Tutorialr.com

Spotify for Developers Follow

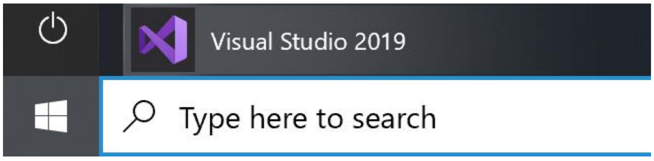
Follow Artists or Users

Add the current user as a follower of one or more artists or other Spotify users.

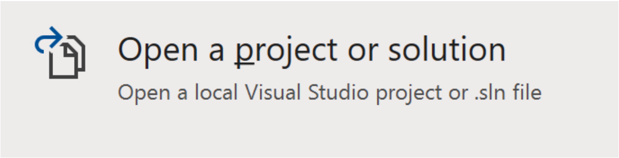
PUT https://api.spotify.com/v1/me/following	
Header	
Authorization	User Token from Spotify Accounts service with **user-follow-modify** scope
Query Parameter	
type	Type of artist or user
ids	Comma-separated list of the artist or the user Spotify Ids - maximum 50

Header	**Response**
Success	
HTTP Status 204 No Content	Empty Response Body
Error	
Error Code	Error Object

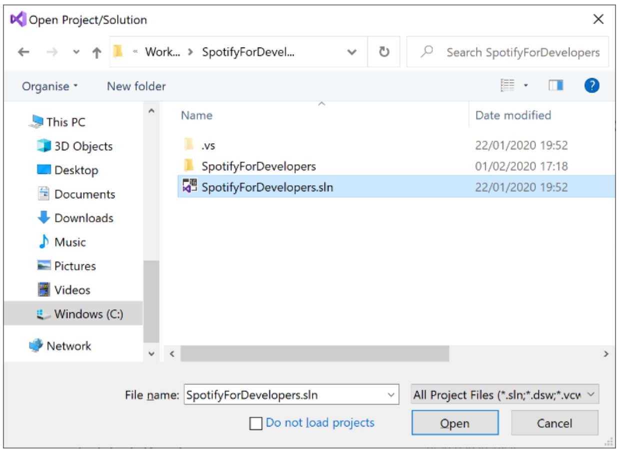
Step 1



In **Windows 10** choose **Start**, and then from the **Start Menu** find and select **Visual Studio 2019**


Open a project or solution
Open a local Visual Studio project or .sln file

Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



Name	Date modified
.vs	22/01/2020 19:52
SpotifyForDevelopers	01/02/2020 17:18
SpotifyForDevelopers.sln	22/01/2020 19:52

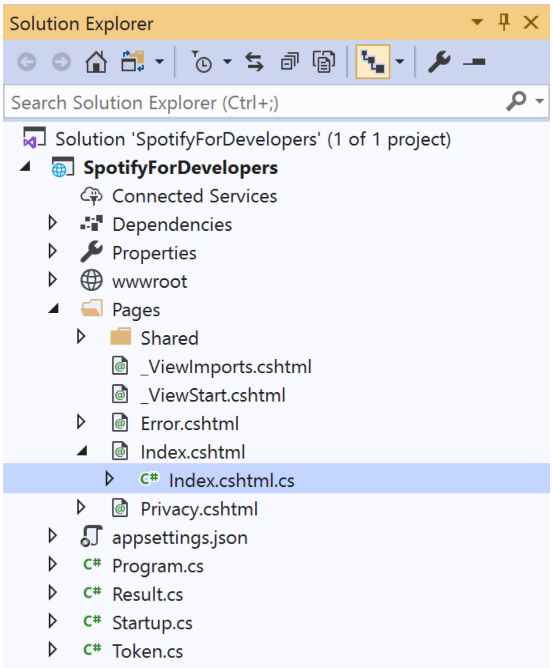
File name: SpotifyForDevelopers.sln

All Project Files (*.sln;*.dsw;*.vcw

□ Do not load projects

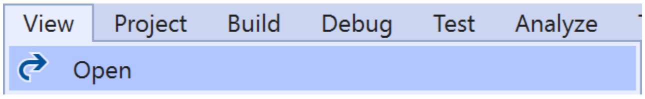
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started**, **Authorisation Guide**, **Search & Browse**, **Playlists & Artists**, **Albums & Tracks** and **Episodes & Shows**

Step 2



Once opened, in the **Solution Explorer** open the **Pages** section, then open the **Index.cshtml** section and select **Index.cshtml.cs**

Step 3



Then from the **Menu** choose **View** and then **Open**

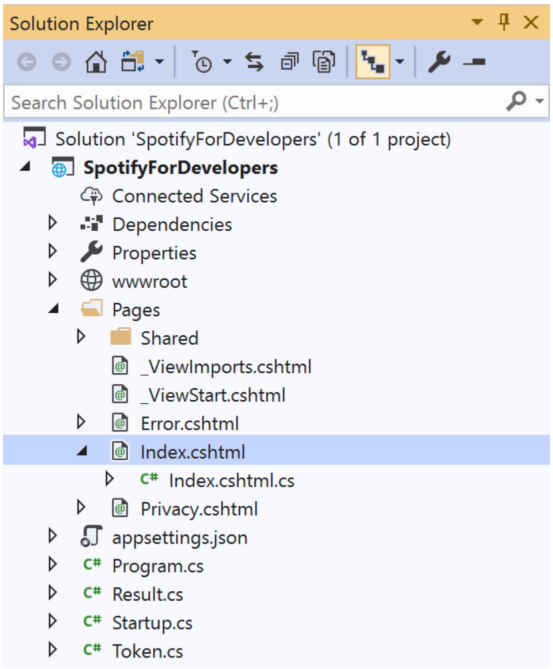
Step 4

In the **Code View** for **Index.cshtml.cs** below the **method** for `public async Task<IActionResult> OnPostGetShowEpisodesAsync(...) { ... }` enter the following **method**:

```
public async Task<IActionResult> OnPostFollowArtistsOrUsersAsync(string value, FollowType
option)
{
    LoadToken();
    var values = value.Split(",").ToList();
    var result = await Api.FollowArtistsOrUsersAsync(values, option);
    if (result != null)
    {
        Results = new List<Result>() { new Result()
        {
            Name = result.Success.ToString()
        }};
    }
    return Page();
}
```

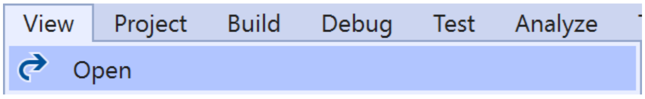
The **method** for `OnPostFollowArtistsOrUsersAsync` is used to set the **following state** for an **artist** or **user** by **Artist Id** or **User Id** on Spotify with the `value` and selected `option` and populate the **property** for `Results` accordingly with the **success** of the operation.

Step 5



In the **Solution Explorer** in the **Pages** section select **Index.cshtml**

Step 6



Then from the **Menu** choose **View** and then **Open**

Step 7

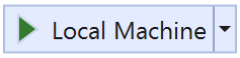
Once in the **Code View** for **Index.cshtml** above `<!-- Spotify Web API User Authorisation -->` enter the following:

```
<ul class="list-group mb-2">
    <li class="list-group-item list-group-item-primary">
        <h5 class="list-group-item-heading">Follow</h5>
    </li>
    <li class="list-group-item">
        <form asp-page-handler="FollowArtistsOrUsers" method="post">
            <select asp-for="Option" class="form-control mb-2"
                asp-items="Html.GetEnumSelectList<Spotify.NetStandard.Enums.FollowType>()">
            </select>
            <input asp-for="Value" placeholder="Ids" class="form-control mb-2" />
            <button class="btn btn-primary mb-2">
                Follow Artists or Users
            </button>
        </form>
    </li>

    <!-- Follow -->
</ul>
```

This `form` will **post** to the **method** for `FollowArtistsOrUsers` with the `Value` of the **Ids** and the `Option` for the **Artist** or **User** and will output to the **Results**.

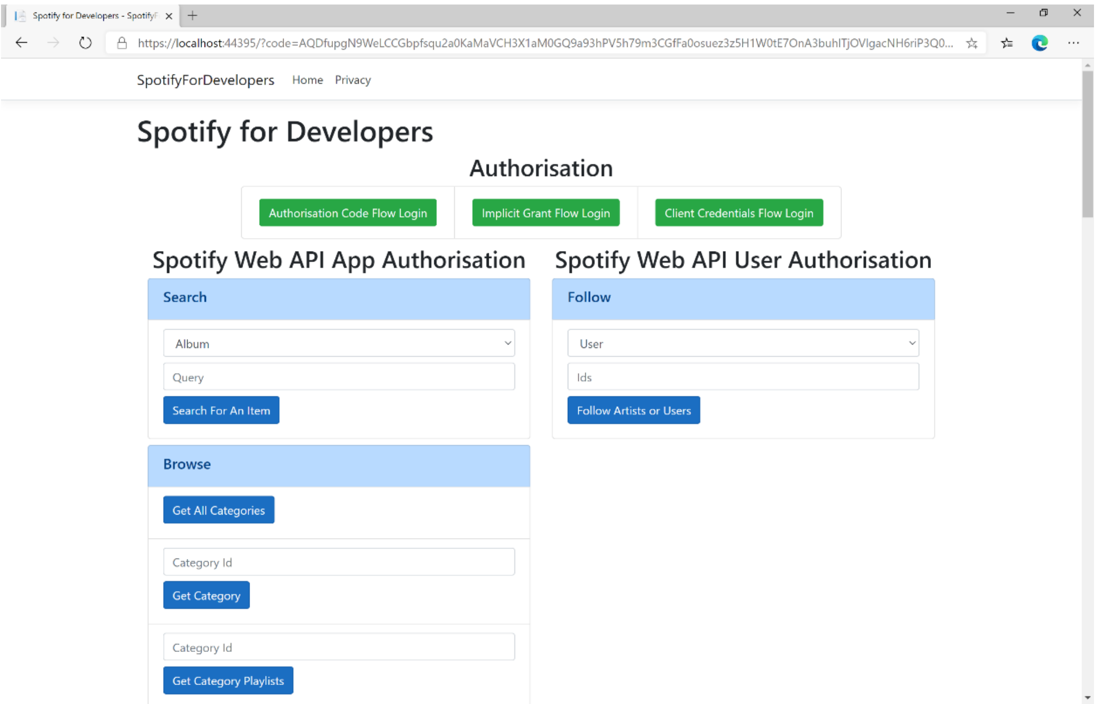
Step 8



Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**

Step 9

Once the **Web Application** is running and you select **Authorisation Code Flow Login** or **Implicit Grant Flow Login** you should see something like the following:



Spotify for Developers - SpotifyF
https://localhost:44395/?code=AQDfupgN9WeLCCGbpfsqu2a0KaMaVCH3X1aM0GQ9a93hPV5h79m3CGfFa0osuez3z5H1W0tE7OnA3buhlTjOVlgacNH6riP3Q0...
SpotifyForDevelopers
Home
Privacy

Spotify for Developers

Authorisation

Authorisation Code Flow Login | Implicit Grant Flow Login | Client Credentials Flow Login

Spotify Web API App Authorisation

Search

Album

Query

Search For An Item

Browse

Get All Categories

Category Id

Get Category

Category Id

Get Category Playlists

Spotify Web API User Authorisation

Follow

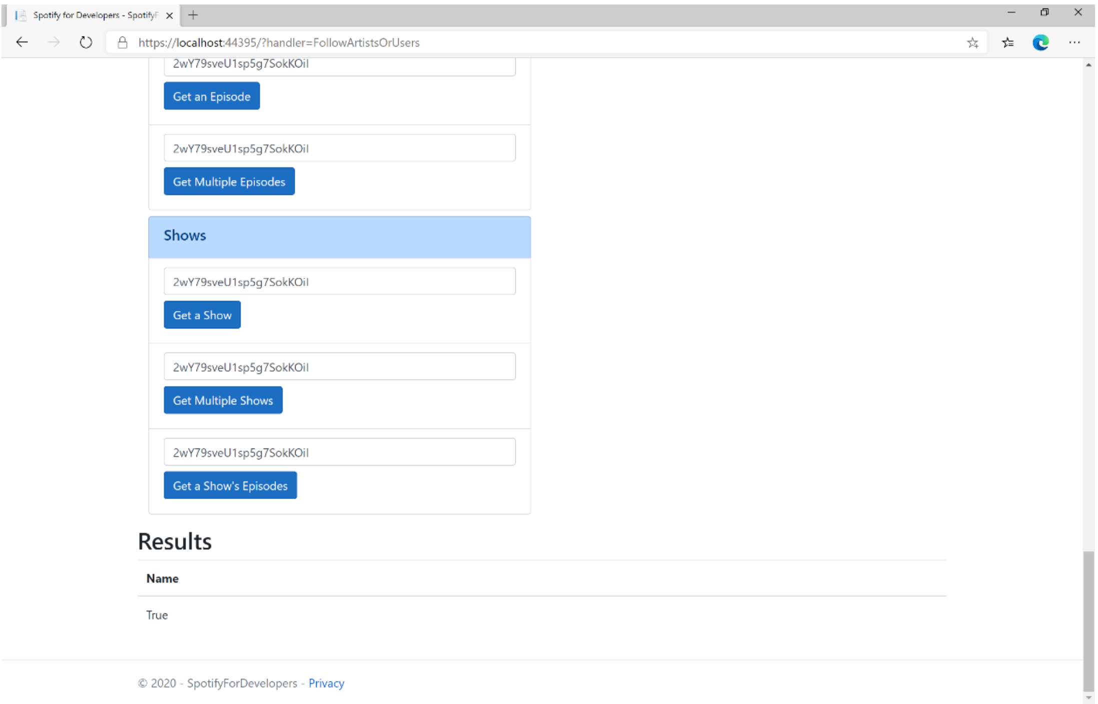
User

Ids

Follow Artists or Users

Step 10

You can then enter an **Artist Id** from **Get All New Releases** and copy an **Artist Id** to use and above **Follow Artists or Users** choose **Artist** and then select **Follow Artists or Users** and scroll down to view **Results** like the following:



