

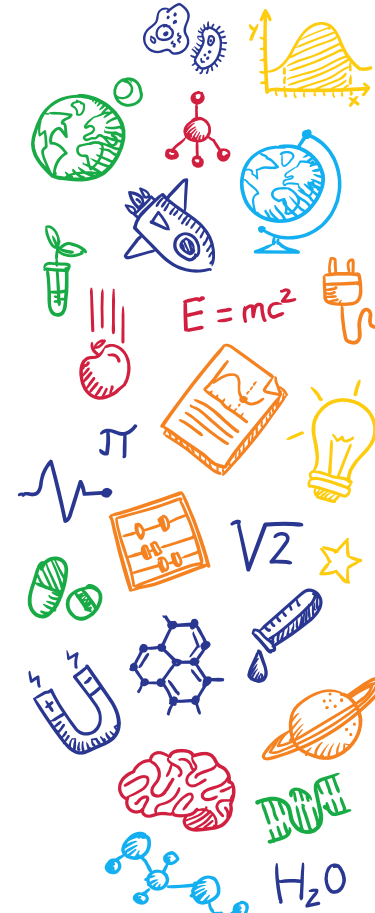
# Internet of Things

## Weather Data from Location



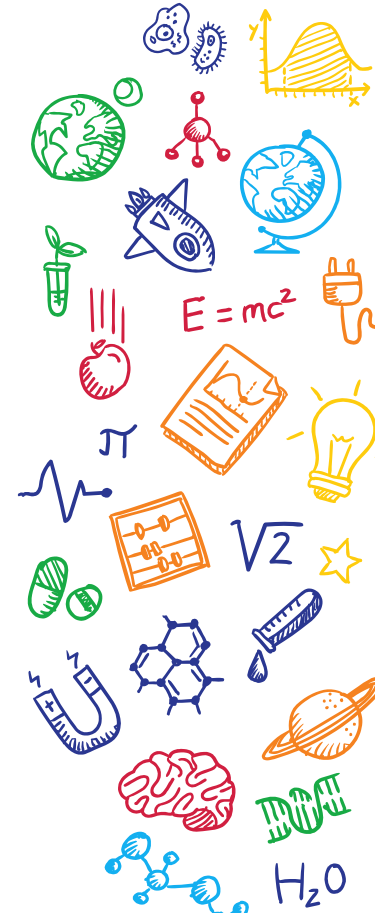
-

- There are many organizations, who gather weather data like temperature, etc. to the cloud and give access to the requested data using API.
- An **application program interface (API)** is code that allows two software programs to communicate with each other.
- The API defines the correct way for a developer to write a program that requests services from an operating system (OS) or could server, which in our case is the weather data.
- In this session we will be using OpenWeatherMap API to get the weather of a particular location.



# OpenWeatherMap API

- **OpenWeatherMap** is one of the leading digital weather information providers.
- Using OpenWeatherMap API you can get Current weather data:
  - Access current weather data for any location including over 200,000 cities
  - Current weather is frequently updated based on global models and data from more than 40,000 weather stations
  - Data is available in JSON, XML, or HTML format
  - Available for Free and all other paid accounts



# Creating Account on OpenWeatherMaps

# Creating Account

- Go to OpenWeatherMap website:  
<https://openweathermap.org/> and create a free account.

### Create New Account

STEMpedia

thestempedia@gmail.com

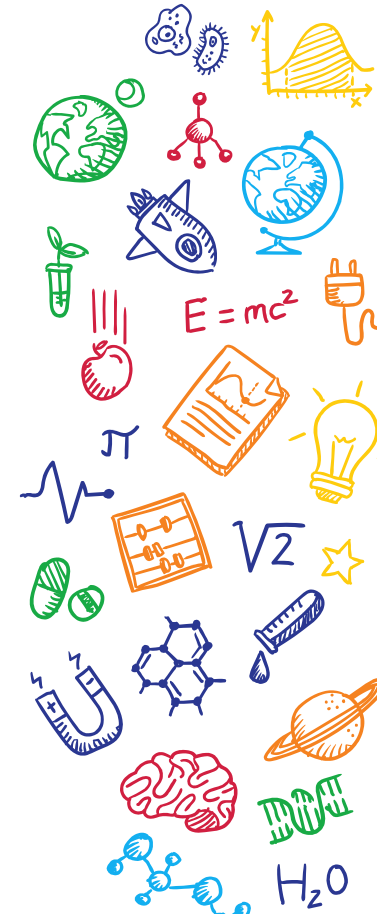
.....

