

[Enable](#) [Try this API ↗](#)

Go to APIs & Service → Credentials

API APIs & Services ☆

**Credentials** + Create credentials Delete Restore deleted credentials

Create credentials to access your enabled APIs. [Learn more ↗](#)

▲ Remember to configure the OAuth consent screen with information about your application. Configure consent screen

**API Keys**

<input type="checkbox"/>	Name	Bound account <span>②</span>	Creation date <span>↓</span>	Restrictions	Actions
No API keys to display					

**OAuth 2.0 Client IDs**

<input type="checkbox"/>	Name	Creation date <span>↓</span>	Type	Client ID	Actions
No OAuth clients to display					

**Service Accounts** Manage service accounts

<input type="checkbox"/>	Email	Name <span>↑</span>	Actions
No service accounts to display			

## Select Create Credentials

The screenshot shows the 'Credentials' page in the Google Cloud Platform. At the top, there's a dropdown menu labeled '+ Create credentials' and buttons for 'Delete' and 'Restore deleted credentials'. Below this, a yellow banner says 'Remember to configure the OAuth consent screen with information about your application.' and has a 'Configure consent screen' button. The main area is titled 'Create credentials to access your enabled APIs' and contains four options: 'API key', 'OAuth client ID', 'Service account', and 'Help me choose'. Under 'API Keys', there's a note 'Identifies your project using a simple API key to check quota and access' and a 'Remember' note. A red arrow points from the text 'A red arrow points to the API Keys section' to the 'API Keys' section. At the bottom, there's a note 'No API keys to display'.

Select Service Account and create one with default settings (choose done for the optional menus)

This screenshot shows the 'Create service account' form. It has a title '1 Create service account'. The 'Service account name' field is filled with 'MyService'. The 'Service account ID' field is filled with 'myservice'. The 'Email address' field shows 'myservice@mynewproject-482210.iam.gserviceaccount.com'. There's a 'Service account description' field with the placeholder 'Describe what this service account will do'. At the bottom is a 'Create and continue' button.

Click on the service account after it's been created

This screenshot shows the 'Credentials' page again, but now with a 'Service Accounts' section. The 'Service Accounts' section has a table with one row: 'Email' (myservice@mynewproject-482210.iam.gserviceaccount.com) and 'Actions' (edit, delete). A large red arrow points from the text 'A red arrow points to the newly created service account' to the 'Email' column of the 'Service Accounts' table.

Go to the keys tab and create a new key

← MyService

Details Permissions **Keys** Metrics Logs Principals with access

## Keys

⚠ Service account keys could pose a security risk if compromised. We recommend you avoid down Federation [\[?\]](#). Learn more about the best way to authenticate service accounts on Google Cloud [\[?\]](#)

ℹ Google automatically disables service account keys detected in public repositories. You can cus 'iam.serviceAccountKeyExposureResponse' organization policy. [Learn more \[?\]](#)

Add a new key pair or upload a public key certificate from an existing key pair.

Block service account key creation using [organization policies \[?\]](#).  
[Learn more about setting organization policies for service accounts \[?\]](#)

Add key ▾	Key	Creation date	Expiration date
Create new key			
Upload existing key			

select JSON and it will be downloaded

Create private key for "MyService"

Downloads a file that contains the private key. Store the file securely because this key can't be recovered if lost.