Spotify for Developers - SpotifyF
https://localhost:44395/?handler=FollowArtistsOrUsers

2wY79sveU1sp5g7SokKOil

Get an Episode

2wY79sveU1sp5g7SokKOil

Get Multiple Episodes

Shows

2wY79sveU1sp5g7SokKOil

Get a Show

2wY79sveU1sp5g7SokKOil

Get Multiple Shows

2wY79sveU1sp5g7SokKOil

Get a Show's Episodes

Results

Name
True

© 2020 - SpotifyForDevelopers - Privacy

Step 11

■

You can stop the **web application** in **Visual Studio 2019** by selecting the **Stop debugging** button

Step 12

×

You can choose to exit **Visual Studio 2019** by selecting the **Close** button in the top right of the **application** as that completes this part of the workshop

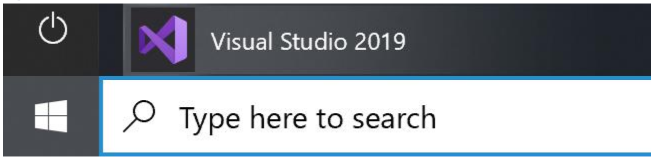
Follow a Playlist

Add the current user as a follower of a playlist.

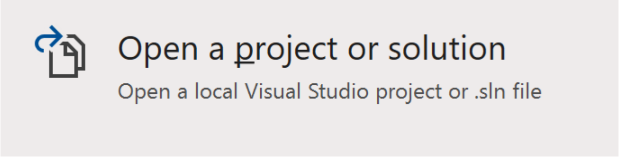
PUT https://api.spotify.com/v1/playlists/{playlist_id}/followers	
Header	
Authorization	User Token from Spotify Accounts service with **user-follow-modify** scope
Path Parameter	
playlist_id	Spotify Id of the playlist
Body Parameter	
public	Defaults to true. If true the Playlist will be included in the User's public Playlists, if false it will remain private

Header	Response
Success	
HTTP Status 200 OK	Empty Response Body
Error	
Error Code	Error Object

Step 1



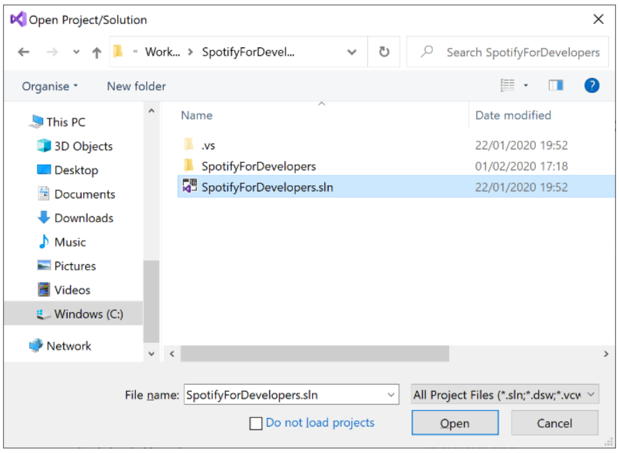
If you chose to close **Visual Studio 2019** previously, in **Windows 10** choose **Start**, and then from the **Start Menu** find and select **Visual Studio 2019**



Open a project or solution

Open a local Visual Studio project or .sln file

Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



Open Project/Solution

Work... › SpotifyForDevel... Search SpotifyForDevelopers

Organise ▾ New folder

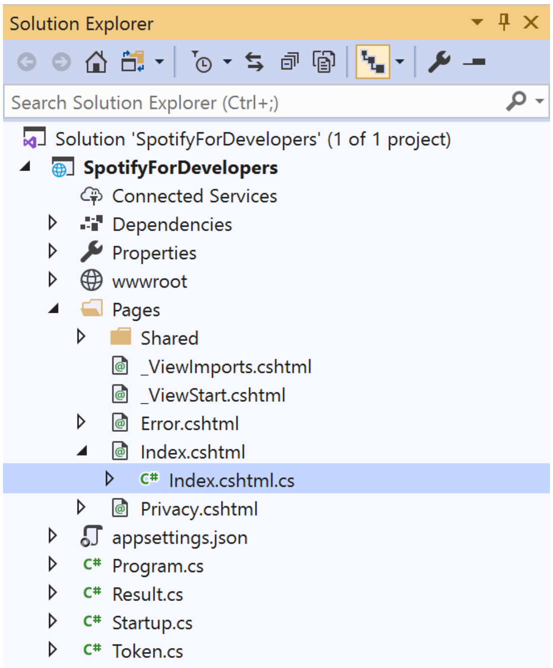
Name	Date modified
.vs	22/01/2020 19:52
SpotifyForDevelopers	01/02/2020 17:18
SpotifyForDevelopers.sln	22/01/2020 19:52

File name: SpotifyForDevelopers.sln All Project Files (*.sln;*.dsw;*.vcw

☐ Do not load projects Open Cancel

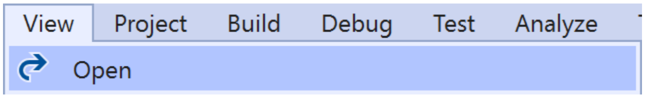
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started**, **Authorisation Guide**, **Search & Browse**, **Playlists & Artists**, **Albums & Tracks**, **Episodes & Shows** and **Follow**

Step 2



Once opened, in the **Solution Explorer** open the **Pages** section, then open the **Index.cshtml** section and select **Index.cshtml.cs**

Step 3



Then from the **Menu** choose **View** and then **Open**

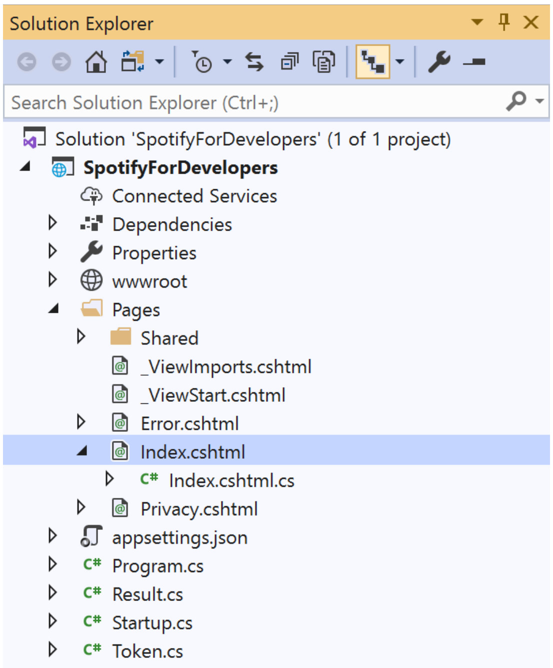
Step 4

In the **Code View** for **Index.cshtml.cs** below the **method** for `public async Task<IActionResult> OnPostFollowArtistsOrUsersAsync(...) { ... }` enter the following **method**:

```
public async Task<IActionResult> OnPostFollowPlaylistAsync(string value)
{
    LoadToken();
    var result = await Api.FollowPlaylistAsync(value);
    if (result != null)
    {
        Results = new List<Result>() { new Result()
        {
            Name = result.Success.ToString()
        }};
    }
    return Page();
}
```

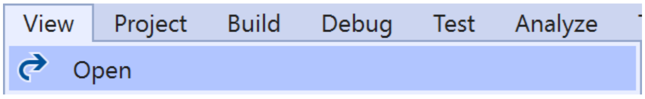
The **method** for `OnPostFollowPlaylistAsync` is used to set the **following state** for a **playlist** by **Playlist Id** on Spotify with the `value` and populate the **property** for `Results` accordingly with the **success** of the operation.

Step 5



In the **Solution Explorer** in the **Pages** section select **Index.cshtml**

Step 6



Then from the **Menu** choose **View** and then **Open**

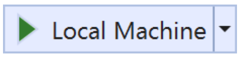
Step 7

Once in the **Code View** for **Index.cshtml** above `<!-- Follow -->` enter the following:

```
<li class="list-group-item">
    <form asp-page-handler="FollowPlaylist" method="post">
        <input asp-for="Value" placeholder="Playlist Id" class="form-control mb-2" />
        <button class="btn btn-primary mb-2">
            Follow a Playlist
        </button>
    </form>
</li>
```

This `form` will **post** to the **method** for `FollowPlaylist` with the `Value` of the **Playlist Id** and will output to the **Results**.

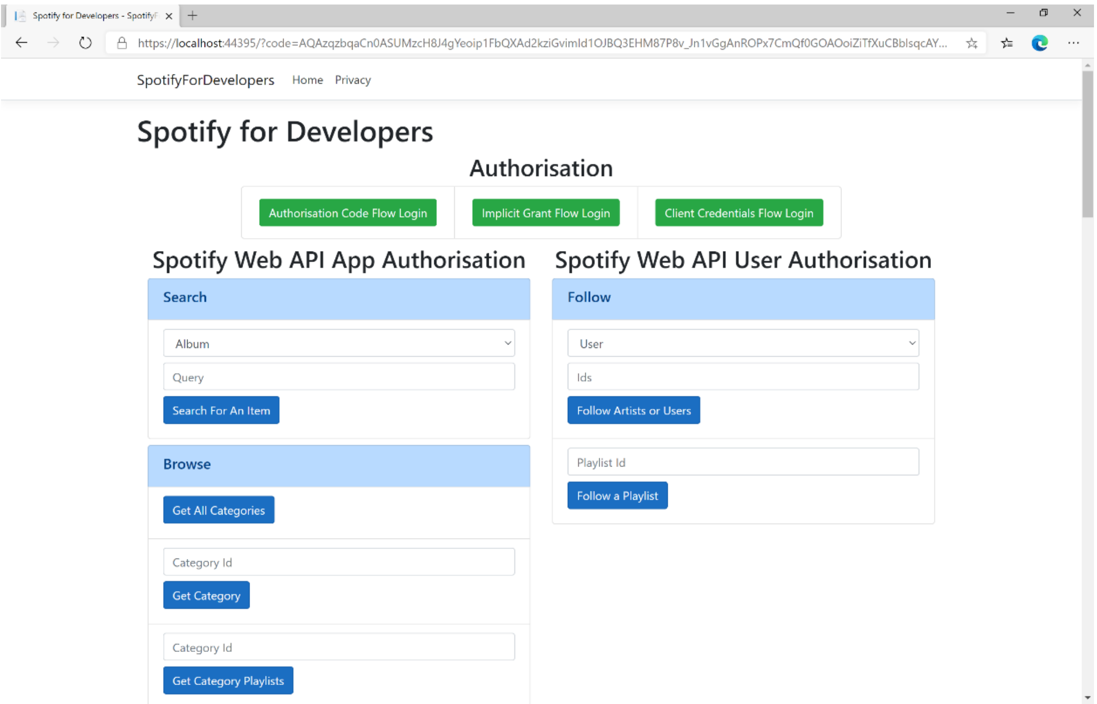
Step 8



Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**

Step 9

Once the **Web Application** is running and you select **Authorisation Code Flow Login** or **Implicit Grant Flow Login** you should see something like the following:



Spotify for Developers - Spotify

https://localhost:44395/?code=AQAzqzbqaCn0ASUMzcH8J4gYeoip1FbQXAd2kziGvimld1OJBQ3EHM87P8v_Jn1vGgAnROPx7CmQf0GOAOoiZiTfXuCBblsqcAY...

