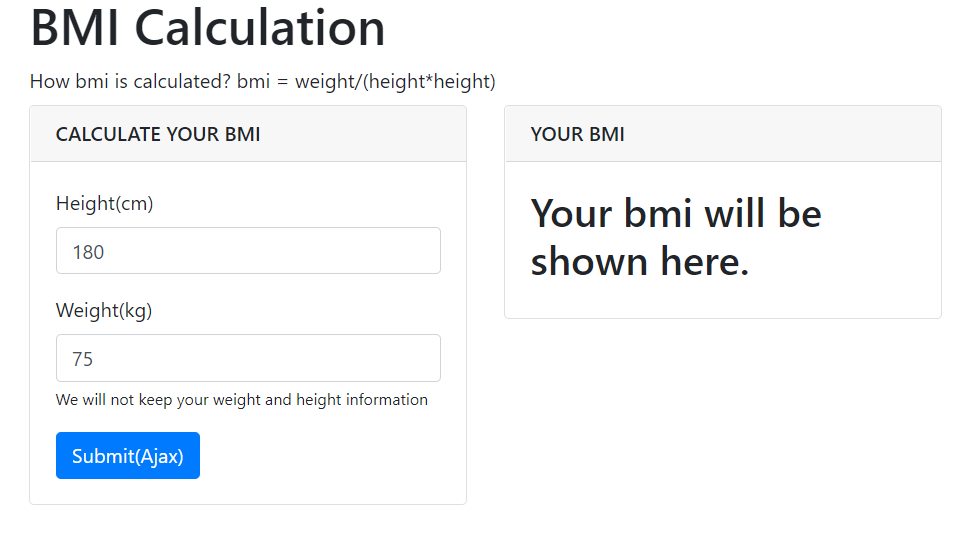
Get BMI via Ajax



Why do we use Ajax?

What is AJAX?

AJAX = Asynchronous JavaScript and XML.

AJAX is a technique for creating fast and dynamic web pages.

AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.

Classic web pages, (which do not use AJAX) must reload the entire page if the content should change.

Examples of applications using AJAX: Google Maps, Gmail, Youtube, and Facebook tabs.

[source](https://www.google.com/search?q=Why+do+we+use+Ajax%3F&oq=Why+do+we+use+Ajax%3F&aqs=chrome..69i57j0i19j0i19i30l3.262j0j7&sourceid=chrome&ie=UTF-8)

Step1: Project configures and App

Make project folder “website\_bmi\_v1” and change directory into it.

django-admin startproject website\_configs .

Create an app:

django-admin startapp app\_bmi\_ajax

or

python manage.py startapp app\_bmi\_ajax

Step2: settings.py

settings.py

|  |
| --- |
| ALLOWED\_HOSTS = ['\*']  INSTALLED\_APPS = [  …..  'app\_bmi\_ajax',  ] |

Step3: urls.py

開放哪一支網頁程式?名稱是甚麼?

urls.py

|  |
| --- |
| from django.contrib import admin  from django.urls import path  #若匯入多個views，必須改名  # from app\_bmi\_ajax import views as views\_bmi\_ajax  from app\_bmi\_ajax import views as views\_ajax  urlpatterns = [  path('', views\_ajax.home), #for homepage  path('api\_get\_bmi/', views\_ajax.api\_get\_bmi), #for ajax  ] |

There are two programs are open to the public. We have to define two routes (paths), namely views.home and views.api\_get\_bmi.

Step4: views.py

伺服器端準備甚麼資料傳送給網頁去呈現?

views.py

Define home()

|  |
| --- |
| *from* django.shortcuts *import* render  *def* home( *request* ):  *return* render(*request*, 'app\_bmi\_ajax/home.html') |

Define API with POST Version

|  |
| --- |
| # Ajax API POST *from* django.http *import* JsonResponse *from* django.views.decorators.csrf *import* csrf\_exempt  @csrf\_exempt *def* api\_get\_bmi(*request*):  *try*:  height = int(*request*.POST.get('height'))  weight = int(*request*.POST.get('weight'))  result = round( weight / pow(height / 100, 2), 2)  print(height, weight)  response = {"yourbmi": result}  *return* JsonResponse(response)  *except*:  print("Not an legal ajax call!")  *return* JsonResponse({"yourbmi": 'error'}) |

Step5: home.html

Where is “home.html” should be placed?

home.html 要置放在哪個目錄?

