\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

INT422 Assignment 8: Build a data model. Implement security. Post to a public web server.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Week 8 Lab: In-class grading: At this point in time, you will have written your

* Data model - design model classes,
* some view model classes-base, withDetails
* mappers.

Make sure the project builds/compiles without errors. Demonstrate this partially-completed project in the lab sessions.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

public class ArtistAddForm

{

public ArtistAddForm() {BirthOrStartDate …}

public string Name { get; set; }

public string BirthName { get; set; }

public DateTime BirthOrStartDate { … }

public string UrlArtist { get; set; }

public SelectList GenreList { get; set; }

}

public class ArtistAdd

{

public string Name { get; set; }

public string BirthName { get; set; }

public DateTime BirthOrStartDate { }

public string UrlArtist { get; set; }

public string Genre { get; set; }

}

public class ArtistBase : ArtistAdd

{

public int Id { get; set; }

public string Executive { get; set; }

}

public class ArtistWithDetails : ArtistBase

{

public int AlbumsCount { get; set; }

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

public class AlbumAddForm

{**// don’t need ArtistId for AddAlbumfor Existing Artist. Because use passed-in ArtistId to set form.ArtistList’s selectedValues**

public AlbumAddForm(){…}

public string Name { get; set; }

public DateTime ReleaseDate { get; set; }

public string UrlAlbum { get; set; }

public SelectList GenreList { get; set; }

public string ArtistName { get; set; }

public MultiSelectList ArtistList { … }

public MultiSelectList TrackList { … }

}

public class AlbumAdd

{**// don’t need ArtistId: the task is just add a new Album with the checkbox groups for the association with Artists and Tracks**

public AlbumAdd() {

ReleaseDate = DateTime.Now;

ArtistIds = new List<int>();

TrackIds = new List<int>();

}

public string Name { get; set; }

public DateTime ReleaseDate { get; set; }

public string UrlAlbum { get; set; }

public string Genre { get; set; }

public List<int> ArtistIds { get; set; }

public List<int> TrackIds { get; set; }

}

public class AlbumBase : AlbumAdd {

public int Id { get; set; }

public string Coordinator { get; set; }

}

public class AlbumWithDetails : AlbumBase

{

public int TracksCount { get; set; }

public int ArtistsCount { get; set; } }\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

public class TrackAddForm{

public string Name { get; set; }

public string Composers { get; set; }

public SelectList GenreList { get; set; }

// Associated item properties

//public string AlbumArtistName { s}

**// Will be a hidden HTML Forms element. Because in method for adding artist, we need to set the added track to the Album (only)**

**public int AlbumId { get; set; }**

public string AlbumName { get; set; }

}

public class TrackAdd

{

public string Name { get; set; }

public string Composers { get; set; }

public string Genre { get; set; }

public int AlbumId { get; set; }

}

public class TrackBase : TrackAdd

{

public int Id { get; set; }

public string Clerk { get; set; }

}

public class TrackWithDetails : TrackBase

{

public TrackWithDetails(){AlbumNames = ..}

public int AlbumsCount { get; set; }

public IEnumerable<string> AlbumNames { g}

}\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Creating data model

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Creating data – Design model classes with association

* follow the diagram.
* Data context properties

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Add new with Associated data**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

When adding a new artist,

* we will not need to configure an associated album.

When adding a new album,

* we WILL need to configure an associated artist
* we will be adding an album for a known artist.

Similarly, when adding a new track,

* we WILL need to configure an associated album.
* we will be adding a track for a known album.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Week 9 Lab - INT422 Assignment 8:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Week 8 Lab: In-class grading: At least one entity (e.g. Artist) works for:

* initial data loading
* GetOneGetAll
* Add new

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Other considerations

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Gener: data type in design model: string

GenerSelectList: return string after selected by user

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Adding Album:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ArtistsController

* public ActionResult AddAlbum(int? id) {}
* public ActionResult AddAlbum(int? id, AlbumAdd newItem) {…}

AlbumAddForm

**// don’t need ArtistId for AddAlbumfor Existing Artist. Because use passed-in ArtistId to set form.ArtistList’s selectedValues**

AlbumAdd

**// don’t need ArtistId: the task is just add a new Album with the checkbox groups for the association with Artists and Tracks**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Adding Track:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

AlbumsController

[HttpPost]

public ActionResult AddTrack(int? id)

public ActionResult AddTrack(int? id, TrackAdd newItem)

TrackAddForm, TrackAdd

**// Need AlbumId, which Will be a hidden HTML Forms element. Because in method for adding artist, we need to set the added track to the Album (only)**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Configure the security settings**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

We will need these role claims:

* Executive – adding Artist
* Coordinator – Adding Album
* Clerk – Adding Track
* Staff

Admin – Loading data

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

setting the secure-related value of the property

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

When you are coding the Add() methods for artist, album, and track, you will need the name of the **current authenticated user** for setting the value of the property Executive, Coordinator or Clerk.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*