### 透過HTTP與伺服器溝通的方法

HTTP協定中定義了多種不同的method，瀏覽器或是其他程式再進行HTTP連線時，會使用這些method來進行連線並取得回覆

HTTP 1.1 的版本定義的方法

POST [新增]

GET [讀取]

PUT [更新]

DELETE [刪除]

OPTIONS [跨域資源共享用得到]

HEAD

TRACE

CONNECT

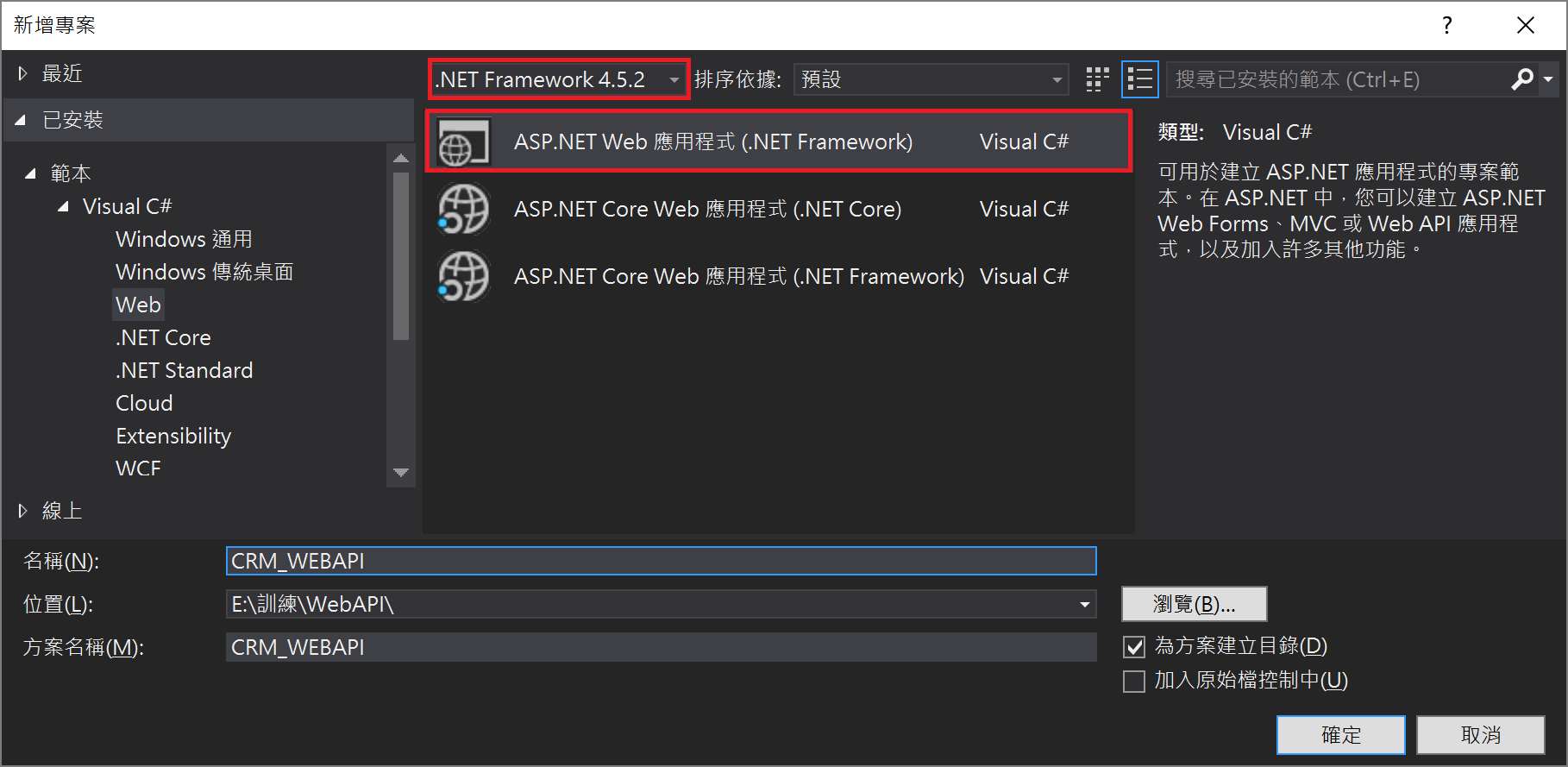
Header

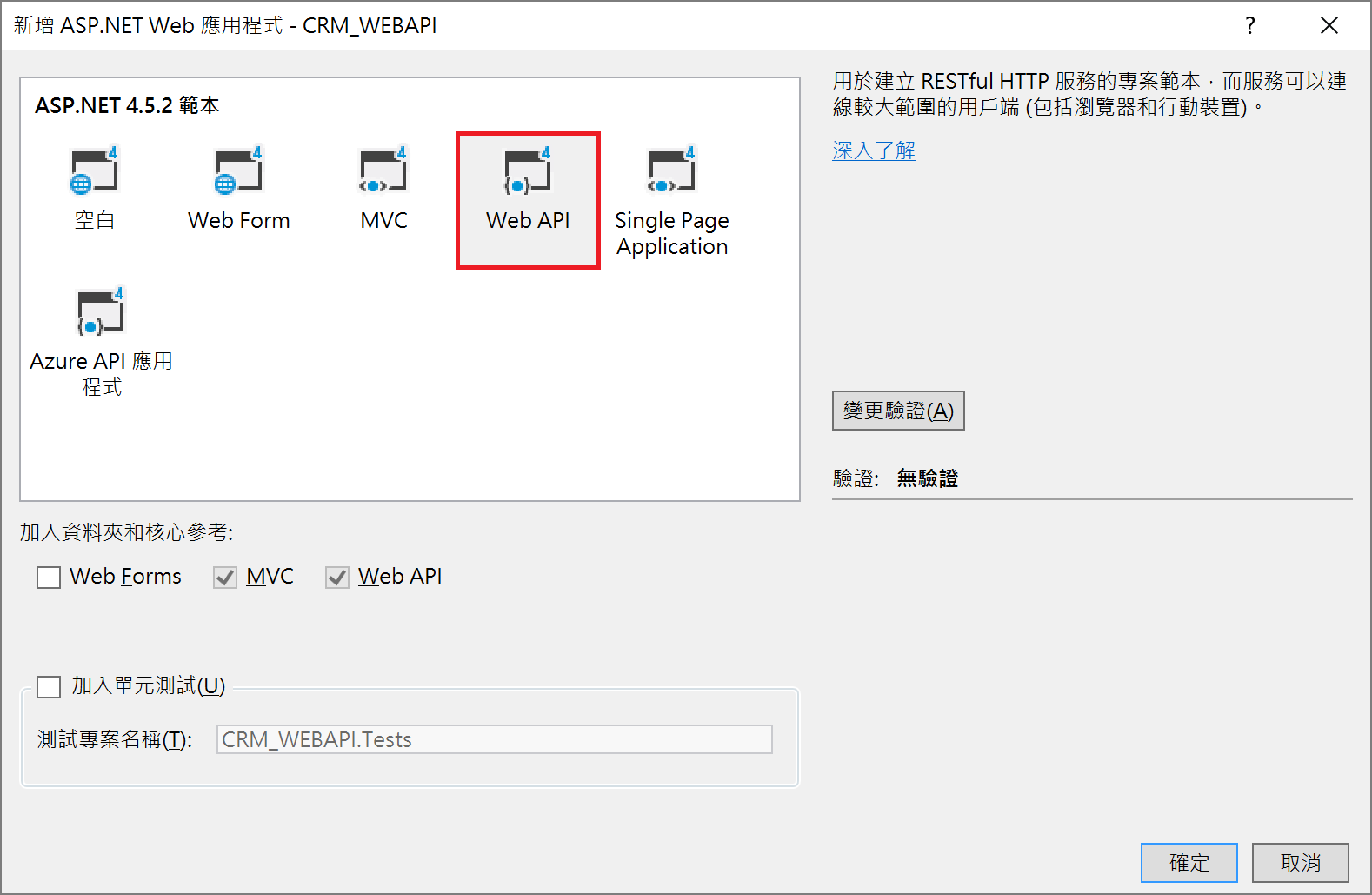
Accept [宣告期望回應的資料類型]

Content-Type [宣告遞送給對方的資料類型]

其他自訂 [存放任何資訊，如驗證的帳號密碼等等(有SSL或加上時戳，才能確保安全性)]

### 建立WebAP專案





### 測試[Test01]

Controllers 右鍵新增>加入>控制器>選擇Web API 2 控制器–空白，取名為[Test01Controller]

編寫Test01Controller.cs內容如下

public string Get()

{

return "Hello World! (Get)";

}

public string Post()

{

return "Hello World! (Post)";

}

修改App\_Start/WebApiConfig.cs裡Register(HttpConfiguration config)的Route設定為

public static void Register(HttpConfiguration config)

{

// Web API 設定和服務

// Web API 路由

config.MapHttpAttributeRoutes();

config.Routes.MapHttpRoute(

name: "DefaultApi",

routeTemplate: "api/{controller}/{userName}",

defaults: new { userName = RouteParameter.Optional }

);

}

在根目錄建立[MyHemlTest]資料夾後，建立檔案[Test01.html]