SpotifyForDevelopers Home Privacy

Spotify for Developers

Authorisation

Authorisation Code Flow Login | Implicit Grant Flow Login | Client Credentials Flow Login

Spotify Web API App Authorisation

Search

Album

Query

Search For An Item

Browse

Get All Categories

Category Id

Get Category

Category Id

Get Category Playlists

Spotify Web API User Authorisation

Follow

User

Ids

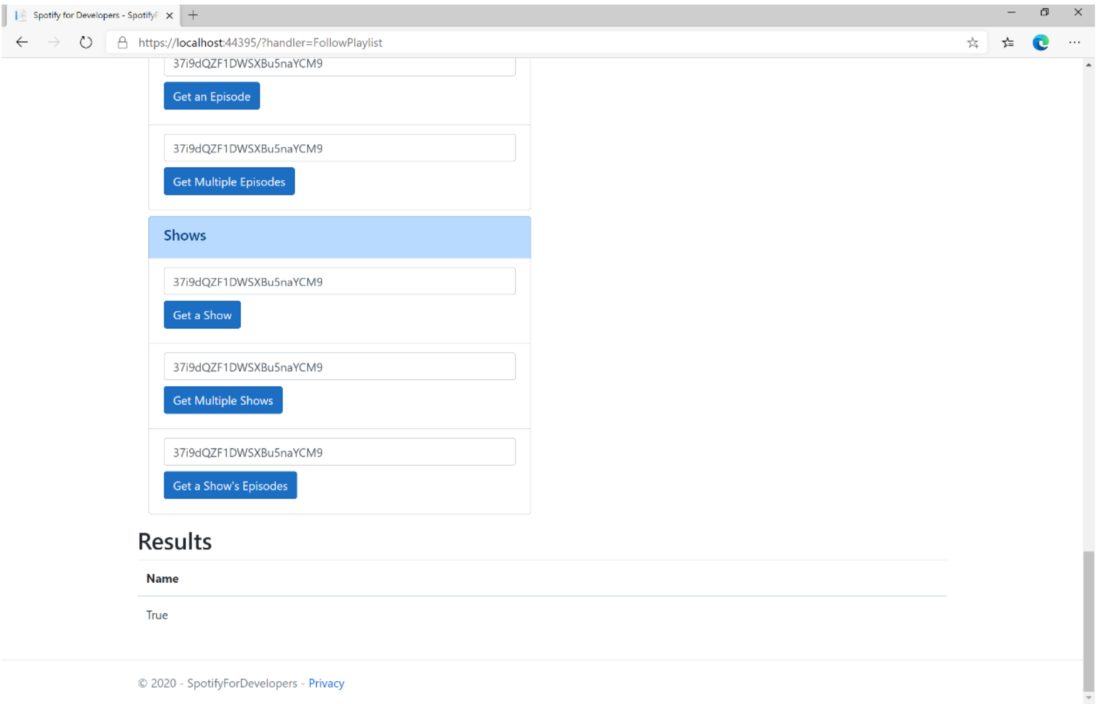
Follow Artists or Users

Playlist Id

Follow a Playlist

Step 10

You can then enter a **Playlist Id** from **Get All Featured Playlists** and copy a **Playlist Id** to use and select **Follow a Playlist** and scroll down to view **Results** like the following:



Spotify for Developers - Spotify

https://localhost:44395/?handler=FollowPlaylist

37i9dQZF1DWSXBu5naYCM9

Get an Episode

37i9dQZF1DWSXBu5naYCM9

Get Multiple Episodes

Shows

37i9dQZF1DWSXBu5naYCM9

Get a Show

37i9dQZF1DWSXBu5naYCM9

Get Multiple Shows

37i9dQZF1DWSXBu5naYCM9

Get a Show's Episodes

Results

Name
True

© 2020 - SpotifyForDevelopers - Privacy

Step 11

■

You can stop the **web application** in **Visual Studio 2019** by selecting the **Stop debugging** button

Step 12

✕

You can choose to exit **Visual Studio 2019** by selecting the **Close** button in the top right of the **application** as that completes this part of the workshop

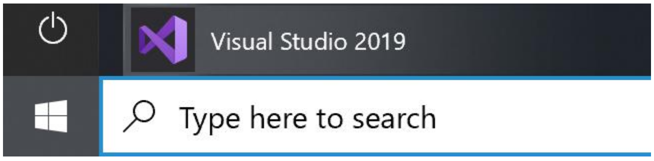
Get Following State for Artists/Users

Check to see if the current user is following one or more artists or other Spotify users.

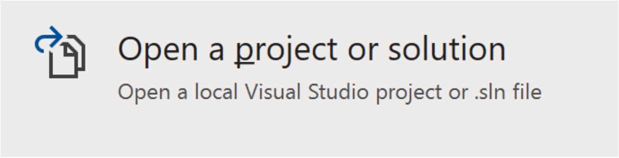
GET https://api.spotify.com/v1/me/following/contains	
Header	
Authorization	User Token from Spotify Accounts service with **user-follow-read** scope
Query Parameter	
type	Type of artist or user
ids	Comma-separated list of the artist or the user Spotify Ids to check

Header	**Response**
Success	
HTTP Status 200 OK	Array of true or false values in same order as in which the Ids were provided
Error	
Error Code	Error Object

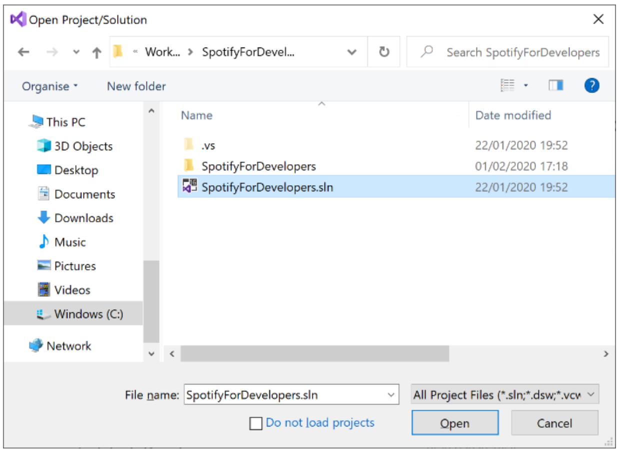
Step 1



If you chose to close **Visual Studio 2019** previously, in **Windows 10** choose **Start**, and then from the **Start Menu** find and select **Visual Studio 2019**


Open a project or solution
Open a local Visual Studio project or .sln file

Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**

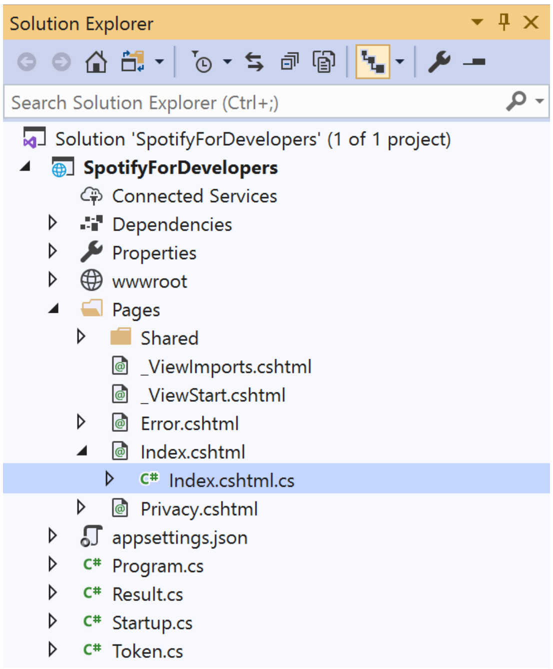


Name	Date modified
.vs	22/01/2020 19:52
SpotifyForDevelopers	01/02/2020 17:18
SpotifyForDevelopers.sln	22/01/2020 19:52

File name: SpotifyForDevelopers.sln

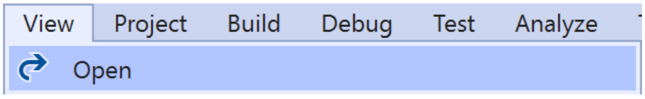
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started**, **Authorisation Guide**, **Search & Browse**, **Playlists & Artists**, **Albums & Tracks**, **Episodes & Shows** and **Follow**

Step 2



Once opened, in the **Solution Explorer** open the **Pages** section, then open the **Index.cshtml** section and select **Index.cshtml.cs**

Step 3



Then from the **Menu** choose **View** and then **Open**

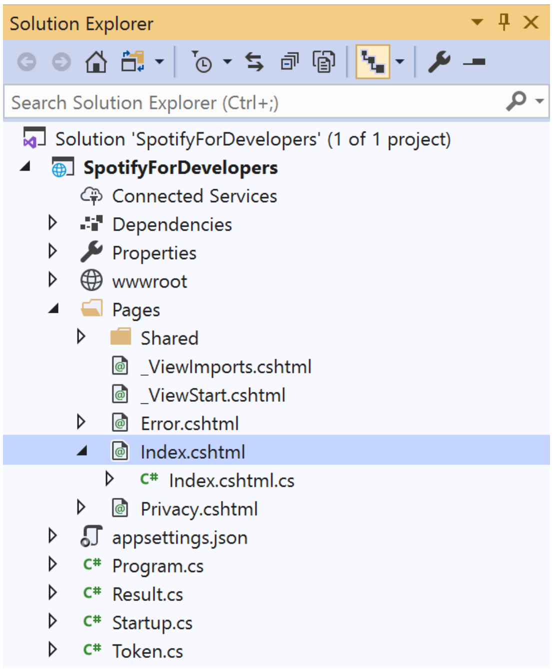
Step 4

In the **Code View** for **Index.cshtml.cs** below the **method** for `public async Task<IActionResult> OnPostFollowPlaylistAsync(...) { ... }` enter the following **method**:

```
public async Task<IActionResult> OnPostGetFollowingStateForArtistsOrUsersAsync(string value,
FollowType option)
{
    LoadToken();
    var values = value.Split(",").ToList();
    var results = await Api.GetFollowingStateForArtistsOrUsersAsync(values, option);
    if (results != null)
    {
        Results = results.Select(result => new Result()
        {
            Name = result.ToString()
        });
    }
    return Page();
}
```

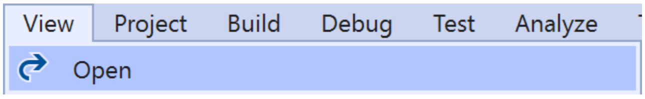
The **method** for `OnPostGetFollowingStateForArtistsOrUsersAsync` is used to get the **following state** for a **artist** or **user** by **Artist Id** or **User Id** on Spotify with the `value` and selected `option` and populate the **property** for `Results` accordingly.

Step 5



In the **Solution Explorer** in the **Pages** section select **Index.cshtml**

Step 6



Then from the **Menu** choose **View** and then **Open**

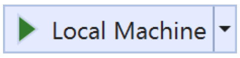
Step 7

Once in the **Code View** for **Index.cshtml** above `<!-- Follow -->` enter the following:

```
<li class="list-group-item">
    <form asp-page-handler="GetFollowingStateForArtistsOrUsers" method="post">
        <select asp-for="Option" class="form-control mb-2"
            asp-items="Html.GetEnumSelectList<Spotify.NetStandard.Enums.FollowType>()">
        </select>
        <input asp-for="Value" placeholder="Ids" class="form-control mb-2" />
        <button class="btn btn-primary mb-2">
            Get Following State for Artists or Users
        </button>
    </form>
</li>
```

This `form` will **post** to the **method** for `GetFollowingStateForArtistsOrUsers` with the `Value` of the **Ids** and the `Option` for the **Artist** or **User** and will output to the **Results**.