* site網站底下的app目錄底下的 templates

website\_bmi\_v1/app\_bmi\_ajax/templates/app\_bmi\_ajax/home.html

|  |  |
| --- | --- |
| <!-- home.html --> <!DOCTYPE html> <html lang="en"> <meta charset="utf-8"> <meta name="viewport" content="width=device-width, initial-scale=1"> <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"> <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script> <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script> <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script>  <head>  <title>Get BMI</title> </head>  <body>  <div class="container">  <h1>BMI Calculation</h1>  <label>How bmi is calculated? bmi = weight/(height\*height)</label>   <div class="row">   <!-- row .. col-lg-6 .. card -->  <div class="col-lg-4 mb-5">  <div class="card">  <div class="card-header">  <h3 class="h6 text-uppercase mb-0">Calculate your BMI</h3>  </div>  <div class="card-body">  <!-- write your HTML code here -->  </div>  </div>  </div><!-- end of card-->    <!-- row .. col-lg-6 .. card -->  <div class="col-lg-4 mb-5">  <div class="card">  <div class="card-header">  <h3 class="h6 text-uppercase mb-0">Your BMI</h3>  </div>  <div class="card-body">  <h2 id="yourbmi">Your bmi will be shown here.</h2>  </div>  </div>  </div><!-- end of card -->    </div><!-- row --> </div> <!-- container -->  </body> </html>  <!-- java scrip--> <!-- jQuery js--> <script src="https://code.jquery.com/jquery-3.1.0.min.js"></script> <!-- chartjs s--> <script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.3/Chart.min.js"></script>  <!-- Your Javascript codes --> <script>  //write your JavaScript code here.  </script> |  |

Height 身高的HTML碼

Alt+Shift+F 格式化熱鍵

|  |
| --- |
| <div class="form-group">  <label>Height(cm)</label>  <input class="form-control" id="height" name="height" value="180" placeholder="Height(cm)">  </div> |

Weight

|  |
| --- |
| <div class="form-group">  <label>Weight(kg)</label>  <input class="form-control" id="weight" name="weight" value="75" placeholder="Weight(kg)">  <small>We will not keep your weight and height information</small> </div> |

Submit button with id="btn\_get\_bmi"

|  |
| --- |
| <button type="submit" class="btn btn-primary" id="btn\_get\_bmi">Submit(Ajax)</button> |

Try the following JS code, and see what happens. Press F12

|  |
| --- |
| * Step 1: Get weight and height and display them on console:   let weight = $('#weight').val(); console.log(weight);  console.log($('#height').val())  $('#yourbmi').empty(); $('#yourbmi').append("This is your bmi: 23.5");  jQuery('#yourbmi').append("You are too thin.");   * Step2: on click event for submit button   $('#btn\_get\_bmi').on('click', function () {  alert("You click this button!");  $('#yourbmi').empty();  $('#yourbmi').append("here is your bmi!");  let height = $('#height').val();  let weight = $('#weight').val();  console.log(height);  // Insert your Ajax function Ajax程式碼    });// btn on click event     * Step3: your Ajax function Ajax程式碼   注意:不要放錯位置，必須置放於裡面: $('#btn\_get\_bmi').on('click', function () {}  $.ajax({   type: "POST",  url: "api\_get\_bmi/",  //url: "http://127.0.0.1:8000/api\_get\_bmi/",  data: {  "height": height,  "weight": weight,  }, //pass to server  success: function (*received*) {  // do somethings here  const result = *received*['yourbmi'];  console.log(result);  $('#yourbmi').empty();  $('#yourbmi').append("Your bmi: " + result);   }, //success function  });//ajax |