Key type

JSON

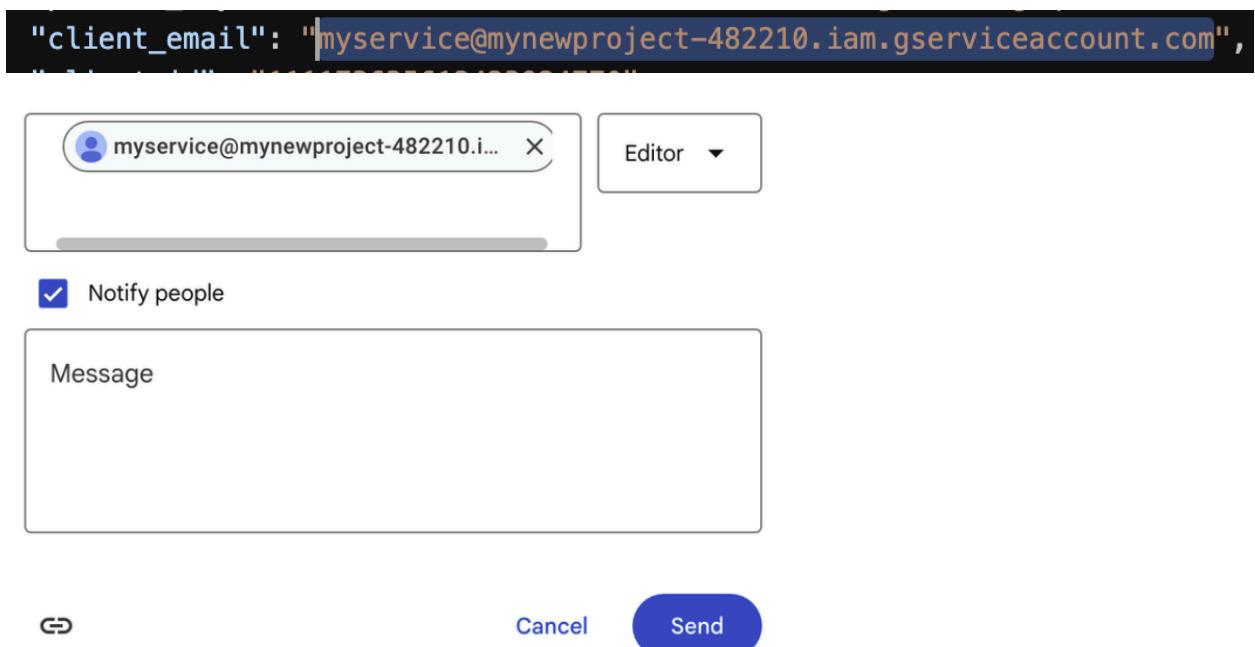
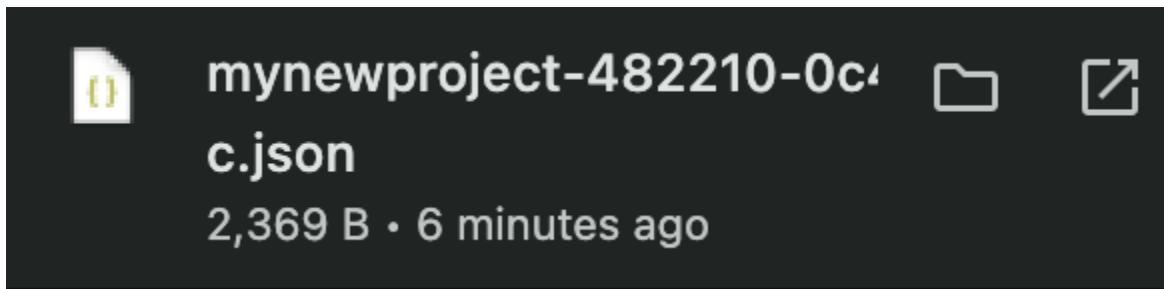
Recommended

P12

For backward compatibility with code using the P12 format

Cancel **Create**

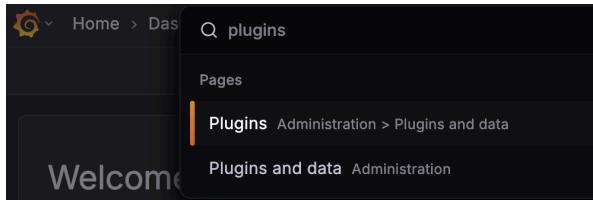
Check the email of your service account from the JSON file, go to your desired Google Sheet, and add it as an Editor



## Accessing Sheets in Grafana

Open Grafana using this link: localhost:3000 (copy paste this exactly in the URL bar)