.....

We will use information you provided for management and administration purposes, and for keeping you informed by mail, telephone, email and SMS of other products and services from us and our partners. You can proactively manage your preferences or opt-out of communications with us at any time using Privacy Centre. You have the right to access your data held by us or to request your data to be deleted. For full details please see the OpenWeather [Privacy Policy](#).

☒ I am 16 years old and over

☒ I agree with [Privacy Policy](#), [Terms and conditions of sale](#) and [Websites terms and conditions of use](#)



# API Key

- Choose the API tab to get the API Key.
- This API Key allows you to request data.

[New Products](#)
[Setup](#)
[API keys](#)
[Services](#)
[Payments](#)
[Billing plans](#)
[Block logs](#)
[History bulk](#)
[Logout](#)

You can generate as many API keys as needed for your subscription. We accumulate the total load from all of them.

Key

Name

8152b2a8be1c2e6b80c00b1c12aa584d

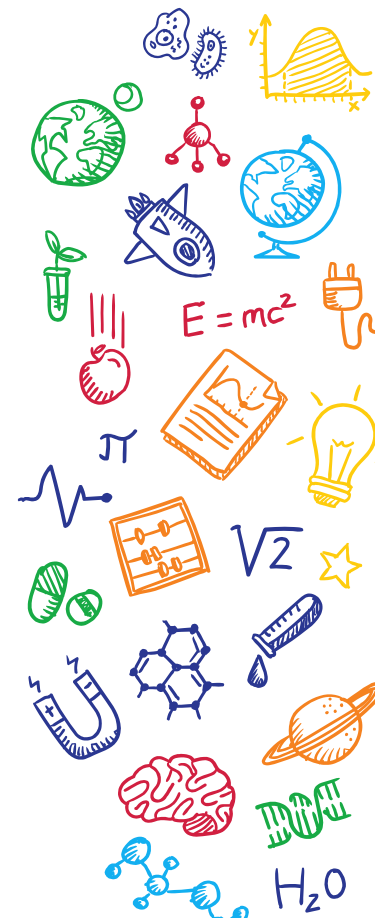
Default



Create key

\* Name

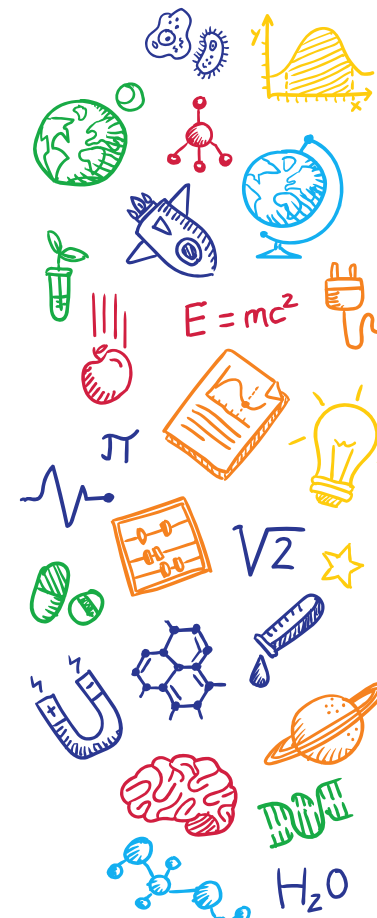
Generate



# Free Plan

- Free plan has some limitations like only 60 calls per minutes, but it is sufficient for individuals.

