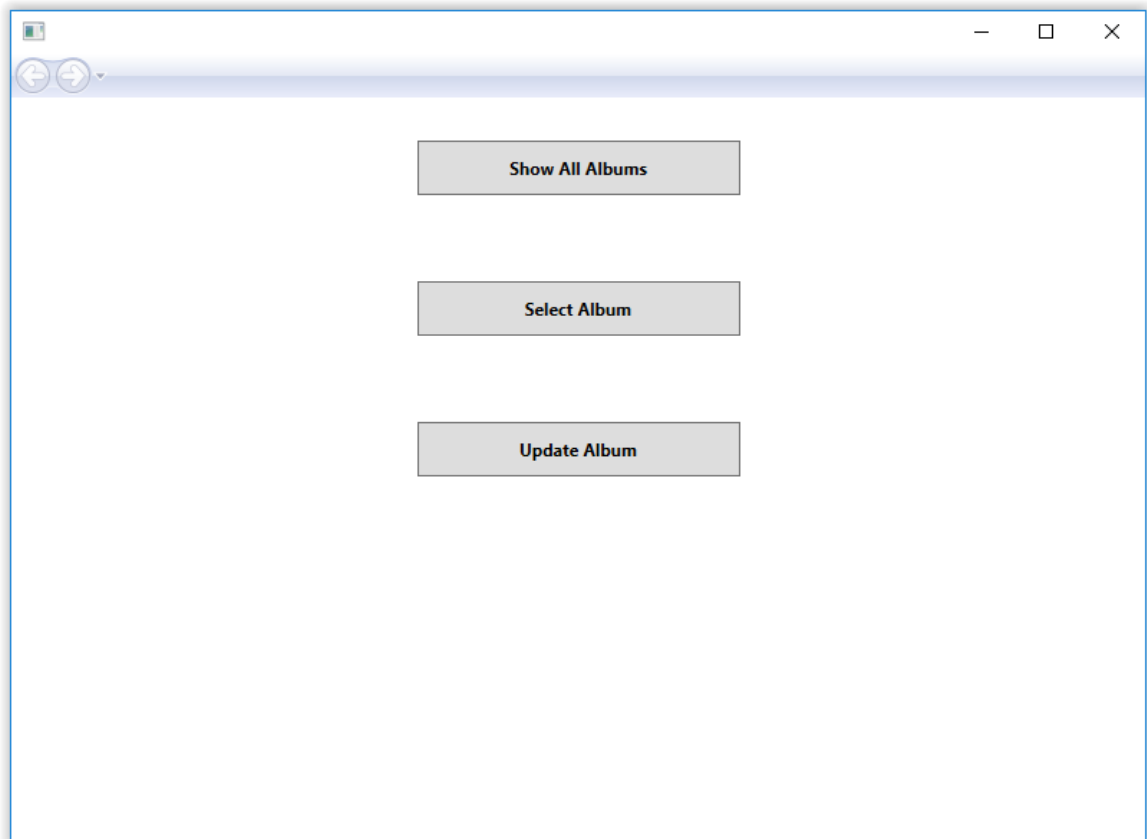


# How to work with parameters and stored procedures

1. Create a new solution called **AlbumAdmin** with a project called **AlbumAdmin** and another project **MusicStoreData**. The first project contains the GUI, the second the database access classes and the DB itself.

The *MainWindow* should be a *Page* (not a *Window*) control. It should contain 3 buttons to navigate to 3 other pages. (Use the *NavigationService* property of the *Page* class to navigate).

The result should look as follows:



- a) When you click the **Show All Albums** button you get the following:

AlbumId	GenreId	ArtistId	Title	Price	AlbumArtUrl
386	1	1	For Those About To Rock We Salute You	8.99	/Content/Images/placeholder.gif
387	1	1	Let There Be Rock	8.99	/Content/Images/placeholder.gif
388	1	100	Greatest Hits	8.99	/Content/Images/placeholder.gif
389	1	102	Misplaced Childhood	8.99	/Content/Images/placeholder.gif
390	1	105	The Best Of Men At Work	8.99	/Content/Images/placeholder.gif
392	1	110	Nevermind	8.99	/Content/Images/placeholder.gif
393	1	111	Compositores	8.99	/Content/Images/placeholder.gif
394	1	114	Bark at the Moon (Remastered)	8.99	/Content/Images/placeholder.gif
395	1	114	Blizzard of Ozz	8.99	/Content/Images/placeholder.gif
396	1	114	Diary of a Madman (Remastered)	8.99	/Content/Images/placeholder.gif
397	1	114	No More Tears (Remastered)	8.99	/Content/Images/placeholder.gif
398	1	114	Speak of the Devil	8.99	/Content/Images/placeholder.gif
399	1	115	Walking Into Clarksdale	8.99	/Content/Images/placeholder.gif
400	1	117	The Beast Live	8.99	/Content/Images/placeholder.gif
401	1	118	Live On Two Legs [Live]	8.99	/Content/Images/placeholder.gif
402	1	118	Riot Act	8.99	/Content/Images/placeholder.gif
403	1	118	Ten	8.99	/Content/Images/placeholder.gif
404	1	118	Vs.	8.99	/Content/Images/placeholder.gif
405	1	120	Dark Side Of The Moon	8.99	/Content/Images/placeholder.gif
406	1	124	New Adventures In Hi-Fi	8.99	/Content/Images/placeholder.gif
407	1	126	Raul Seixas	8.99	/Content/Images/placeholder.gif
408	1	127	By The Way	8.99	/Content/Images/placeholder.gif
409	1	127	Californication	8.99	/Content/Images/placeholder.gif
410	1	128	Retrospective I (1974-1980)	8.99	/Content/Images/placeholder.gif
411	1	130	Maquinarama	8.99	/Content/Images/placeholder.gif
412	1	130	O Samba Poconé	8.99	/Content/Images/placeholder.gif
413	1	132	A-Sides	8.99	/Content/Images/placeholder.gif
414	1	134	Core	8.99	/Content/Images/placeholder.gif
415	1	136	[1997] Black Light Syndrome	8.99	/Content/Images/placeholder.gif
416	1	139	Beyond Good And Evil	8.99	/Content/Images/placeholder.gif
418	1	140	The Doors	8.99	/Content/Images/placeholder.gif
419	1	141	The Police Greatest Hits	8.99	/Content/Images/placeholder.gif
420	1	142	Hot Rocks, 1964-1971 (Disc 1)	8.99	/Content/Images/placeholder.gif
421	1	142	No Security	8.99	/Content/Images/placeholder.gif