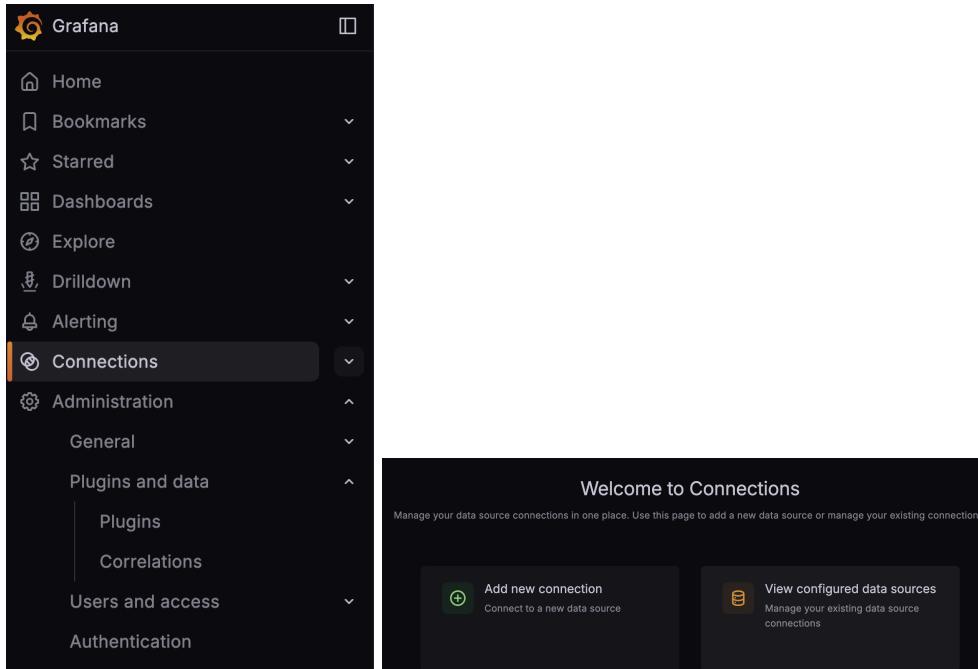
If you were unable to install the google sheets plugin, go to the grafana search bar, search “plugins” and then in the plugins section search “google sheets” and install the google sheets plugin. The click on add new data source on the same screen.

A screenshot of the "Plugins" page in Grafana. At the top, there's a search bar with "google" typed in and a "Clear" button. To the right, there's a "Type" dropdown set to "All". The main area shows a grid of six plugin cards:

- Google Analytics** by blackcowmoo, signed.
- Google BigQuery** by Grafana Labs, signed.
- Google Cloud Logging** by googlecloud, signed.
- Google Cloud Trace** by googlecloud, signed.
- Google Sheets** by Grafana Labs, signed.
- Infinity** by Grafana Labs, signed.

A modal window is open at the bottom for the "Google Sheets" plugin, showing its description: "The Google Sheets data source plugin for Grafana lets you to visualize your Google spreadsheets in Grafana" and a blue "Install" button.

Navigate back to the grafana homepage, go to Connections and click on “View configured data sources.

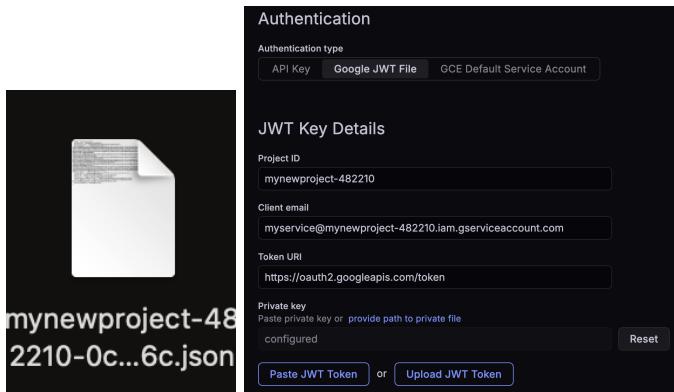


Add Data source(Search Google Sheets and select that)

