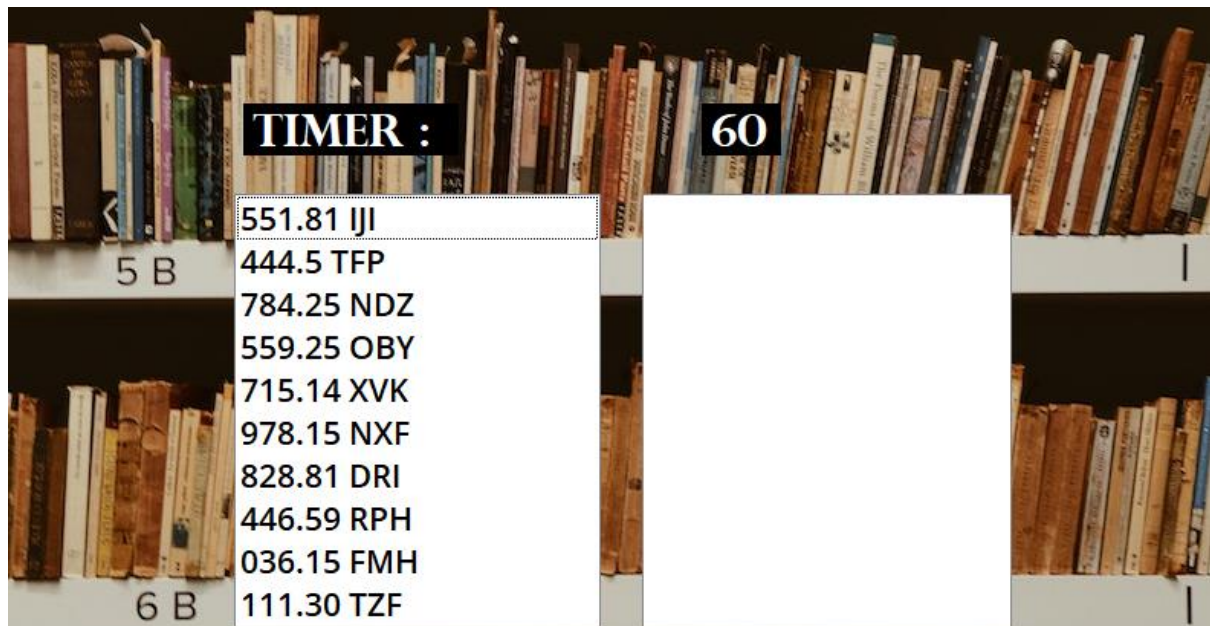


<p>App Functionality: Allow the user to choose which task to practice.</p>	<p>9—10 The feature works perfectly without any errors.</p>	<p>7—8 The feature is well implemented with only one or two bugs.</p>	<p>4—6 The feature is implemented, but there are lots of bugs.</p>	<p>0—3 The feature is not implemented or does not work at all.</p>
---	--	--	---	---



App Functionality: Display ten randomly generated call numbers.	9—10 The feature works perfectly without any errors.	7—8 The feature is well implemented with only one or two bugs.	4—6 The feature is implemented, but there are lots of bugs.	0—3 The feature is not implemented or does not work at all.
--	--	---	--	--