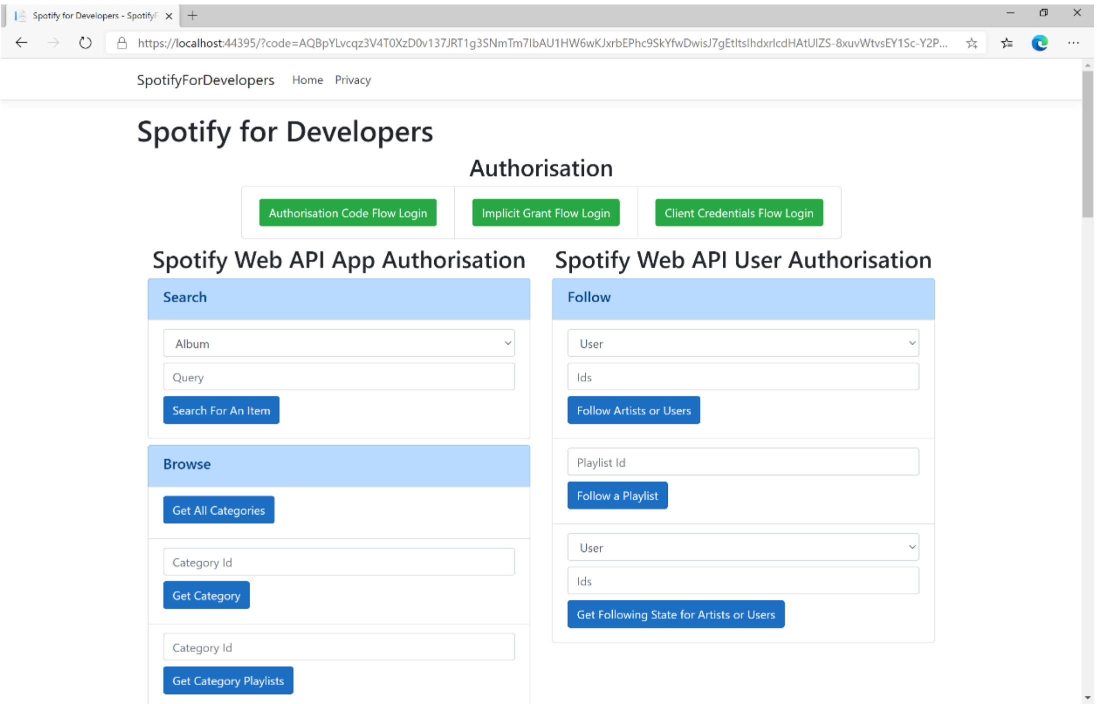
Step 8



Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**

Step 9

Once the **Web Application** is running and you select **Authorisation Code Flow Login** or **Implicit Grant Flow Login** you should see something like the following:



https://localhost:44395/?code=AQBpYLvcqz3V4T0XzD0v137JRT1g3SNmTm7IbAU1HW6wKJxrbEPhc9SkYfwDwisJ7gEtltsIhdxrlcdHAtUIZS-8xuvWtvsEY1Sc-Y2P...

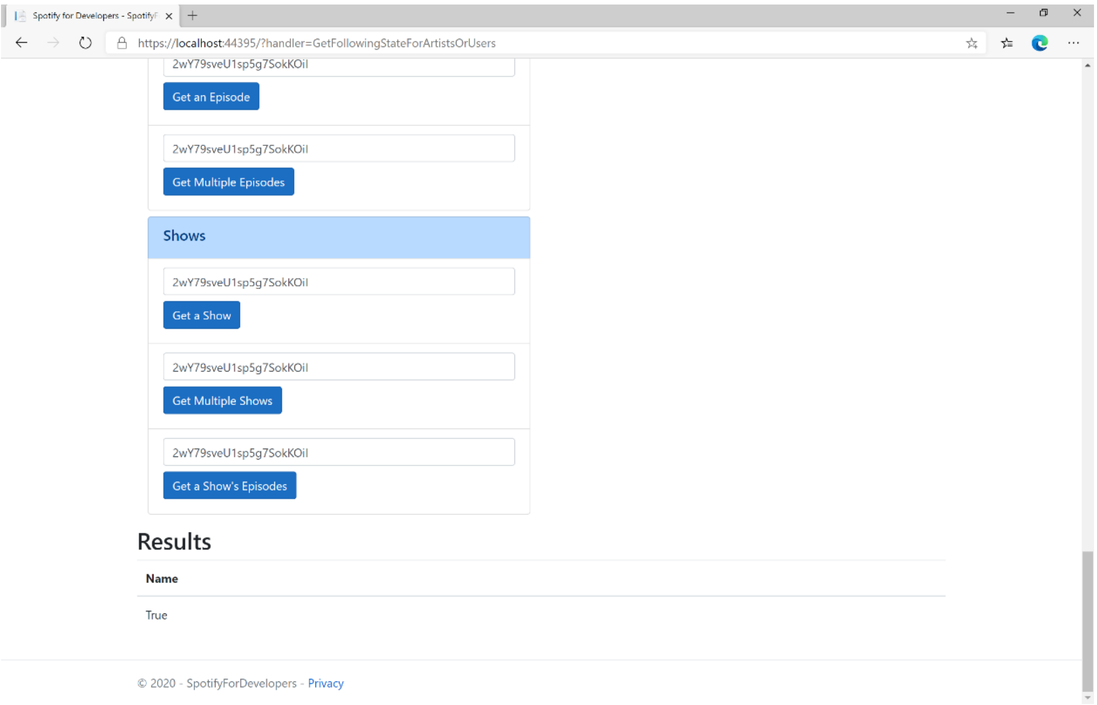
Album

User

User

Step 10

You can then enter an **Artist Id** from **Get All New Releases** or copy the same **Artist Id** from **Follow Artists or Users** to use and above **Get Following State for Artists or Users** choose **Artist** and then select **Get Following State for Artists or Users** and scroll down to view **Results** like the following:



https://localhost:44395/?handler=GetFollowingStateForArtistsOrUsers

2wY79sveU1sp5g7SokKOil

2wY79sveU1sp5g7SokKOil

2wY79sveU1sp5g7SokKOil

2wY79sveU1sp5g7SokKOil

2wY79sveU1sp5g7SokKOil

Results

Name
True

© 2020 - SpotifyForDevelopers - Privacy

Step 11

■

You can stop the **web application** in **Visual Studio 2019** by selecting the **Stop debugging** button

Step 12

×

You can choose to exit **Visual Studio 2019** by selecting the **Close** button in the top right of the **application** as that completes this part of the workshop

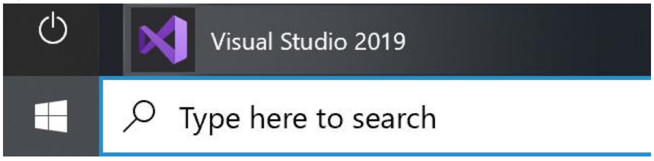
Check if Users Follow a Playlist

Check to see if one or more Spotify users are following a specified playlist.

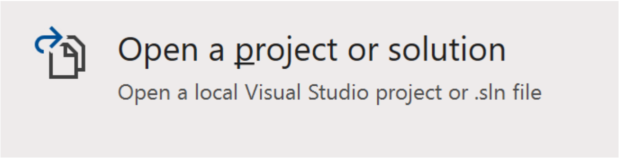
GET https://api.spotify.com/v1/playlists/{playlist_id}/followers/contains	
Header	
Authorization	User Token from Spotify Accounts service with **playlist-read-private** scope if a private playlist is requested
Path Parameter	
playlist_id	Spotify Id of the playlist.
Query Parameter	
ids	Comma-separated list of Spotify User Ids you want to check to see if they follow the playlist

Header	Response
Success	
HTTP Status 200 OK	Array of true or false values in same order as in which the Ids were provided
Error	
Error Code	Error Object

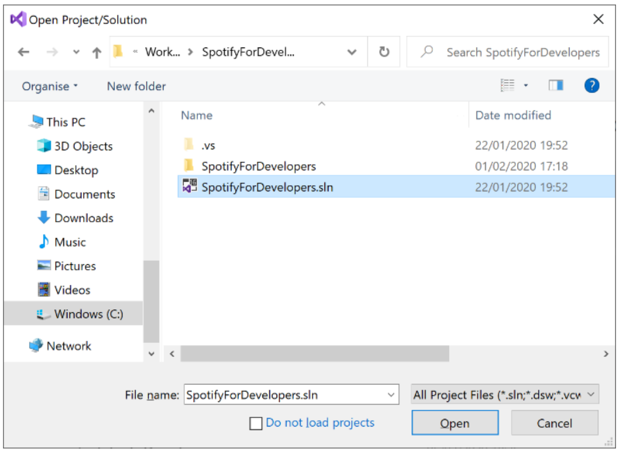
Step 1



If you chose to close **Visual Studio 2019** previously, in **Windows 10** choose **Start**, and then from the **Start Menu** find and select **Visual Studio 2019**


Open a project or solution
Open a local Visual Studio project or .sln file

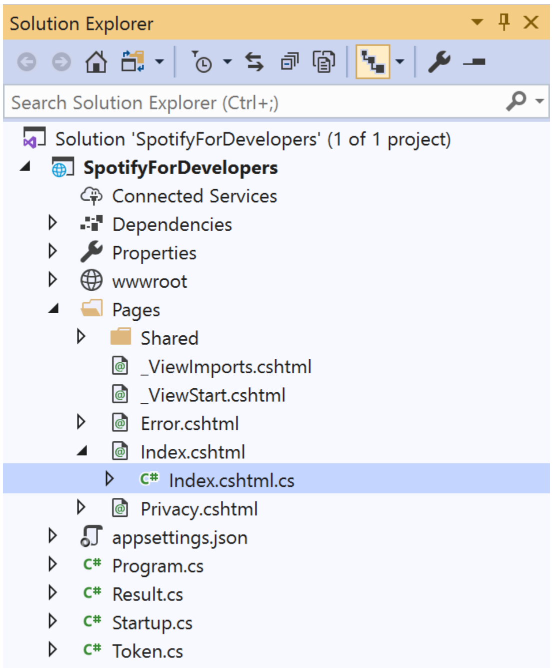
Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



File name: SpotifyForDevelopers.sln

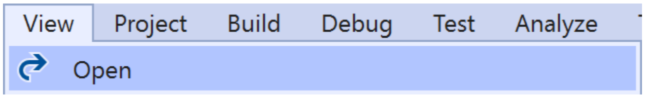
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started**, **Authorisation Guide**, **Search & Browse**, **Playlists & Artists**, **Albums & Tracks**, **Episodes & Shows** and **Follow**

Step 2



Once opened, in the **Solution Explorer** open the **Pages** section, then open the **Index.cshtml** section and select **Index.cshtml.cs**

Step 3



Then from the **Menu** choose **View** and then **Open**

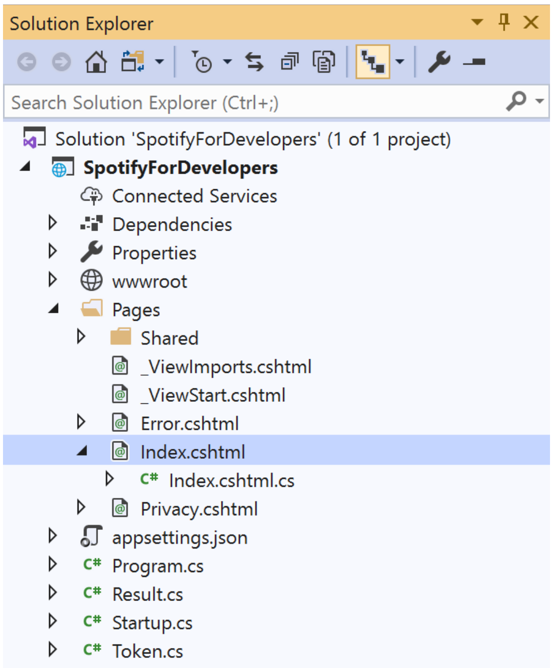
Step 4

In the **Code View** for **Index.cshtml.cs** below the **method** for `public async Task<IActionResult> OnPostGetFollowingStateForArtistsOrUsersAsync(...) { ... }` enter the following **method**:

```
public async Task<IActionResult> OnPostCheckUsersFollowingPlaylistAsync(string value, string
option)
{
    LoadToken();
    var values = value.Split(",").ToList();
    var results = await Api.CheckUsersFollowingPlaylistAsync(values, option);
    if (results != null)
    {
        Results = results.Select(result => new Result()
        {
            Name = result.ToString()
        });
    }
    return Page();
}
```

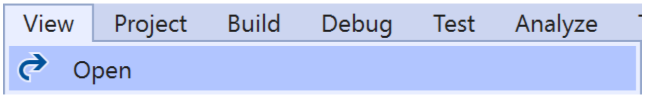
The **method** for `OnPostCheckUsersFollowingPlaylistAsync` is used to get the **following state** for a **playlist** by **Artist Id** for the **users** by **User Id** on Spotify with the `value` and `option` and populate the **property** for `Results` accordingly.

Step 5



In the **Solution Explorer** in the **Pages** section select **Index.cshtml**

Step 6



Then from the **Menu** choose **View** and then **Open**

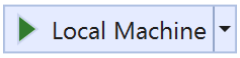
Step 7

Once in the **Code View** for **Index.cshtml** above `<!-- Follow -->` enter the following:

```
<li class="list-group-item">
    <form asp-page-handler="CheckUsersFollowingPlaylist" method="post">
        <input asp-for="Option" placeholder="Playlist Id" class="form-control mb-2" />
        <input asp-for="Value" placeholder="User Ids" class="form-control mb-2" />
        <button class="btn btn-primary mb-2">
            Check if Users Follow a Playlist
        </button>
    </form>
</li>
```

This `form` will **post** to the **method** for `CheckUsersFollowingPlaylist` with the `Value` of the **Playlist Id** and the `Option` for the **User Id** and will output to the **Results**.

