



**G's ACADEMY**  
**TOKYO**

# Web API



# Web API 沢山あります

私達の作ったアプリとWeb上に公開されているWeb APIを  
Request/Responseでデータの送受信をおこない、  
必要なデータ情報を利用することが可能です

# Mapsサービスサイト



こんなMapサービスや・・・

# ぐるなび API

[新規アカウント発行](#)>利用規約  
>設定変更

[ぐるなび Web サービス](#)    [はじめての方へ](#)    [API 仕様](#) ▾    [API テストツール](#)    [API 利用ルール](#)    [FAQ](#)

## 新しいアイデア、発信しませんか？

ぐるなびWebサービスは、ぐるなびが所有する豊富な飲食店情報を提供する、無料のAPIです。

ぐるなびWebサービスは、新たなコンテンツを制作する開発者をサポートします。

[はじめての方へ](#)

# リアルタイム データベース API

chatのようなリアルタイム通信やMessage通知が可能



Firebase はモバイルアプリの改善とビジネスの成長に貢献します。

[スタートガイド](#) [動画を見る](#)

アプリをすばやく作成、インフラストラクチャの管理

Google のインフラが支える、多数の人気アプリが信頼

連係する全プロダクトを1つのコンソールで管理

# VideoChat API

chatから音声電話、ビデオ電話を簡単に使用可能



ホーム 開発者 料金 パートナー サポート

新規登録 ログイン English



リアルタイムコミュニケーションで  
イノベーションを起こそう

アプリやWebサイト、IoTデバイスにビデオ通話とP2P通信を追加できるAPI

# Microsoft AI API

## Cognitive Services

自然なコミュニケーション手段を通して、見たり、聞いたり、話したり、理解したり、ユーザーのニーズを解釈したりできるインテリジェントなアルゴリズムを、アプリや Web サイト、ボットに取り入れましょう。AI で貴社のビジネスを今すぐ変革しましょう。

[Cognitive Services を無料で試す >](#)

Cognitive Services についてさらに詳しく: [ディレクトリ](#) [価格](#) [ドキュメント](#)

## AI を使用してビジネスの問題を解決しましょう

### 視覚



イメージ処理アルゴリズムで、写真をスマートに識別したり、キャプションを追加したり、モーデレートしたりできます。

### 音声



音声をテキストに変換したり、確認に音声を使用したり、アプリに話者の認識機能を追加したりできます。

### 知識



インテリジェントなお勧め機能やセマンティック サーチといったタスクを解決するために、複雑な情報やデータをマッピングします。

### Search



Bing Search APIs をアプリに追加して、1 つの API 呼び出しで何十億という Web ページやイメージ、ビデオ、ニュースをくまなく調べる機能を実装しましょう。

