



Tutorialr.com

# Spotify for Developers Library

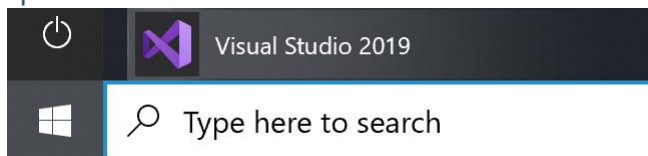
## Save Albums for Current User

Save one or more albums to the current user's 'Your Music' library.

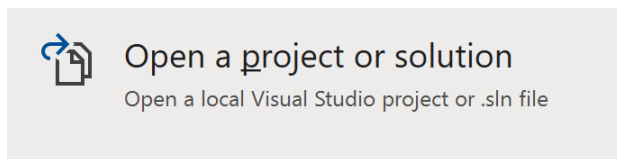
PUT <a href="https://api.spotify.com/v1/me/albums?ids={ids}">https://api.spotify.com/v1/me/albums?ids={ids}</a>	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-modify</b> scope
Query Parameter	
<b>ids</b>	Comma-separated list of the Spotify Ids for the Albums. Maximum: 50 Ids.

Header	Response
Success	
Http Status 201 Created	
Error	
HTTP Status 403 Forbidden	Returned when not Authorised
Error Code	Error Object

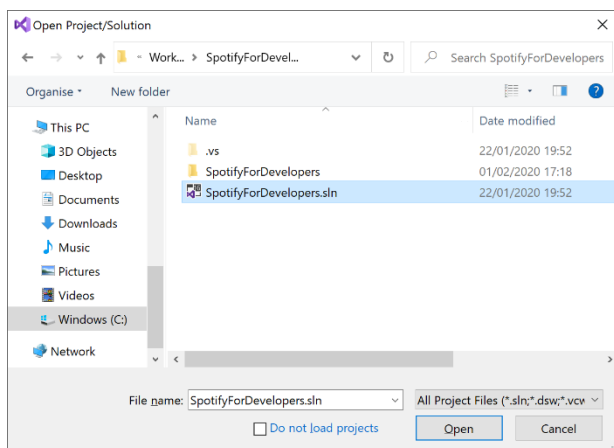
### Step 1



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

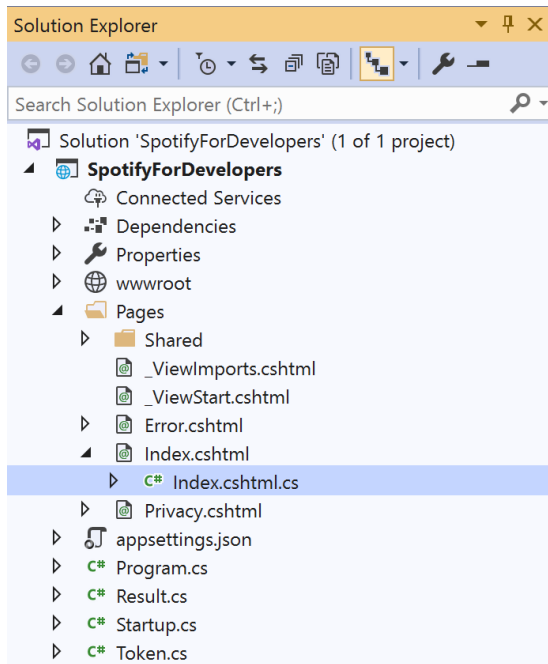


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



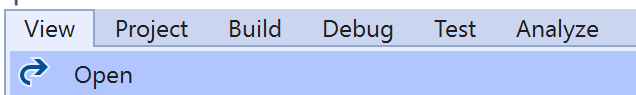
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow** and **Playlists**

## 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**

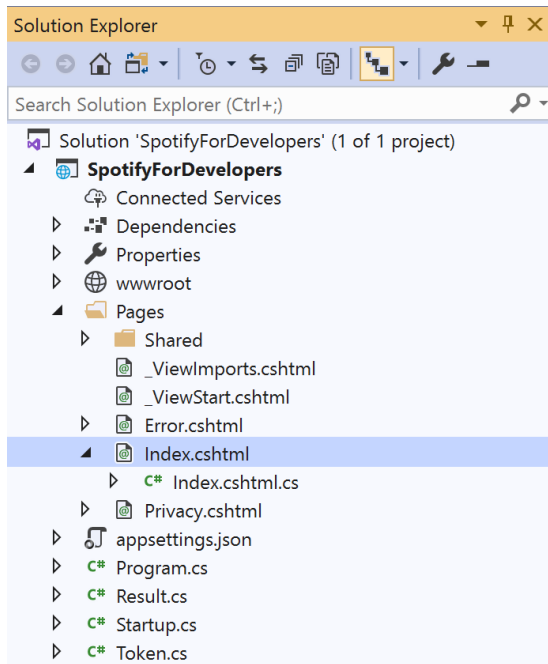
## Step 4

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

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

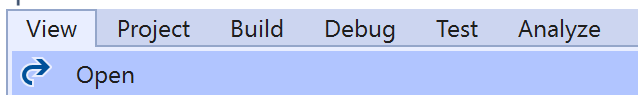
The **method** for `OnPostSaveUserAlbumsAsync` **saves** the **albums** to the **Library** with Value of the **Album Ids** of the **album** and populate the **property** for Results of the **success** of the operation accordingly.

## Step 5



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

## Step 6



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

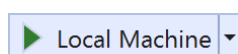
## Step 7

Once in the **Code View** for **Index.cshtml** above `<!-- 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">Library</h5>
  </li>
  <li class="list-group-item">
    <form asp-page-handler="SaveUserAlbums" method="post">
      <input asp-for="Value" placeholder="Album Ids" class="form-control mb-2" />
      <button class="btn btn-primary mb-2">
        Save Albums for Current User
      </button>
    </form>
  </li>
  <!-- Library -->
</ul>
```

This form will **post** to the **method** for SaveUserAlbums with the Value of the **Album Ids** and will output to the **Results**.

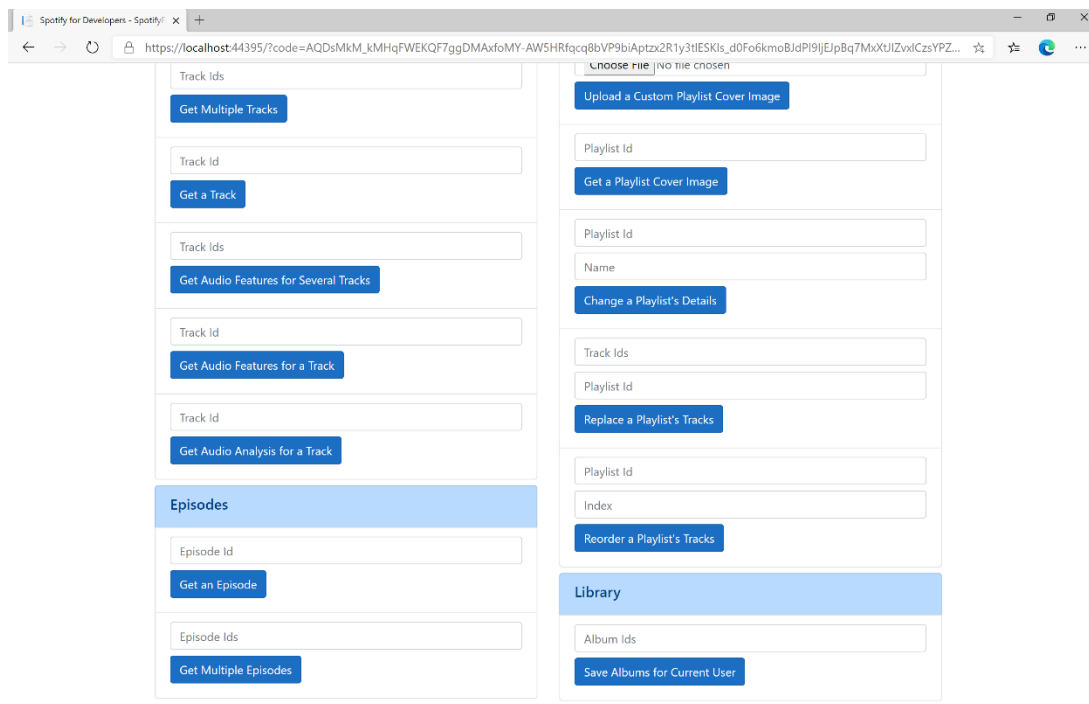
## Step 8



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

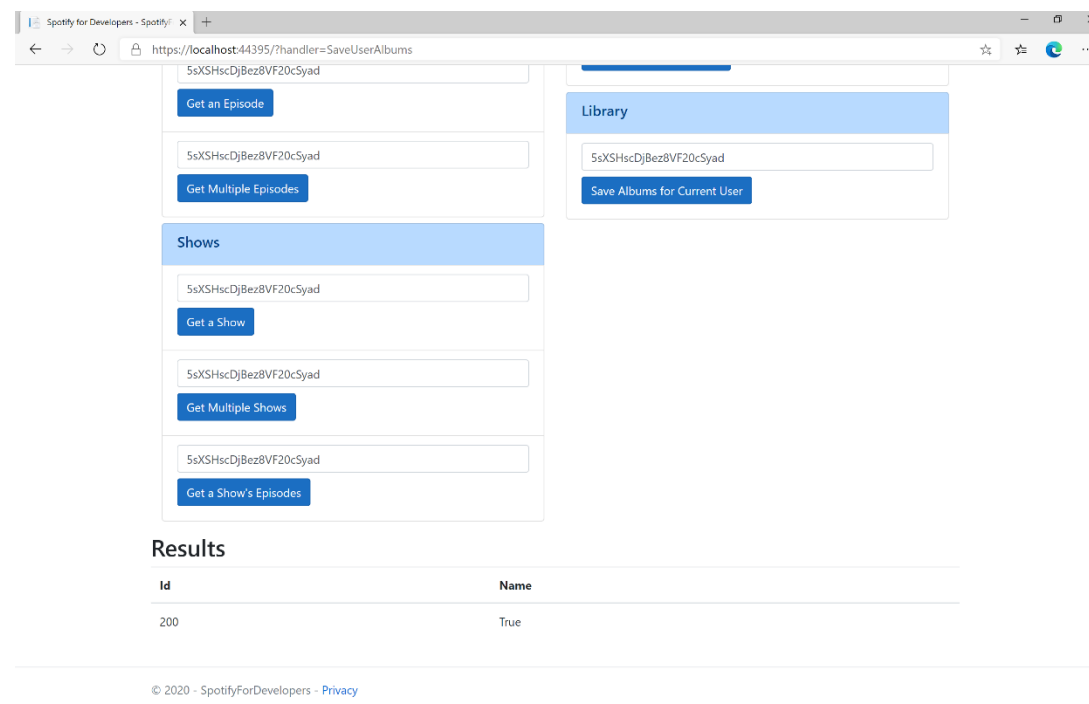
## Step 9

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:



## Step 10

You can then enter some **Album Ids** from **Get All New Releases** above **Save Albums for Current User** and then select **Save Albums for Current User** and scroll down to view **Results** like the following:



## 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

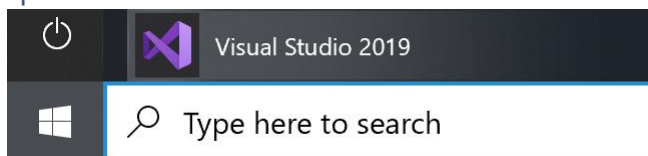
## Save Tracks for User

Save one or more tracks to the current user's 'Your Music' library.

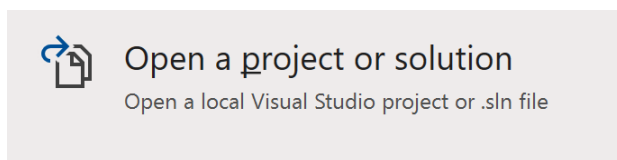
PUT <a href="https://api.spotify.com/v1/me/tracks">https://api.spotify.com/v1/me/tracks</a>	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-modify</b> scope
Query Parameter	
<b>ids</b>	Comma-separated list of the Spotify Ids for the Tracks. Maximum: 50 Ids.

Header	Response
Success	
Http Status 200 OK	
Error	
HTTP Status 403 Forbidden	Returned when not Authorised or more than 10,000 tracks in Your Music
Error Code	Error Object

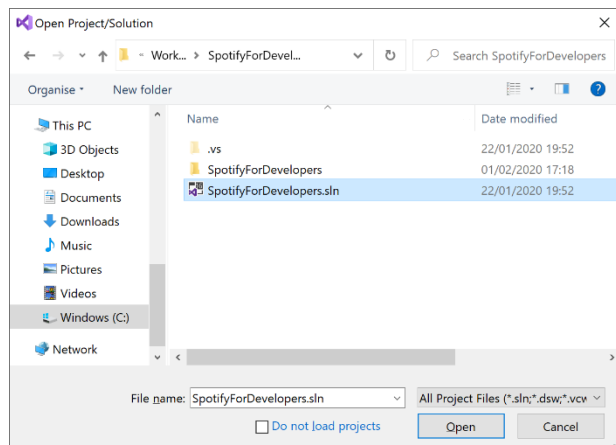
### Step 1



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**

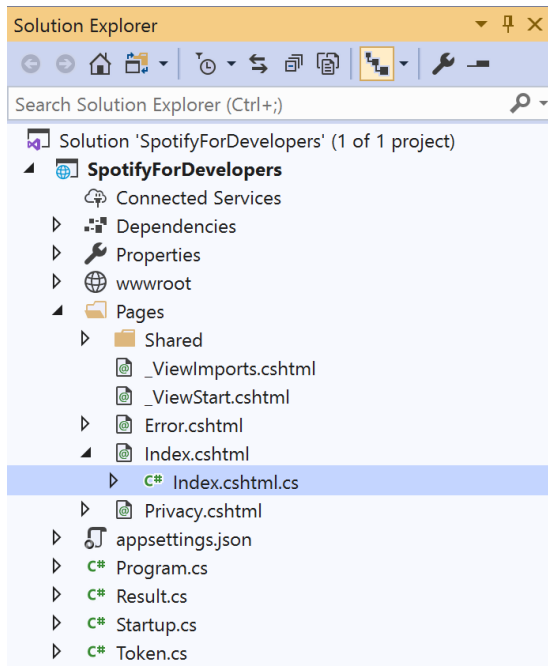


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



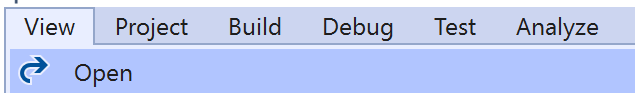
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow, Playlists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

## 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**

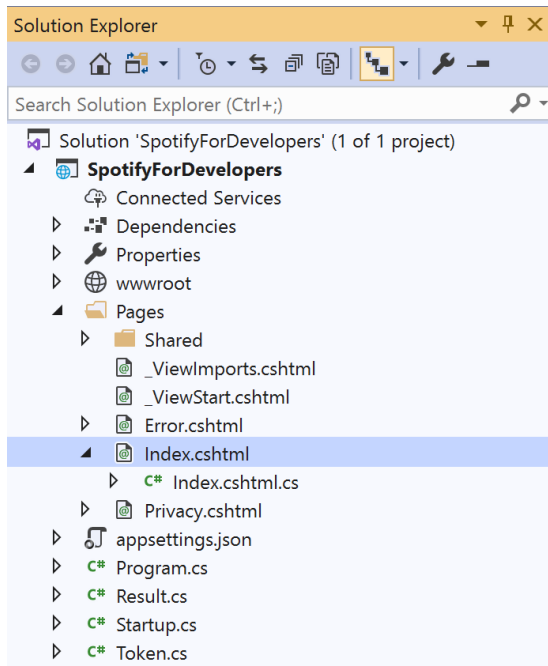
## Step 4

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

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

The **method** for `OnPostSaveUserTracksAsync` **saves** the **tracks** to the **Library** with Value of the **Track Ids** of the **track** and populate the **property** for Results of the **success** of the operation 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**

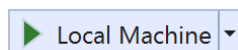
## Step 7

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