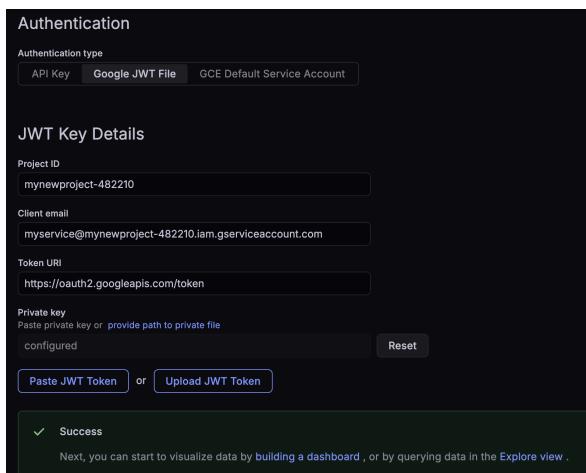
select Auth type to Google JWT file.

A screenshot of the "Authentication" configuration screen. The "Authentication type" dropdown is set to "Google JWT File". Below it, the "JWT Key Details" section shows a placeholder for a "JWT token" with the instruction "Upload or paste Google JWT token". There is a dashed box for dropping files, with the text "Drop the Google JWT file here" above it. A "Click to browse files" button and the note "Accepted file type: json" are also present. At the bottom, there are two buttons: "Paste JWT Token" and "Fill In JWT Token manually".

Drag and drop the JSON downloaded when setting up the service account



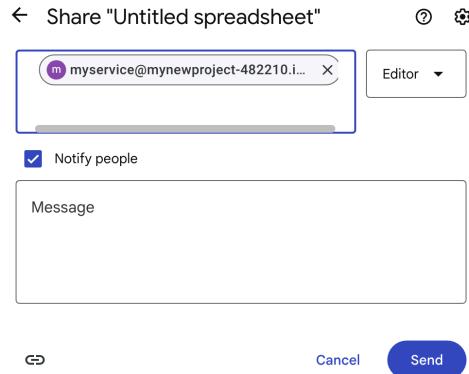
Save and test. (if the bottom displays “permissions failed” refer back to the google service account section and check if your google drive api and the google sheets api were enabled or not).



From the JSON file extract the `client_email`.

```
"client_email": "myservice@mynewproject-482210.iam.gserviceaccount.com",
```

In the case that the Google sheet you're trying to access is private you will have to give access to the `client_email` by giving it viewership rights(Go to the sheet and share).



Navigate back to the homepage and go to the dashboards section and click create dashboard

A screenshot of the Grafana homepage. The top navigation bar has the Grafana logo and a search bar. The main menu on the left includes "Home", "Bookmarks", "Starred", "Dashboards" (which is highlighted with an orange bar), "Explore", and "Drilldown". The main content area features a cartoon character wearing a VR headset and the text "You haven't created any dashboards yet". A blue button at the bottom says "+ Create dashboard".

Add a visualization

## Start your new dashboard by adding a visualization

Select a data source and then query and visualize your data with charts, stats and tables or create lists, markdowns and other widgets.