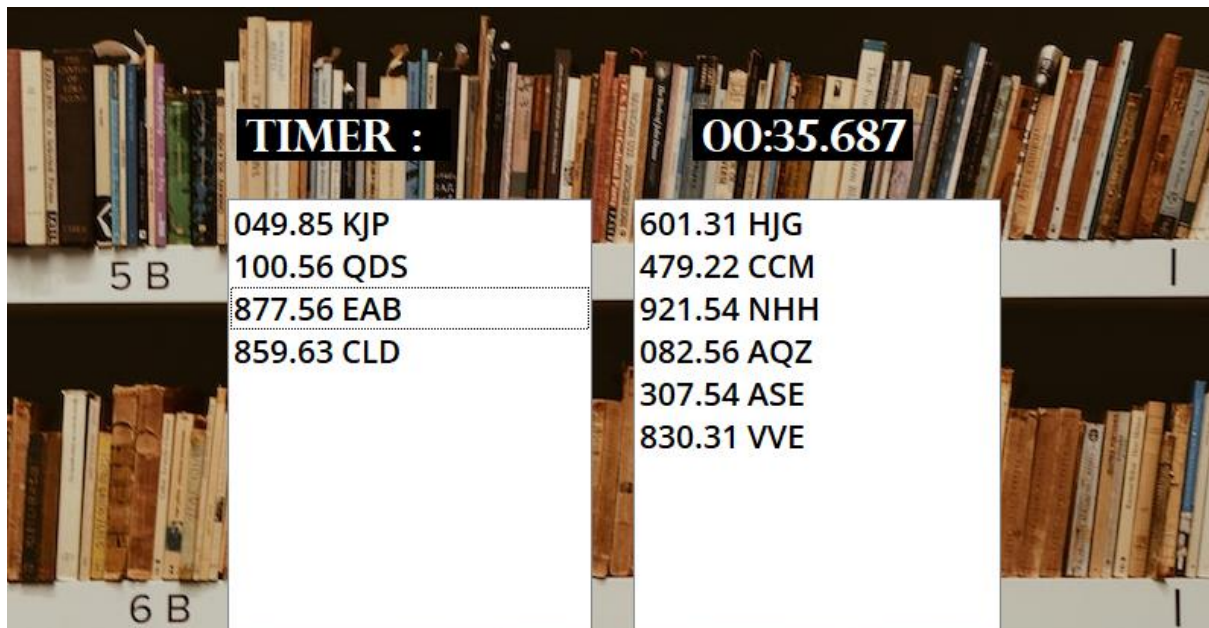
App Functionality: The user can change the order of the call numbers.

9—10
The feature works perfectly without any errors.

7—8
The feature is well implemented with only one or two bugs.

4—6
The feature is implemented, but there are lots of bugs.

0—3
The feature is not implemented or does not work at all.



App Functionality: The app checks whether the user got the ordering correct.

9—10
The feature works perfectly without any errors.

7—8
The feature is well implemented with only one or two bugs.

4—6
The feature is implemented, but there are lots of bugs.

0—3
The feature is not implemented or does not work at all.

Congratulations! You Solved Correctly 🏆

Your Score:

100

Bonus Points:

0

Final Score:

100

Username:

Save Score

OR

Aim for a better score?

Play Again!

Unlucky! You Solved Incorrectly 😞

Your Score:

0

Bonus Points:

0

Final Score:

0

Aim for a better score?

Play Again!

OR

Username:

Save Score

App Functionality: A gamification feature is implemented.	9—10 The feature works perfectly without any errors.	7—8 The feature is well implemented with only one or two bugs.	4—6 The feature is implemented, but there are lots of bugs.	0—3 The feature is not implemented or does not work at all.
--	--	--	---	---

Level Difficulty



Countdown Timers



Points System

Your Score: 100	Bonus Points: 0	Final Score: 100
----------------------------------	----------------------------------	-----------------------------------

Leaderboards



Username	Score
Liam	100
Tyrese	96
Trent	89
Kiaan	84
Ebrahim	79
Abdullah	66
Dhiya	63
Tyler	54
Sarina	15

	5	3—4	1—2	0
App Logic: A list is used to store the call numbers.	A list is consistently used throughout the application to store the call numbers.	A list is used in most places in the app to store the call numbers.	A list is used only in some places with arrays or a different data structure being used in others.	A list is not used at all to store the call numbers.

```

2 references
public class ReplacingTheBooks
{
    #region Generate Game Call Numbers
    1 reference
    public List<string> GenerateCallNos()
    {
        string period = "";
        Random rnd = new Random();
        LinkedList<BookModel> books = new LinkedList<BookModel>();
        List<string> callnums = new List<string>();

        for (int i = 0; i < 10; i++)
        {
            //generate a random number 1 >= n <= 999
            int number = rnd.Next(1, 1000);

            //generate a random number between 1 and 10
            int pCheck = rnd.Next(1, 11);

            if (pCheck > 4) { period = $"{rnd.Next(1, 100)}"; }

            string author = RandomString(3);

            BookModel b = new BookModel($"{number.ToString().PadLeft(3, '0')}{period} {author}");

            books.Append(b);
        }

        books.QuickSort(books.Head);

        foreach (BookModel book in books)
        {
            callnums.Add(book.CallNumber);
        }

        return callnums;
    }
    #endregion

    #region Generate Random String
    1 reference
    public string RandomString(int size)
    {
        Random rnd = new Random();

        const string chars = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";

        return new string(Enumerable.Repeat(chars, size).Select(s => s[rnd.Next(s.Length)]).ToArray());
    }
    #endregion

    #region Shuffle List
    1 reference
    public List<string> ShuffleList(List<string> myList)
    {
        Random rnd = new Random();

        return myList.OrderBy(iten => rnd.Next()).ToList();
    }
    #endregion
}

