

COP 4814 Lab: BookFinderService, Part 2

This lab continues work you began during the previous Web services lab.

Step 1. Define a class named *Book* in your *IService.cs* file by changing the class name to *Book*. Add properties to the class for the ID (string), Title (string), Author (string), and Year (int). Use the *DataContract* and *DataMember* attributes to make the class and its properties visible to client programs.

Step 2. Modify your constructor in *Service.cs* to build a list of 10 books (having the properties listed in Step #1).

Step 3. Modify the *GetBookList* service method so it returns a sorted array of *Book* objects. The calling program indicates which field should be used for the sort, by passing a string parameter equal to one of the following single letters: ("I" = ID, "T" = title, "A" = author, "Y" = year).

In the Web service, define a separate class for each sort field. Each class implements the *IComparer* interface, as shown in this sample for the ID sort field:

```
public class BookIdComparer : IComparer<Book>
{
    int IComparer<Book>.Compare(Book x, Book y)
    {
        return x.ID.CompareTo(y.ID);
    }
}
```

And this is how you would sort the list by ID:

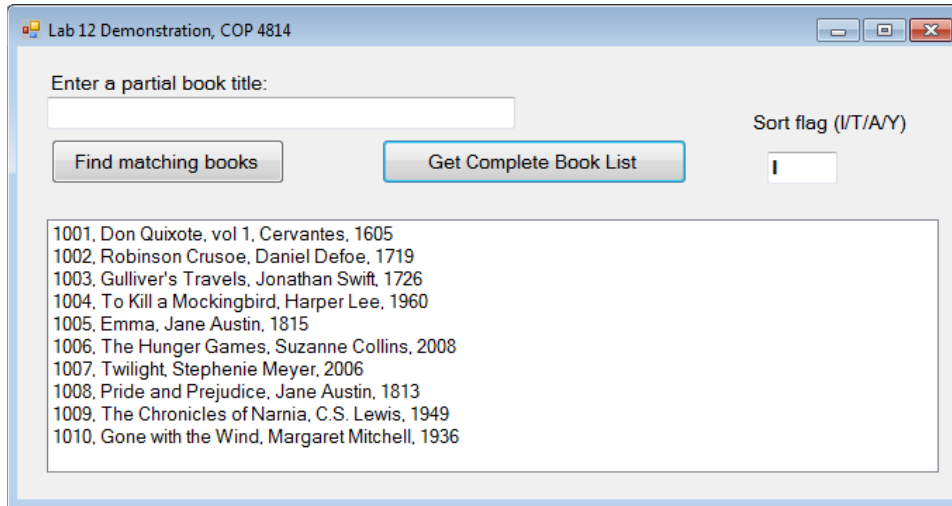
```
bookList.Sort(new BookIdComparer());
```

You can read much more about the *IComparer* interface on the Microsoft web site.

The *Book* class, because it is defined as a *DataContract* inside the web service, could contain a *ToString* method, but there's no way to make that method visible to the client program. On the other hand, you can add another method to the *BookService* web service named **ToBookString(Book b)** that returns a string representation of a book. Then you can call it from the client program, passing it a *Book* object.

Step 3: Update the Client Program

1. Change the user interface to this:



Lab 12 Demonstration, COP 4814

Enter a partial book title:

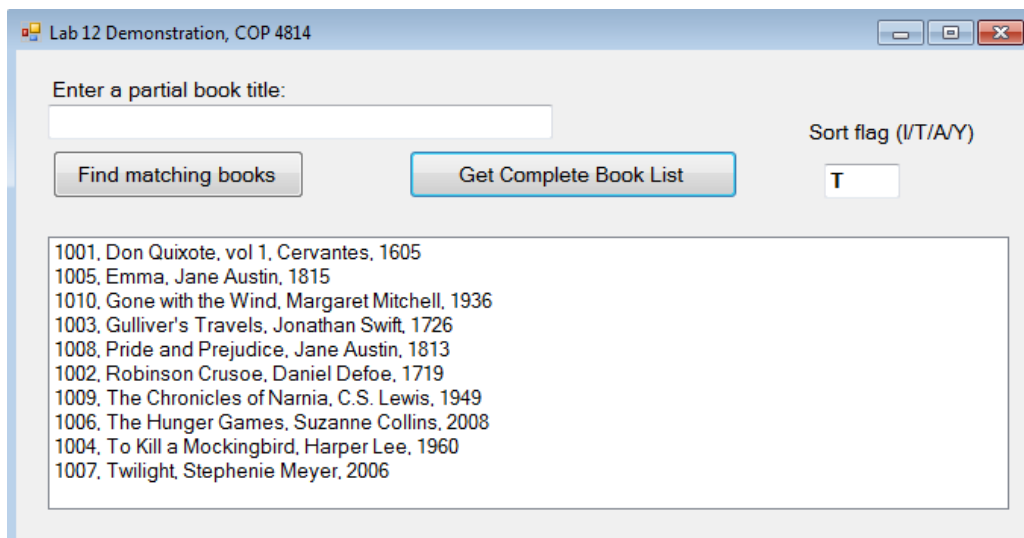
Find matching books Get Complete Book List

