Web Based Technologies – HW2

In this homework, you will design a website that displays movie information and allows user to add movies into shopping cart.

You have to design **four** web pages (**Default.aspx**, **Login.aspx**, **MovieInfo.aspx**, and **Cart.aspx**) and a class page (**Movie.cs**) that holds movie information.

Preliminary Work

1. Create a web project for the homework.
2. Download the sample moive-cover images from the web site and put these images into a folder named **images** in your web folder.

Movie.cs

1. Add a class file named **Movie.cs** into your project (Add New Item, Class). This file will contain information about the movies. If you encounter a message which states that this file should be in **App_Code** folder, then select **Yes** and let Visual Studio create the **App_Code** folder for you and put **Movie.cs** file in this folder. After then, you will be able to use the **Movie** class in your application like any other classes of .NET library.
2. Add member variables to the class **Movie** which are **MovieID** (integer), **Title** (string), **Writer** (string), **Director** (string), **Year** (int), and **MovieUrl** (string).
3. Define a constructor for the **Movie** class which takes all of its member variables as parameter (**Hint**: In Visual Studio, type **ctor** and press **TAB** key. This constructs an empty constructor for the class. After then, change the constructor parameters and implementation).
4. Finally, your Movie class should be like this:

```
public class Movie
{
        public int MovieID;
        public string Title;
        public string Writer;
        public string Director;
        public int Year;
        public string MovieUrl;

        public Movie(int MovieID, string Title, string Director,
                        string Writer, int Year, string MovieUrl)
        {
            this.MovieID = MovieID;
            this.Title = Title;
            this.Director = Director;
            this.Writer= Writer;
            this.Year = Year;
            this.MovieUrl = MovieUrl;
        }
}
```

Design of Default.aspx

When loaded, **Default.aspx** page should check whether the user is logged in or not. This check should be accomplished by a **cookie** check. The cookie should contain only the **first name** and **last name** of

the user. If no such cookie is defined, then **Default.aspx** should display a **message** and a **link** to the **Login.aspx** as in Fig. 1.
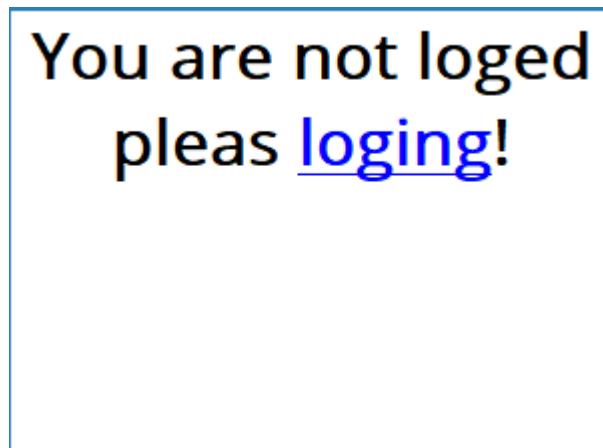

Fig. 1 – Default.aspx when user is not logged in.

Fig. 2 – Login.aspx page

Design of Login.aspx

Login.aspx is very simple web page; it contains only **two textboxes** for **first name** and **last name**, and a **Login** button as in Fig. 2. There will be no password check. Any user that completes this form will be considered as a logged in user.

When the **Login** button is clicked , **first name** and **last name** will be written into a cookie named **UserInfo** for **one month** and the page will be redirected to **Default.aspx** and **Default.aspx** will display **user info** on the left and a **list of movies** on the right as in Fig. 3. The information about the movies is as the following:

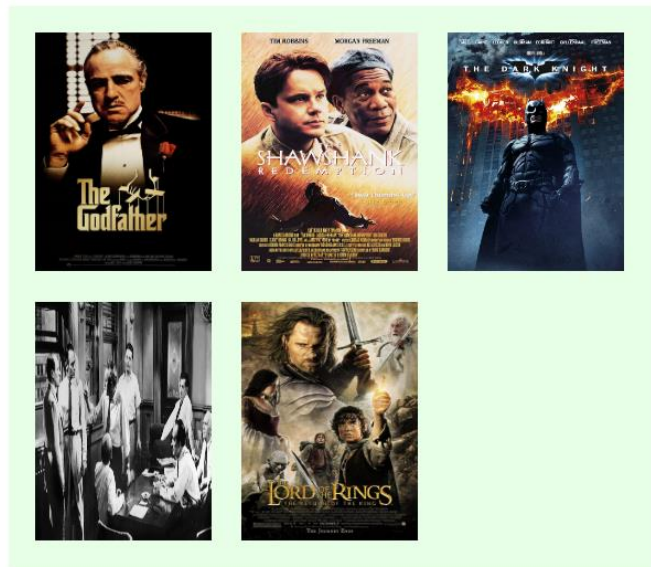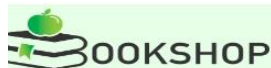| Movie ID	Title	Writer	Director	Year Number
1	The God Father	Mario Puzo	Francis Ford Coppola	1972
2	The shawshank redemption	Stephen King	Frank Darabont	1994
3	The Dark Night	Jonathan Nolan	Christopher Nolan	2008
4	12 Angry Men	Reginald Rose	Sidney Lumet	1957
5	The Lord of the Rings: The Return of the King	J.R.R. Tolkien	Peter Jackson	2003

Table 1 – Movie info

Fig. 3 – Default.aspx page after user is logged in

Default.aspx should display a link to **Cart.aspx** and a **Logout** button on the left. When the **Logout** button is clicked, the **UserInfo** cookie should be remove d and Fig. 1 should be displayed.

Default.aspx should display the titles of the movies on the right with a **link** on each one. Each link should navigate to **MovieInfo.aspx** with query string named **id** (e.g. **MovieInfo.aspx?id=5**).

When **Default.aspx** page is first loaded, it should create five **Movie** objects and put all them in to the **Session state**.

Design of MovieInfo.aspx

When a movie title is clicked in **Default.aspx**, the information about the movie should be displayed in **MovieInfo.aspx** page as in Fig. 4. Movie ID should be taken from **query string** with name **id**.
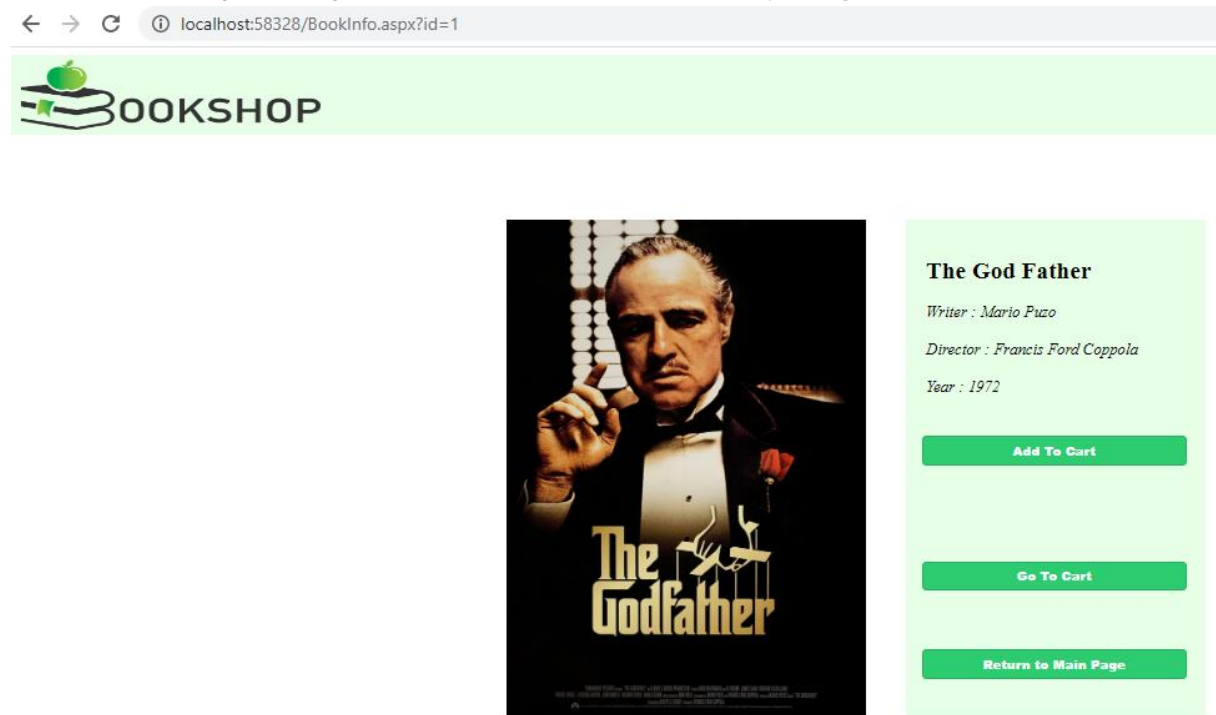