**Completed code:**

|  |
| --- |
| <!-- home.html --> <!DOCTYPE html> <html lang="en"> <meta charset="utf-8"> <meta name="viewport" content="width=device-width, initial-scale=1"> <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"> <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script> <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script> <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script>  <head>  <title>Get BMI</title> </head>  <body>  <div class="container">  <h1>BMI Calculation</h1>  <label>How bmi is calculated? bmi = weight/(height\*height)</label>   <div class="row">   <!-- row .. col-lg-6 .. card -->  <div class="col-lg-4 mb-5">  <div class="card">  <div class="card-header">  <h3 class="h6 text-uppercase mb-0">Calculate your BMI</h3>  </div>  <div class="card-body">   <div class="form-group">  <label>Height(cm)</label>  <input class="form-control" id="height" name="height" value="180" placeholder="Height(cm)">   </div>  <div class="form-group">  <label>Weight(kg)</label>  <input class="form-control" id="weight" name="weight" value="75" placeholder="Weight(kg)">  <small>We will not keep your weight and height information</small>  </div>  <button type="submit" class="btn btn-primary" id="btn\_get\_bmi">Submit(Ajax)</button>    </div>  </div>  </div><!-- end of card-->    <!-- row .. col-lg-6 .. card -->  <div class="col-lg-4 mb-5">  <div class="card">  <div class="card-header">  <h3 class="h6 text-uppercase mb-0">Your BMI</h3>  </div>  <div class="card-body">  <h2 id="yourbmi">Your bmi will be shown here.</h2>  </div>  </div>  </div><!-- end of card -->    </div><!-- row --> </div> <!-- container -->  </body> </html>  <!-- java scrip--> <!-- jQuery js--> <script src="https://code.jquery.com/jquery-3.1.0.min.js"></script> <!-- chartjs s--> <script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.3/Chart.min.js"></script>  <!-- Your Javascript codes --> <script>   //\*\* on click event for submit button  $('#btn\_get\_bmi').on('click', function () {  // alert("You click this button!")  // $('#myForm').trigger('submit'); # form submit   $('#yourbmi').empty();  $('#yourbmi').append("here is your bmi!");  let height = $('#height').val();  let weight = $('#weight').val();   jQuery.ajax({   type: "POST",  url: "api\_get\_bmi/",  //url: "http://163.18.23.21:8000/api\_get\_bmi/",  //url: "http://127.0.0.1:8000/api\_get\_bmi/",  data: {  "height": height,  "weight": weight,  }, //pass to server  success: function (*received*) {  const result = *received*['yourbmi'];  console.log(result);  $('#yourbmi').empty();  $('#yourbmi').append("Your bmi: " + result);   }, //success function   });//ajax   });// btn on click event </script> |

|  |
| --- |
| $表示jQuery指令  #元素id  <!-- Your Javascript codes -->  <script>  4種js變數宣告方式:  const不可修改、不可重新宣告  let可改內容、不可重新宣告  var可改內容、可重新宣告  none不寫，全域變數  //\*\* on click event for submit button  $('#btn\_get\_bmi').on('click', function () {  // alert("You click this button!")  // $('#myForm').trigger('submit'); # form submit  $('#yourbmi').empty();  $('#yourbmi').append("here is your bmi!");  let height = $('#height').val();  let weight = $('#weight').val();  Ajax寫法:全定義在json格式中{}  url: 呼叫哪一個路徑?在urls.py定義  data:傳給伺服器的變數  success:呼叫成功後執行的函數function (received)  received是接收到變數(名稱自訂)  jQuery.ajax({  type: "POST",  url: "http://127.0.0.1:8000/api\_get\_bmi/",  data: {  "height": height,  "weight": weight,  }, //pass to server  success: function (received) {  const result = received['yourbmi'];  $('#yourbmi').empty();  $('#yourbmi').append("Your bmi: " + result);  }, //success function  });//ajax  });// btn on click event  </script> |

Step6: Start server

python manage.py runserver

http://localhost:8000/

看到畫面的內容了嗎?

How to set Django urls route(path) and Ajax url

The setting of urls route in Django is a little bit difficult to understand.

Version #1:

The home.html is rendered in the

home.html的url是空字串，表示根路徑

urls.py

urlpatterns = [

path('', views.home), #for homepage

path('api\_get\_bmi/', views.api\_get\_bmi), #for ajax

]

Notice the URL paths

Ajax寫法1 建議!

jQuery.ajax({

type: "POST",

url: "http://127.0.0.1:8000/api\_get\_bmi/",

…

});

or Ajax寫法2 不建議!

Notice: 前面有一個斜槓

jQuery.ajax({

type: "POST",

url: "/api\_get\_bmi/", //Notice: 前面有一個斜槓，表示從根目錄找api程式

…

});

Notice: 前面沒有斜槓

or Ajax寫法3 不建議!

jQuery.ajax({

type: "POST",

url: "api\_get\_bmi/", //Notice: 前面沒有一個斜槓，表示從此App之相對目錄找api程式

…

});

home.html的url路徑是bmi

Version #2:

urls.py

urlpatterns = [

path('bmi/', views.home), #for homepage

path('api\_get\_bmi/', views.api\_get\_bmi), #for ajax

]