Tip: use a *DataGrid* and set the *AutoGenerateColumns* property to *true*.

b) When you click the **Select Album** button you get the following:

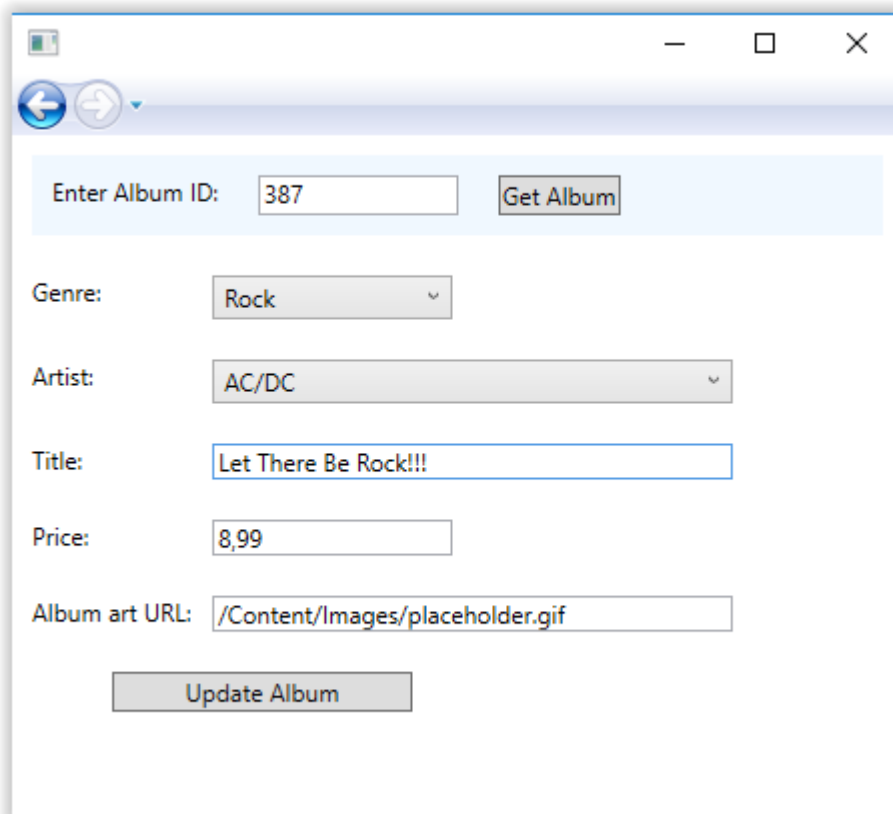
Enter Album ID:	<input type="text" value="387"/>	<input type="button" value="Get Album"/>
Genre:	<input type="text" value="Rock"/>	
Artist:	<input type="text" value="AC/DC"/>	
Title:	<input type="text" value="Let There Be Rock"/>	
Price:	<input type="text" value="8,99"/>	
Album art URL:	<input type="text" value="/Content/Images/placeholder.gif"/>	

When you have entered a correct **album ID**, you can click the **Get Album** Button to retrieve the album. Create a method **GetAlbumById** in the **ArtistRepository** class.

Attention: you have also to retrieve the Genre Name and Artist Name using the genreId and the artistId of the Album. Therefore create a method **GetGenreById** in the **GenreRepository** class and a method **GetArtistById** in the **ArtistRepository** class.

Show a *MessageBox* when the album could not be found.

c) When you click the **Update Album** button you get the following Form:



The screenshot shows a Windows application window with a standard title bar (minimize, maximize, close buttons). Below the title bar is a navigation bar with a back button and a forward button. The main content area contains a form with the following elements:

- A label "Enter Album ID:" followed by a text input field containing the value "387".
- A "Get Album" button.
- A "Genre:" label followed by a dropdown menu showing "Rock".
- An "Artist:" label followed by a dropdown menu showing "AC/DC".
- A "Title:" label followed by a text input field containing the value "Let There Be Rock!!!".
- A "Price:" label followed by a text input field containing the value "8,99".
- An "Album art URL:" label followed by a text input field containing the value "/Content/Images/placeholder.gif".
- An "Update Album" button.

When you have entered a correct **album ID**, you can click the **Get Album** Button to retrieve the album (when you enter an albumId that doesn't exist, you get the error message that no Album was found). You can change the album data and save the changes to the database by clicking the **Update Album** button.

Create a method **UpdateAlbum** in the **AlbumDB** class.

Attention: All the genres and all the artists have to appear in the comboboxes so the user can choose a new genre / artist.

Therefore create a method **GetAllGenres** in the **GenreRepository** class and a method **GetAllArtists** in the **ArtistRepository** class.

2. Apply parameters wherever possible!
3. Experiment with Insert, Update and Delete. E.g. provide a screen for adding, deleting and modifying genres.