+ Add visualization

Select Google Sheets as the data source

### Select data source

Q Select data source



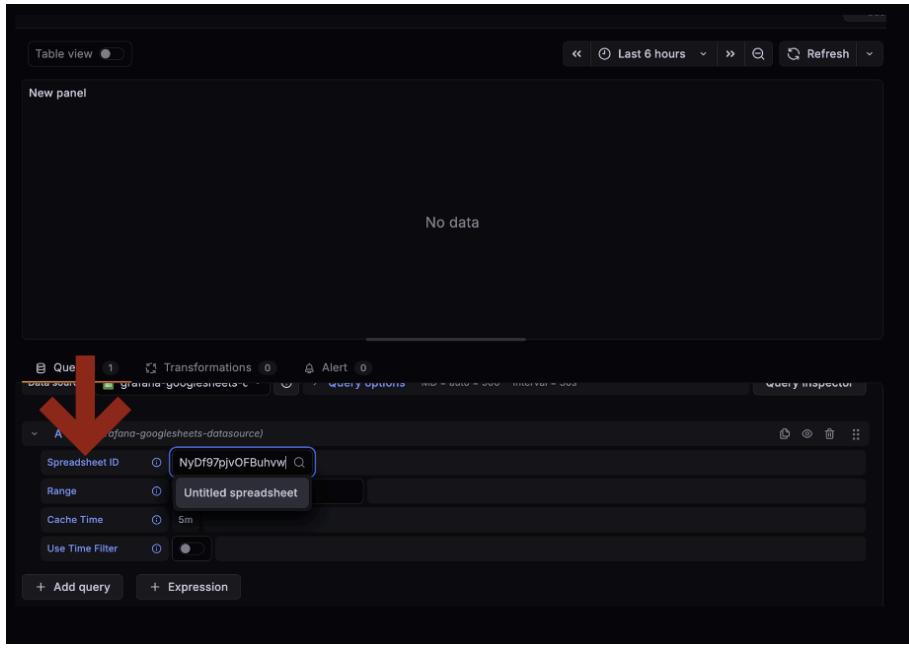
grafana-googlesheets-datasource

default

Google Sheets

For Spreadsheet ID add the sheet ID found in the URL between the d/ and the /edit

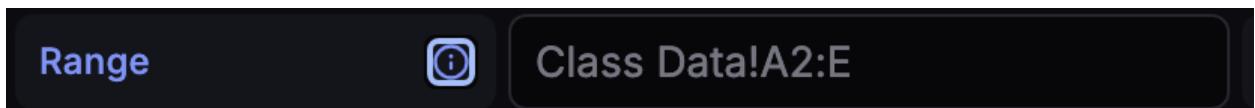
<https://docs.google.com/spreadsheets/d/1mUIRs9wh0BhiWC9S8PEI0nQ7seiNyDf97pjvOFBuhvw/edit?gid=0#gid=0>



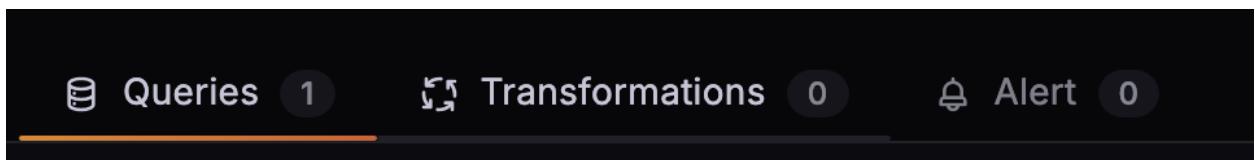
Save the dashboard. Grafana should now have access to the sheet and you can visualise the data from it!

## Data Visualisation

After Entering the Spreadsheet ID under the Queries tab you can select range from the spreadsheet to utilize. You may set the range as whatever you want. Use this syntax:  
“<sheet\_name!<lower\_limit>:<upper\_limit>”



(optional section) Visit the tab next to queries labelled "Transformations" and select organized fields by name to choose which fields to include in any particular visualisation.



To the right you can select the visualization type and customize it.

### Example Visualisation:

This is specific only to the table visualisation structure provided by grafana, there are many more options available for you and you are free to try any.

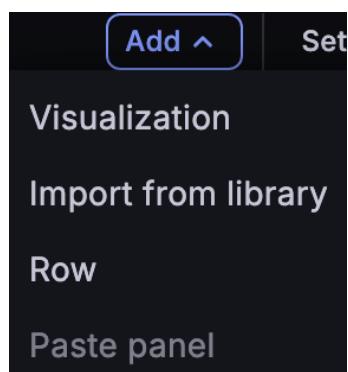
Firstly, go to your Google Sheet and input data in a table-like manner as shown:

	A	B	C	D
1	Physics	Math	CS	
2		78	100	87
3		76	99	86
4		56	25	45
5		90	74	74
6		56	89	90
7				
8				
q				

Then specify the range of your data; in this case, the data set goes from cell A1 to cell C6. You will find this option at the bottom of your dashboard.