	Free
<b>Price</b> Price is fixed, no other hidden costs (VAT is not included)	Free
<b>Subscribe</b>	<a href="#">Get API key and Start</a>
Calls per minute (no more than)	60
Current weather API	✓
5 days/3 hour forecast API	✓
16 days/daily forecast API	-
Weather maps 2.0: Current, Forecast, Historical <sup>NEW</sup>	-
Relief maps <sup>NEW</sup>	-
Weather maps 1.0	✓
Bulk download	-
UV index	✓
Weather alerts	✓





# Activity

## Getting Temperature Data of a Location

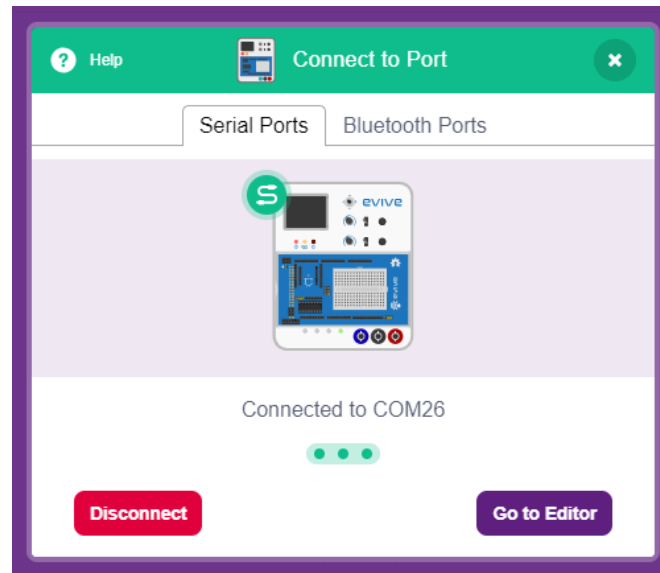
- 



# Programming the evive

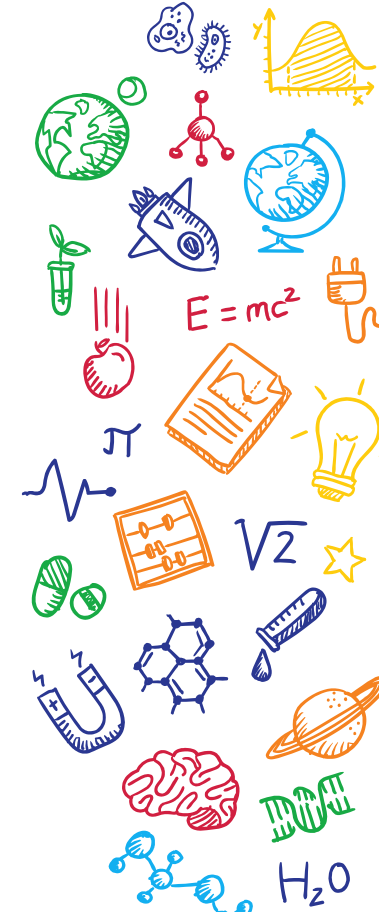


- 
- A collection of colorful, hand-drawn icons representing various fields of science and technology. The icons include: a green globe with continents, a small green circle, a blue microorganism, a yellow line graph with a peak, a red molecular structure, a blue globe with continents, a green test tube with a plant, a blue fish-like robot, a blue power plug, a red apple, the equation
- $E=mc^2$
- , a yellow lightbulb, a blue pi symbol, a blue heartbeat line, a blue abacus, a blue square root symbol
- $\sqrt{2}$
- , a green pill, a blue hexagonal molecular structure, a blue syringe, a blue horseshoe magnet, a blue brain, a blue DNA helix, a blue water molecule
- $H_2O$
- , a blue star, and a blue planet with a ring.



# PictoBlox Program

- Add Internet of Things extension in PictoBlox by clicking on the add extension button on the bottom left corner.