### 言語



アプリが、事前に構築済みのスクリプトで自然言語を処理したり、センチメントを評価したり、ユーザーが望んでいることの認識方法を学習したりできます。

# 画像認識 API

画像から背景、人、場所など様々な情報を取得

Home

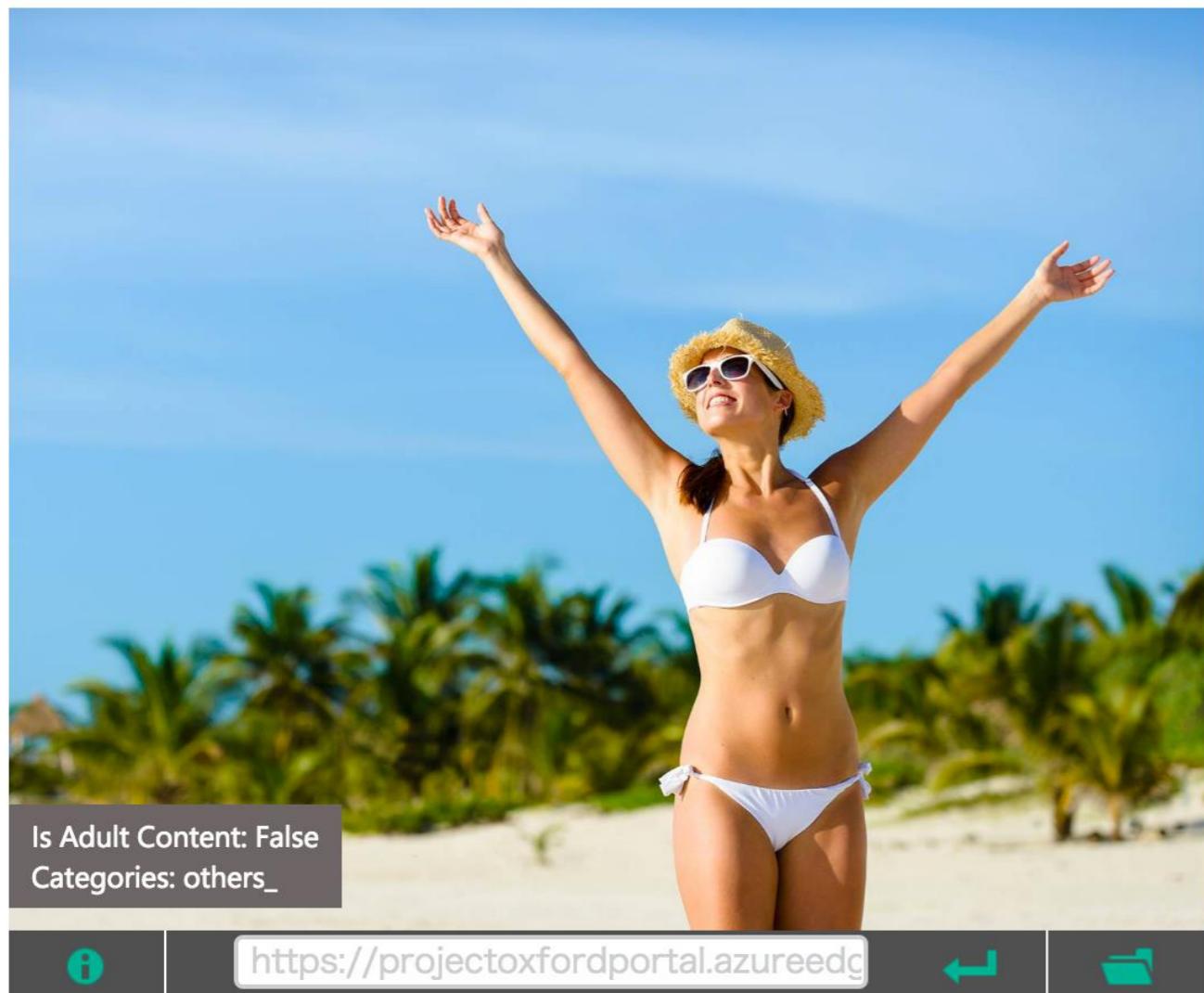
APIs ▾

Applications

Developers ▾

Pricing

My account



## Features:

Feature Name	Value
Description	{ "type": 0, "captions": [ { "text": "a beautiful woman standing on a beach", "confidence": 0.679803189466983 } ] }
Tags	[ { "name": "sky", "confidence": 0.999806821346283 }, { "name": "outdoor", "confidence": 0.9973123073577881 }, { "name": "woman", "confidence": 0.9627310633659363 }, { "name": "person", "confidence": 0.9388522505760193 }, { "name": "beach", "confidence": 0.9086331725120544 }, { "name": "swimsuit", "confidence": 0.5581952929496765 }, { "name": "beautiful", "confidence": 0.5157498717308044 } ]
Image Format	jpg
Image Dimensions	1500 x 1155
Clip Art Type	0 Non-clipart
Line Drawing Type	0 Non-LineDrawing
Black & White Image	Unknown

# 感情認識 API

画像から顔の場所を判定し、感情を数値化する

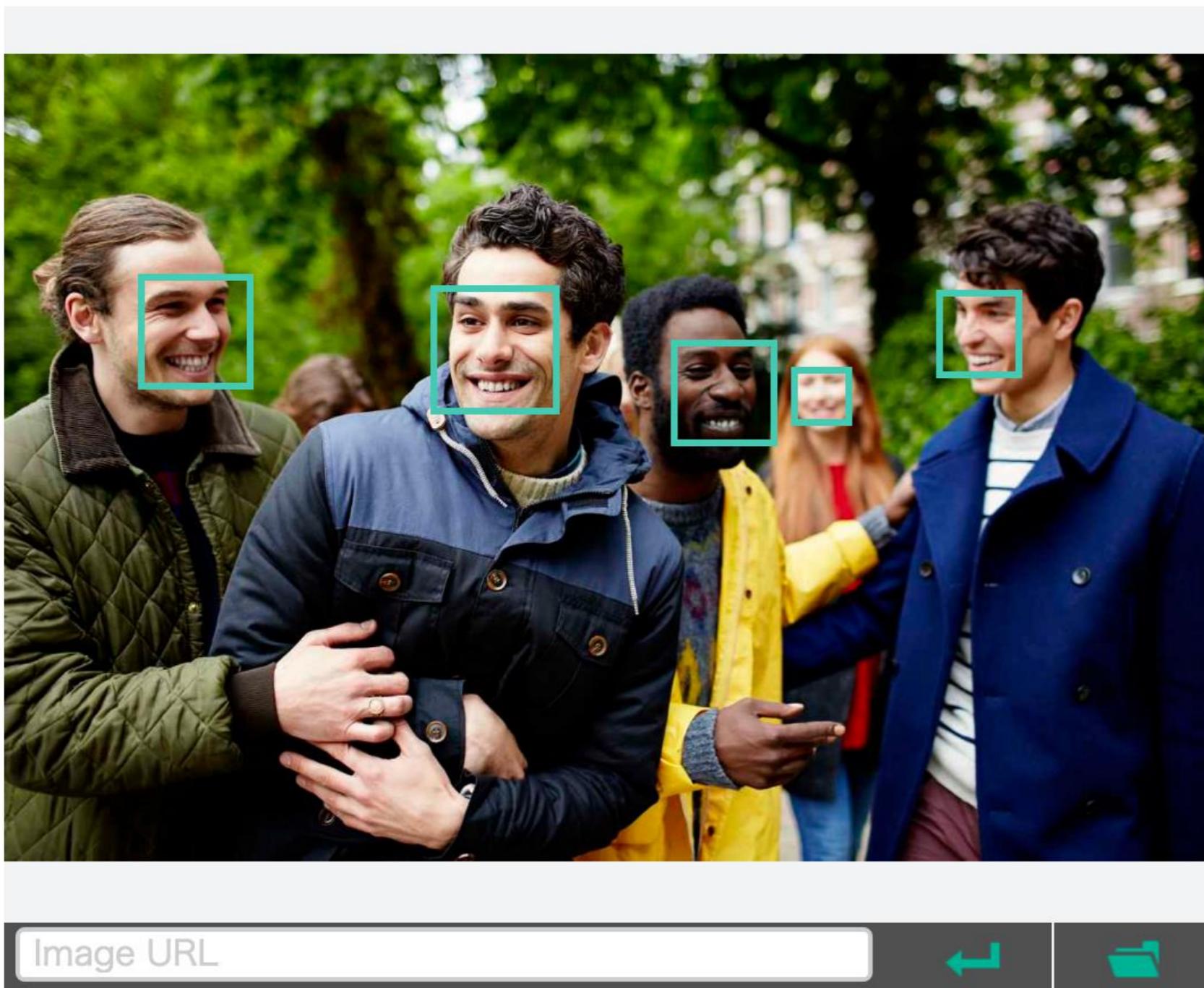
Home

APIs ▾

Applications

Developers ▾

Pricing



The image shows a group of four young men outdoors, smiling and laughing. Their faces are detected by the API and highlighted with green bounding boxes. The men are wearing various jackets: a green quilted jacket, a blue and black拼接 jacket, a yellow jacket, and a dark blue coat.