On the top right of your dashboard, you will see an “Add” button; click on it and select visualisation



Choose the table visualisation in the visualisation menu

The screenshot shows the 'Visualizations' tab selected in the Tableau interface. A search bar at the top contains the placeholder 'Search for...'. Below the search bar, there are two tabs: 'Visualizations' (selected) and 'Suggestions'. The main area displays ten visualization options, each with an icon and a brief description:

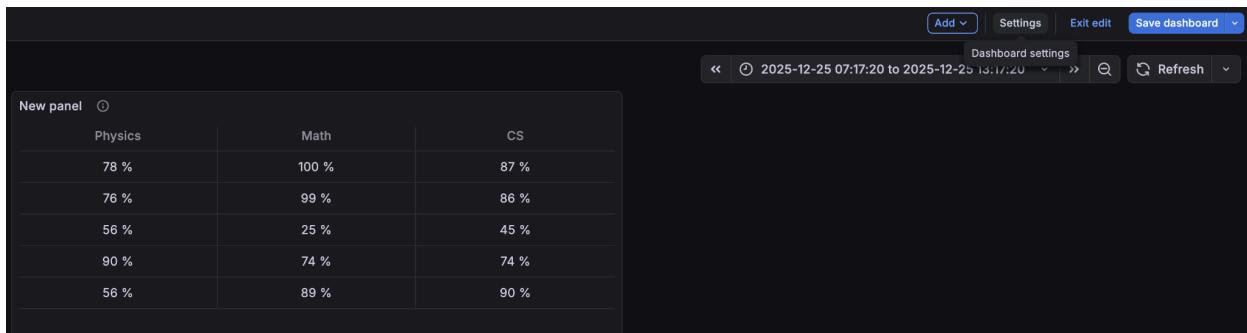
- Time series**: Time based line, area and bar charts
- Bar chart**: Categorical charts with group support
- 12.4 Stat**: Big stat values & sparklines
- Gauge**: Standard gauge visualization
- Bar gauge**: Horizontal and vertical gauges
- Table**: Supports many column styles
- Pie chart**: The new core pie chart visualization
- State timeline**: State changes and durations
- Heatmap**: Like a histogram over time
- Status history**

A table of this sorts will appear on your dashboard: you have created your visualisation!

Physics	Math	CS
78	100	87
76	99	86
56	25	45
90	74	74
56	89	90

You can use the menu on the right hand side to format this in the exact way you wish.

Go back to the dashboards page and click on settings (a formatted table for reference)



Select “JSON Model”

## Settings

---

- [General](#)
- [Annotations](#)
- [Variables](#)
- [Links](#)
- [Versions](#)
- [Permissions](#)
- [JSON Model](#)