- connect to Wifi  with password

- get weather data for latitude 23.02 & longitude 23.02 with API your API



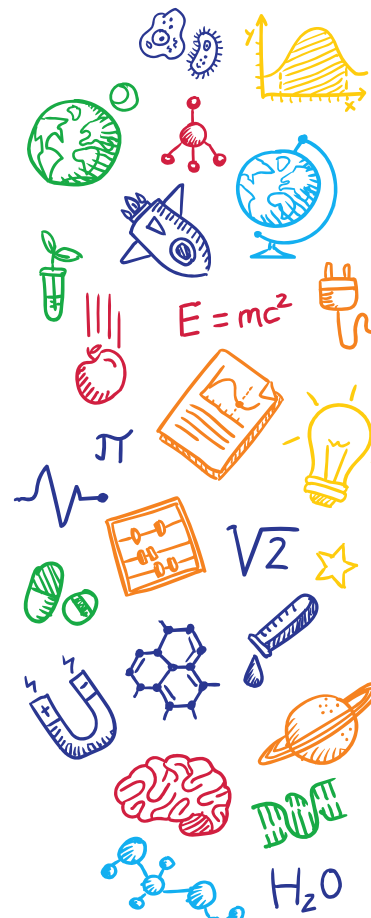
- get **Temperature in Celcius** data





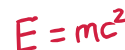
# Internet of Things Extension

- **get (string) data:** The block reports the weather data specified in the input:
  - weather,
  - weather description,
  - country code,
  - city name,
  - time of captured data,
  - sunrise time and
  - sunset time.



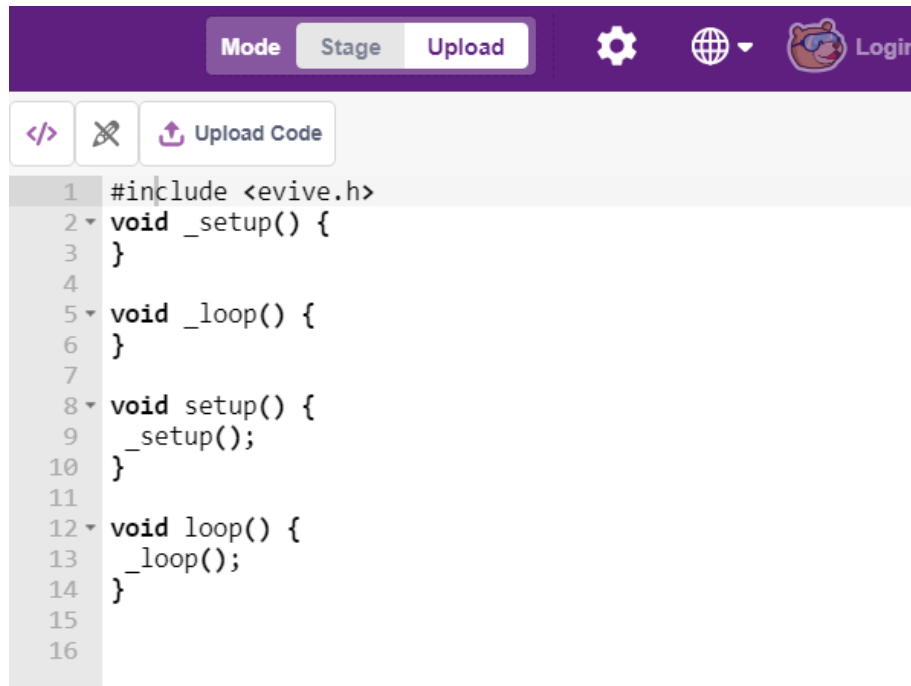
- 
- The image shows a Scratch script designed to connect to a Wi-Fi network and fetch weather data. The script is as follows:
- ```

when green flag clicked
  when environment starts up
  fill screen with black color
  connect to Wifi Wifi Name with password Password
  forever loop
    get weather data for latitude 23.02 & longitude 23.02 with API your API
    set cursor at 10, 10
    write join Temperature: join get Temperature in Celcius data C
    set cursor at 10, 10
    write join Runrise Time: join get Sun Rise Time data
  