```
<li class="list-group-item">
  <form asp-page-handler="SaveUserTracks" method="post">
    <input asp-for="Value" placeholder="Track Ids" class="form-control mb-2" />
    <button class="btn btn-primary mb-2">
      Save Tracks for User
    </button>
  </form>
</li>
```

This form will **post** to the **method** for SaveUserTracks with the Value of the **Track Ids** and will output to the **Results**.

## Step 8

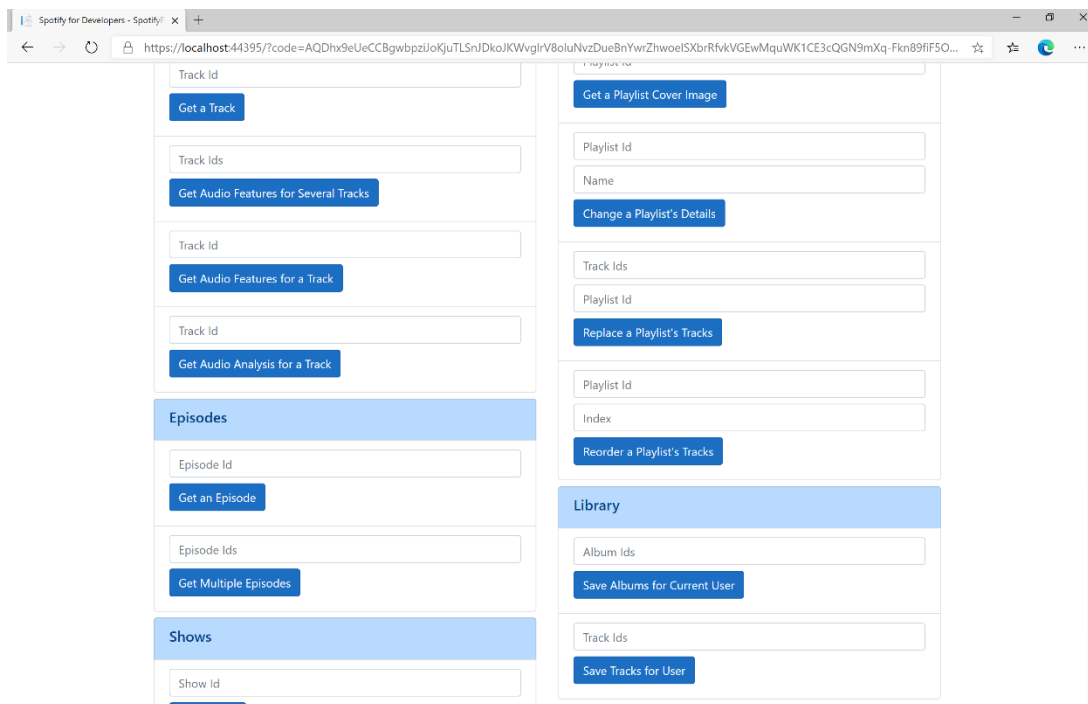


Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**



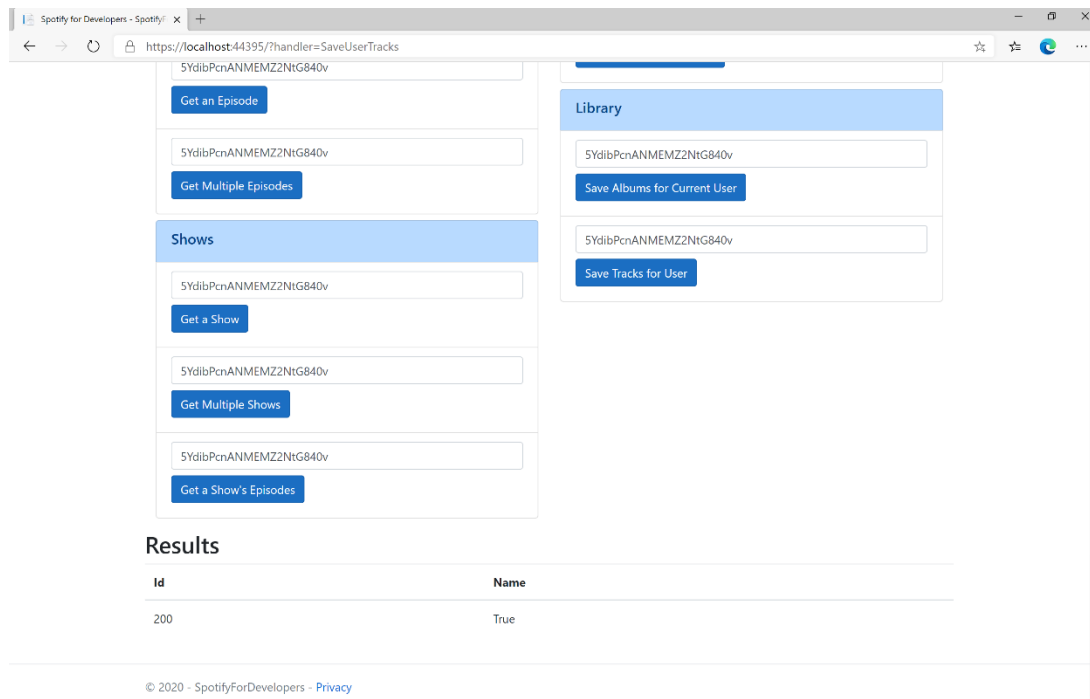
## Step 9

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:



## Step 10

You can then enter some **Track Ids** from an **Album Id** from **Get All New Releases** using **Get an Album's Tracks** into **Track Ids** above **Save Tracks for User** and then select **Save Tracks for User** and scroll down to view **Results** like the following:



## 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

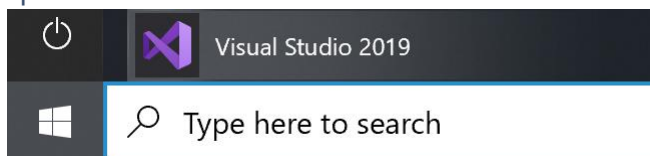
## Save Shows for Current User

Save one or more shows to current Spotify user's library.

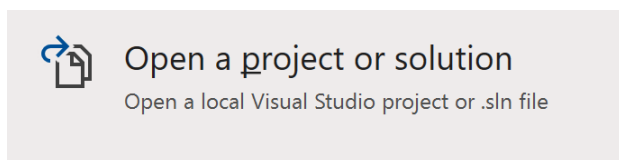
PUT <a href="https://api.spotify.com/v1/me/shows">https://api.spotify.com/v1/me/shows</a>	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-modify</b> scope
Query Parameter	
<b>ids</b>	Comma-separated list of Spotify Ids for the shows to be added to the user's library.

Header	Response
Success	
Http Status 200 OK	
Error	
HTTP Status 403 Forbidden	Returned when not Authorised or more than 10,000 items saved in library
Error Code	Error Object

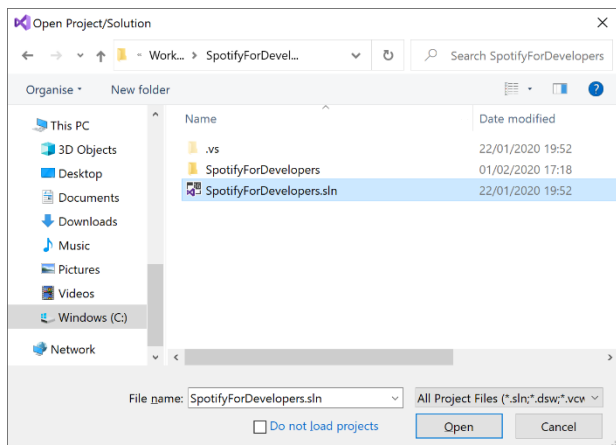
### Step 1



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**

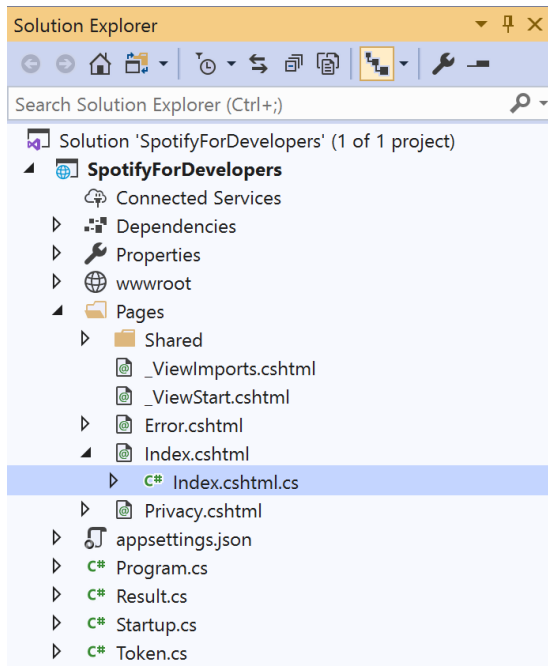


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



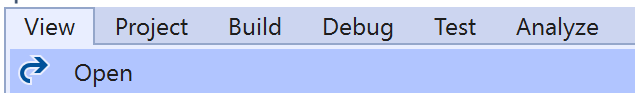
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow, Playlists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

## 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> OnPostSaveUserTracksAsync(...)` { ... } enter the following **method**:

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

The **method** for `OnPostSaveUserShowsAsync` **saves** the **shows** to the **Library** with Value of the **Show Ids** of the **show** and populate the **property** for Results of the **success** of the operation 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**

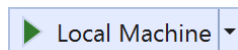
## Step 7

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