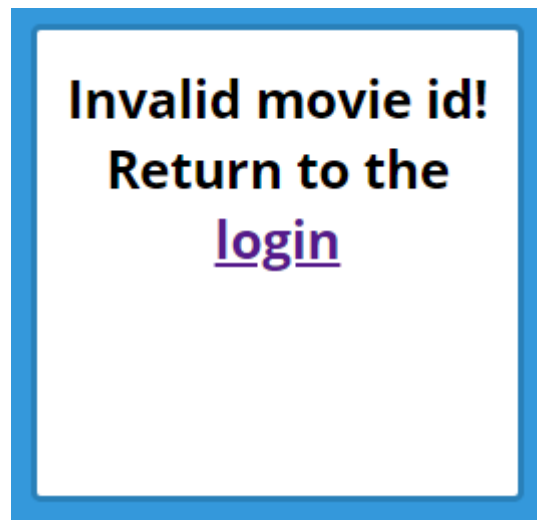**Fig. 4 – MovieInfo.aspx page with a valid movie id**

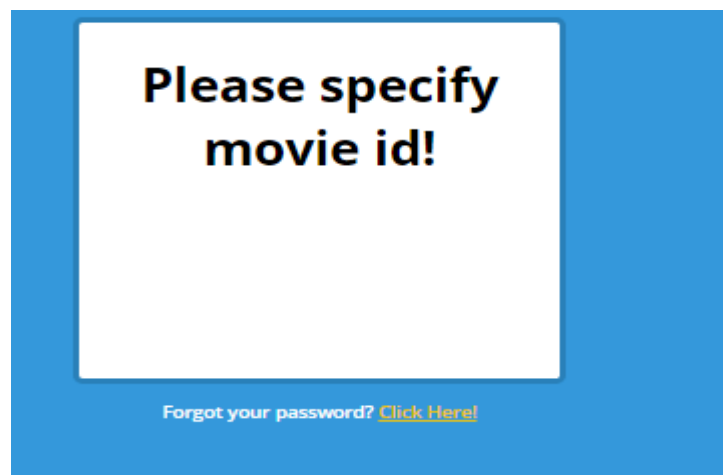**Fig. 5 – MovInfo.aspx when an invalid ID is provided**

Fig. 6 – Movinfo.aspx when no query string is provided

If **MovieInfo.aspx** is requested without any query string or the movie ID does not exist, then it should display an appropriate message as in Figures 5 and 6.

If a valid movie ID is presented, then **MovieInfo.aspx** page should bring the desired movie info from the Session state and di splay the movie info with its cover image.

There also should be an **Add to Cart** button, a link to **Cart.aspx** and a link to **Default.aspx** as in Fig. 4.

When **Add to Cart** button is clicked, the **movie** ID should be added in an **ArrayList** object in Session state that holds the selected movie IDs. After the movie is added into this **ArrayList** object, the user should be informed about this as in Fig. 7.
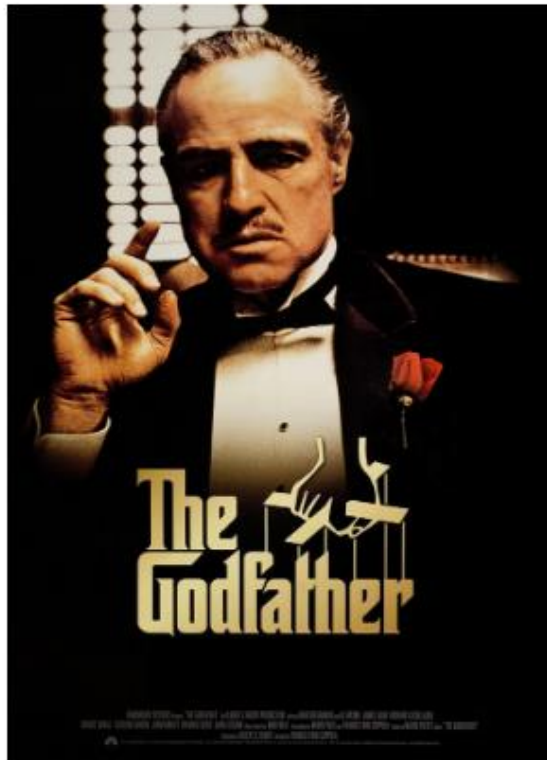
If that movie is already in the cart, an appropriate message should be displayed as in Fig. 7 and Fig. 8.