Sort flag (I/T/A/Y): I

- 1001, Don Quixote, vol 1, Cervantes, 1605
- 1002, Robinson Crusoe, Daniel Defoe, 1719
- 1003, Gulliver's Travels, Jonathan Swift, 1726
- 1004, To Kill a Mockingbird, Harper Lee, 1960
- 1005, Emma, Jane Austin, 1815
- 1006, The Hunger Games, Suzanne Collins, 2008
- 1007, Twilight, Stephenie Meyer, 2006
- 1008, Pride and Prejudice, Jane Austin, 1813
- 1009, The Chronicles of Narnia, C.S. Lewis, 1949
- 1010, Gone with the Wind, Margaret Mitchell, 1936

2. The *Get Complete Book List* button calls a web service method to get a list of Book objects. Display the list of books in a ListBox control.

3. Let the user enter a capital letter into the textbox, to specify the sort order of the books (“I” = ID, “T” = title, “A” = author, “Y” = year). Pass this letter to the GetBookList method. Here, for example, the books are sorted by title:



Lab 12 Demonstration, COP 4814

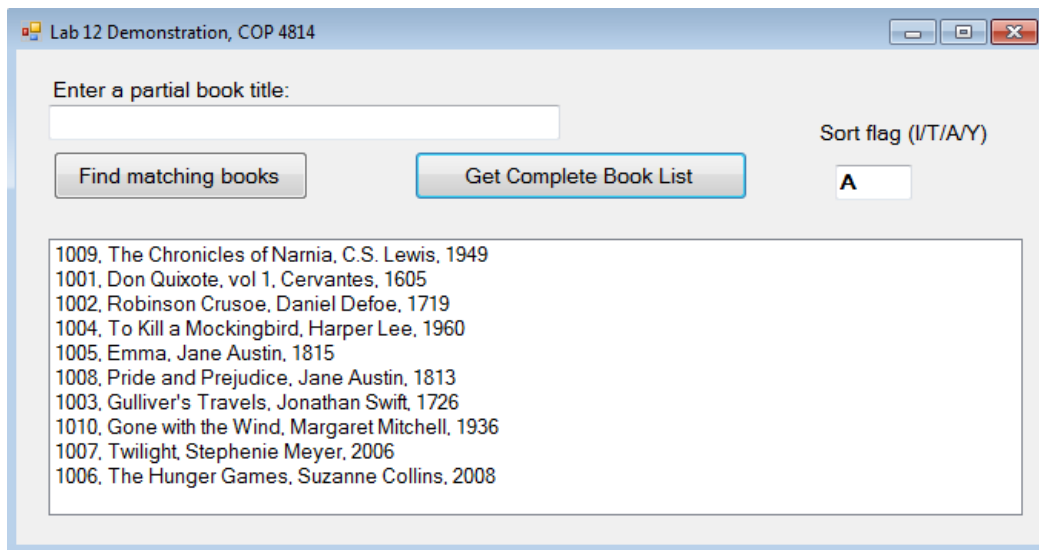
Enter a partial book title:

Find matching books Get Complete Book List

Sort flag (I/T/A/Y): T

- 1001, Don Quixote, vol 1, Cervantes, 1605
- 1005, Emma, Jane Austin, 1815
- 1010, Gone with the Wind, Margaret Mitchell, 1936
- 1003, Gulliver's Travels, Jonathan Swift, 1726
- 1008, Pride and Prejudice, Jane Austin, 1813
- 1002, Robinson Crusoe, Daniel Defoe, 1719
- 1009, The Chronicles of Narnia, C.S. Lewis, 1949
- 1006, The Hunger Games, Suzanne Collins, 2008
- 1004, To Kill a Mockingbird, Harper Lee, 1960
- 1007, Twilight, Stephenie Meyer, 2006

And now they are sorted by author name:



Lab 12 Demonstration, COP 4814

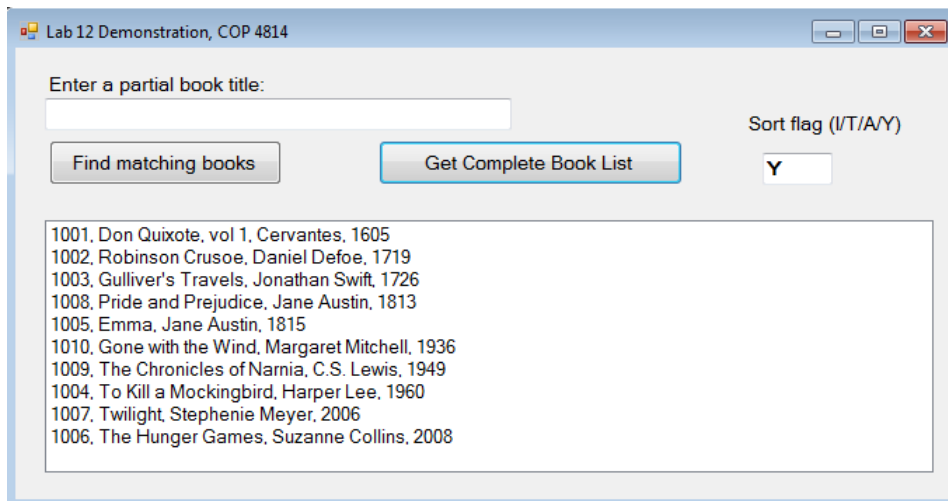
Enter a partial book title:

Find matching books Get Complete Book List

Sort flag (I/T/A/Y) A

1009, The Chronicles of Narnia, C.S. Lewis, 1949
1001, Don Quixote, vol 1, Cervantes, 1605
1002, Robinson Crusoe, Daniel Defoe, 1719
1004, To Kill a Mockingbird, Harper Lee, 1960
1005, Emma, Jane Austin, 1815
1008, Pride and Prejudice, Jane Austin, 1813
1003, Gulliver's Travels, Jonathan Swift, 1726
1010, Gone with the Wind, Margaret Mitchell, 1936
1007, Twilight, Stephenie Meyer, 2006
1006, The Hunger Games, Suzanne Collins, 2008

And finally, by year:



Lab 12 Demonstration, COP 4814

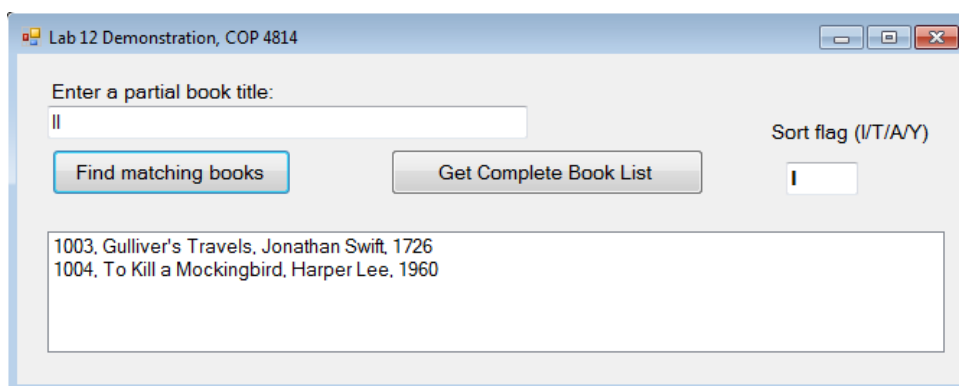
Enter a partial book title:

Find matching books Get Complete Book List

Sort flag (I/T/A/Y) Y

1001, Don Quixote, vol 1, Cervantes, 1605
1002, Robinson Crusoe, Daniel Defoe, 1719
1003, Gulliver's Travels, Jonathan Swift, 1726
1008, Pride and Prejudice, Jane Austin, 1813
1005, Emma, Jane Austin, 1815
1010, Gone with the Wind, Margaret Mitchell, 1936
1009, The Chronicles of Narnia, C.S. Lewis, 1949
1004, To Kill a Mockingbird, Harper Lee, 1960
1007, Twilight, Stephenie Meyer, 2006
1006, The Hunger Games, Suzanne Collins, 2008

4. The *Find Matching Books* button calls the web service method that tries to match a partial string with each of the book titles. The comparisons must be case-insensitive. Here's an example:



Lab 12 Demonstration, COP 4814

Enter a partial book title:
ll

Find matching books Get Complete Book List

Sort flag (I/T/A/Y) I

1003, Gulliver's Travels, Jonathan Swift, 1726
1004, To Kill a Mockingbird, Harper Lee, 1960