```
<li class="list-group-item">
  <form asp-page-handler="SaveUserShows" method="post">
    <input asp-for="Value" placeholder="Show Ids" class="form-control mb-2" />
    <button class="btn btn-primary mb-2">
      Save Shows for User
    </button>
  </form>
</li>
```

This form will **post** to the **method** for SaveUserTracks with the Value of the **Track Ids** and will output to the **Results**.

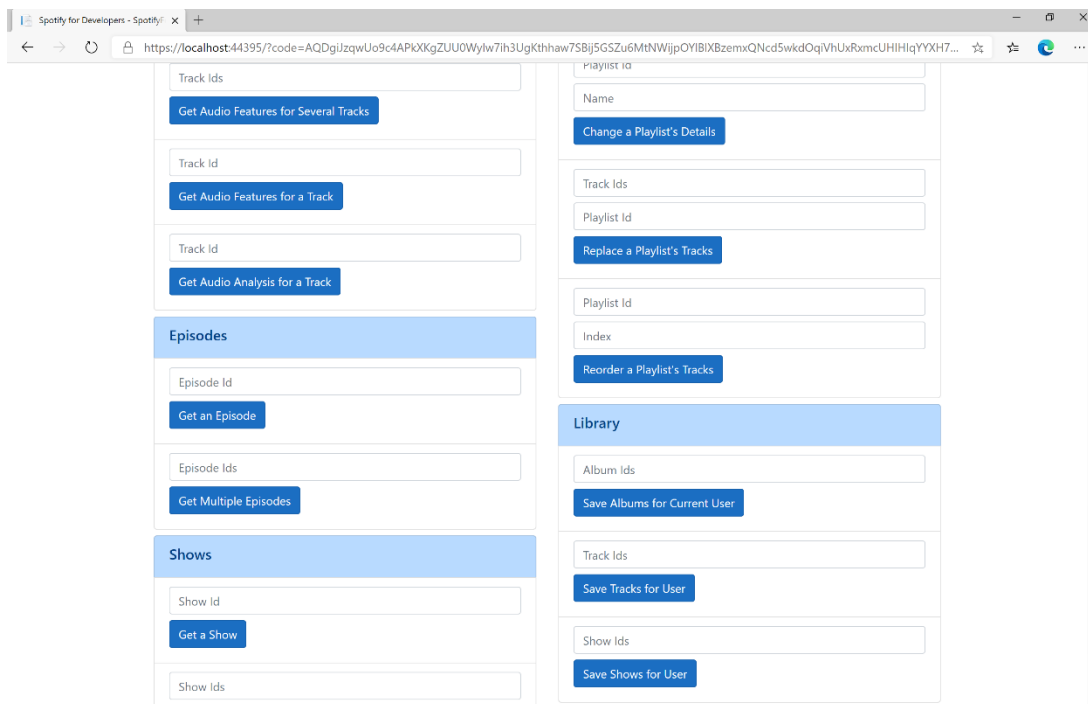
## Step 8



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

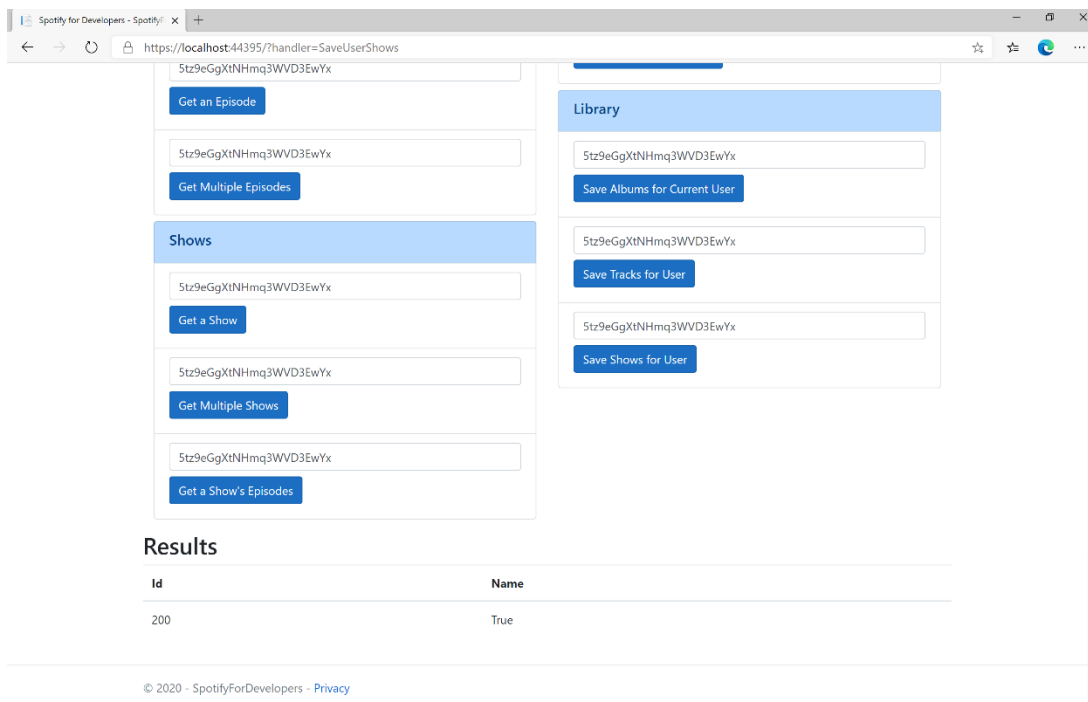
## Step 9

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:



## Step 10

You can get a **Show Id** from **Search For An Item** using **Shows** and then enter this into **Show Ids** above **Save Shows for User** and then select **Save Shows for User** and scroll down to view **Results** like the following:



## 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

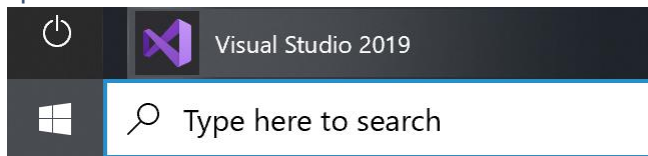
## Check User's Saved Albums

Check if one or more albums is already saved in the current Spotify user's 'Your Music' library.

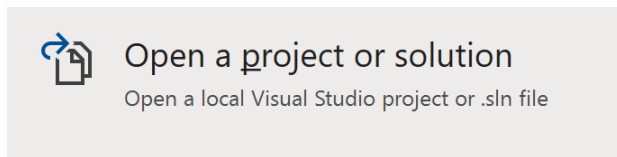
GET <a href="https://api.spotify.com/v1/me/albums/contains">https://api.spotify.com/v1/me/albums/contains</a>	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-read</b> scope
Query Parameter	
<b>ids</b>	Comma-separated list of the Spotify Ids for the Albums. Maximum: 50 Ids.

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

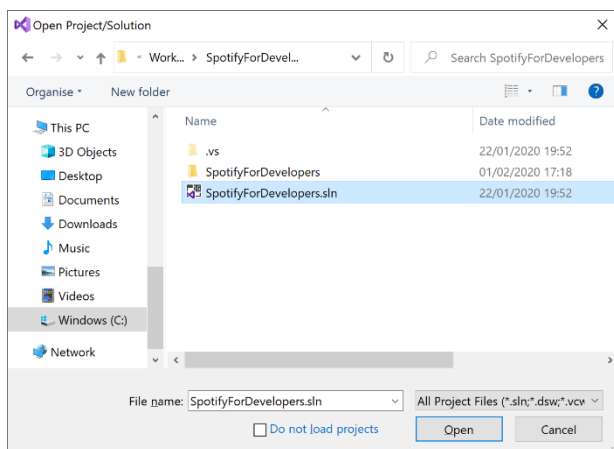
### Step 1



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**

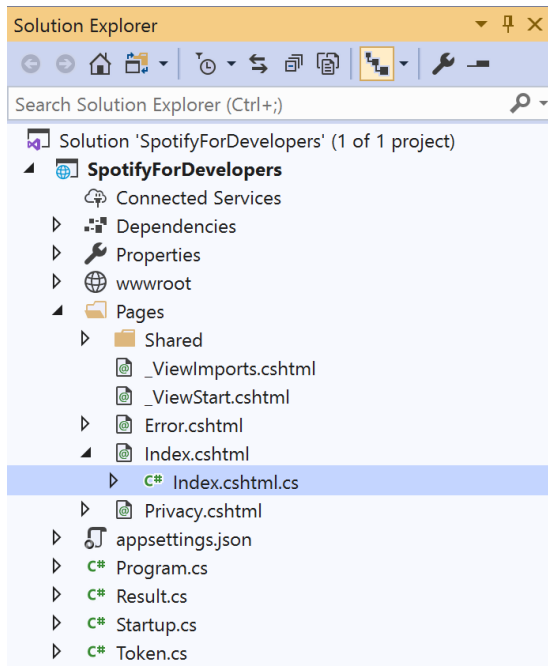


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



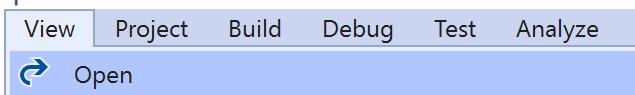
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow, Playlists**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

## 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**

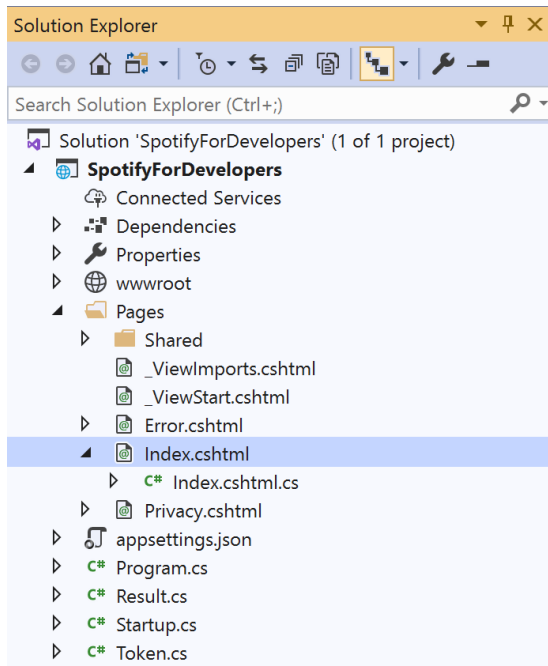
## Step 4

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

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

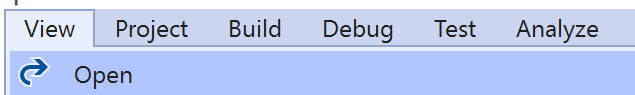
The **method** for `OnPostCheckUserSavedAlbumsAsync` is used to get the **saved state** of **albums** in the **Library** with Value of the **Album Ids** of the **track** and populate the **property** for `Results` accordingly.

## Step 5



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

## Step 6



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

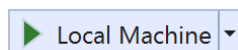
## Step 7

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

```
<li class="list-group-item">
  <form asp-page-handler="CheckUserSavedAlbums" method="post">
    <input asp-for="Value" placeholder="Album Ids" class="form-control mb-2" />
    <button class="btn btn-primary mb-2">
      Check User's Saved Albums
    </button>
  </form>
</li>
```

This form will **post** to the **method** for `CheckUserSavedAlbums` with the Value of the **Album Ids** and will output to the **Results**.

## Step 8

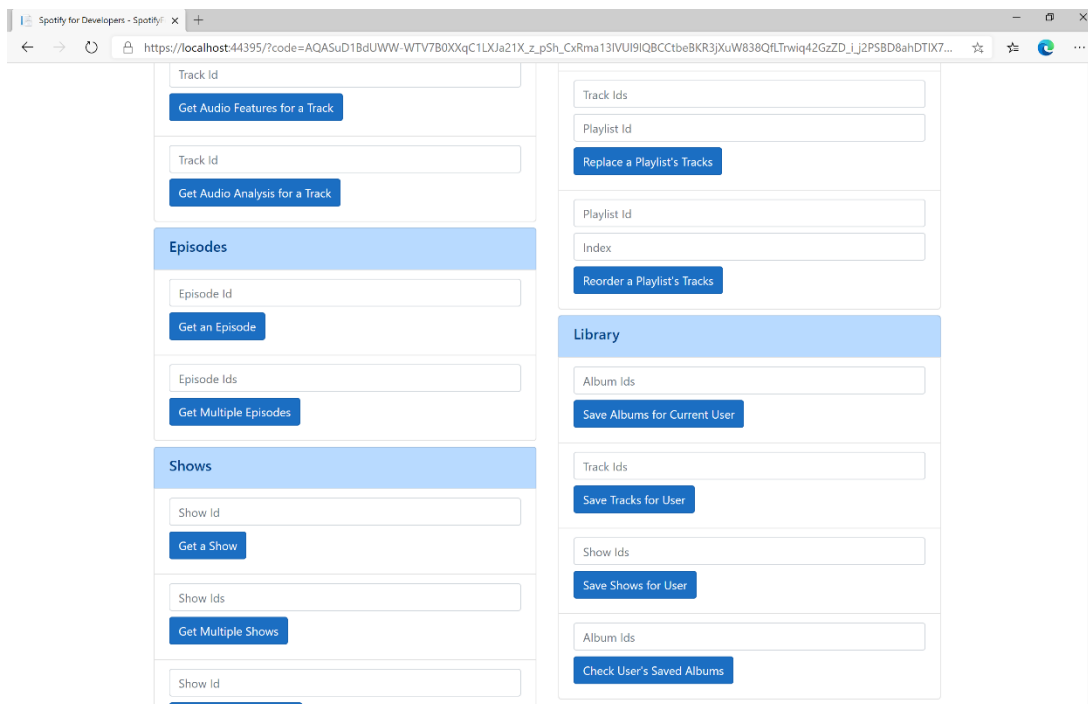


Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**



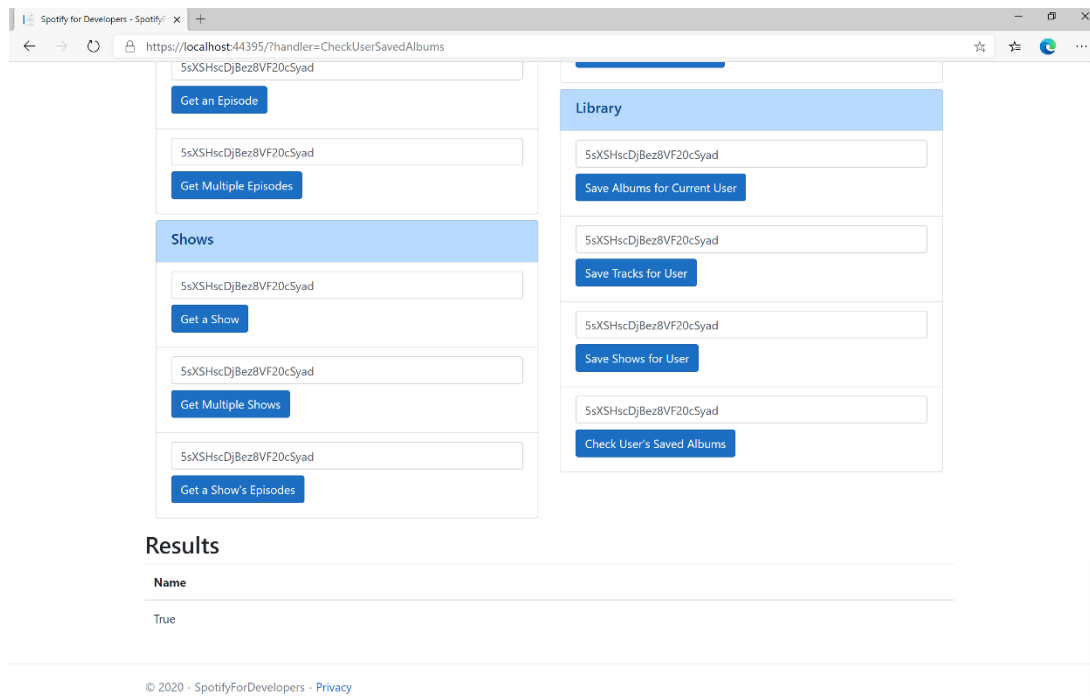
## Step 9

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:



## Step 10

You can then enter some **Album Ids** from **Get All New Releases** or copy the same **Album Ids** used in **Save Albums for Current User** into **Album Ids** above **Check User's Saved Albums** and then select **Check User's Saved Albums** and scroll down to view **Results** like the following:



## 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

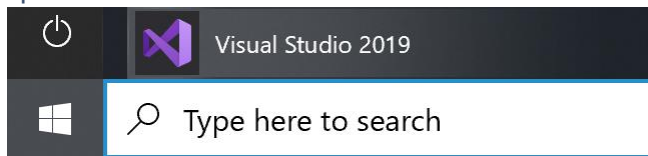
## Check User's Saved Tracks

Check if one or more tracks is already saved in the current Spotify user's 'Your Music' library.

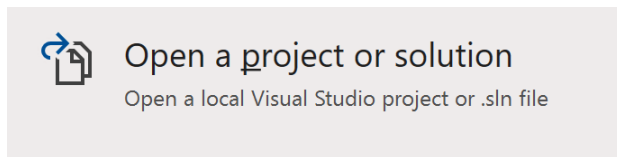
GET <a href="https://api.spotify.com/v1/me/tracks/contains">https://api.spotify.com/v1/me/tracks/contains</a>	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-read</b> scope
Query Parameter	
<b>ids</b>	Comma-separated list of the Spotify Ids for the Tracks. Maximum: 50 Ids.

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

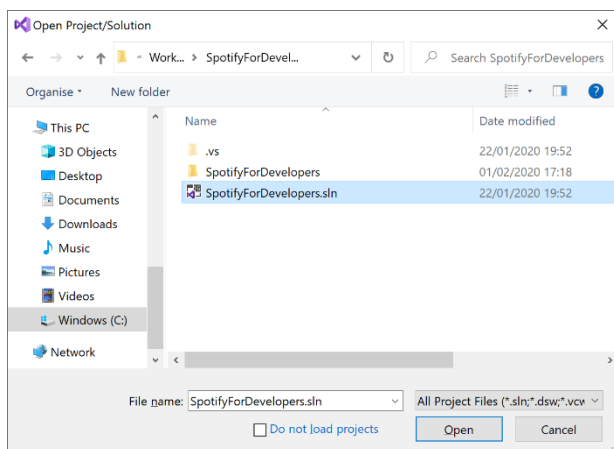
### Step 1



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**

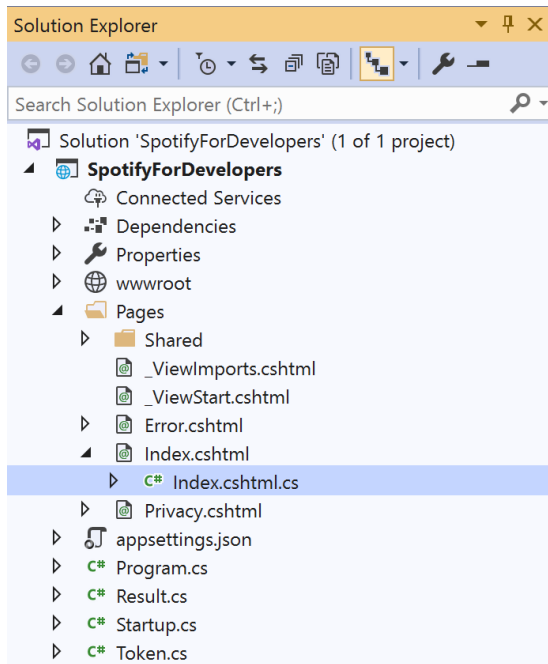


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



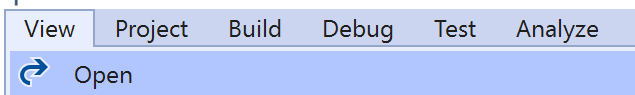
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow, Playlists**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

## 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> OnPostCheckUserSavedAlbumsAsync(...)` { ... } enter the following **method**:

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

The **method** for `OnPostCheckUserSavedTracksAsync` is used to get the **saved state** of **tracks** in the **Library** with Value of the **Track Ids** of the **track** 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**

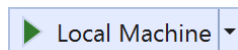
## Step 7

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