```
- Annotations and Callouts:
- Connect to Wifi:** A callout box explains that "Wifi Name" should be replaced with the name of your Wi-Fi and "Password" should be replaced with your Wi-Fi password.
  - Get weather data:** A callout box provides instructions: 1. Register a free account on <https://openweathermap.org/>, 2. Go to your API and copy paste the API in the block, 3. To retrieve weather data of a particular location, get the latitude and the longitude from Google and write it in the block. The block then stores all the data locally.
  - Temperature in Celcius:** A callout box explains that this block allows you to get number data from local variables like: 1. Temperature in Celcius, 2. Latitude, 3. Longitude, 4. Temperature in Farenhiet, 5. Humidity, 6. Visibility, 7. Wind Speed, 8. Wind Direction, 9. Clouds.
  - Sun Rise Time:** A callout box explains that this block allows you to get string data from local variables like: 1. Weather, 2. Weather Description, 3. Country Code, 4. City Name, 5. Data Capture Time, 6. Sun Rise Time, 7. Sun Set Time.



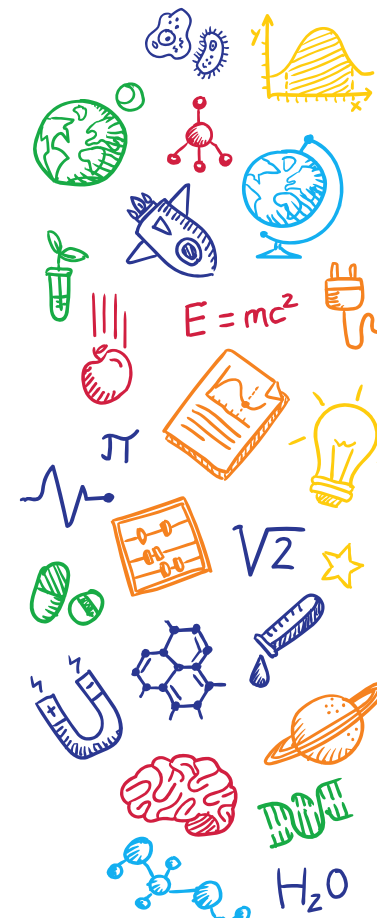
# PictoBlox Script

- Upload the code onto evive by clicking on the Upload Code Button:



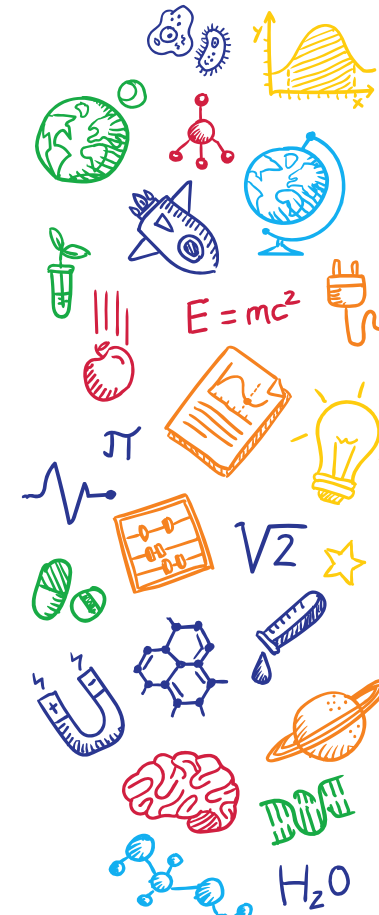
```

1 #include <evive.h>
2 void _setup() {
3 }
4
5 void _loop() {
6 }
7
8 void setup() {
9   _setup();
10 }
11
12 void loop() {
13   _loop();
14 }
15
16
  
```



# Receiving Data

- When you start your evive, you will see two things happening:
- M1 LED glowing:
  - Yellow: Connected to Wi-Fi.
  - Red: Not connected to Wi-Fi. Check if your Wi-Fi name and password are correct.
- M2 LED glowing:
  - Yellow: Connected to ThingSpeak.
  - Red: Not connected to ThingSpeak.
- Once both the LEDs are Yellow, you will see the requested data on the evive Screen.



THANK  
YOU