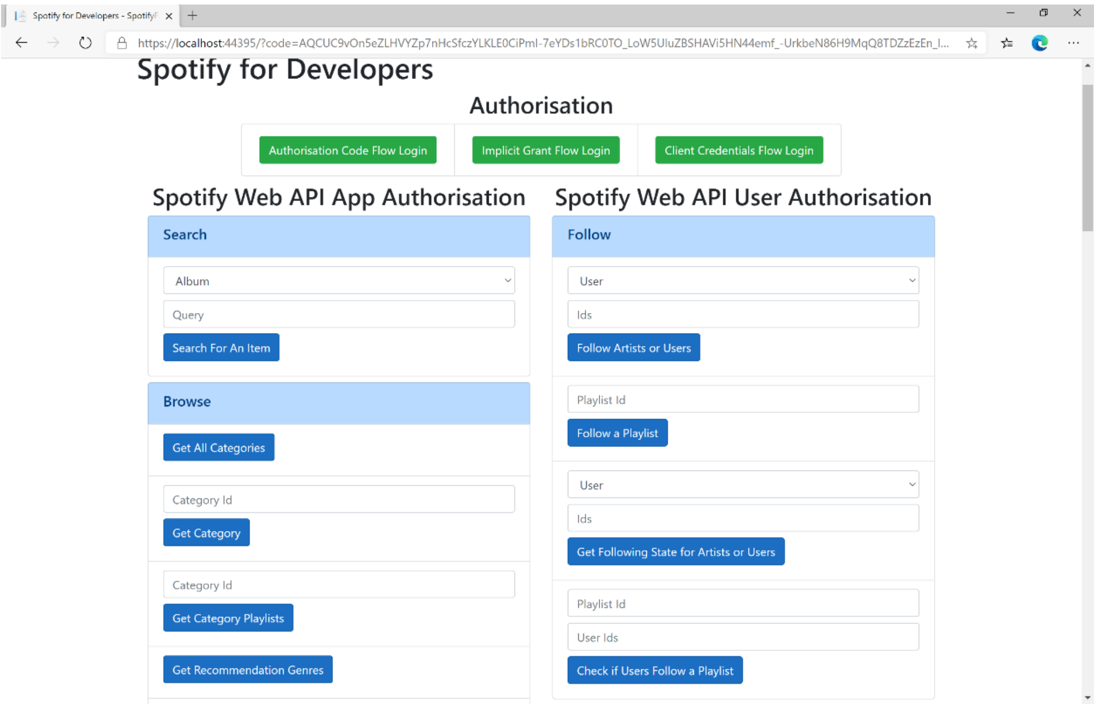
Step 8



Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**

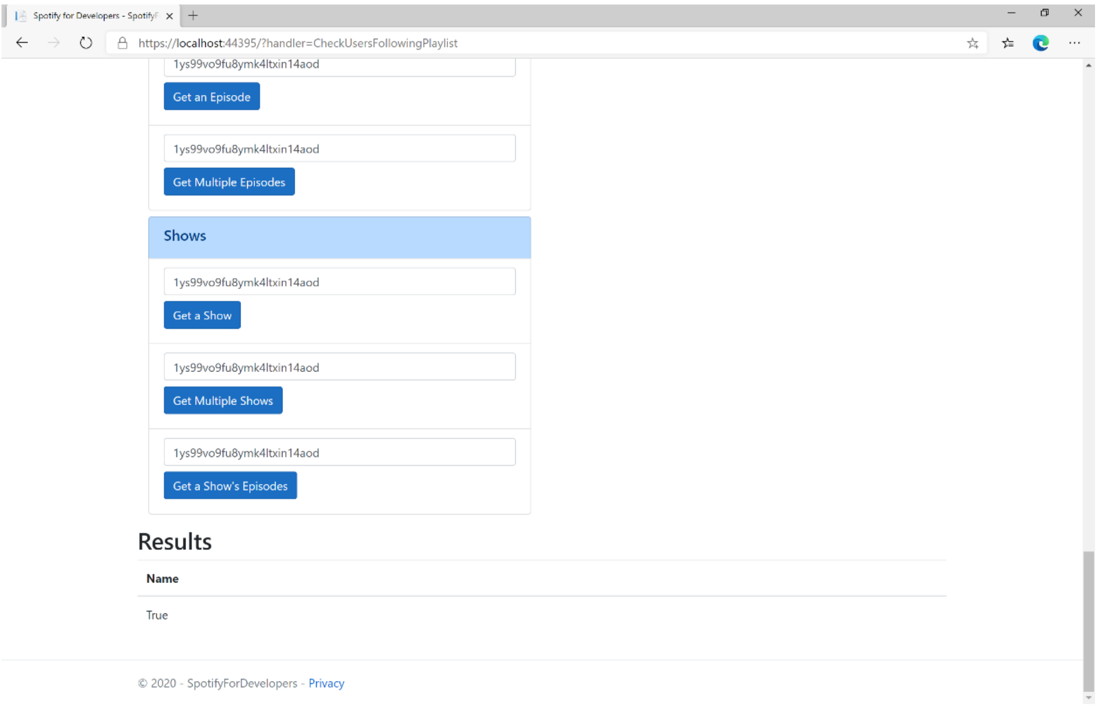
Step 9

Once the **Web Application** is running and you select **Authorisation Code Flow Login** or **Implicit Grant Flow Login** and scroll down you should see something like the following:


Spotify for Developers - Spotify
https://localhost:44395/?code=AQCUC9vOn5eZLHVYZp7nHcSfczYLKLE0CiPml-7eYDs1bRC0TO_LoW5UluZBSHAVi5HN44emf_-UrkbeN86H9MqQ8TDZzEzEn_I...
Spotify for Developers
Authorisation
Authorisation Code Flow Login
Implicit Grant Flow Login
Client Credentials Flow Login
Spotify Web API App Authorisation
Search
Album
Query
Search For An Item
Browse
Get All Categories
Category Id
Get Category
Category Id
Get Category Playlists
Get Recommendation Genres
Spotify Web API User Authorisation
Follow
User
Ids
Follow Artists or Users
Playlist Id
Follow a Playlist
User
Ids
Get Following State for Artists or Users
Playlist Id
User Ids
Check if Users Follow a Playlist

Step 10

Use the same **Playlist Id** from **Follow a Playlist** and use your own **User Id**, to get this in **Spotify** select your **Username** then **...** and **Copy Spotify URI** your **User Id** will be after **spotify:user:** then enter this above **Check if Users Follow a Playlist** and then select **Check if Users Follow a Playlist** and scroll down to view **Results** like the following:


Spotify for Developers - Spotify
https://localhost:44395/?handler=CheckUsersFollowingPlaylist
Get an Episode
Get Multiple Episodes
Shows
Get a Show
Get Multiple Shows
Get a Show's Episodes

1ys99vo9fu8ymk4ltxin14aod

1ys99vo9fu8ymk4ltxin14aod

1ys99vo9fu8ymk4ltxin14aod

1ys99vo9fu8ymk4ltxin14aod

1ys99vo9fu8ymk4ltxin14aod

1ys99vo9fu8ymk4ltxin14aod

Results

Name
True

© 2020 - SpotifyForDevelopers - Privacy

Step 11

You can stop the **web application** in **Visual Studio 2019** by selecting the **Stop debugging** button

Step 12

You can choose to exit **Visual Studio 2019** by selecting the **Close** button in the top right of the **application** as that completes this part of the workshop

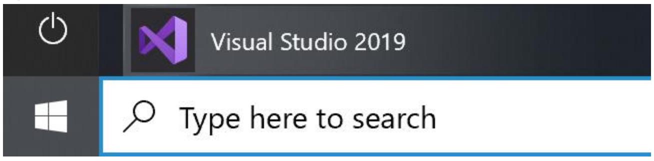
Get User's Followed Artists

Get the current user's followed artists.

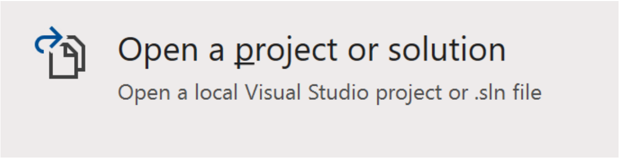
GET https://api.spotify.com/v1/me/following	
Header	
Authorization	User Token from Spotify Accounts service with **user-follow-modify** scope
Query Parameter	
type	Id type, only artist is currently supported
limit	Maximum number of items to return. Default: 20. Minimum: 1. Maximum: 50.
after	Last artist Id retrieved from the previous request.

Header	**Response**
Success	
HTTP Status 200 OK	Object containing a cursor-based paging object of Artist Objects
Error	
Error Code	Error Object

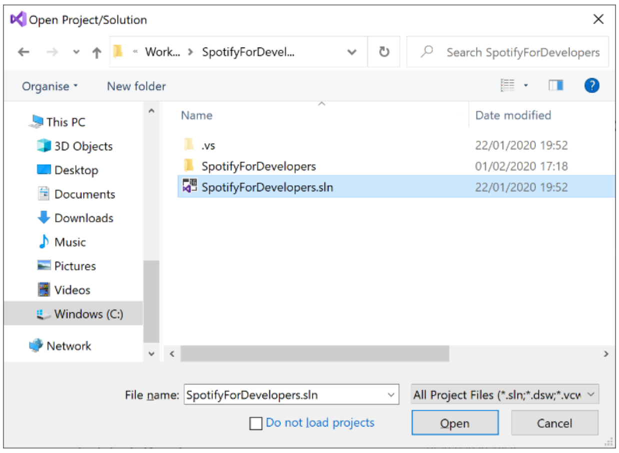
Step 1



If you chose to close **Visual Studio 2019** previously, in **Windows 10** choose **Start**, and then from the **Start Menu** find and select **Visual Studio 2019**


Open a project or solution
Open a local Visual Studio project or .sln file

Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



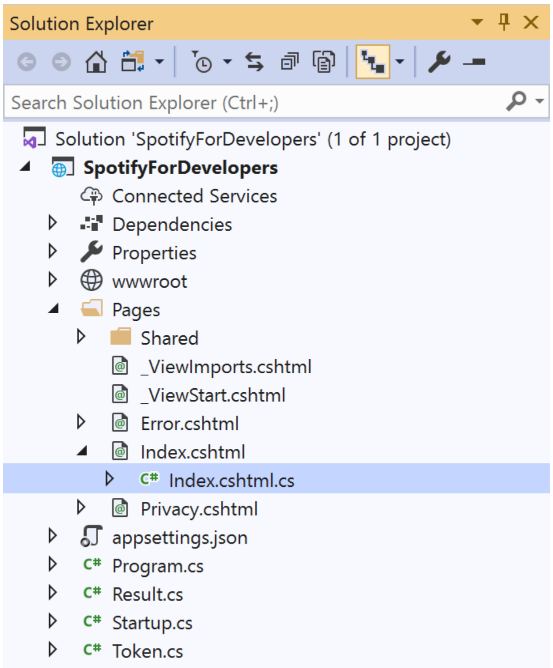
Name	Date modified
.vs	22/01/2020 19:52
SpotifyForDevelopers	01/02/2020 17:18
SpotifyForDevelopers.sln	22/01/2020 19:52

File name: SpotifyForDevelopers.sln

All Project Files (*.sln;*.dsw;*.vcw

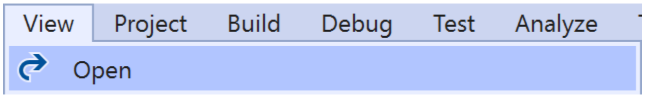
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started**, **Authorisation Guide**, **Search & Browse**, **Playlists & Artists**, **Albums & Tracks**, **Episodes & Shows** and **Follow**

Step 2



Once opened, in the **Solution Explorer** open the **Pages** section, then open the **Index.cshtml** section and select **Index.cshtml.cs**

Step 3



Then from the **Menu** choose **View** and then **Open**

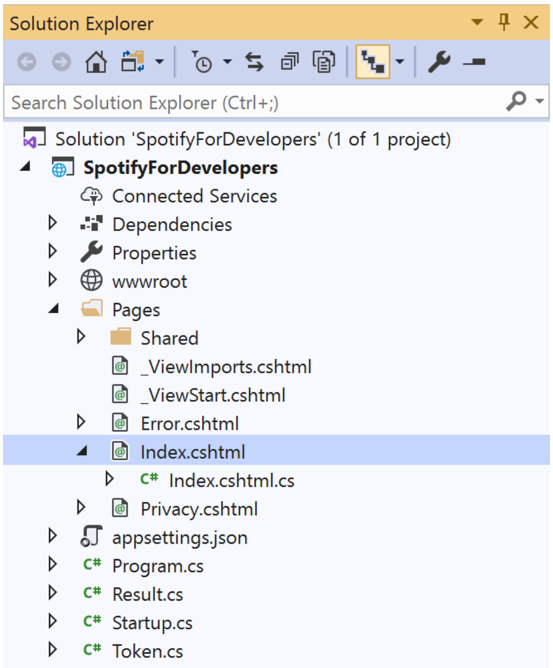
Step 4

In the **Code View** for **Index.cshtml.cs** below the **method** for `public async Task<IActionResult> OnPostCheckUsersFollowingPlaylistAsync(...) { ... }` enter the following **method**:

```
public async Task<IActionResult> OnPostGetUsersFollowedArtistsAsync()
{
    LoadToken();
    var results = await Api.GetUsersFollowedArtistsAsync();
    if (results != null)
    {
        Results = results.Items.Select(result => new Result()
        {
            Id = result.Id,
            Name = result.Name,
            Image = result?.Images?.FirstOrDefault()?.Url
        });
    }
    return Page();
}
```

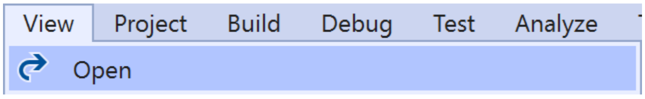
The **method** for `OnPostGetUsersFollowedArtistsAsync` is used to get the **followed artists** for the logged in **user** and populate the **property** for `Results` accordingly.