```
<li class="list-group-item">
  <form asp-page-handler="CheckUserSavedTracks" method="post">
    <input asp-for="Value" placeholder="Track Ids" class="form-control mb-2" />
    <button class="btn btn-primary mb-2">
      Check User's Saved Tracks
    </button>
  </form>
</li>
```

This form will **post** to the **method** for `CheckUserSavedTracks` with the Value of the **Track Ids** and will output to the **Results**.

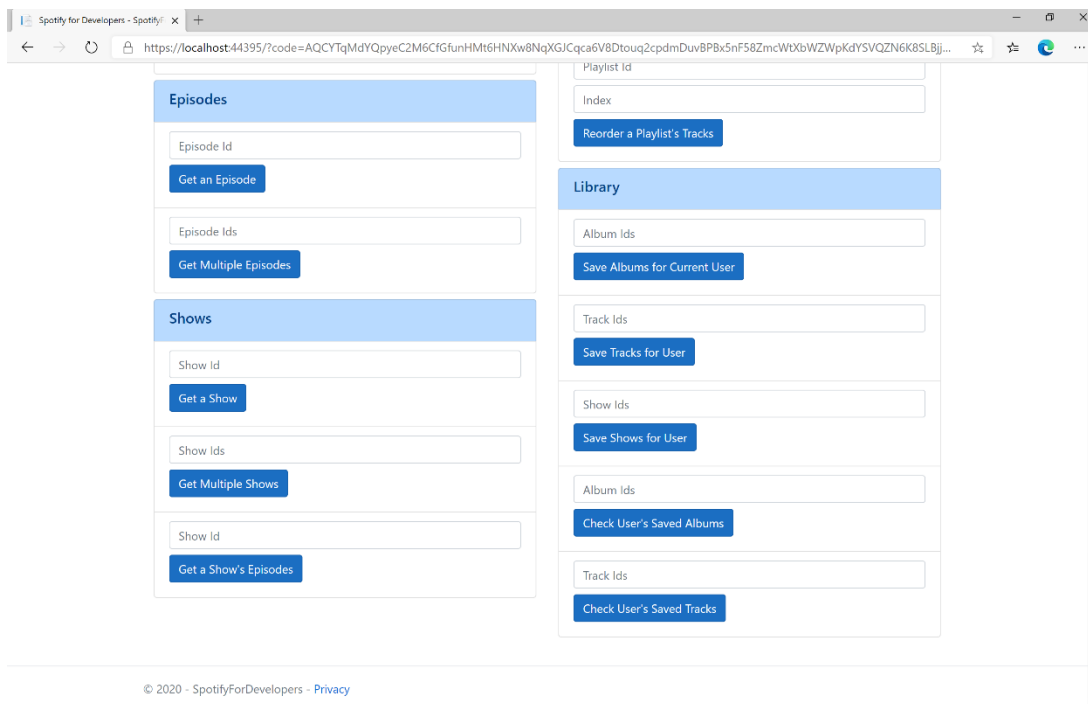
## Step 8



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

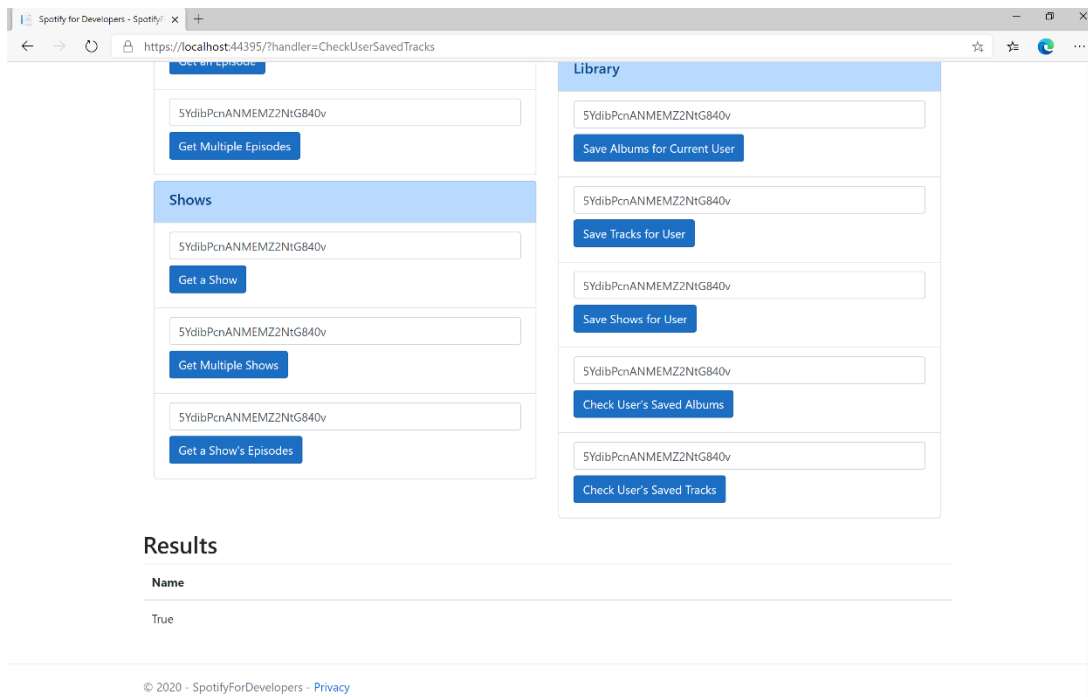
## Step 9

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:



## Step 10

You can then enter some **Track Ids** from an **Album Id** from **Get All New Releases** using **Get an Album's Tracks** or copy the same **Track Ids** used in **Save Tracks for User** in **Track Ids** above **Check User's Saved Tracks** and then select **Check User's Saved Tracks** and scroll down to view **Results** like the following:



## 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

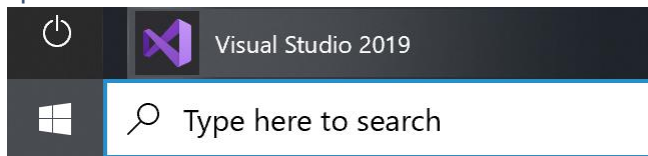
## Check User's Saved Shows

Check if one or more shows is already saved in the current Spotify user's library

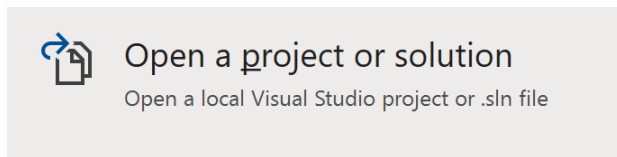
GET https://api.spotify.com/v1/me/shows/contains	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-read</b> scope
Query Parameter	
<b>ids</b>	Comma-separated list of the Spotify Ids for the shows. Maximum: 50 ids

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

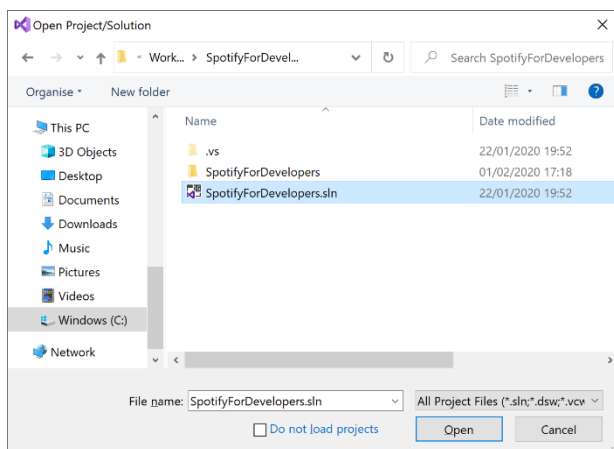
### Step 1



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**

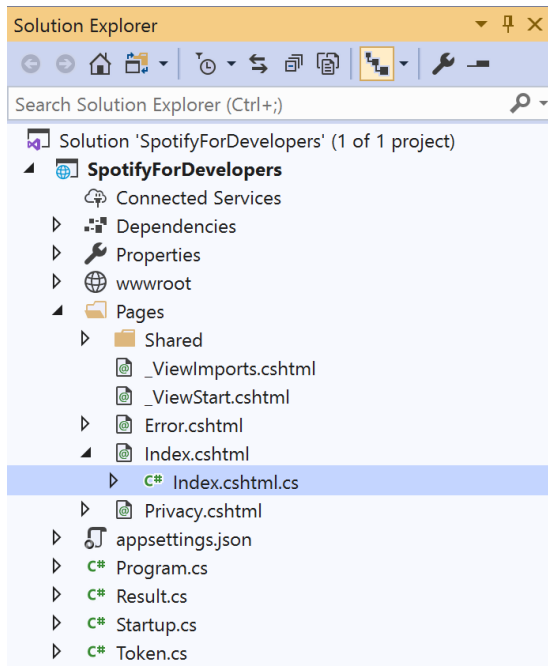


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



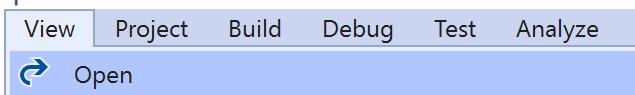
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow, Playlists**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

## 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**

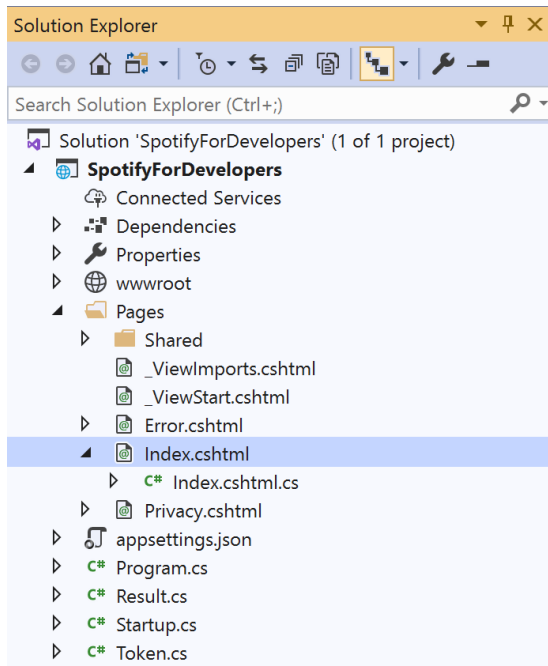
## Step 4

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

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

The **method** for `OnPostCheckUserSavedShowsAsync` is used to get the **saved state** of **shows** in the **Library** with Value of the **Shows Ids** of the **show** 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**

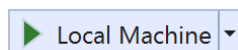
## Step 7

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