Image URL  ⬅ ➡

Detection Result:  
5 faces detected

JSON:

```
[  
  {  
    "faceRectangle": {  
      "left": 488,  
      "top": 263,  
      "width": 148,  
      "height": 148  
    },  
    "scores": {  
      "anger": 9.075572e-13,  
      "contempt": 7.048959e-9,  
      "disgust": 1.02152783e-11,  
      "fear": 1.778957e-14,  
      "happiness": 0.9999999,  
      "neutral": 1.31694478e-7,  
      "sadness": 6.04054263e-12,  
      "surprise": 3.92249462e-11  
    }  
  },  
  {
```

# 顔認識 API

画像から顔、目、鼻、口など複数のポイントあらゆる情報を取得可能

Home

APIs ▾

Applications

Developers ▾

Pricing



Detection Result:

JSON:

```
[  
  {  
    "faceId": "80ec7366-da66-4413-b463-5631e73b813d",  
    "faceRectangle": {  
      "width": 68,  
      "height": 68,  
      "left": 134,  
      "top": 47  
    },  
    "faceLandmarks": {  
      "pupilLeft": {  
        "x": 148.5,  
        "y": 73  
      },  
      "pupilRight": {  
        "x": 178.7,  
        "y": 62.1  
      },  
      "noseTip": {  
        "x": 169.2,  
        "y": 83  
      },  
      "mouth": {  
        "x": 169.2,  
        "y": 103  
      }  
    }  
  }]
```



# 多言語翻訳 API

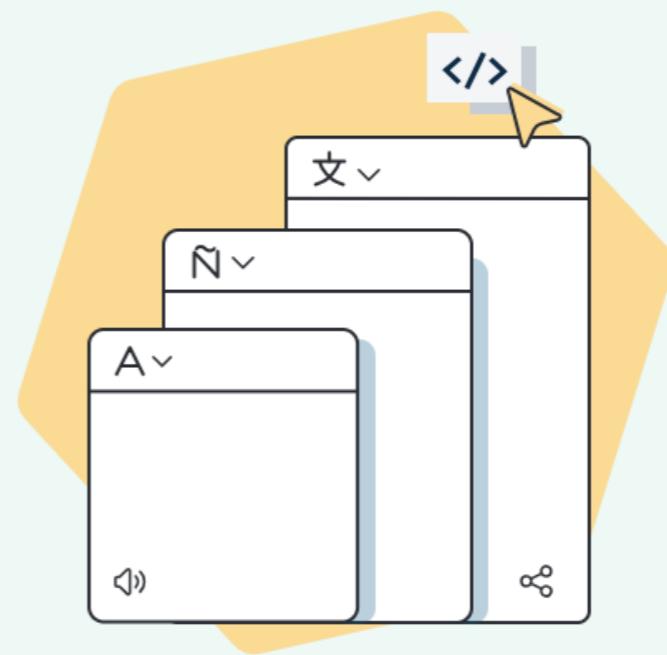


## DeepLのAPIを 使い多言語環境を構築

クラス最高の翻訳、カスタマイズ性、万全のセキュリティ対策

無料で始める

技術資料を読む



# Gemini AI API

ドキュメント API リファレンス クックブック Google AI Studio プロンプト ギャラリー 料金

Gemini 1.5 Flash の値下げ、すべてのデベロッパー向けのファインチューニング アクセスなど

詳細

# Gemini API で構築する

Google の最大かつ最も高性能な AI モデルをアプリに簡単に統合

<ph type="x-smartling-placeholder">

</ph>Google AI Studio で API キーを取得する

をご覧ください。

API ドキュメントを読む

<ph type="x-smartling-placeholder"></ph>  Google Cloud のエンタープライズ対応 AI をお試しください >

[Google AI Gemini API](#) | [Google AI Studio](#) | [Google for Developers](#) | [Google AI for Developers](#)

<https://ai.google.dev/gemini-api?hl=ja>

# 天気 API



Q Weather in your city

Guide API Dashboard Marketplace Pricing Maps Our Initiatives Partners Blog For Business Sign in Support▼

## OpenWeather

Weather forecasts, nowcasts and history in a fast and elegant way

Search city

Search



Different Weather?

Metric: °C, m/s

Imperial: °F, mph

Aug 15, 08:56am

**London, GB**

18°C

Feels like 17°C. Broken clouds. Gentle Breeze

5.1m/s SW 1016hPa

Humidity: 67% UV: 2

Dew point: 12°C Visibility: 10.0km



### Hourly forecast

8am 9am 10am 11am 12pm 1pm 2pm 3pm 4pm

30°

### 8-day forecast

Thu, Aug 15

24 / 17°C

light rain

Fri, Aug 16

26 / 18°C

clear sky

<https://openweathermap.org/api>

# APIを使用するメリット

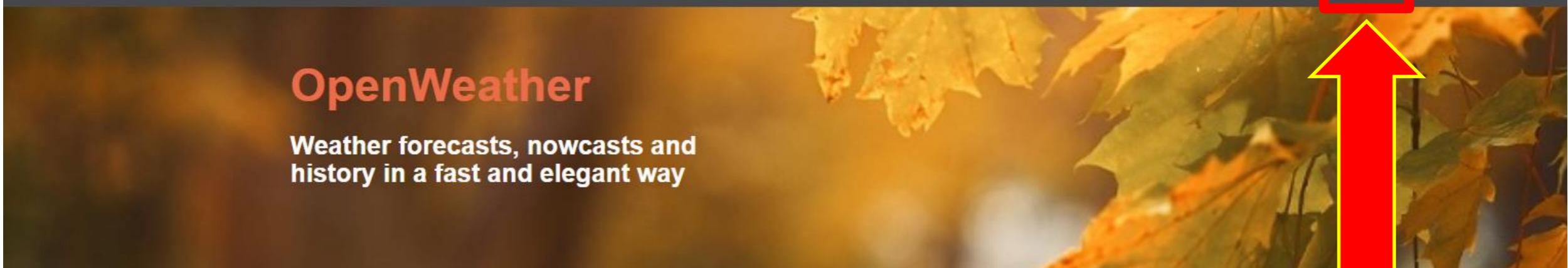
- ◇自分たちで全てを作らない。
- ◇外部サービスを利用してすることで素早く開発

プロトタイプを作成するには必須の知識です。



# OpenWeather

天氣API



# OpenWeather

Weather forecasts, nowcasts and history in a fast and elegant way

Search city

Search



Different Weather?