```

App Logic: An appropriate sorting algorithm is used to sort the call numbers.	9—10 The sorting algorithm works perfectly.	7—8 The sorting algorithm works with only one or two issues.	4—6 The sorting algorithm works some of the time, but there are lots of issues.	0—3 The call numbers are not sorted, or the algorithm does not work for call numbers.
---	--	---	--	--

Recursive Quicksort Algorithm

Adapted from : <https://www.geeksforgeeks.org/quicksort-for-linked-list/>

```

/* A recursive implementation of
quicksort for linked list */
3 references
private void RecursiveQuickSort(Node<T> last, Node<T> head)
{
    if (head != null && last != head && last != head.Next)
    {
        Node<T> temp = Partition(last, head);
        RecursiveQuickSort(last, temp.Previous);
        RecursiveQuickSort(temp.Next, head);
    }
}

// The main function to sort a linked list.
// It mainly calls _quickSort()
1 reference
public void QuickSort(Node<T> node)
{
    // Find last node
    Node<T> head = LastNode(node);

    // Call the recursive QuickSort
    RecursiveQuickSort(node, head);
}

```

	5	3—4	1—2	0
Coding Standards: The code is well structured and documented.	The code is excellently structured, easy to read, and with sufficient detail in the comments.	The code structure can be somewhat improved or too few comments included.	The code is not well structured but somewhat readable, and very few comments are included.	The code is poorly structured, no naming convention used and no comments included.

Code Attribution and Comments to be found in source code

```

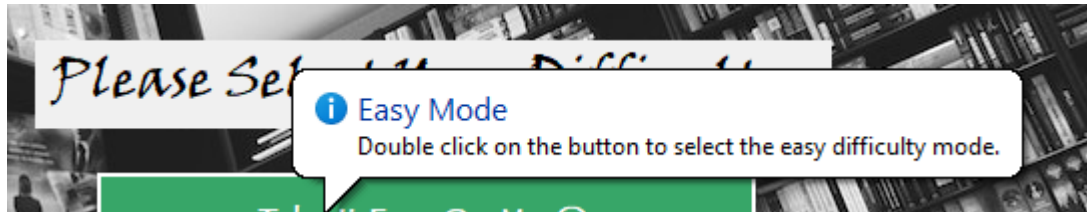
/*
  Author   : Gen Grievous [Youtube]
  Topic    : Windows Forms -ListView add items and read selected
  Resource : [https://www.youtube.com/watch?v=0AvMTu-LFGs]
  Date     : Jun 23, 2017
*/

/*
  Author   : Arkadiusz Raszeja [Stackoverflow]
  Topic    : How to add list items to a ListView in C#winform?
  Resource : [https://stackoverflow.com/questions/43841962/how-to-add-list-items-to-a-listview-in-cwinform]
  Date     : May 8, 2017
*/

```


Other marks: The app is easy to use.	<p>5</p> <p>The app is straightforward and intuitive to use.</p>	<p>3—4</p> <p>The app is relatively easy to use, with only one or two usability issues.</p>	<p>1—2</p> <p>The app can use used, but it is hard to know how to access its features.</p>	<p>0</p> <p>The app is impossible to use.</p>
--------------------------------------	---	--	---	--

Tool tips and instructions can be found along the use of the application



Other Marks: Advanced features not covered in class (Bonus Marks).	[5] Extensive use of advanced features.	[3–4] Good use of advanced features.	[1–2] Limited use of advanced features.	[0] No advanced features were used.
--	--	---	--	--

Bonus features implemented:

- Single Linked Lists
- Json Serialization and Deserialization
- Json Files for data storage
- Advanced LINQ queries
- Sorting Algorithms
- Advanced UI Gestures – Drag and Drop feature
- Custom Countdown Timer Class
- Implementation of Task 2 - Completed ahead of due date
- Multiple Gamification Features – Timer , Leaderboards, Points System , Levels Difficulty