

## Web Based Technologies – HW2

In this homework, you will design a news crawling website that parse news from a particular RSS and instantly updates itself.

You have to design **one** web page (**RSSparsing.aspx**) and a class page (**News.cs**) that holds news information.

### News.cs

1. Add a class file named **News.cs** into your project (Add New Item, Class). This file will contain information about the news. 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 **News.cs** file in this folder. After then, you will be able to use the **News** class in your application like any other classes of .NET library.

```
public class News
{
    public int NewsID;
    public string Title;
    public string Description;
    public string Category;
    public string Author;

    public string PubDate;
    public string ImageUrl;

    public News(int NewsID, string Title, string Description,
string Category, string Author, string PubDate, string ImageUrl)
    {
        this.NewsID = NewsID;
        this.Title = Title;
        this.Description = Description;
        this.Category = Category;
        this.Author = Author;
        this.PubDate = PubDate;
        this.ImageUrl = ImageUrl;
    }
}
```

### Design of RSSparsing.aspx

This website will be use to parse the xml in the [http://ajans.dha.com.tr/dha\\_public\\_rss.php](http://ajans.dha.com.tr/dha_public_rss.php).

The following information about news will be downloaded and stored in a ArrayList. You can consider up to 20 news. Each news should be saved with unique id, which is called as **NewsID**. Information that are needed to be parsed are:

- **Description**
- **Category**
- **Author**
- **PubDate**
- **ImageUrl**

This information of news are located in <item> tag. Therefore, you should consider only <item> tags when parsing the xml file.

### Check List of HW2:

1. You should retrieve all information of news from xml file.

2. Then store each news into a news object by using News.cs
3. Store each news object into an ArrayList.
4. Then bind the ArrayList into your gridview.
5. Your gridview should only show the `NewsID`, `Title` and `Category`.
6. You should use a Nuget logging package. You should write the exception message into logger file by using Nuget logging Package.

You can use any Nuget logger package, but I found a special one, which is <https://github.com/serilog/serilog-extensions-logging-file>  
<https://www.nuget.org/packages/Serilog.Extensions.Logging.File/>

## Submitting the HW2

**Example; Name-Surname-HW2#.Zip**