Metric: °C, m/s

Imperial: °F, mph

Nov 14, 08:32am

## London, GB

 10°C

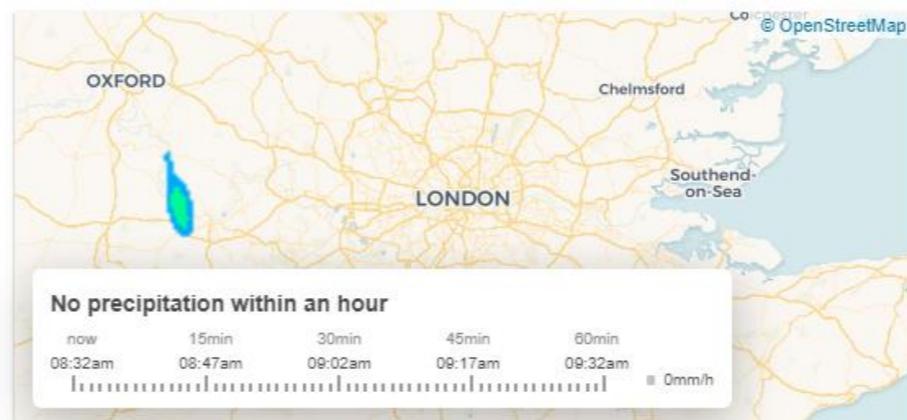
Yellow fog warning

Feels like 10°C. Fog. Gentle Breeze

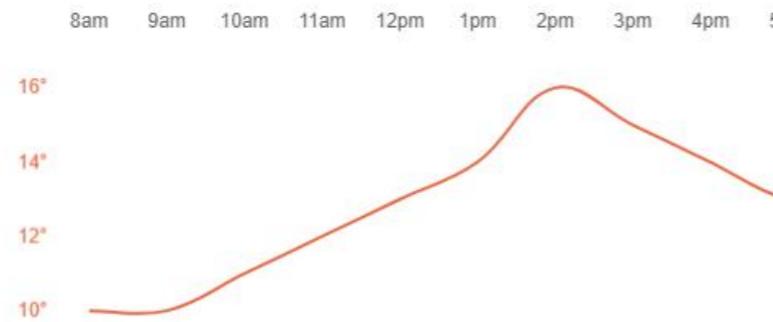
◀ 3.6m/s E ⚛ 1009hPa

Humidity: 97% UV: 0

Dew point: 10°C Visibility: 0.3km



### Hourly forecast



### 8-day forecast

Mon, Nov 14	 16 / 10°C	broken clouds	▼
Tue, Nov 15	 13 / 9°C	moderate rain	▼
Wed, Nov 16	 13 / 8°C	moderate rain	▼
Thu, Nov 17	 13 / 9°C	moderate rain	▼
Fri, Nov 18	 10 / 7°C	few clouds	▼



Q Weather in your city

Guide API Dashboard Marketplace Pricing Maps Our Initiatives Partners Blog For Business Sign In Support▼

### Sign In To Your Account

Enter email

Password

Remember me

Not registered? [Create an Account.](#)

Lost your password? [Click here to recover.](#)

Not registered? [Create an Account.](#)

### Create New Account

1

······

······

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 [Website terms and conditions of use](#)

I consent to receive communications from OpenWeather Group of Companies and their partners:

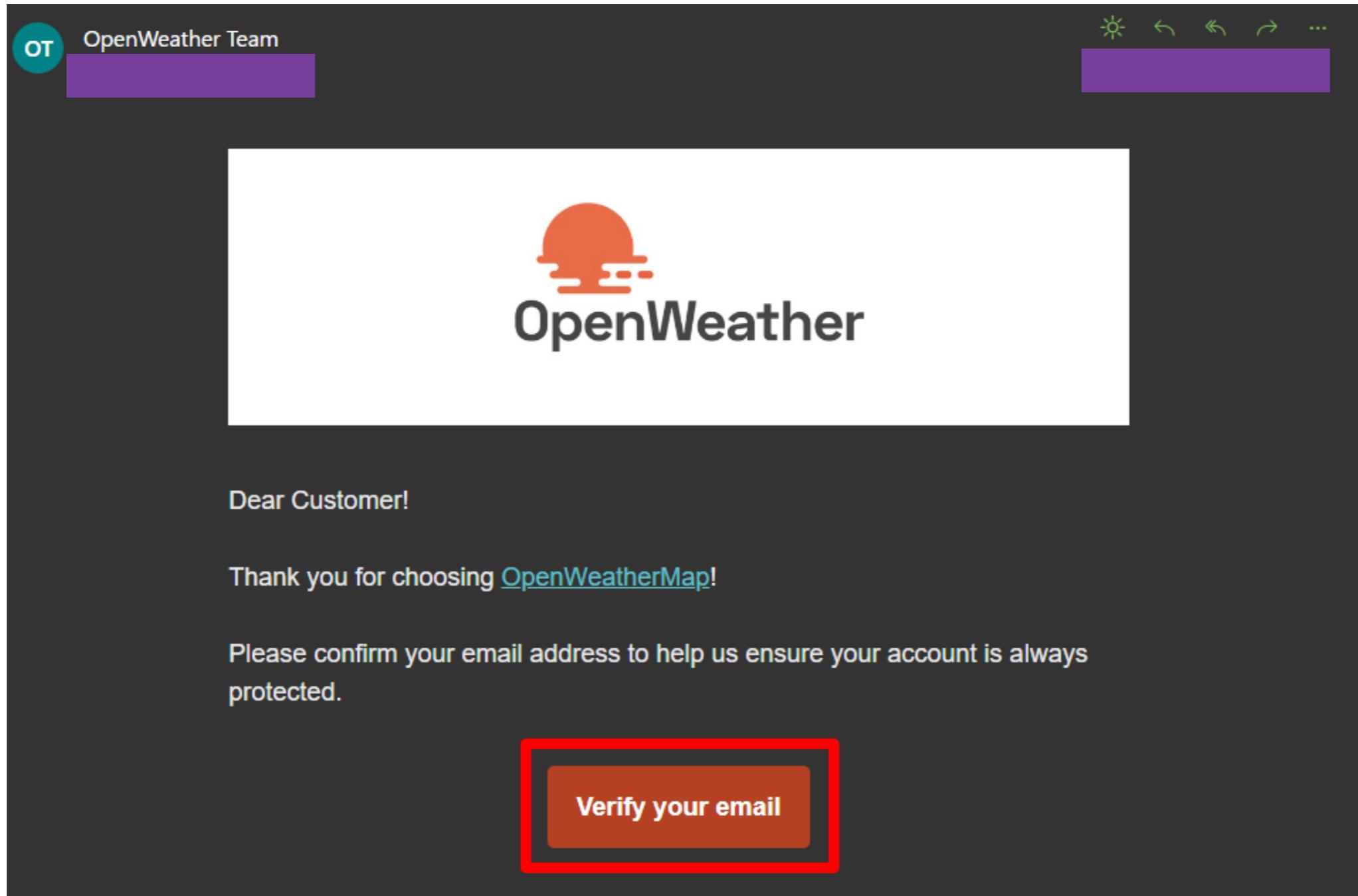
System news (API usage alert, system update, temporary system shutdown, etc)