Paste the following code in the space of this section to access an example formatted table for you to experiment with:

```
{  
  "annotations": {  
    "list": [  
      {  
        "builtIn": 1,  
        "datasource": {  
          "type": "grafana",  
          "uid": "-- Grafana --"  
        },  
        "enable": true,  
        "hide": true,  
        "iconColor": "rgba(0, 211, 255, 1)",  
        "name": "Annotations & Alerts",  
        "type": "dashboard"  
      }  
    ]  
  },
```

```
"editable": true,
"fiscalYearStartMonth": 0,
"graphTooltip": 0,
"id": 0,
"links": [],
"panels": [
{
  "datasource": {
    "type": "grafana-googlesheets-datasource",
    "uid": "df85p3ob0308wa"
  },
  "description": "hello\\n",
  "fieldConfig": {
    "defaults": {
      "custom": {
        "align": "center",
        "cellOptions": {
          "type": "auto"
        },
        "footer": {
          "reducers": []
        },
        "inspect": false,
        "styleField": "100",
        "wrapHeaderText": false
      }
    },
    "mappings": [],
    "thresholds": {
      "mode": "absolute",
      "steps": [
        {
          "color": "green",
          "value": 0
        },
        {
          "color": "red",
          "value": 1
        }
      ]
    }
  }
}
```

```
        "value": 80
    }
]
},
"unit": "%"
},
"overrides": []
},
"gridPos": {
    "h": 8,
    "w": 12,
    "x": 0,
    "y": 0
},
"id": 2,
"options": {
    "cellHeight": "sm",
    "showHeader": true
},
"pluginVersion": "12.3.1",
"targets": [
    {
        "cacheDurationSeconds": 300,
        "range": "Sheet1!A1:C6",
        "refId": "A",
        "spreadsheet": "1mUIRs9wh0BhiWC9S8PEI0nQ7seiNyDf97pjv0FBuhvw"
    }
],
"title": "New panel",
"type": "table"
},
{
    "datasource": {
        "type": "grafana-googlesheets-datasource",
        "uid": "df85p3ob0308wa"
    }
},
```

```
"fieldConfig": {  
    "defaults": {  
        "color": {  
            "mode": "palette-classic"  
        },  
        "custom": {  
            "axisBorderShow": false,  
            "axisCenteredZero": false,  
            "axisColorMode": "text",  
            "axisLabel": "",  
            "axisPlacement": "auto",  
            "barAlignment": 0,  
            "barWidthFactor": 0.6,  
            "drawStyle": "line",  
            "fillOpacity": 0,  
            "gradientMode": "none",  
            "hideFrom": {  
                "legend": false,  
                "tooltip": false,  
                "viz": false  
            },  
            "insertNulls": false,  
            "lineInterpolation": "linear",  
            "lineWidth": 1,  
            "pointSize": 5,  
            "scaleDistribution": {  
                "type": "linear"  
            },  
            "showPoints": "auto",  
            "showValues": false,  
            "spanNulls": false,  
            "stacking": {  
                "group": "A",  
                "mode": "none"  
            },  
            "thresholdsStyle": {  
                "color": {  
                    "mode": "classic"  
                },  
                "rule": {  
                    "color": "black",  
                    "dash": [4, 4],  
                    "width": 1  
                },  
                "ruleLabel": {  
                    "color": "black",  
                    "fontStyle": "italic",  
                    "fontWeight": "bold",  
                    "fontSize": 12  
                },  
                "ruleValue": {  
                    "color": "black",  
                    "fontStyle": "italic",  
                    "fontWeight": "bold",  
                    "fontSize": 12  
                }  
            }  
        }  
    }  
}
```

```
        "mode": "off"
    }
},
"mappings": [],
"thresholds": {
    "mode": "absolute",
    "steps": [
        {
            "color": "green",
            "value": 0
        },
        {
            "color": "red",
            "value": 80
        }
    ]
},
"overrides": []
},
"gridPos": {
    "h": 8,
    "w": 12,
    "x": 0,
    "y": 8
},
"id": 1,
"options": {
    "legend": {
        "calcs": [],
        "displayMode": "list",
        "placement": "bottom",
        "showLegend": true
    },
    "tooltip": {
        "hideZeros": false,
```

```
        "mode": "single",
        "sort": "none"
    },
},
"pluginVersion": "12.3.1",
"targets": [
{
    "cacheDurationSeconds": 300,
    "datasource": {
        "type": "grafana-googlesheets-datasource",
        "uid": "df85p3ob0308wa"
    },
    "refId": "A",
    "spreadsheet": "1mUIRs9wh0BhiWC9S8PEI0nQ7seiNyDf97pjv0FBuhvw"
},
],
"title": "New panel",
"type": "timeseries"
},
],
"preload": false,
"schemaVersion": 42,
"tags": [],
"templating": {
    "list": []
},
"time": {
    "from": "2025-12-25T02:17:20.873Z",
    "to": "2025-12-25T08:17:20.873Z"
},
"timepicker": {},
"timezone": "browser",
"title": "New dashboard",
"uid": "adpm224",
"version": 4
}
```

## Next Steps

Now you may refer to Guide-3 for Node-RED and API usage:

[https://docs.google.com/document/d/1aDf66Jmn8E-IRz1RGE8FL9GGfKQbnUxi2WLiL\\_XtPs/edit?usp=sharing](https://docs.google.com/document/d/1aDf66Jmn8E-IRz1RGE8FL9GGfKQbnUxi2WLiL_XtPs/edit?usp=sharing)