```
<li class="list-group-item">
  <form asp-page-handler="CheckUserSavedShows" method="post">
    <input asp-for="Value" placeholder="Show Ids" class="form-control mb-2" />
    <button class="btn btn-primary mb-2">
      Check User's Saved Shows
    </button>
  </form>
</li>
```

This form will **post** to the **method** for CheckUserSavedShows with the Value of the **Show Ids** and will output to the **Results**.

## Step 8

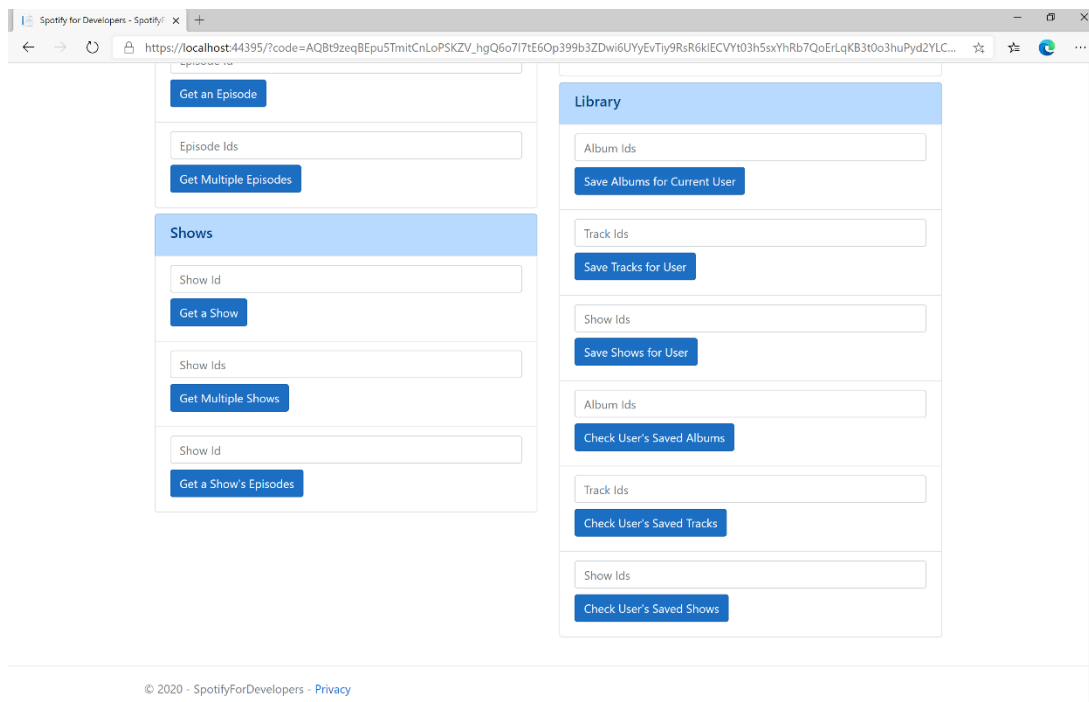


Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**



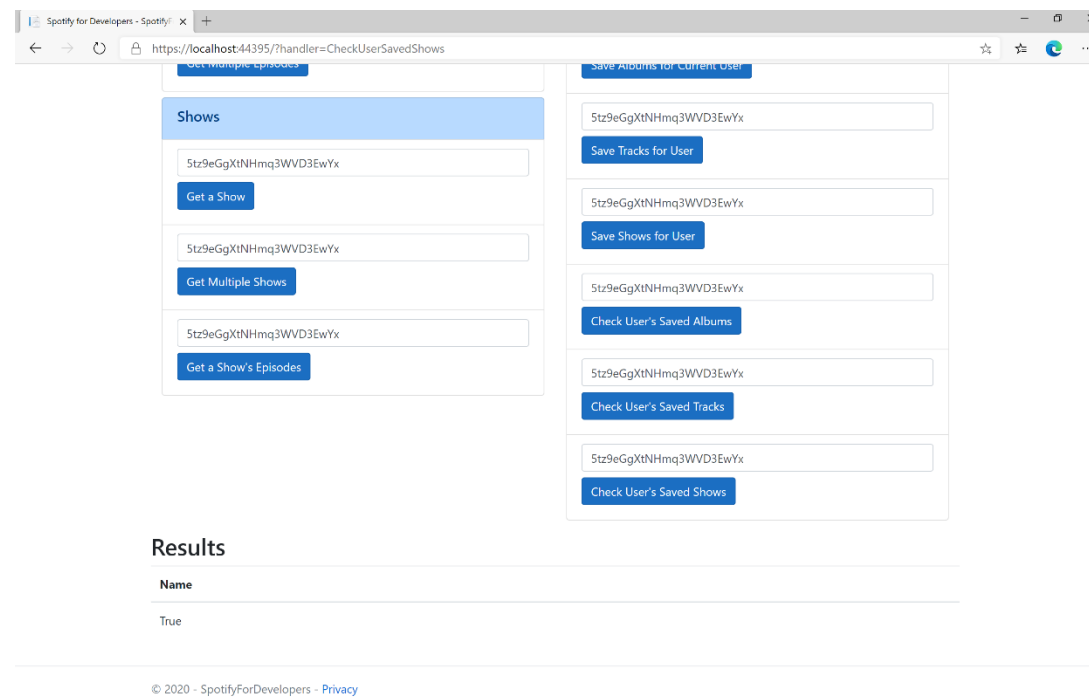
## Step 9

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:



## Step 10

You can get a **Show Id** from **Search For An Item** using **Shows** or copy the same **Shows Ids** used in **Save Shows for User** in **Show Ids** above **Check User's Saved Shows** and then select **Check User's Saved Shows** and scroll down to view **Results** like the following:



## 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

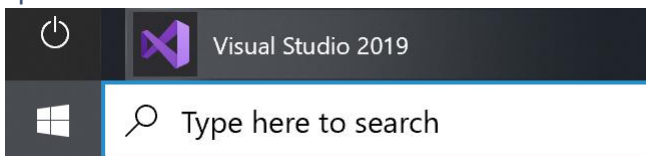
## Get User's Saved Albums

Get a list of the albums saved in the current Spotify user's 'Your Music' library.

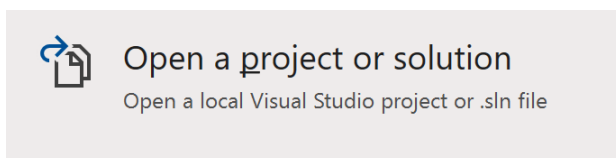
GET <a href="https://api.spotify.com/v1/me/albums">https://api.spotify.com/v1/me/albums</a>	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-read</b> scope
Query Parameter	
limit	Maximum number of objects to return. Default: 20. Minimum: 1. Maximum: 50
offset	Index of the first object to return. Default: 0
market	ISO 3166-1 alpha-2 country code. Provide to apply Track Relinking

Header	Response
Success	
Http Status 200 OK	Array of Album Objects accompanied by a Timestamp wrapped in a Paging Object
Error	
Error Code	Error Object

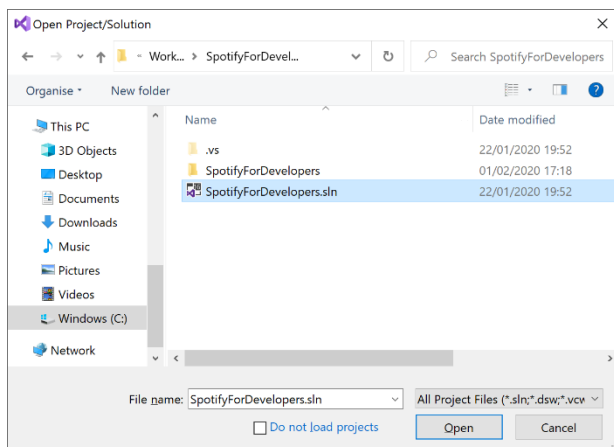
### Step 1



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**

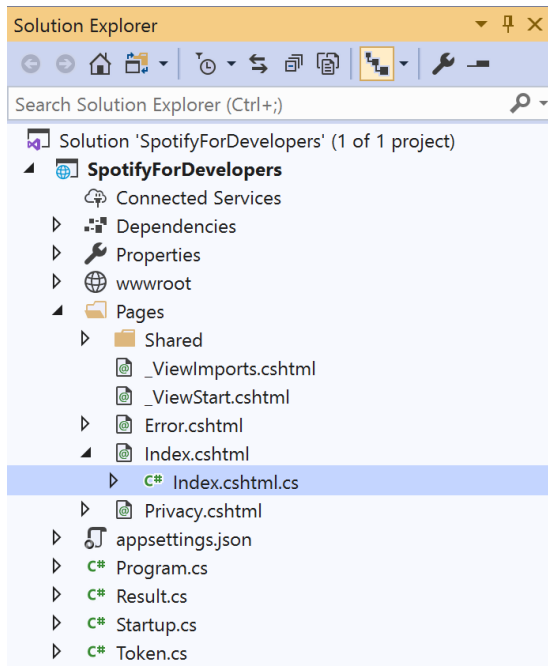


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



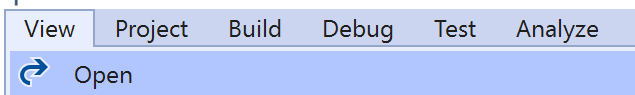
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow, Playlists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

## 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> OnPostCheckUserSavedShowsAsync(...)` { ... } enter the following **method**:

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

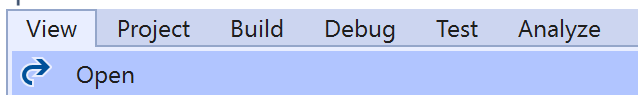
The **method** for `OnPostGetUserSavedAlbumsAsync` is used to get the **albums** in the **Library** and populate the **property** for `Results` accordingly.

## Step 5



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

## Step 6



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

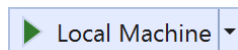
## Step 7

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

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

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

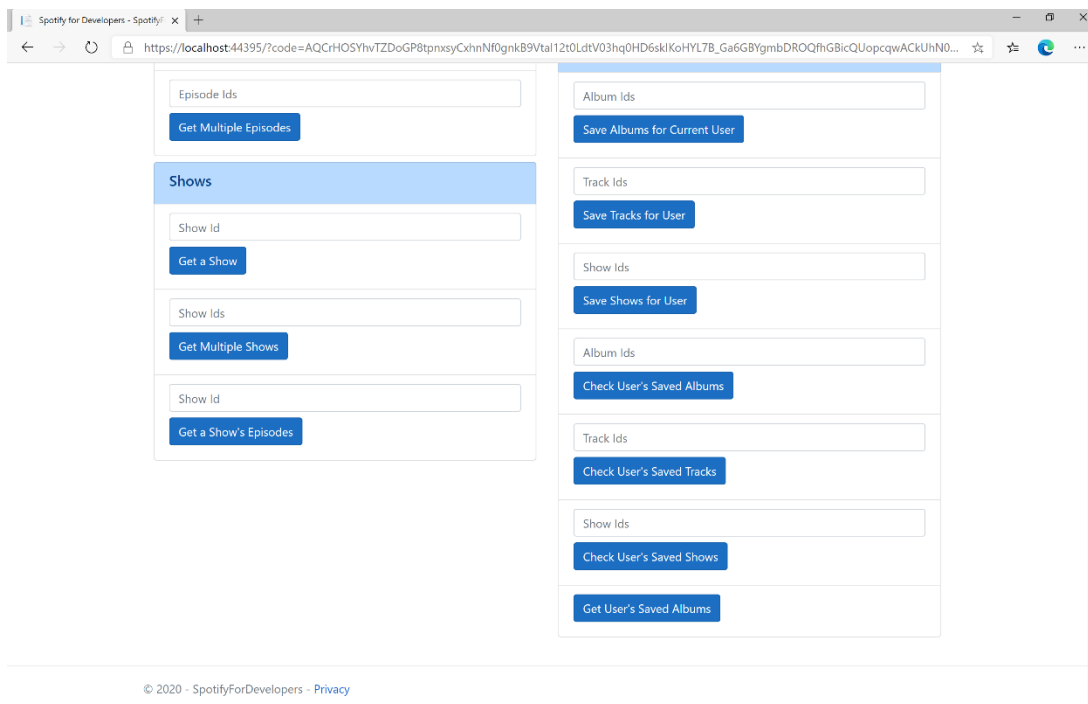
## Step 8



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

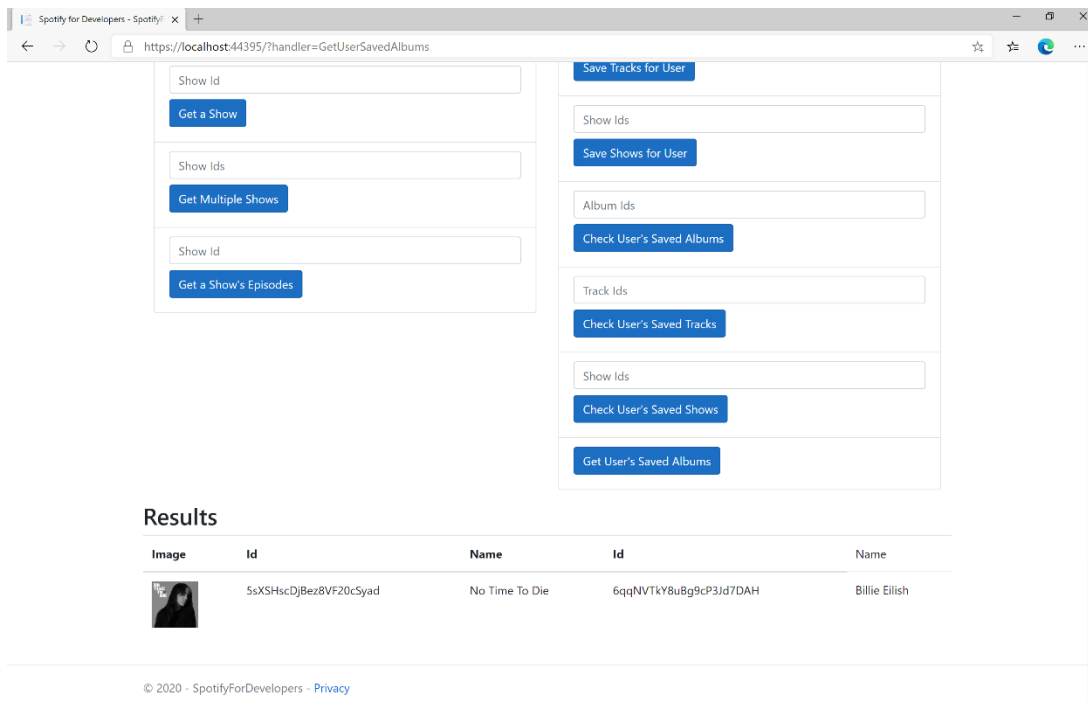
## Step 9

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:



## Step 10

You can then select **Get User's Saved Albums** and scroll down to view **Results** like the following:



## 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

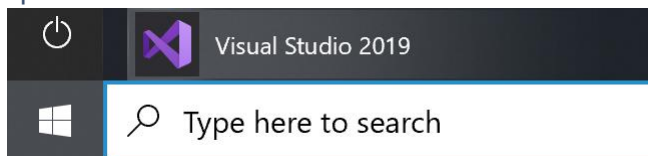
## Get User's Saved Tracks

Get a list of the songs saved in the current Spotify user's 'Your Music' library.

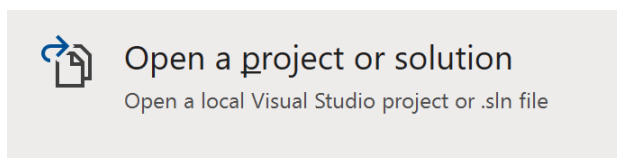
GET <a href="https://api.spotify.com/v1/me/tracks">https://api.spotify.com/v1/me/tracks</a>	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-read</b> scope
Query Parameter	
limit	Maximum number of objects to return. Default: 20. Minimum: 1. Maximum: 50
offset	Index of the first object to return. Default: 0
market	ISO 3166-1 alpha-2 country code. Provide to apply Track Relinking

Header	Response
Success	
Http Status 200 OK	Array of Saved Track Objects wrapped in a Paging Object
Error	
Error Code	Error Object

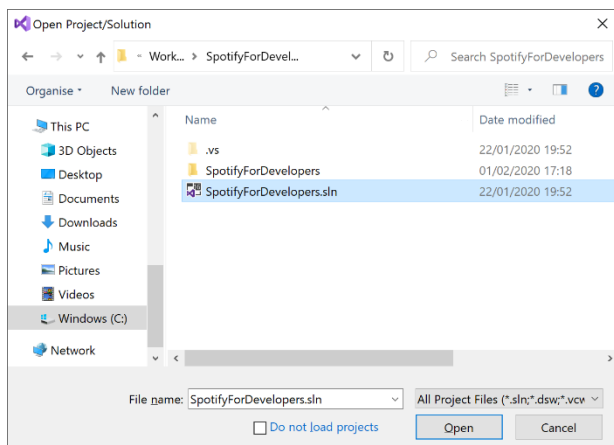
### Step 1



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**

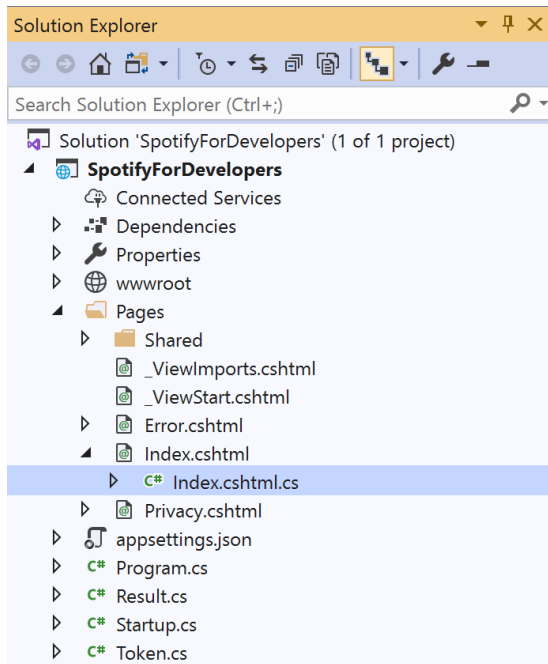


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



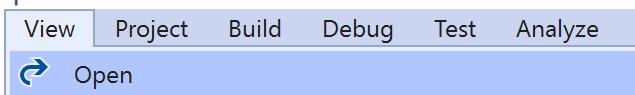
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow, Playlists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

## 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**

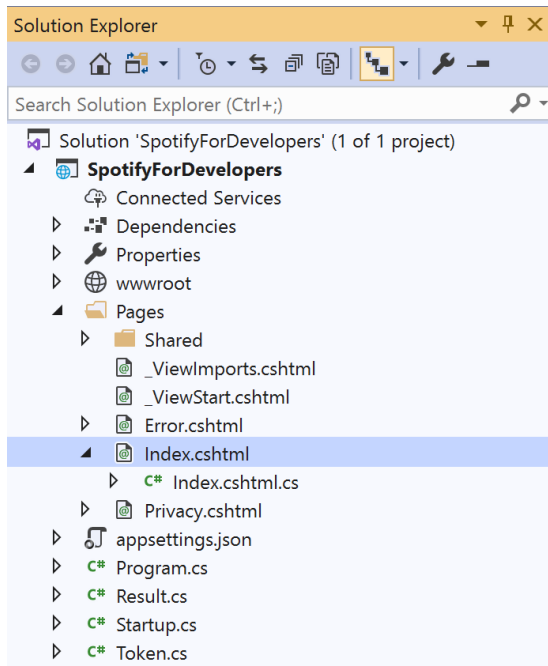
## Step 4

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

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

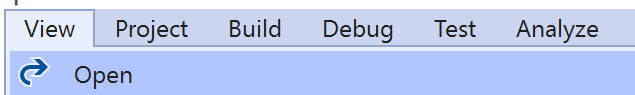
The **method** for `OnPostGetUserSavedTracksAsync` is used to get the **tracks** in the **Library** and populate the **property** for `Results` accordingly.

## Step 5



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

## Step 6



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

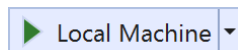
## Step 7

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

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

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

## Step 8

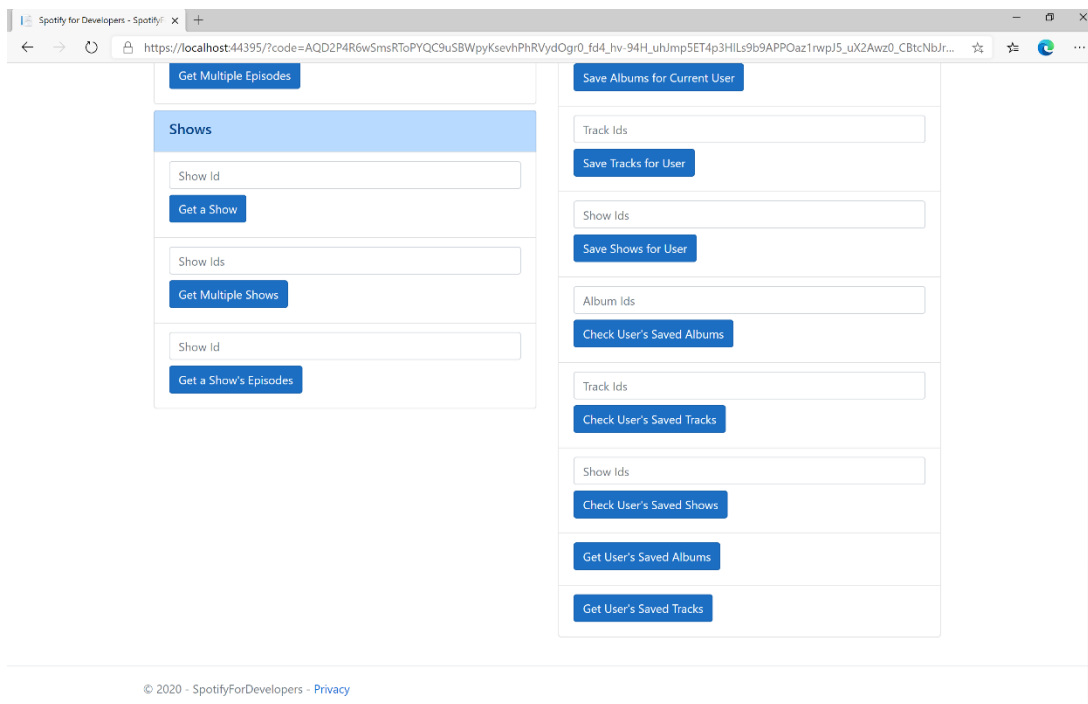


Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**



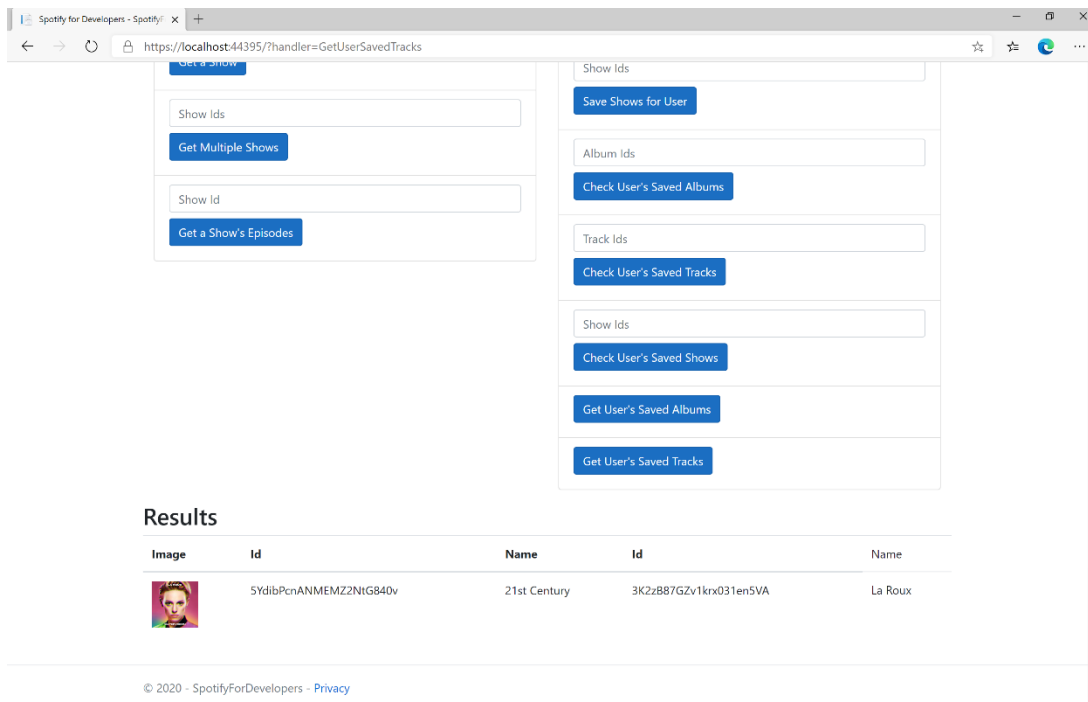
## Step 9

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:



## Step 10

You can then select **Get User's Saved Tracks** and scroll down to view **Results** like the following:



## 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

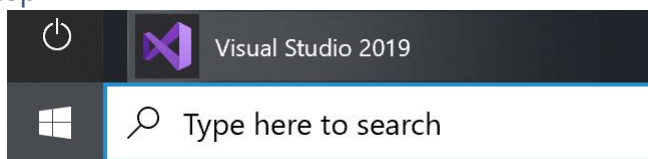
## Get User's Saved Shows

Get a list of shows saved in the current Spotify user's library.

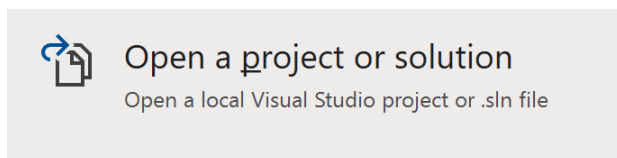
GET <a href="https://api.spotify.com/v1/me/shows">https://api.spotify.com/v1/me/shows</a>	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-read</b> scope
Query Parameter	
limit	Maximum number of shows to return. Default: 20. Minimum: 1. Maximum: 50
offset	Index of the first object to return. Default: 0

Header	Response
Success	
Http Status 200 OK	Array of Saved Show Objects wrapped in a Paging Object
Error	
Error Code	Error Object

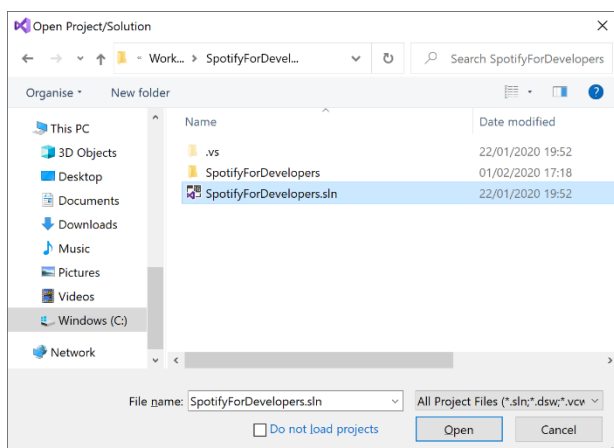
### Step 1



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**

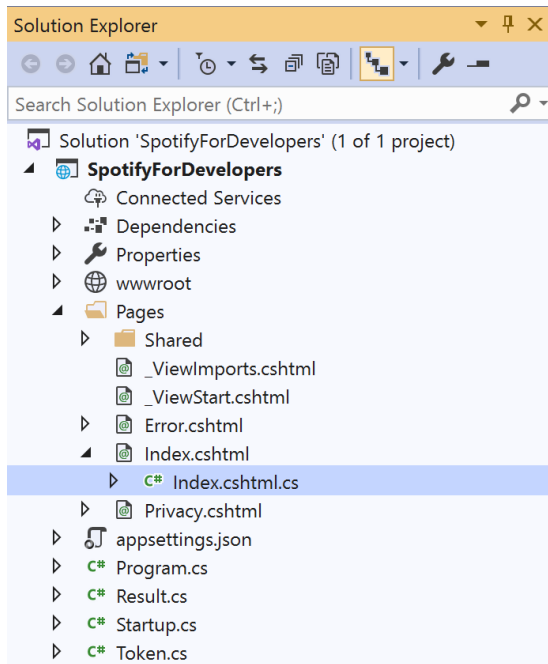


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



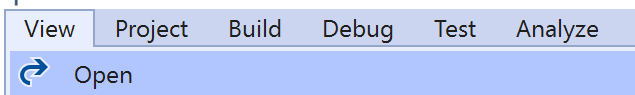
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow, Playlists**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

## 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> OnPostGetUserSavedTracksAsync() { ... }` enter the following **method**:

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

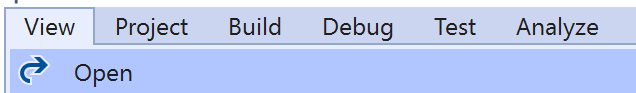
The **method** for `OnPostGetUserSavedShowsAsync` is used to get the **shows** in the **Library** and populate the **property** for `Results` accordingly.

## Step 5



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

## Step 6



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

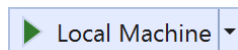
## Step 7

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

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

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

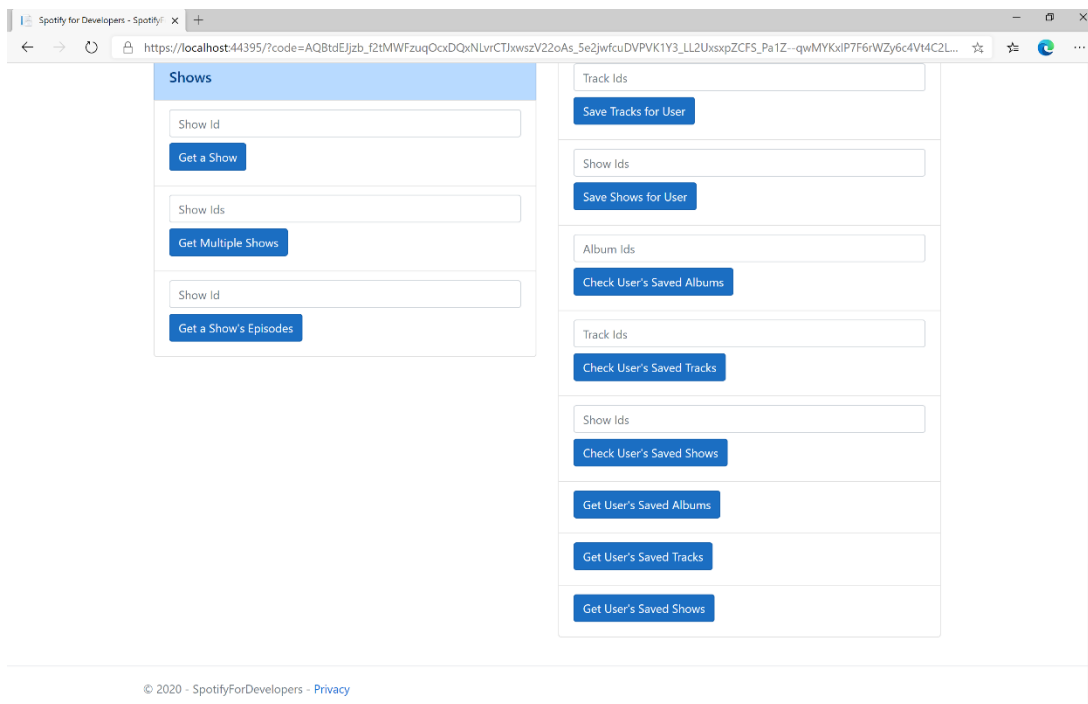
## Step 8



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

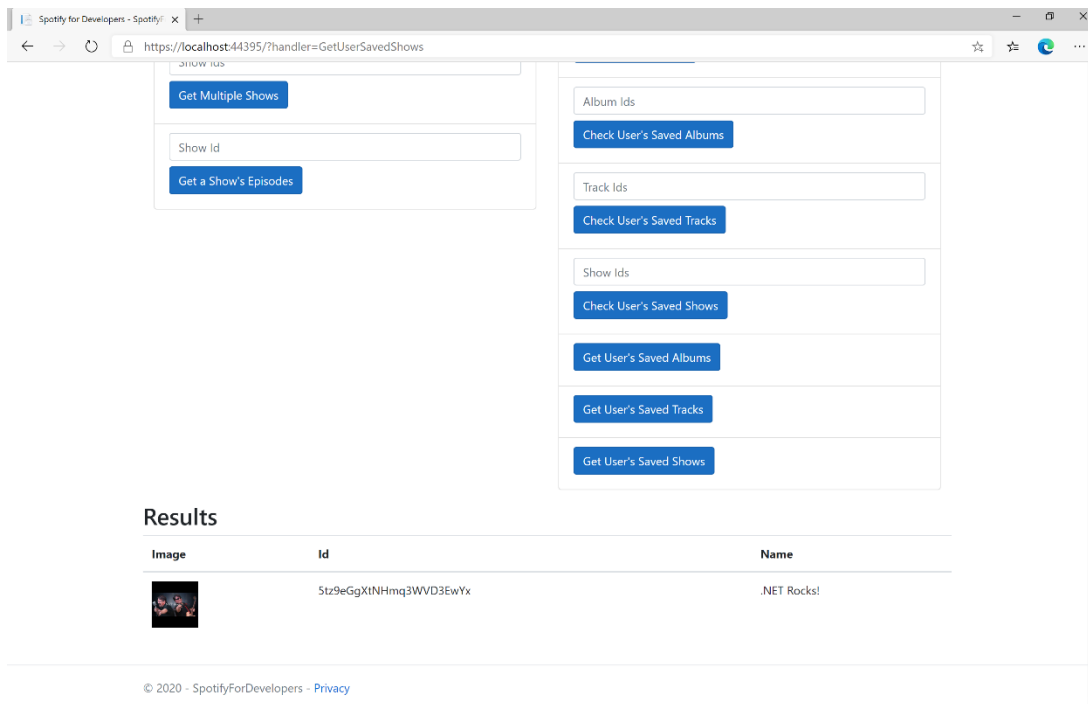
## Step 9

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:



## Step 10

You can then select **Get User's Saved Shows** and scroll down to view **Results** like the following:



## 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

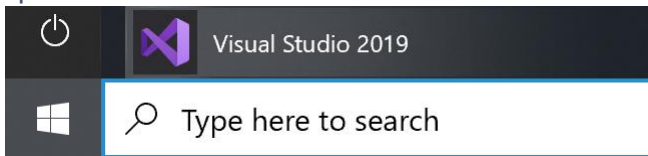
## Remove Albums for Current User

Remove one or more albums from the current user's 'Your Music' library.

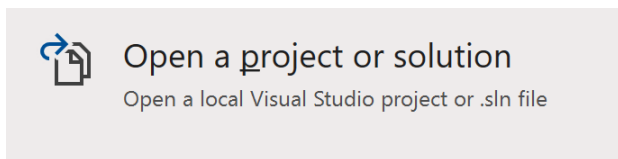
DELETE <a href="https://api.spotify.com/v1/me/albums?ids={ids}">https://api.spotify.com/v1/me/albums?ids={ids}</a>	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-modify</b> scope
Query Parameter	
<b>ids</b>	Comma-separated list of the Spotify Ids for the Albums. Maximum: 50 Ids.

Header	Response
Success	
Http Status 201 Created	
Error	
HTTP Status 403 Forbidden	Returned when not Authorised
Error Code	Error Object

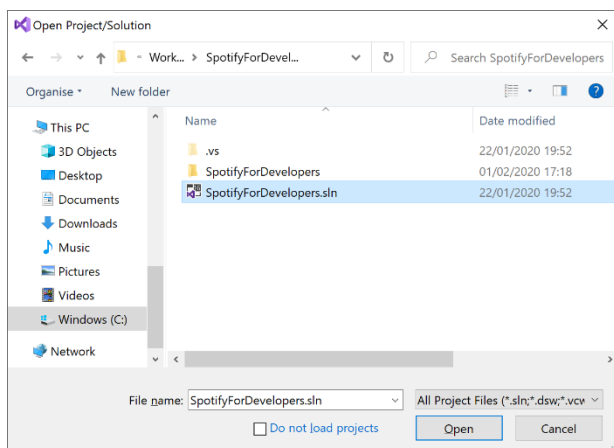
### Step 1



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**

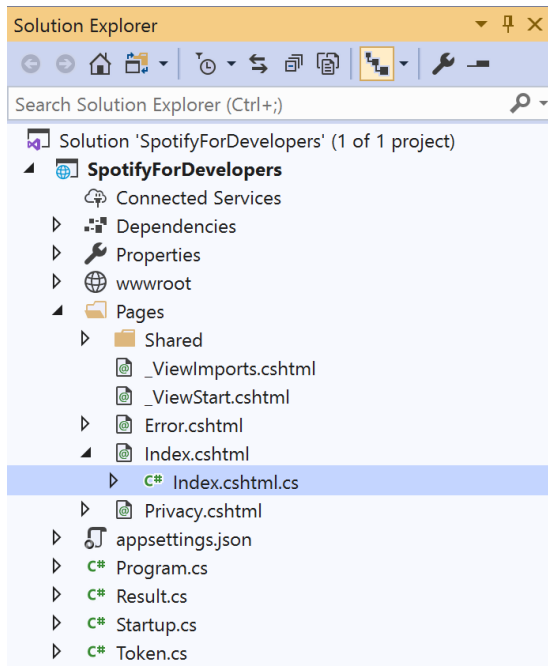


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



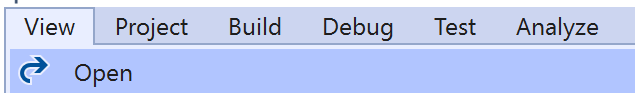
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow, Playlists**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

## 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**

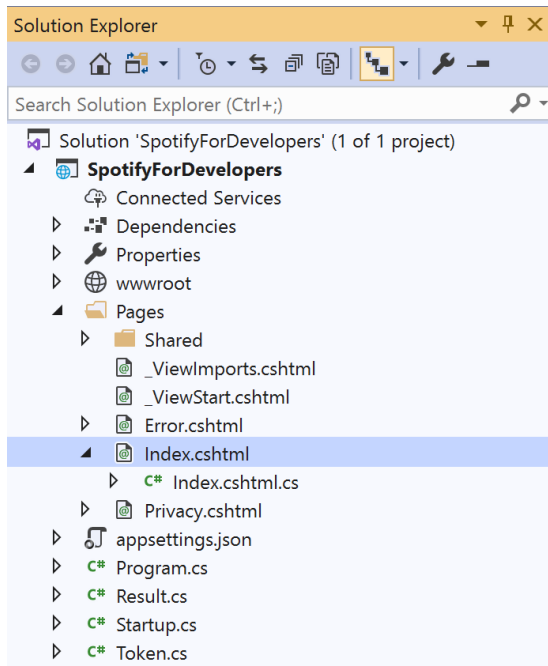
## Step 4

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

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

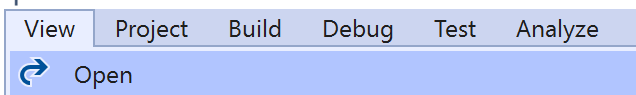
The **method** for `OnPostRemoveUserAlbumsAsync` **removes** the **saved albums** from the **Library** with Value of the **Album Ids** of the **album** and populate the **property** for Results of the **success** of the operation accordingly.

## Step 5



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

## Step 6



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

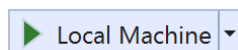
## Step 7

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