MovieInfo.aspx should also present links to **Cart.aspx** and **Default.aspx.**

Fig. 7 – MovieInfo.aspx after Add to Cart button is clicked and the movie is successfully added to the cart

If the movie is already in the chart, then display the following message at the below of button:
Movie was already in the cart

Fig. 8 – MovieInfo.aspx after Add to Cart button is clicked and it says that the Movie is already in the shopping cart.

Basic Information About *ArrayList* Class

ArrayList class can be used to store items in an array in object-oriented manner. You can store movie IDs in an **ArrayList** object. An **ArrayList** objec t is created by the new operator:

```
ArrayList SelectedMovieIndices = new ArrayList();
```

An item can be adde d to the **ArrayList** by **Add()** method:

```
SelectedMovieIndices.Add(id);
```

Here, **id** can be of any class. For the demonstration, we assume that **id** is of type **string**.

In order to check whether an item exists in the **ArrayList** or not, **Contains()** method can be used:

```
if (SelectedMovieIndices.Contains(id))
{
    ...
}
```

To remove an item, **Remove()** method can be used. If you want to remove all items, you can use **Clear()** method.

Design of Cart.aspx

Cart.aspx should display all movies that are added to the shopping cart as in Fig. 9 . The IDs of the selected movies should be taken from **Session** state as described above and information about the movies should also be taken from the **Session** state which is put into the **Session** state by **Default.aspx** when it is first displayed. If there is no movie in the shopping cart, a message should be displayed as in Fig. 10.

Cart.aspx page should also presents a link to the main page.

Shopping Cart

Return to Main Page
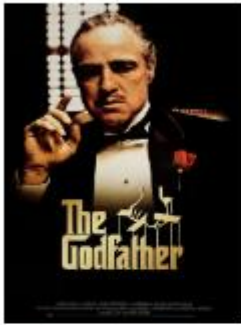
Writer : Mario Puzo

Director : Francis Ford Coppola

Year : 1972

Writer : J.R.R. Tolkien

Director : Peter Jackson

Year : 2003

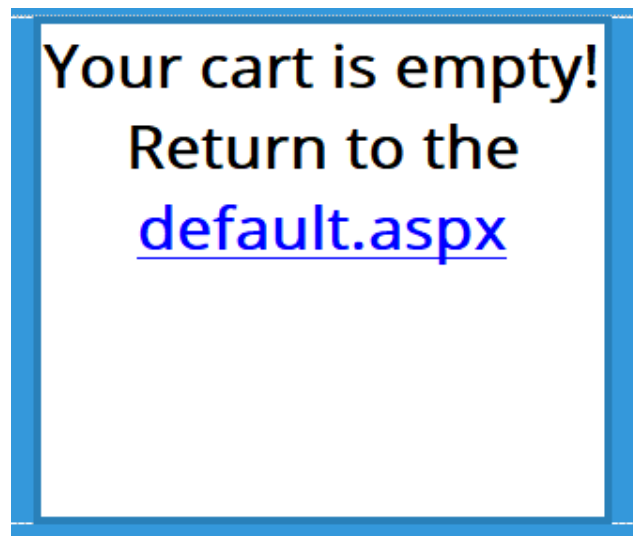**Fig. 9 –** display all movies that are added to the shopping cart **cart**

Your cart is empty!
Return to the
default.aspx

Fig. 10 – if the cart is empty show this message.

Submitting the homework

Attach your images folder to zip file

Examle; Name-Surname-HW#.Zip

🌐 Default.aspx	2019-02-19 15:46	ASP.NET Server Pa	1 KB
Default.aspx.cs	2019-02-20 11:16	CS File	2 KB
Default.aspx.designer.cs	2019-02-19 15:43	CS File	3 KB