Product news (change to price, new product features, etc)

Corporate news (our life, the launch of a new service, etc)

私はロボットではありません  
 reCAPTCHA  
プライバシー・利用規約

Create Account





Q Weather in your city

Guide API Dashboard Marketplace Pricing Maps Our Initiatives Partners Blog For Business yam... Support

New Products Services API keys Billing plans Payments Block logs My orders My profile Ask a question

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

Key	Name	Status	Actions
[REDACTED]	Default	Inactive	

Create key

Generate

① My API keys

② keyをコピー

KEY取得完了

# 次にAPIページへ



A screenshot of a web browser showing the OpenWeatherMap API landing page. The URL in the address bar is <https://openweathermap.org/api>. The page features a dark header with the OpenWeather logo, a search icon, and navigation links for Products, Dashboard, Pricing, Industries, Customer stories, Blog, Technology, About us, Careers, Contact Us, and Sign In. The main content area has a light background. It starts with a message encouraging users to sign up for their fast and easy-to-work weather APIs. Below this, a section for the One Call API 3.0 is shown, featuring a large orange "Subscribe" button. A list of features includes current weather forecasts, historical weather data, daily aggregation, and weather overviews. To the right, a box highlights a free plan of 1,000 API calls per day and 0.0012 GBP per API call over the limit. A red box highlights the first step: "1. 以下URLへアクセス (Slackで共有します)".

## Weather API

[Home](#) / Weather API

Please, [sign up](#) to use our fast and easy-to-work weather APIs. As a start to use OpenWeather products, we recommend our [One Call API 3.0](#). For more functionality, please consider our products, which are included in [professional collections](#).

### One Call API 3.0

[API doc](#)

[Subscribe](#)

Pay as you call

Make an API call to receive access to the various data:

- [Current weather and forecasts](#):
  - minute forecast for 1 hour
  - hourly forecast for 48 hours
  - daily forecast for 8 daysand government weather alerts
- [Weather data for any timestamp](#) for 45 years historical archive and 4 days ahead forecast
- [Daily aggregation](#) of weather data for 45 years archive and 1.5 years ahead forecast
- [Weather overview](#) with today and tomorrow's fo

[1,000 API calls per day for free](#)

[0.0012 GBP](#) per API call over the daily limit

[Subscribe](#)

This is a separate subscription plan, which includes only One Call API.

#### 1. 以下URLへアクセス (Slackで共有します)

Read more about this API and

<https://openweathermap.org/api>

OpenWeat  
Meteorological

# 5 Day / 3 Hour

<https://openweathermap.org/api>

The screenshot shows the OpenWeatherMap API landing page. A red box highlights the '5 Day / 3 Hour Forecast' section. Inside this box, a red line connects the 'API doc' button to the '5 Day / 3 Hour Forecast' heading. Another red line connects the '5 Day / 3 Hour Forecast' heading to the 'Road Risk API' heading.

**stations**

- JSON, XML, and HTML formats
- Included in both free and paid subscriptions

**Climatic Forecast 30 days**

[API doc](#) [Subscribe](#)

- Forecast weather data for 30 days
- JSON format

**Bulk Download**

[API doc](#) [Subscribe](#)

- Current weather, a variety of weather forecasts and their 7-day archive via CSV and JSON formats
- Included in the Professional and Enterprise subscription plans

**Global Weather Alerts**

**Push notifications**

[Doc](#) [Subscribe](#)

- Get all the **warnings from national weather agencies**
- Weather alerts are pushed to your endpoint as soon as they occur
- Data feed provides all active weather alerts from the entire world
- Each alert contains date, time, location, and detailed description
- **Monthly subscription.** Please [contact us](#) to get access.

**1. API一覧ページ表示**

**2. Call 5 day / 3hour の「API doc」をクリック**

**5 Day / 3 Hour Forecast**

[API doc](#) [Subscribe](#)

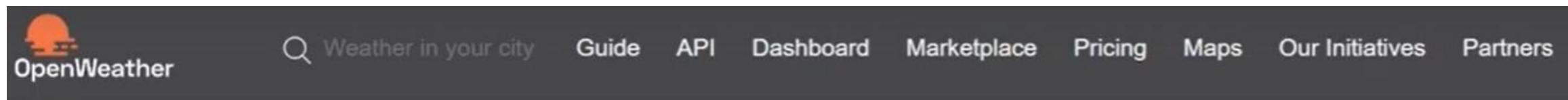
- 5 day forecast for any location on the globe
- 5 day forecast with a 3-hour step
- JSON and XML formats
- Included in both free and paid subscriptions

**Road Risk API**

[Doc](#) [Get a quote](#)

- Specify your route and get weather data and national alerts for the point of destination and along the route
- Current, forecast for 5 days and historical weather data for 1 year for your route
- Weather data are available for any point on the globe
- To receive information on price and get

# API call / URL ①



5 day weather forecast

5 day forecast is available at any location on the globe. You can get 5 day forecast data with 3-hour step. Forecast is available in JSON or XML format.