```
<li class="list-group-item">
  <form asp-page-handler="RemoveUserAlbums" method="post">
    <input asp-for="Value" placeholder="Album Ids" class="form-control mb-2" />
    <button class="btn btn-primary mb-2">
      Remove Albums for Current User
    </button>
  </form>
</li>
```

This form will **post** to the **method** for RemoveUserAlbums with the Value of the **Album Ids** and will output to the **Results**.

## Step 8

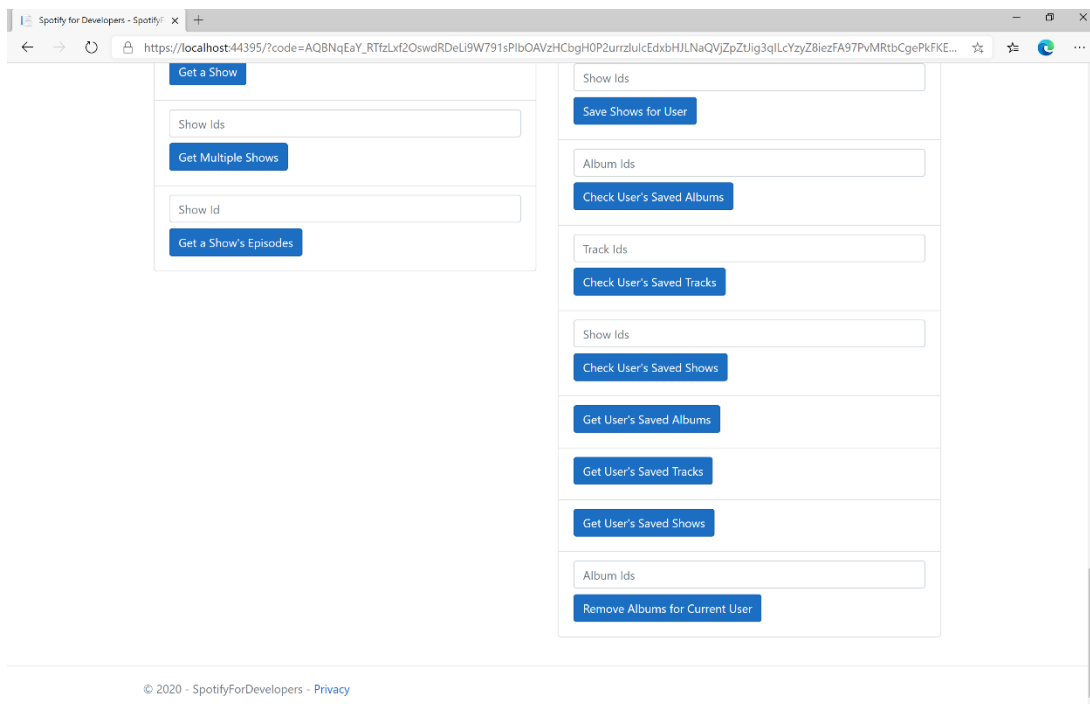


Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**



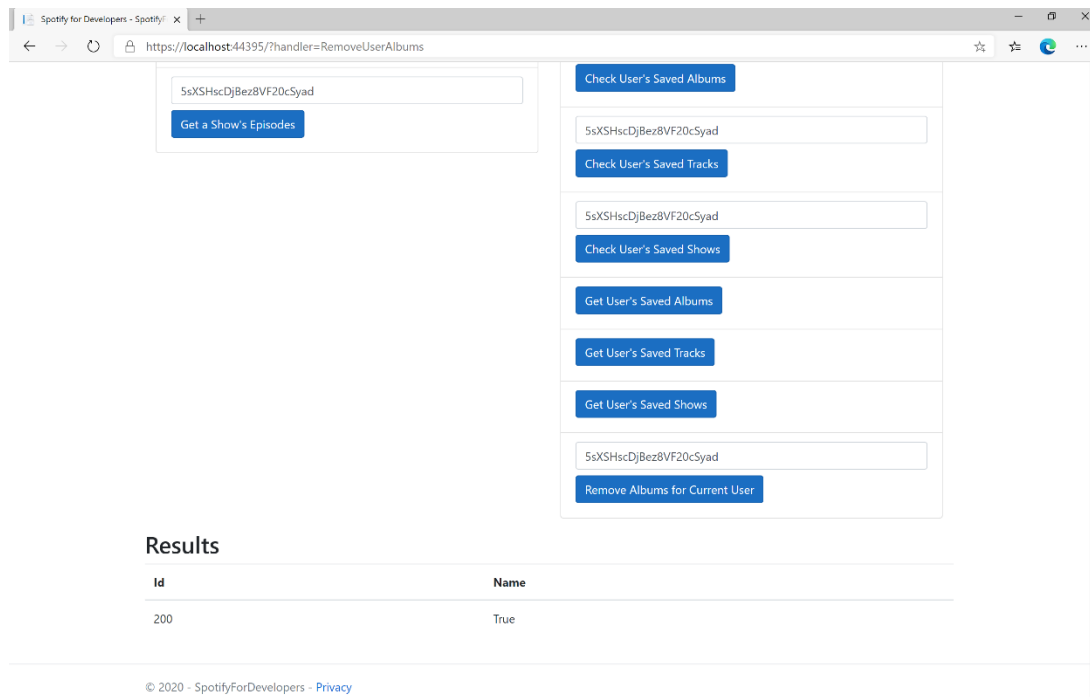
## Step 9

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:



## Step 10

You can then enter some **Album Ids** from **Get All New Releases** or copy the same **Album Ids** used in **Save Albums for Current User** into **Album Ids** above **Remove Albums for Current User** and then select **Remove Albums for Current User** and scroll down to view **Results** like the following:



## 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

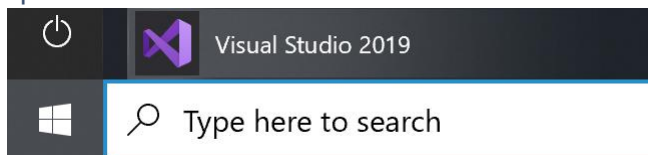
## Remove User's Saved Tracks

Remove one or more tracks from the current user's 'Your Music' library.

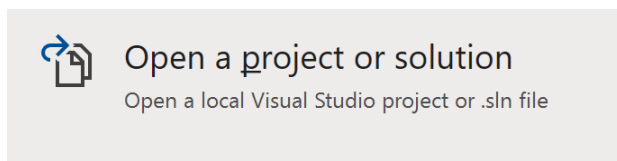
DELETE <a href="https://api.spotify.com/v1/me/tracks">https://api.spotify.com/v1/me/tracks</a>	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-modify</b> scope
Query Parameter	
<b>ids</b>	Comma-separated list of the Spotify Ids for the Tracks. Maximum: 50 Ids.

Header	Response
Success	
Http Status 200 OK	
Error	
HTTP Status 403 Forbidden	Returned when not Authorised
Error Code	Error Object

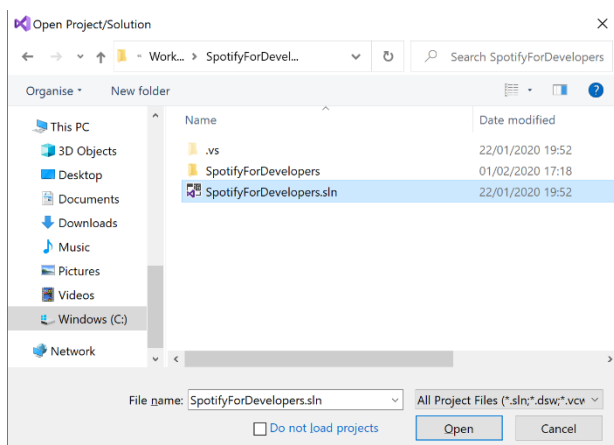
### Step 1



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**

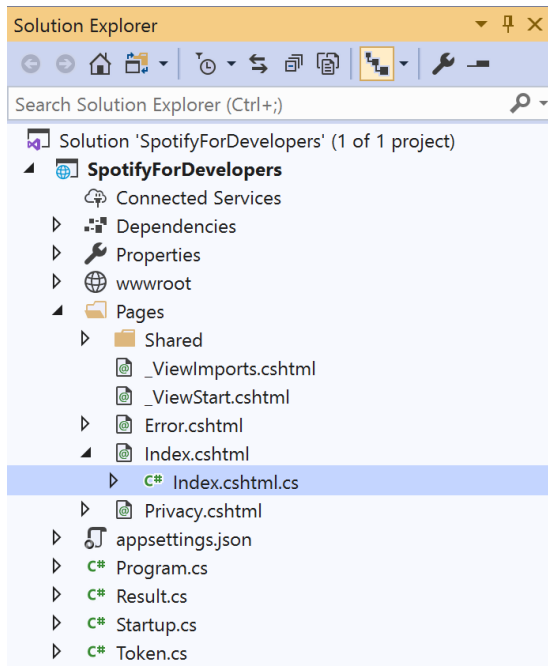


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



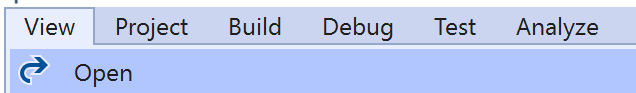
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow, Playlists**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

## 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> OnPostRemoveUserAlbumsAsync(...)` { ... } enter the following **method**:

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

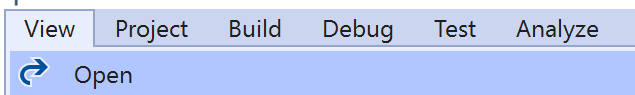
The **method** for `OnPostRemoveUserTracksAsync` **removes** the **saved track** from the **Library** with Value of the **Track Ids** of the **track** and populate the **property** for Results of the **success** of the operation accordingly.

## Step 5



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

## Step 6



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

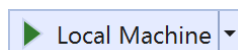
## Step 7

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