Step 5



In the **Solution Explorer** in the **Pages** section select **Index.cshtml**

Step 6



Then from the **Menu** choose **View** and then **Open**

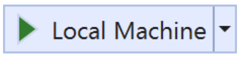
Step 7

Once in the **Code View** for **Index.cshtml** above `<!-- Follow -->` enter the following:

```
<li class="list-group-item">
    <form asp-page-handler="GetUsersFollowedArtists" method="post">
        <button class="btn btn-primary mb-2">
            Get User's Followed Artists
        </button>
    </form>
</li>
```

This `form` will **post** to the **method** for `GetUsersFollowedArtists` and will output to the **Results**.

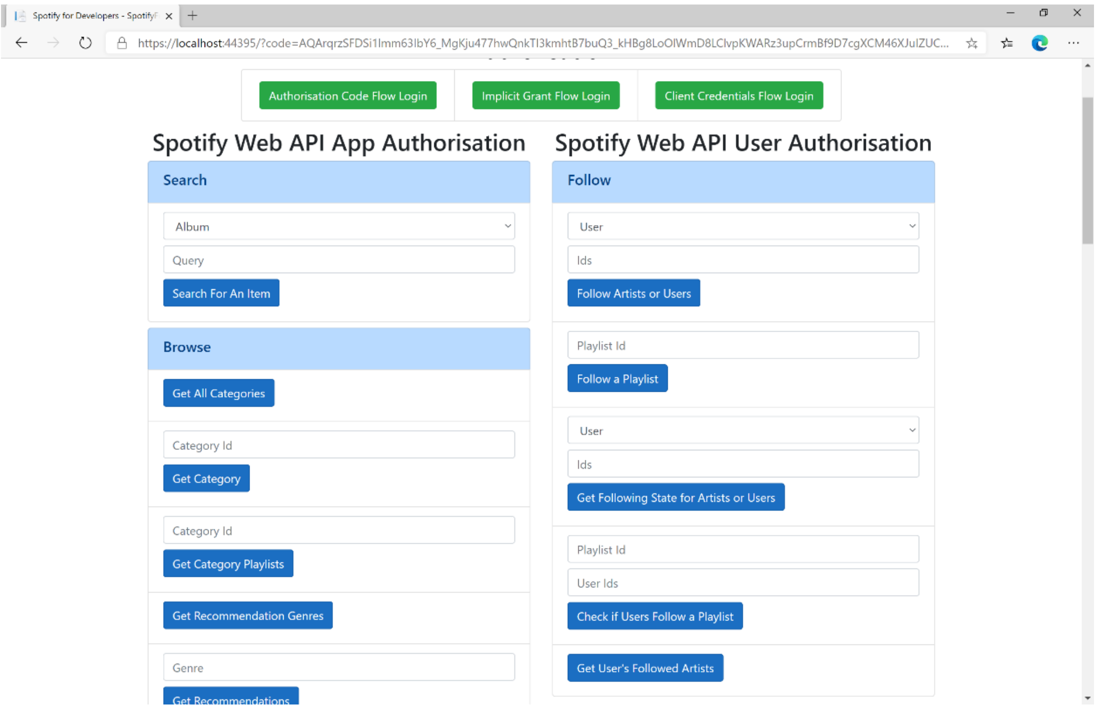
Step 8



Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**

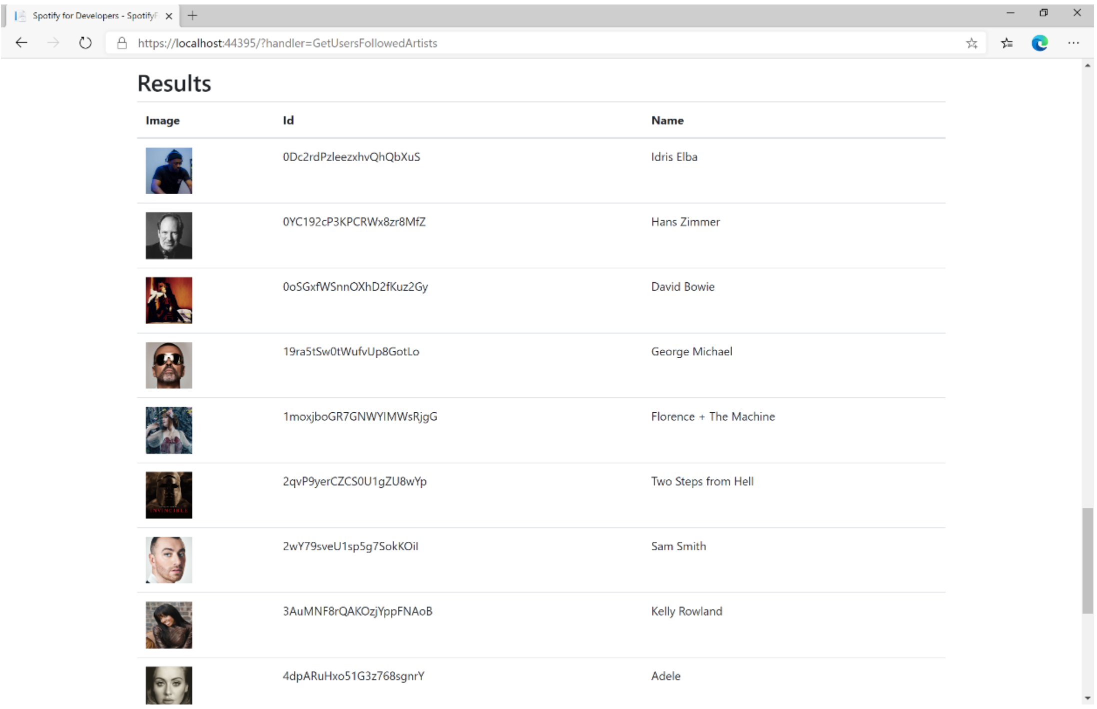
Step 9

Once the **Web Application** is running and you select **Authorisation Code Flow Login** or **Implicit Grant Flow Login** and scroll down you should see something like the following:


Spotify for Developers - Spotify
https://localhost:44395/?code=AQArqrzSFDSi1Imm63lbY6_MgKju477hwQnkTI3kmhtB7buQ3_kHBg8LoOlWmD8LClvpKWARz3upCrmBf9D7cgXCM46XJuIZUC...
Authorisation Code Flow Login
Implicit Grant Flow Login
Client Credentials Flow Login
Spotify Web API App Authorisation
Spotify Web API User Authorisation
Search
Album
Query
Search For An Item
Browse
Get All Categories
Category Id
Get Category
Category Id
Get Category Playlists
Get Recommendation Genres
Genre
Get Recommendations
Follow
User
Ids
Follow Artists or Users
Playlist Id
Follow a Playlist
User
Ids
Get Following State for Artists or Users
Playlist Id
User Ids
Check if Users Follow a Playlist
Get User's Followed Artists

Step 10

You can select **Get User's Followed Artists** and scroll down to view **Results** like the following:


Spotify for Developers - Spotify
https://localhost:44395/?handler=GetUsersFollowedArtists

Results

Image	Id	Name
	0Dc2rdPzleezxhvQhQbXuS	Idris Elba
	0YC192cP3KPCRWx8zr8MfZ	Hans Zimmer
	0oSGxfWSnnOXhD2fKuz2Gy	David Bowie
	19ra5tSw0tWufvUp8GotLo	George Michael
	1moxjboGR7GNWYIMWsRjgG	Florence + The Machine
	2qvP9yerCZCS0U1gZU8wYp	Two Steps from Hell
	2wY79sveU1sp5g7SokKOil	Sam Smith
	3AuMNF8rQAKOzjYppFNAoB	Kelly Rowland
	4dpARuHxo51G3z768sgnrY	Adele

Step 11

You can stop the **web application** in **Visual Studio 2019** by selecting the **Stop debugging** button

Step 12

You can choose to exit **Visual Studio 2019** by selecting the **Close** button in the top right of the **application** as that completes this part of the workshop

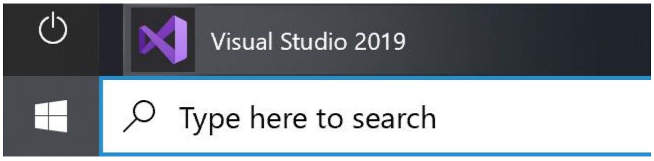
Unfollow Artists or Users

Remove the current user as a follower of one or more artists or other Spotify users.

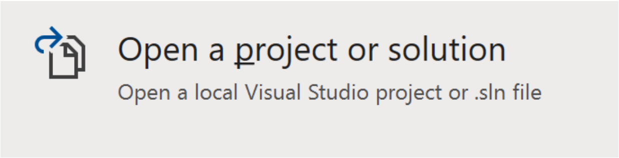
DELETE https://api.spotify.com/v1/me/following	
Header	
Authorization	User Token from Spotify Accounts service with **user-follow-modify** scope
Query Parameter	
type	Type of artist or user
ids	Comma-separated list of the artist or the user Spotify Ids - maximum 50

Header	**Response**
Success	
HTTP Status 204 No Content	Empty Response Body
Error	
Error Code	Error Object

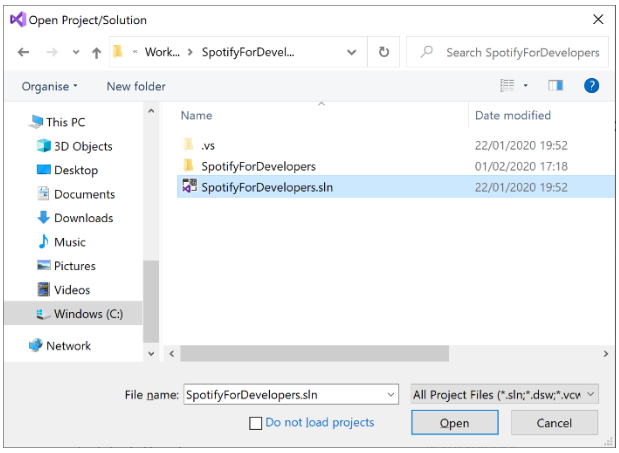
Step 1



If you chose to close **Visual Studio 2019** previously, in **Windows 10** choose **Start**, and then from the **Start Menu** find and select **Visual Studio 2019**


Open a project or solution
Open a local Visual Studio project or .sln file

Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



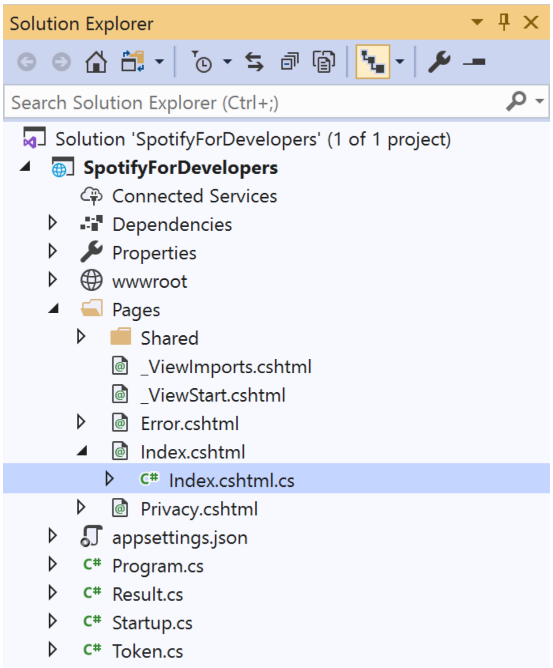
Name	Date modified
.vs	22/01/2020 19:52
SpotifyForDevelopers	01/02/2020 17:18
SpotifyForDevelopers.sln	22/01/2020 19:52