Ajax寫法1: the same as Version#1

jQuery.ajax({

type: "POST",

url: "http://127.0.0.1:8000/api\_get\_bmi/",

…

});

Ajax寫法2: the same as Version#1

Notice: 前面有一個斜槓

jQuery.ajax({

type: "POST",

url: "/api\_get\_bmi/", //Notice: 前面有一個斜槓，表示從根目錄開始，

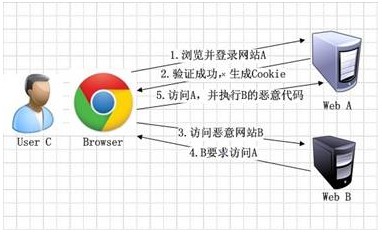
…

});

CSRF(Cross Site Request Forgery)跨站請求偽造

認識CSRF漏洞

CSRF 是一種 Web 上的攻擊手法，全稱是 Cross Site Request Forgery，跨站請求偽造



[圖片](https://www.jianshu.com/p/a178f08d9389)來源

Django內定啟動跨站請求偽造（Cross-site Request Forgery，CSRF）防護機制。

若沒有關閉，則以第三方軟體呼叫API，會出現以下錯誤訊息!

Forbidden!

CSRF verification failed. Request aborted.

因為是API程式，我們必須將CSRF關閉，只針對某個API程式關閉CSRF，方式如下:

from django.views.decorators.csrf import csrf\_exempt

@csrf\_exempt

def test\_api( request ):

為甚麼RESTful API必須關閉CSRF防護?

API就是開放給第三方程式呼叫讀取資料之用，有跨站的需要。

對於讀取資料的服務，無安全之虞，不需要進行CSRF 的保護。

例如:

銀行系統中轉帳的請求會直接改變帳戶的金額，會遭到 CSRF 攻擊，需要保護。而查詢餘額是對金額的讀取操作，不會改變資料，CSRF 攻擊無法解析伺服器返回的結果，無需保護。

資料來源

An API pretty much precludes CSRF, because its entire purpose is generally to allow 3rd-party entities to access and manipulate data on your site (the "cross-site" in CSRF). So, yes, I think as a rule any API view should be CSRF exempt. However, you should still follow best practices and protect every API-endpoint that actually makes a change with some form of authentication, such as OAuth.

一般廠商開放給大眾呼叫的 API

一般廠商開放給大眾呼叫的 API 通常\*\*不會\*\*要求 CSRF token，原因如下：

1. \*\*CSRF 保護的目的不同\*\*：CSRF 保護主要是防止惡意網站在用戶不知情的情況下利用用戶已登入的身份執行操作。但公開 API 通常不依賴於瀏覽器的 cookie 認證。

2. \*\*API 認證機制不同\*\*：公開 API 通常使用其他認證方式：

- API Key / App ID

- OAuth 令牌

- JWT (JSON Web Tokens)

- HMAC 簽名認證

3. \*\*跨域考量\*\*：公開 API 設計為允許從任何來源呼叫，CSRF 保護會阻礙這種使用場景。

4. \*\*無狀態設計\*\*：RESTful API 通常設計為無狀態的，每次請求都需要完整的認證資訊，不依賴於 cookie 中的會話。

如果你正在開發的是：

- 面向瀏覽器，使用 cookie 認證的 API

- 網站自身使用的 API 端點

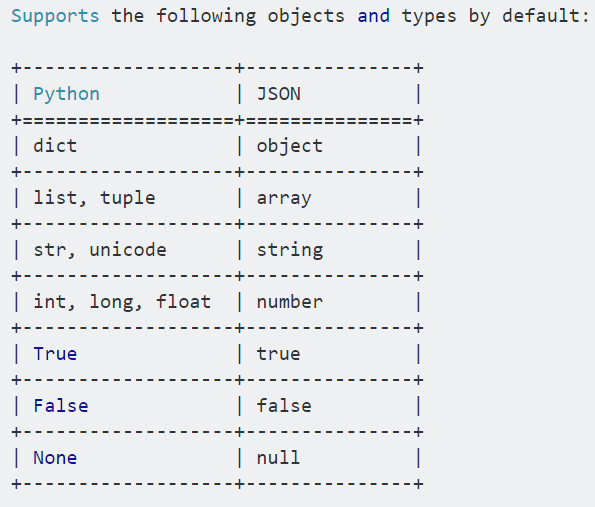
- 需要執行敏感操作的 API

這些情況才需要考慮 CSRF 保護。

對於本例中的 BMI 計算 API，如果只是內部使用且不涉及敏感操作，可以保留 `@csrf\_exempt`；但作為良好實踐，建議為所有基於 cookie 認證的系統實施 CSRF 保護。

API程式為何對於資料的傳送使用json格式?