Call 5 day / 3 hour forecasts data

How to make an API call

You can search weather forecast for 5 days with data every 3 hours by geographic coordinates. All weather data can be obtained in JSON and XML formats.

**API call**

```
api.openweathermap.org/data/2.5/forecast?lat=[lat]&lon=[lon]&appid=[API key]
```

1. Call 5 day / 3hour …が表示されます  
2. API call のURLで情報を送受信します

Call 5 day / 3 hour forecasts data

How to make an API call

Other features

Geocoding API

Built-in geocoding

Built-in API

Built-in API

Format

Limitation of results

Units of measurement

Multilingual support

Call 5 day / 3 hour forecasts data

How to make an API call

Other features

Geocoding API

Built-in geocoding

Built-in API

Built-in API

Format

Limitation of results

Units of measurement

Multilingual support

# API call / URL ②

API call

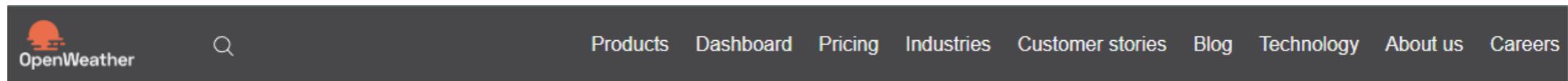
api.openweathermap.org/data/2.5/forecast?lat=[Lat]&lon=[Lon]&appid=[API key]

1. このURLが重要
2. [lat],[lon],[API key]をURLで渡す

coordinates (latitude, longitude). If you need the geocoder to automatically convert city names and zip-codes to geo coordinates and the other way around, please use our [Geocoding API](#).

appid	required	Your unique API key (you can always find it on your account page under the "API key" tab)
<b>lat</b>	optional	緯度
units		Units of measurement. standard, metric and imperial units are available. If you do not use the units parameter, units will be applied by default. <a href="#">Learn more</a>
<b>lon</b>	optional	軽度
mode	optional	Response format. JSON format is used by default. To get data in XML format, set mode to XML. <a href="#">Learn more</a>
<b>appid</b>	optional	自分のKEY

# API call / URL ③



## API response

If you do not see some of the parameters in your API response it means that these weather phenomena are just not happened for the time of measurement for the city or location chosen. Only really measured or calculated data is displayed in API response.

### JSON

#### JSON format API response

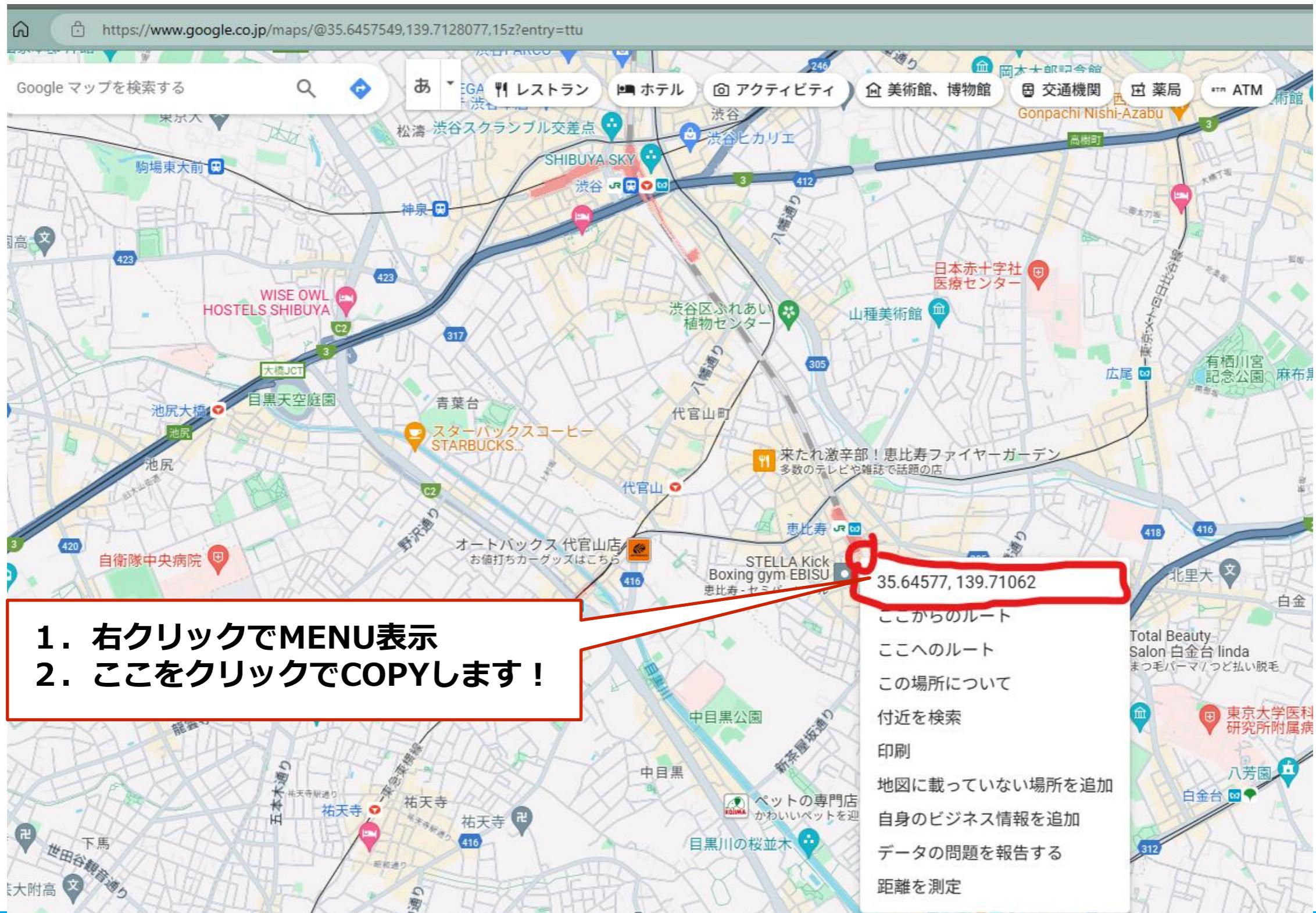
```
{  
  "cod": "200",  
  "message": 0,  
  "cnt": 40,  
  "list": [  
    {  
      "dt": 1661871600,  
      "main": {  
        "temp": 296.76,  
        "feels_like": 296.98,  
        "temp_min": 296.76,  
        "temp_max": 297.87,  
        "pressure": 1015,  
        "sea_level": 1015,  
        "grnd_level": 933,  
        "humidity": 69,  
        "temp_kf": -1.11  
      },  
      "weather": [  
        {  
          "id": 500,  
          "main": "R  
        }  
      ]  
    }  
  ]  
}
```