```
<li class="list-group-item">
  <form asp-page-handler="RemoveUserTracks" method="post">
    <input asp-for="Value" placeholder="Track Ids" class="form-control mb-2" />
    <button class="btn btn-primary mb-2">
      Remove User's Saved Tracks
    </button>
  </form>
</li>
```

This form will **post** to the **method** for RemoveUserTracks with the Value of the **Track Ids** and will output to the **Results**.

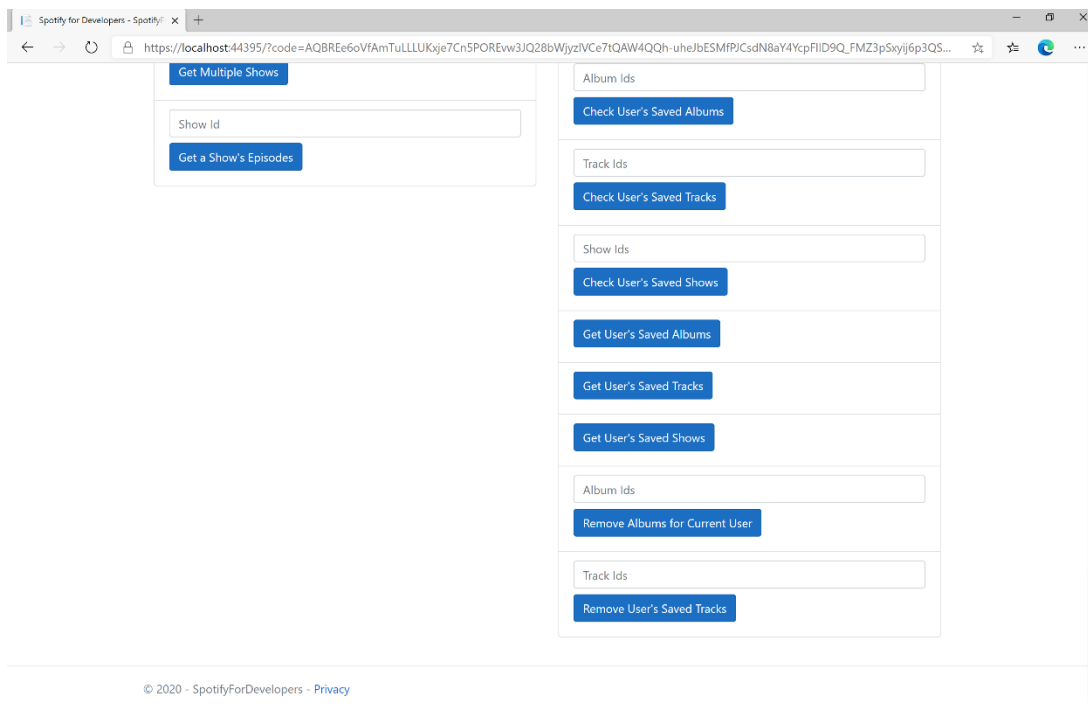
## Step 8



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

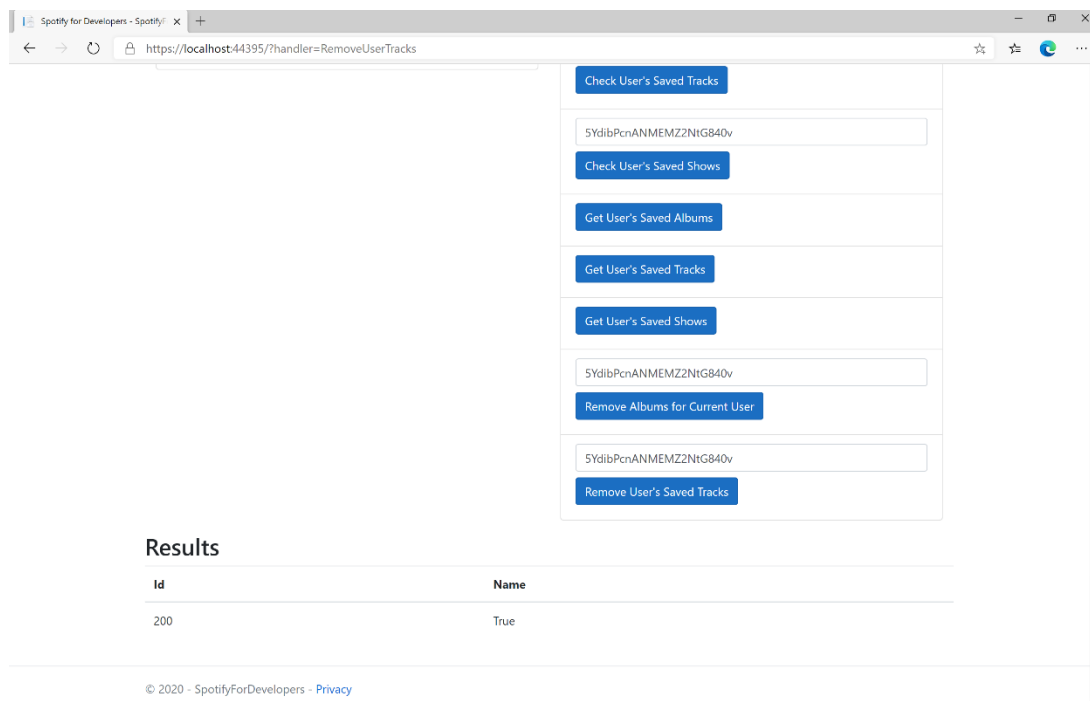
## Step 9

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:



## Step 10

You can then enter some **Track Ids** from an **Album Id** from **Get All New Releases** using **Get an Album's Tracks** or copy the same **Track Ids** used in **Save Tracks for User** in **Track Ids** above **Remove User's Saved Tracks** and then select **Remove User's Saved Tracks** and scroll down to view **Results** like the following:



## 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

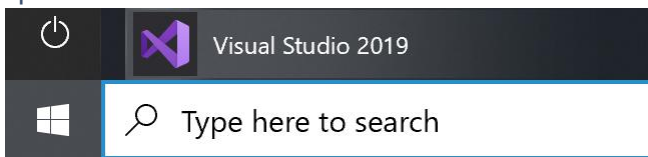
## Remove User's Saved Shows

Delete one or more shows from current Spotify user's library.

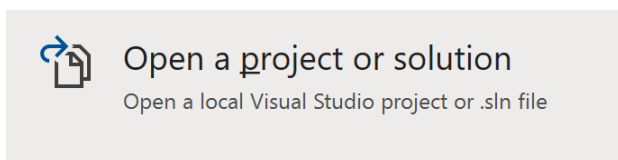
DELETE <a href="https://api.spotify.com/v1/me/shows">https://api.spotify.com/v1/me/shows</a>	
Header	
<b>Authorization</b>	User Token from Spotify Accounts service with <b>user-library-modify</b> scope
Query Parameter	
<b>ids</b>	Comma-separated list of Spotify Ids for the shows to be deleted from the user's library
<b>market</b>	<p>An ISO 3166-1 alpha-2 country code. If a country code is specified, only shows that are available in that market will be removed.</p> <p>If a valid user access token is specified in the request header, the country associated with the user account will take priority over this parameter.</p> <p>If neither market nor user country are provided, the content is considered unavailable for the client. Users can view the country that is associated with their account in the account settings</p>

Header	Response
Success	
Http Status 200 OK	
Error	
HTTP Status 403 Forbidden	Returned when not Authorised
Error Code	Error Object

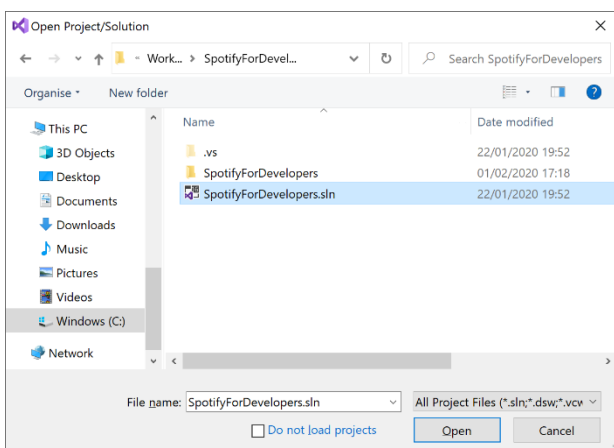
### Step 1



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**

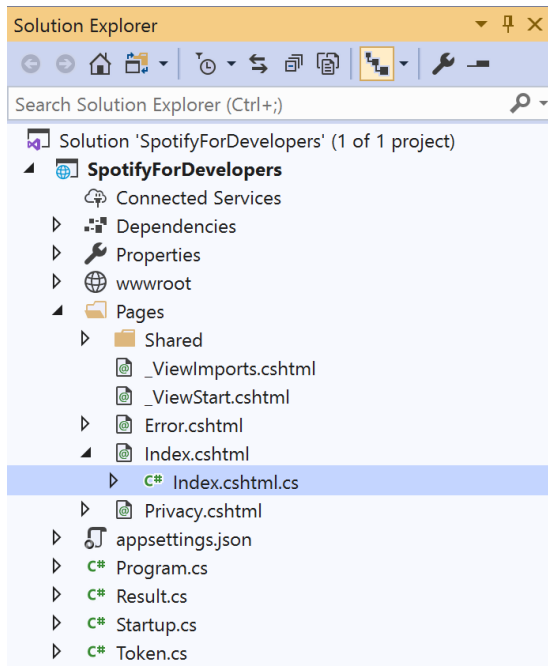


Once done, from the **Get started** screen for **Visual Studio 2019** select **Open a project or solution**



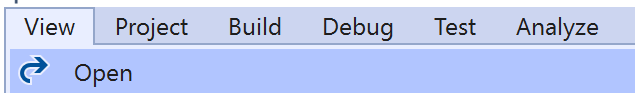
Then locate and select **SpotifyForDevelopers.sln** and select **Open** if you don't have this file already then please follow the previous parts of the workshop including **Getting Started, Authorisation Guide, Search & Browse, Playlists & Artists, Albums & Tracks, Episodes & Shows, Follow, Playlists**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

## 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**

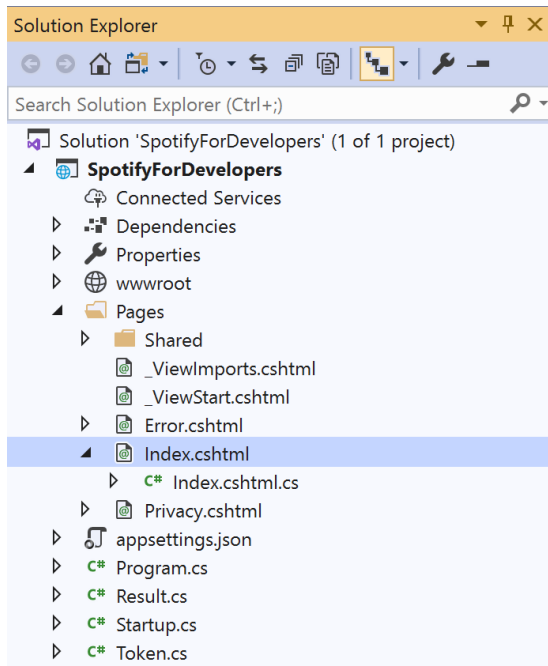
## Step 4

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

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

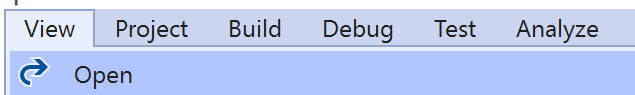
The **method** for `OnPostRemoveUserShowsAsync` **removes** the **saved show** from the **Library** with Value of the **Show Ids** of the **show** and populate the **property** for Results of the **success** of the operation accordingly.

## Step 5



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

## Step 6



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

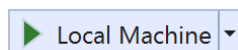
## Step 7

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