<!doctype html>

<html>

<head>

<meta charset="UTF-8">

<title>Untitled Document</title>

</head>

<body>

<form action="http://localhost:3958/api/Test01" method="post">

<a href="http://localhost:3958/api/Test01">Get</a><br>

<input type="submit" value="Post">

</form>

</body>

</html>

執行後瀏覽Test01.html，測試Get與Post按鈕

### 測試[Test02]

建立Controller名為[Test02Controller]

編寫Test01Controller.cs內容如下

public string Get(string userName)

{

return $"Hello World! {userName} (Get)";

}

public string Post()

{

string userName = string.Empty;

HttpContext httpContext = HttpContext.Current; // 取得目前的 HttpContext

NameValueCollection formData = httpContext.Request.Form; // 取得 From 資料集合

userName = formData["userName"];

return $"Hello World! {userName} (Post)";

}

建立檔案[Test02.html]

<!doctype html>

<html>

<head>

<meta charset="UTF-8">

<title>Test02</title>

</head>

<body>

<form action="http://localhost:3958/api/Test02" method="post">

<p>Test02</p>

<a href="http://localhost:3958/api/Test02/Jet">Get</a><br><br>

<input type="text" name="userName">

<input type="submit" value="Post">

</form>

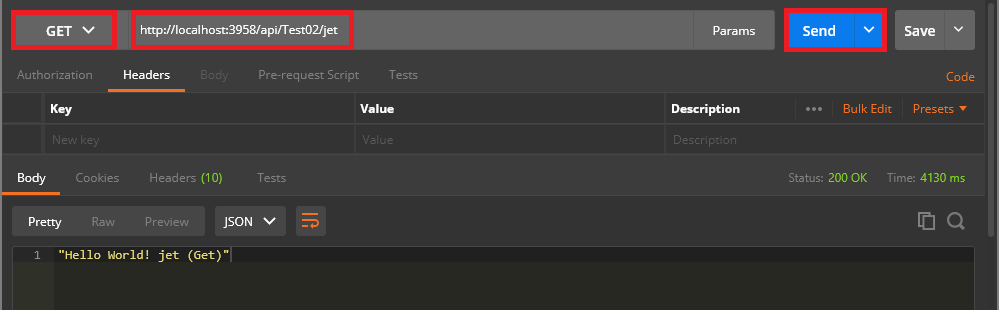
</body>

</html>

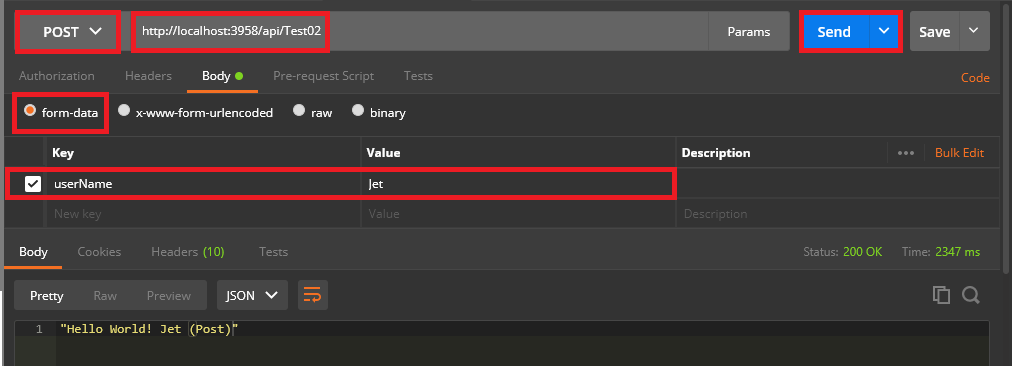
執行後瀏覽Test02.html，測試Get與Post按鈕

### 使用PostMan測試[Test02]

Get <http://localhost:3958/api/Test02/jet>



Get <http://localhost:3958/api/Test02/jet>



注意!資料內容請選擇form-data

### 測試[Test03]

注意資源描述的結構不一樣了

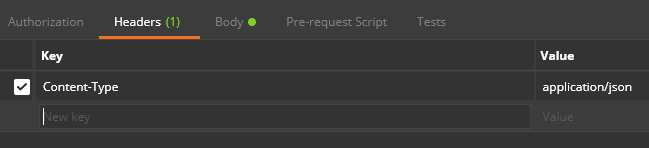
不同瀏覽器回應不同格式的資料

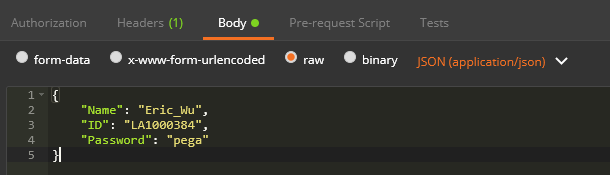
### 測試[Test04]

注意IIS對於存放注意資源描述的結構不一樣了

不同瀏覽器回應不同格式的資料

上傳JSON





上傳檔案(應注意Header不應設定Content-Type)