File name: SpotifyForDevelopers.sln

All Project Files (*.sln;*.dsw;*.vcw

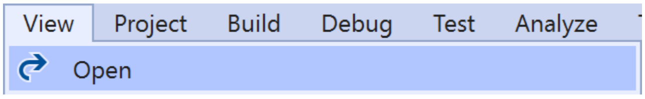
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started**, **Authorisation Guide**, **Search & Browse**, **Playlists & Artists**, **Albums & Tracks**, **Episodes & Shows** and **Follow**

Step 2



Once opened, in the **Solution Explorer** open the **Pages** section, then open the **Index.cshtml** section and select **Index.cshtml.cs**

Step 3



Then from the **Menu** choose **View** and then **Open**

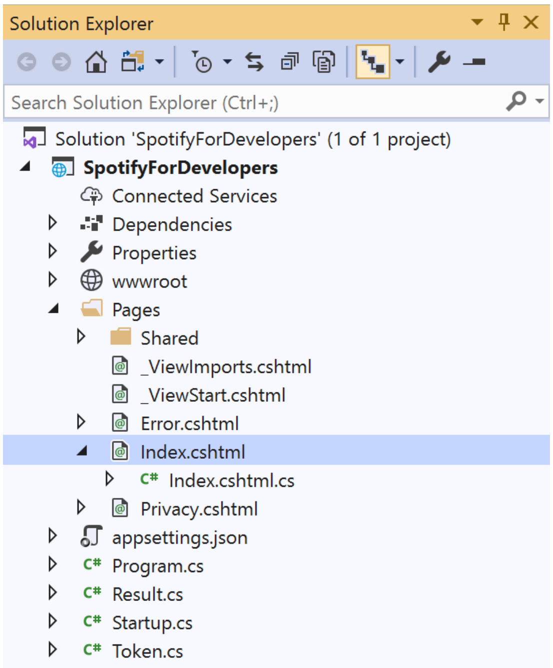
Step 4

In the **Code View** for **Index.cshtml.cs** below the **method** for `public async Task<IActionResult> OnPostGetUsersFollowedArtistsAsync(...) { ... }` enter the following **method**:

```
public async Task<IActionResult> OnPostUnfollowArtistsOrUsersAsync(string value, FollowType
option)
{
    LoadToken();
    var values = value.Split(",").ToList();
    var result = await Api.UnfollowArtistsOrUsersAsync(values, option);
    if (result != null)
    {
        Results = new List<Result>() { new Result()
        {
            Name = result.Success.ToString()
        }};
    }
    return Page();
}
```

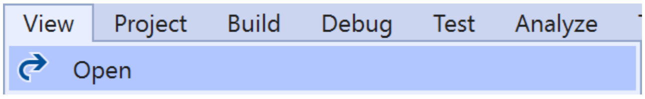
The **method** for `OnPostUnfollowArtistsOrUsersAsync` is used to **unfollow** an **artist** or **user** by **Artist Id** or **User Id** on Spotify with the `value` and selected `option` and populate the **property** for `Results` accordingly with the **success** of the operation.

Step 5



In the **Solution Explorer** in the **Pages** section select **Index.cshtml**

Step 6



Then from the **Menu** choose **View** and then **Open**

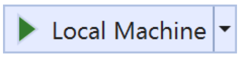
Step 7

Once in the **Code View** for **Index.cshtml** above `<!-- Follow -->` enter the following:

```
<li class="list-group-item">
    <form asp-page-handler="UnfollowArtistsOrUsers" method="post">
        <select asp-for="Option" class="form-control mb-2"
            asp-items="Html.GetEnumSelectList<Spotify.NetStandard.Enums.FollowType>()">
        </select>
        <input asp-for="Value" placeholder="Id" class="form-control mb-2" />
        <button class="btn btn-primary mb-2">
            Unfollow Artists or Users
        </button>
    </form>
</li>
```

This `form` will **post** to the **method** for `UnfollowArtistsOrUsers` with the `Value` of the **Ids** and the `Option` for the **Artist** or **User** and will output to the **Results**.

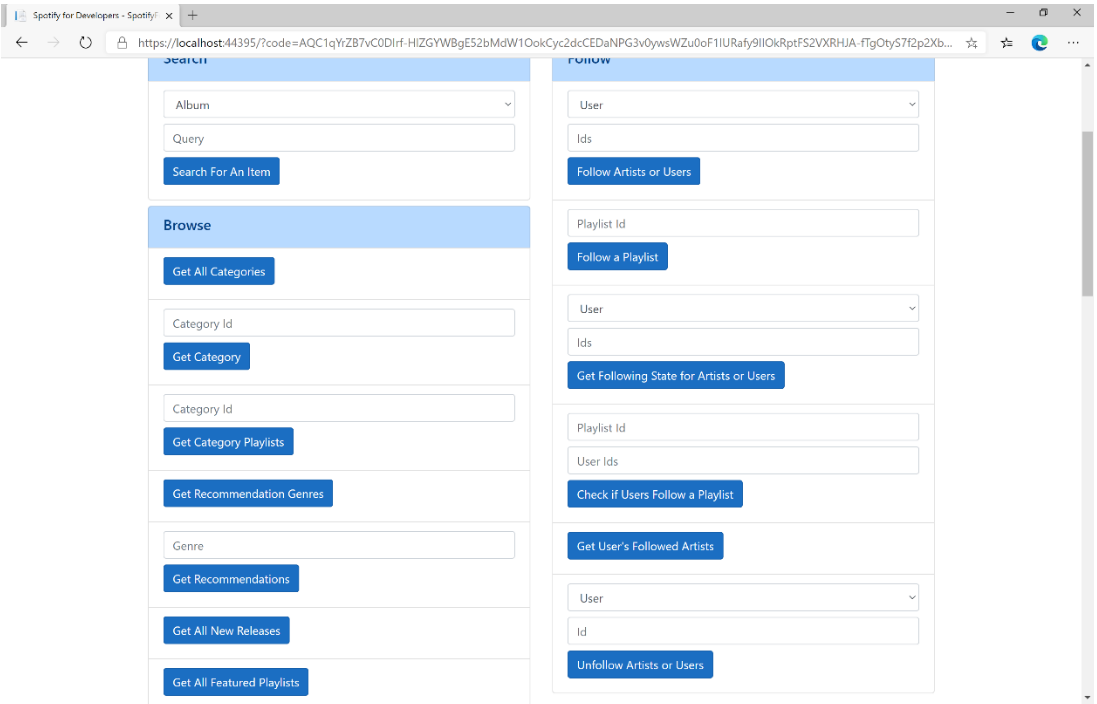
Step 8



Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**

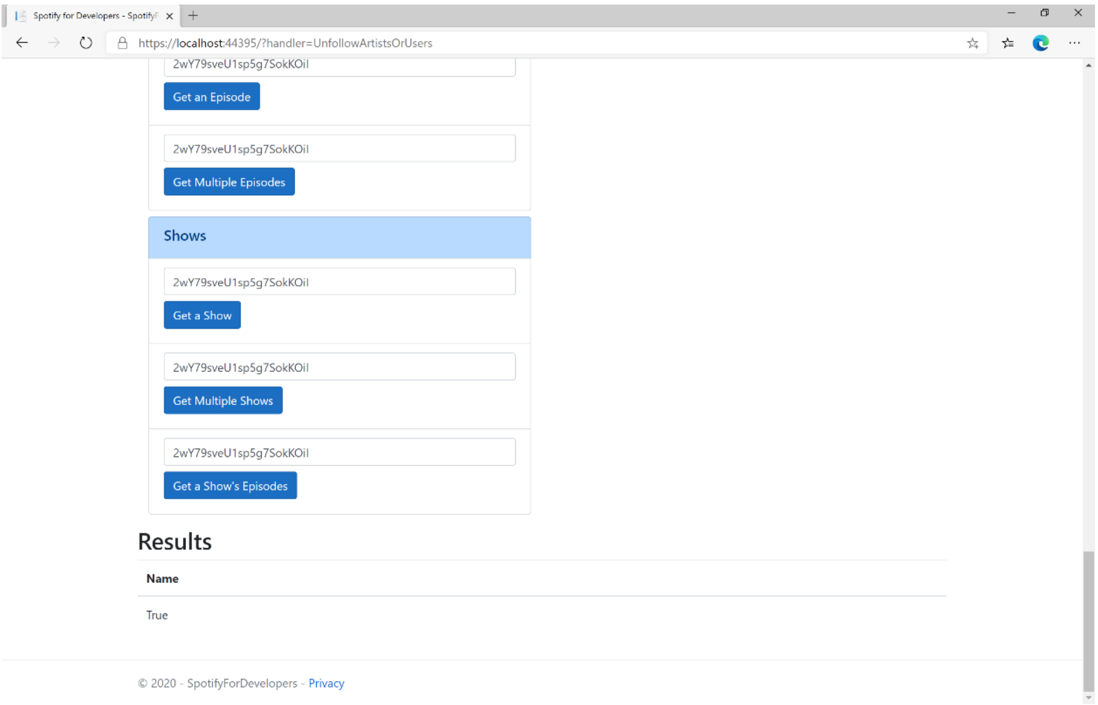
Step 9

Once the **Web Application** is running and you select **Authorisation Code Flow Login** or **Implicit Grant Flow Login** and scroll down you should see something like the following:


Spotify for Developers - Spotify
https://localhost:44395/?code=AQC1qYrZB7vC0DIrf-HIZGYWBgE52bMdW1OokCyc2dcCEDaNPG3v0ywsWZu0oF1lURafy9llOkRptFS2VXRHJA-fTgOtyS7f2p2Xb...
Search
Album
Query
Search For An Item
Browse
Get All Categories
Category Id
Get Category
Category Id
Get Category Playlists
Get Recommendation Genres
Genre
Get Recommendations
Get All New Releases
Get All Featured Playlists
Follow
User
Ids
Follow Artists or Users
Playlist Id
Follow a Playlist
User
Ids
Get Following State for Artists or Users
Playlist Id
User Ids
Check if Users Follow a Playlist
Get User's Followed Artists
User
Id
Unfollow Artists or Users

Step 10

You can then enter an **Artist Id** from **Get All New Releases** or the same **Artist Id** from **Follow Artists or Users** to use and above **Unfollow Artists or Users** choose **Artist** and then select **Unfollow Artists or Users** and scroll down to view **Results** like the following:



Results

Name
True

© 2020 - SpotifyForDevelopers - Privacy

Step 11

You can stop the **web application** in **Visual Studio 2019** by selecting the **Stop debugging** button

Step 12

You can choose to exit **Visual Studio 2019** by selecting the **Close** button in the top right of the **application** as that completes this part of the workshop

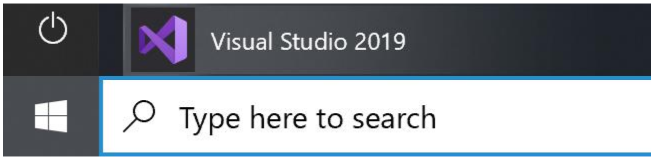
Unfollow Playlist

Remove the current user as a follower of a playlist.