```
<li class="list-group-item">
  <form asp-page-handler="RemoveUserShows" method="post">
    <input asp-for="Value" placeholder="Show Ids" class="form-control mb-2" />
    <button class="btn btn-primary mb-2">
      Remove User's Saved Shows
    </button>
  </form>
</li>
```

This form will **post** to the **method** for RemoveUserShows with the Value of the **Show Ids** and will output to the **Results**.

## Step 8

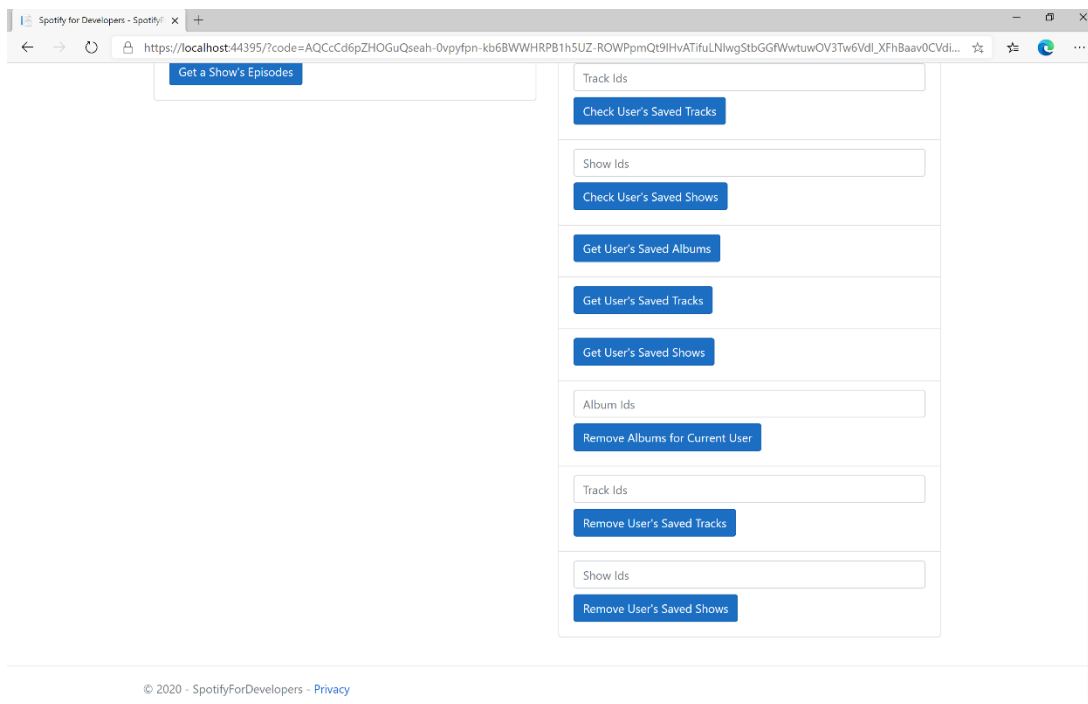


Finally, in **Visual Studio 2019** select **IIS Application** to run the **Web Application**



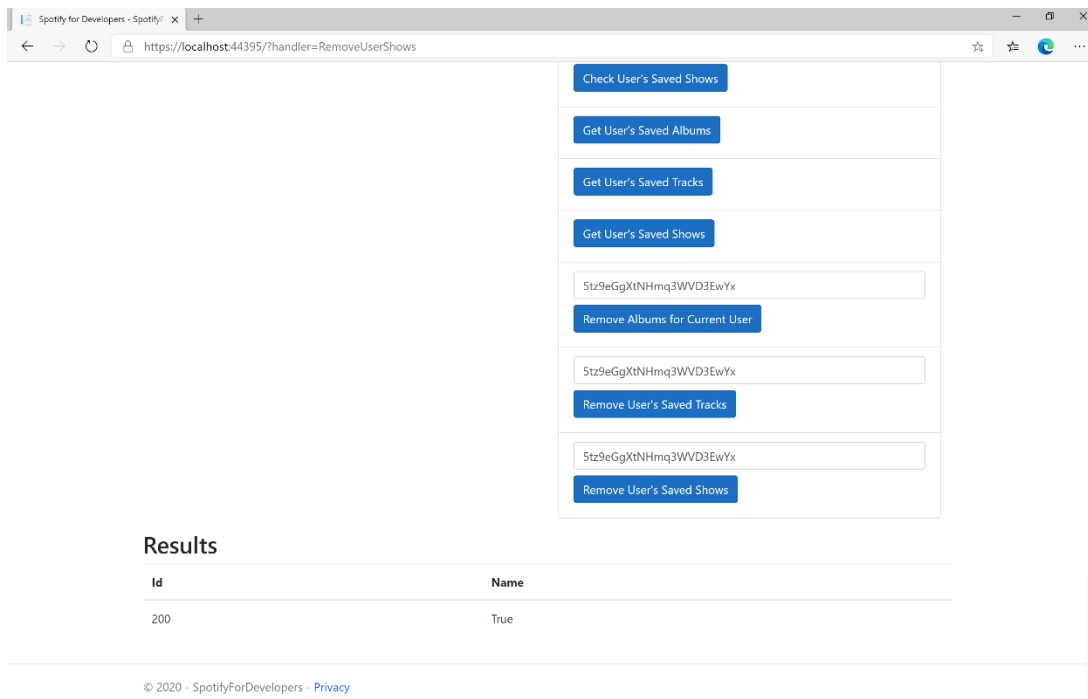
## Step 9

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:



## Step 10

You can get a **Show Id** from **Search For An Item** using **Shows** or copy the same **Show Ids** used in **Save Shows for User** in **Show Ids** above **Remove User's Saved Shows** and then select **Remove User's Saved Shows** and scroll down to view **Results** like the following:



## 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