1. 「API response」
2. JSON (受信データ) 内訳が書いてある  
※ここは後でみていきましょう！ \*\*重要

Product concept  
Call 5 day / 3 hour forecast data  
How to make an API call  
API response  
JSON format API response example  
JSON format API response fields  
XML format API response example  
XML format API response fields  
List of weather condition codes  
Min/max temperature in current weather  
API and forecast API  
Bulk downloading

Units of measurement  
Multilingual support  
Call back function for JavaScript code

# Google Map/緯度経度取得



# VS Code / weather.html を開く

Open Weather API KEYを張る

```
33 <body>
34   <div>
35     <h1>
36       <span id="area"></span> 天気5日間!
37     </h1>
38     <div id="view"></div>
39   </div>
40
41   <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.1/jquer
42   <script>
43     //=====
44     //MAP表示処理
45     //=====
46     const key    = "";                                //***自分のKEY
47     const lat   = "35.713456";           //原宿の緯度
48     const lon   = "139.663682";           //原宿の経度
49     const icon  = "http://openweathermap.org/img/wn/"; //ICONにアクセス
```

1. ここに取得した「KEY」を貼り付けます！

# VS Code / weather.html を開く

## 表示領域、緯度経度の確認

```
33 <body>
34   1. id="view" にJSで表示していきます !
35
36     <span id="area"></span> 天気5日間 !
37   </h1>
38   <div id="view"></div>
39 </div>
40
41 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.1/jquer
42 <script>
43   //=====
44   //MAP表示処理
45   //=====
46   const key    = " ";                                // ***自分のKEY
47   const lat    = "35.713456";   //原宿の緯度
48   const lon    = "139.663682"; //原宿の経度
49   const icon   = "http://openweathermap.org/img/wn/"; //ICONにアクセス
```

# VS Code / weather.html

## URLを確認 (lat,lon,key)



weather.html weather\_complete.html X

map\_weather > weather\_complete.html > html > body > script > key

```
43 //=====
44 //MAP表示処理
45 //=====
46 const key    = ""; //***自分のKEYを!!!!!!!!!!!!!! ***
47 const lat   = "35.713456"; //原宿の緯度
48 const lon   = "139.663682"
49 const icon  = "http://ope
50 //=====
51 //OpenWeatherAPI (天気情報を取得) 5DAY/3hour
52 //=====
53
54 const url = 'https://api.openweathermap.org/data/2.5/forecast?lat=' + lat + '&lon=' + lon + '&appid=' + key + '&lang=ja';
55 $.ajax({
56     url:url,
57     type:'get',
58     cache:false,
59     dataType:'json'
60 }).done(function(data){
```

1. このURLで「lat/lon/key」情報をサイトに渡します

# VS Code / weather.html

## ajaxのブロックを記述

```
51 //=====
52 //OpenWeatherAPI (天気情報を取得) 5DAY/3hour
53 //=====
54 const url = 'https://api.openweathermap.or
55 $.ajax({
56
57 }).done(function(data){
58
59 });
60 
```

このブロックを追加！！

```
$ajax({
  //送受信情報
}).done(function(data)){
  //送受信成功時この中が処理される
});
```

# VS Code / weather.html

## ajax送受信情報を記述

```
51 //=====
52 //OpenWeatherAPI (天気情報を取得) 5DAY/3hour
53 //=====
54 const url = 'https://api.openweathermap.org/data/2.5/
55 $.ajax({
56   url:url,
57   type:'get',
58   cache:false,
59   dataType:'json'
60 }).done(function(data){
61   // console.log(data.list[0].dt_txt); //オブジェクト確認
62   console.log(data); //オブジェクト変数を確認
63
64 });

65
```

- 1. url: “送受信先サイトのURL”
- 2. type: “get” or “post” //今回は“get”
- 3. cache: false or true //キャッシュしないfalse
- 4. dataType: “json” //データ形式“json”

# VS Code / weather.html

data変数の中を確認！

```
51 //=====
52 //OpenWeatherAPI (天気情報を取得) 5DAY/3hour
53 //=====
54 const url = 'https://api.openweathermap.org/data/2.5/
55 $.ajax({
56   url:url,
57   type:'get',
58   cache:false,
59   dataType:'json'
60 }).done(function(data){
61   // console.log(data.list[0].dt_txt); //オブジェクト
62   console.log(data); //オブジェクト変数を確認
63
64 });

65
```

} ).done(function(**data**){  
 //変数**data**にデータがマルッと入ってきます！  
 console.log(**data**); //で確認しましょう！  
});

// console.log(data.list[0].dt\_txt); //オブジェクト

console.log(**data**); //オブジェクト変数を確認

# VS Code / weather.html

## 表示文字列を作成

```
64 let html="";
65 for(let i=0; i<data.list.length; i++){
66     html += [ ]—— [Shift + @]で`バッククオート`で括る!!
67     <div>
68         <div>時刻 : ${data.list[i].dt_txt}</div>
69         <div>風速 : ${data.list[i].wind.speed}</div>
70         <div>天気 : ${data.list[i].weather[0].description}</div>
71         <div>降水確率 : ${data.list[i].pop*100}</div>
72         <div></div>
73     </div>
74 };—— [Shift + @]で`バッククオート`で括る!!
75 }
76 //作成したHTML埋め込む
77 $("#view").append(html);
```

バッククオートの場合 **`\${変数名}`** で埋め込み!!