JSON encoding is supported for the following native python types out of the box.



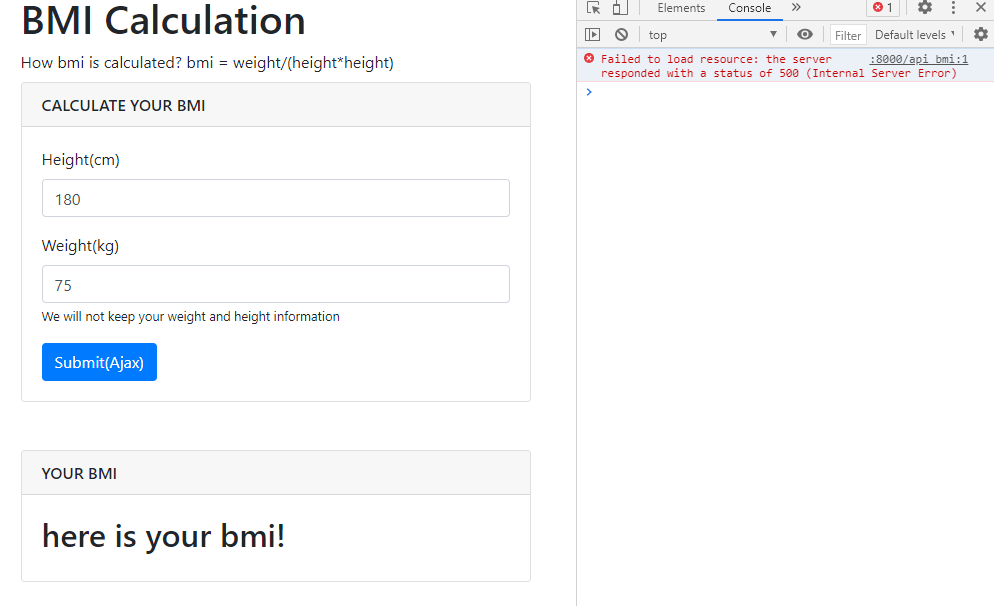
views.py函數中python數字、字串、list可以直接都給jsonResponse沒問題。

但是，

views.py函數中的dict, numpy等會被視為物件，必須進行以下處理方式如下:

* dict字典要用json.dumps()處理轉成字串str格式之後，再用JsonResponse()送出。(因為在網頁中會被轉成字串後其格式不對)
* pandas的欄位若為數字(文字沒問題): Pandas的欄位若為數字，則其格式是numpy格式，不能傳給網頁、或是給jsonResponse使用。必須預先轉成一般的int, float等格式。
* 中文字會呈現亂碼解決方式: json.dumps(sentiment\_prob, ensure\_ascii=False)

Trouble shooting



Internal Server Error

urls.py

|  |
| --- |
| urlpatterns = [  path('', views.home),  path('api\_get\_bmi/', views.api\_get\_bmi),  ]  Notice the URL paths |

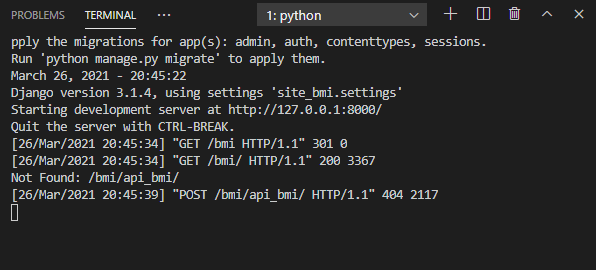
home.html

|  |
| --- |
| jQuery.ajax({  type: "POST",  url: "http://127.0.0.1:8000/api\_get\_bmi/",  //url: "/api\_get\_bmi/", //This is OK.  data: {  "height": height,  "weight": weight,  }, //pass to server  ……  });//ajax |

Error Message:

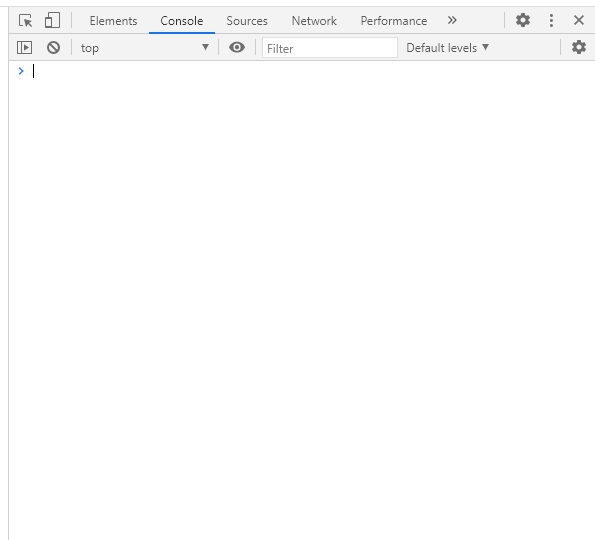
RuntimeError: You called this URL via POST, but the URL doesn't end in a slash and you have APPEND\_SLASH set. Django can't redirect to the slash URL while maintaining POST data. Change your form to point to 127.0.0.1:8000/api\_get\_bmi/ (note the trailing slash), or set APPEND\_SLASH=False in your Django settings.

Not Found: /bmi/api\_get\_bmi/



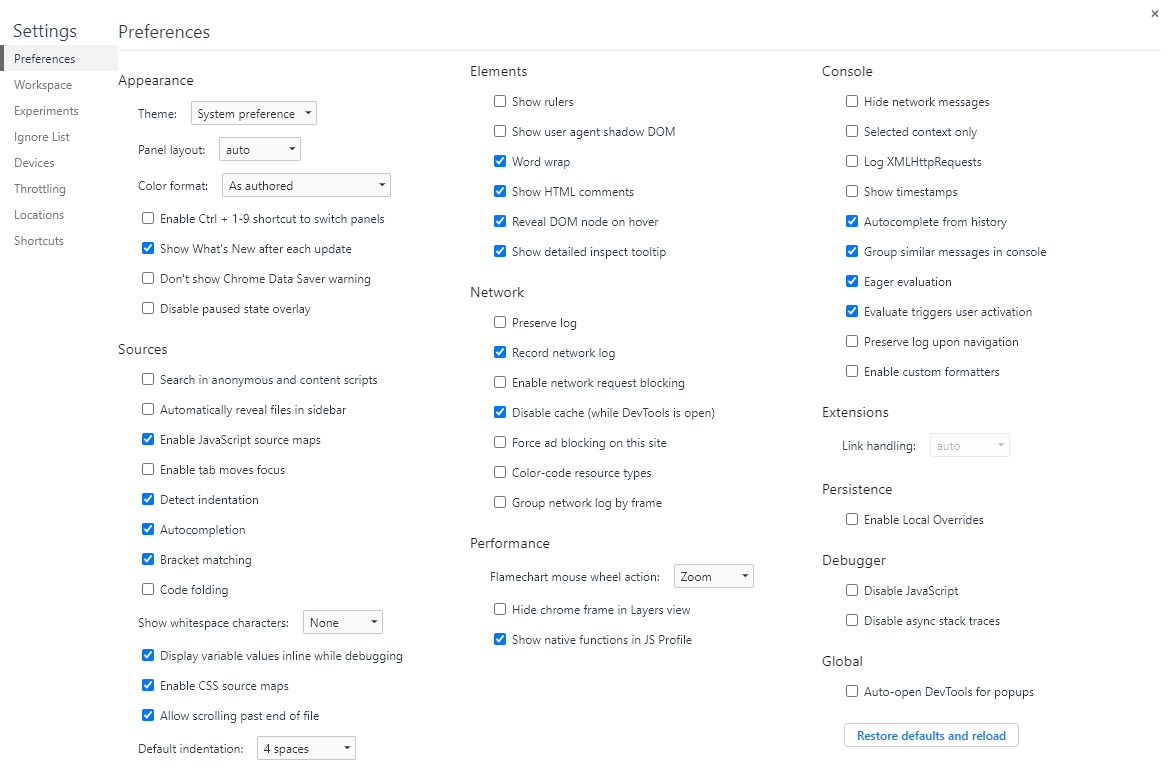
If everything is fine, but it still can’t work (e.g., “Not Found: /bmi/api\_get\_bmi/”), what can you do?

1. 清除Chrome瀏覽器快取資料
2. Disable cache不要快取



Click it.

按一下這裡設定



Check it.

Disable cache

不要快取