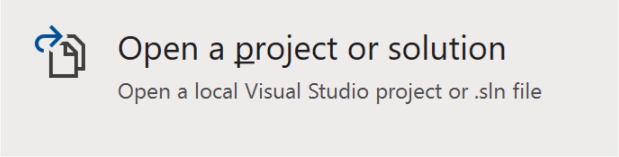
DELETE https://api.spotify.com/v1/playlists/{playlist_id}/followers	
Header	
Authorization	User Token from Spotify Accounts service with **playlist-modify-public** scope for publicly followed playlist and **playlist-modify-private** for a privately followed playlist
Path Parameter	
playlist_id	Spotify Id of the playlist

Header	**Response**
Success	
HTTP Status 200 OK	Empty Response Body
Error	
Error Code	Error Object

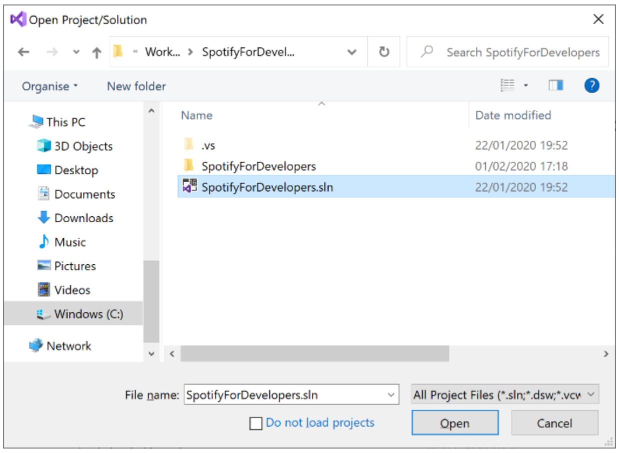
Step 1



If you chose to close **Visual Studio 2019** previously, in **Windows 10** choose **Start**, and then from the **Start Menu** find and select **Visual Studio 2019**


Open a project or solution
Open a local Visual Studio project or .sln file

Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



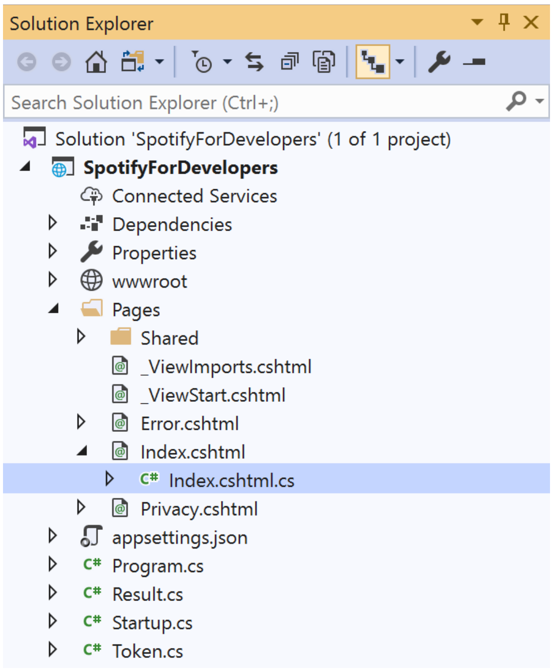
Name	Date modified
.vs	22/01/2020 19:52
SpotifyForDevelopers	01/02/2020 17:18
SpotifyForDevelopers.sln	22/01/2020 19:52

File name: SpotifyForDevelopers.sln

All Project Files (*.sln;*.dsw;*.vcw

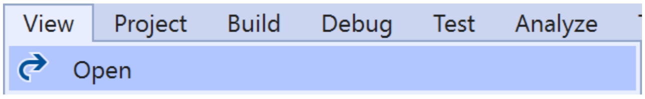
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started**, **Authorisation Guide**, **Search & Browse**, **Playlists & Artists**, **Albums & Tracks**, **Episodes & Shows** and **Follow**

Step 2



Once opened, in the **Solution Explorer** open the **Pages** section, then open the **Index.cshtml** section and select **Index.cshtml.cs**

Step 3



Then from the **Menu** choose **View** and then **Open**

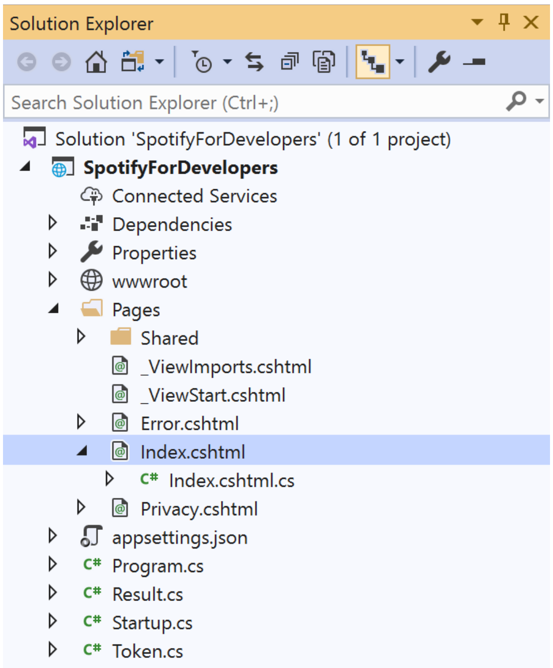
Step 4

In the **Code View** for **Index.cshtml.cs** below the **method** for `public async Task<IActionResult> OnPostUnfollowArtistsOrUsersAsync(...) { ... }` enter the following **method**:

```
public async Task<IActionResult> OnPostUnfollowPlaylistAsync(string value)
{
    LoadToken();
    var result = await Api.UnfollowPlaylistAsync(value);
    if (result != null)
    {
        Results = new List<Result>() { new Result()
        {
            Name = result.Success.ToString()
        }};
    }
    return Page();
}
```

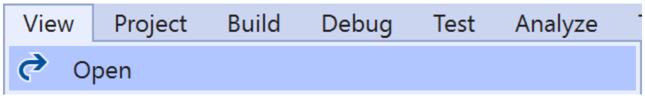
The **method** for `OnPostUnfollowPlaylistAsync` is used to **unfollow** a **playlist** by **Playlist Id** on Spotify with the `value` and populate the **property** for `Results` accordingly with the **success** of the operation.

Step 5



In the **Solution Explorer** in the **Pages** section select **Index.cshtml**

Step 6



Then from the **Menu** choose **View** and then **Open**

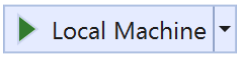
Step 7

Once in the **Code View** for **Index.cshtml** above `<!-- Follow -->` enter the following:

```
<li class="list-group-item">
    <form asp-page-handler="UnfollowPlaylist" method="post">
        <input asp-for="Value" placeholder="Playlist Id" class="form-control mb-2" />
        <button class="btn btn-primary mb-2">
            Unfollow a Playlist
        </button>
    </form>
</li>
```

This `form` will **post** to the **method** for `UnfollowPlaylist` with the `Value` of the **Playlist Id** and will output to the **Results**.

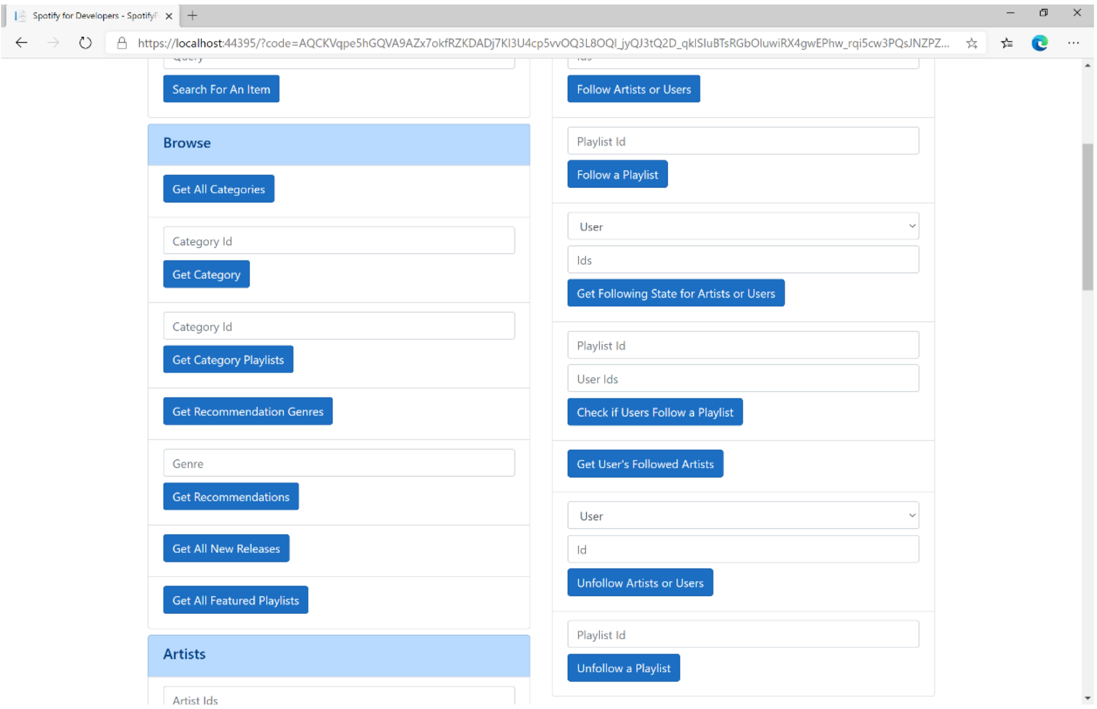
Step 8



Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**

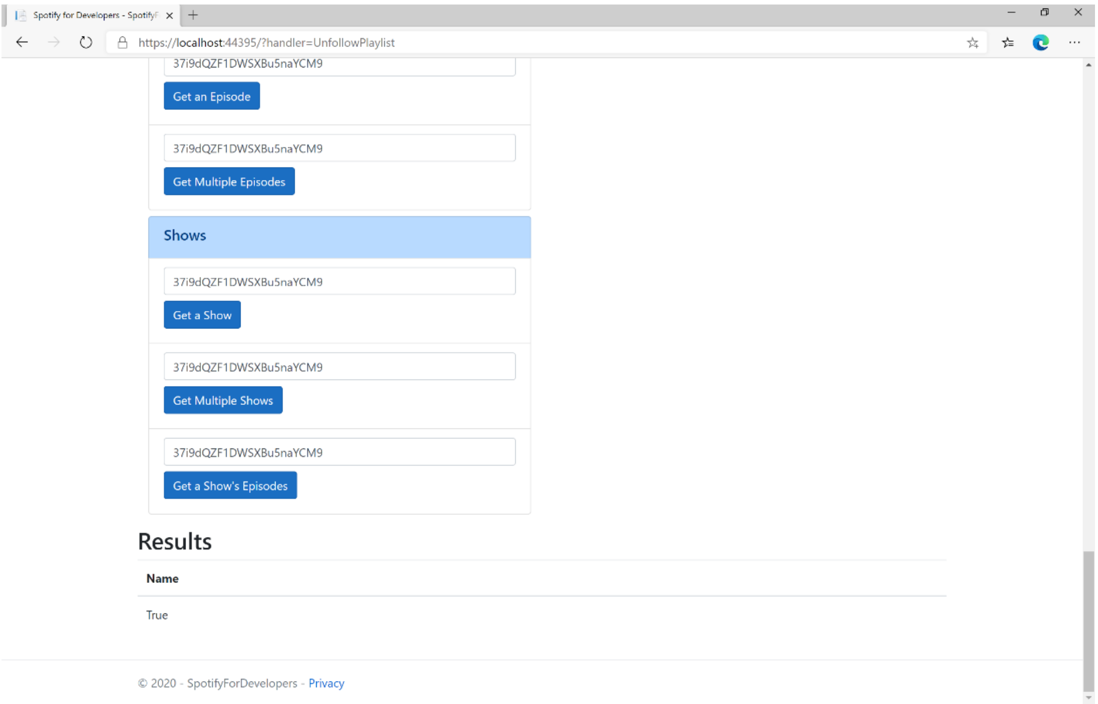
Step 9

Once the **Web Application** is running and you select **Authorisation Code Flow Login** or **Implicit Grant Flow Login** and scroll down you should see something like the following:


Spotify for Developers - Spotify
https://localhost:44395/?code=AQCKVqpe5hGQVA9AZx7okfRZKDADj7KI3U4cp5vvOQ3L8OQI_jyQJ3tQ2D_qklSluBTsRGbOluwiRX4gwEPhw_rqi5cw3PQsJNZPZ...
Search For An Item
Browse
Get All Categories
Category Id
Get Category
Category Id
Get Category Playlists
Get Recommendation Genres
Genre
Get Recommendations
Get All New Releases
Get All Featured Playlists
Artists
Artist Ids
Follow Artists or Users
Playlist Id
Follow a Playlist
User
Ids
Get Following State for Artists or Users
Playlist Id
User Ids
Check if Users Follow a Playlist
Get User's Followed Artists
User
Id
Unfollow Artists or Users
Playlist Id
Unfollow a Playlist

Step 10

You can then enter a **Playlist Id** from **Get All Featured Playlists** or the same **Playlist Id** from **Follow a Playlist** to use and select **Unfollow a Playlist** and scroll down to view **Results** like the following:



Results

Name
True

© 2020 - SpotifyForDevelopers - Privacy

Step 11

■

You can stop the **web application** in **Visual Studio 2019** by selecting the **Stop debugging** button

Step 12

✕

You can exit **Visual Studio 2019** by selecting the **Close** button in the top right of the **application** as that completes this part of the workshop