# VS Code / weather.html

## 作成した文字列を表示

```
64 let html="";
65 for(let i=0; i<data.list.length; i++){
66     html += [
67         <div>
68             <div>時刻 : ${data.list[i].dt_txt}</div>
69             <div>風速 : ${data.list[i].wind.speed}</div>
70             <div>天気 : ${data.list[i].weather[0].description}</div>
71             <div>降水確率 : ${data.list[i].pop*100}</div>
72             <div></div>
73     </div>
74 ];
75
76 }
77 //作成したHTML埋め込む
78 $("#view").append(html);
```

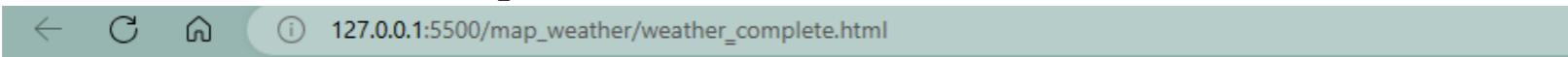
作成した文字列を  
HTML: id="view"の場所に  
「append」で表示追加！

# VS Code / weather.html

## 地名を表示

```
75 //取得したデータを表示する
76 //緯度経度の場所を「地名」で取得
77 //作成したHTML埋め込む
78 //$( "#view" ).append( html );
79 //場所
80 //$( "#area" ).html( data.city.name )
81 });
82
83 );
```

# 完成！！APIはデータ取得して どう料理するか？が大事!!



## 阿佐ヶ谷 天気5日間！

時刻：2024-08-19 09:00:00

風速：2.9

天気：小雨

降水確率：48



時刻：2024-08-19 12:00:00

風速：0.42

天気：小雨

降水確率：64



時刻：2024-08-19 15:00:00

風速：4.46

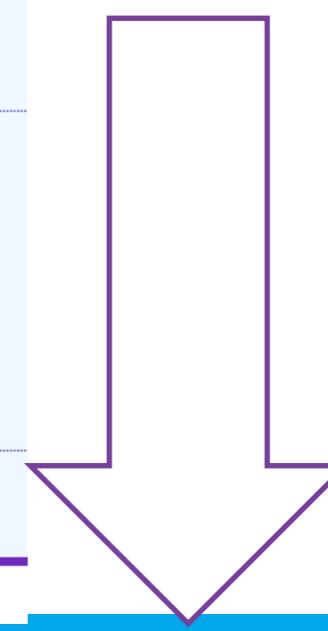
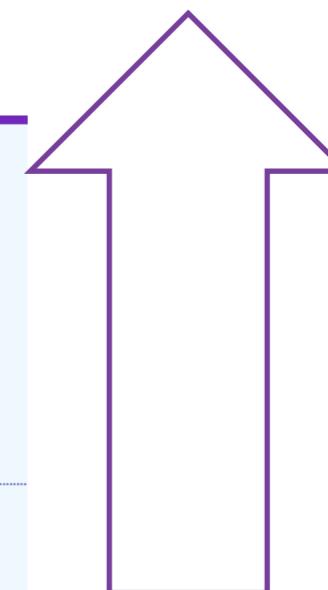
天気：小雨

降水確率：97



時刻：2024-08-19 18:00:00

風速：3.84



# Google Map API サンプル



<https://findout.azurewebsites.net/Gmap.html>

# 課題

Web APIを  
使ったなにか？

わからないとき！手を動かしてやってみる！

# Google Books API チャレンジしてみたい人へ

GoogleBooks API  
[https://youtu.be/Yo0gz7DN2xM?si=\\_NgsH7ojX1Z\\_XHeu](https://youtu.be/Yo0gz7DN2xM?si=_NgsH7ojX1Z_XHeu)

サンプルファイル `google_books.html`

以下時間があれば

# Geolocation

geolocation.html

# geolocationとは

---

## ◇ geolocation

位置情報を取得することができるAPIです。

geolocationがサポートされているブラウザであれば、デバイス関係なしで使用することができます。

座標位置からMAPと連携や、ピンポイント天気予報の作成、  
ローカルニュースなど位置情報からサービスを広げることも可能です

サポート状況の確認

<http://caniuse.com/#feat=geolocation>

# geolocationとは

---

## ◇ 位置を取得

位置を取得する方法が2つあります。

- ・現在の位置を取得する

`navigator.geolocation.getCurrentPosition(成功,失敗,option)`

- ・位置を定期的に取得する

`navigator.geolocation.watchPosition(成功,失敗,option)`

成功時には「position」オブジェクトを取得します

取得内容は次ページの通りです。

## ◇ 取得停止

定期的に取得する「watchPosition」を停止する方法

「watchPosition」は制作時にidを返します。

idを使用して「clearWatch」で停止させます

`navigator.geolocation.clearWatch(watchID)`

# geolocationとは

---

## ◆ positionプロパティ

位置取得が成功すると以下プロパティを取得できます。

プロパティ	説明
position.coords	緯度経度などの情報
position.timestamp	位置を取得した時刻

## ◆ 位置を取得

position.coords で取得した経度緯度の情報について

プロパティ	説明
coords.latitude	緯度
coords.longitude	経度
coords.altitude	高度
coords.accuracy	正確性
coords.altitudeAccuracy	高度の正確性
coords.heading	方位
coords.speed	速度

# geolocationとは

## ◇ optionの指定

3つめの引数では関数を実行する時のオプションを指定します  
指定しなくても可能

プロパティ	説明
option.enableHighAccuracy	より高い正確性を求める(取得時間が遅くなる) <small>L SEP</small> スマフォ ではGPSを利用する (バッテリー消費が早い)
option.timeout	タイムアウトまでの時間
option.maximumAge	位置情報の有効期限

## ◇ 失敗時の取得

失敗時には「errorオブジェクト」を取得します  
取得内容は以下の通りです。

プロパティ	説明
error.code	エラーコード 1.位置情報の取得が許可されていない 2.位置情報の取得が利用できない 3.タイムアウト
error.message	エラーメッセージ