**Display Event Details**

The event details we have got are in a comma-separated format. But suddenly there is a need for the details in a different format. So get details in csv format and change comma into '|'.   
  
Write a program to get event details separated by a comma and change it into a format in which the details are separated by Pipe.

**[Note:  Strictly adhere to the object-oriented specifications given as a part of the problem statement.  
Follow the naming conventions as mentioned. Create separate classes in separate files.]**  
  
Consider the class **Event**with the following private attributes

|  |  |
| --- | --- |
| **Data type** | **Attributes** |
| string | \_eventName |
| string | \_organiserName |
| double | \_eventCost |

The methods for **getters**, **setters** and **constructors** are given in the template code.  
Override **ToString()** method in the Event class to return the event details in the specified format.  
  
Define the following public methods in the **Event**class,

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| void Display(List<Event> eventList) | This method takes event list as parameter and iterates over the list using**Lambda Expression**and displays the event details. Display the statement ‘**List of events**’ inside this method. |

Consider the class **Program**. It includes the method Main.  
In the Main method,

* read the event details from the user, display the list of events using lambda expression.

**Input Format :**  
  
The first line of input corresponds to the number of events 'n'.  
The next 'n' line of inputs corresponds to the event details in the CSV format of (Event Name,Organiser Name,Event Cost).  
  
**Output Format :**  
  
The output consists of event details separated by **"|"**.  
Refer sample input and output for formatting specifications.  
  
**[All text in bold corresponds to input and rest corresponds to output]**  
  
**Sample Input and Output 1:**

Enter the number of events  
**3**  
Enter event details in CSV(Event Name,Organiser Name,Event Cost)  
**Hip Hop,Wright Brothers,300000  
Jaqulos,Einstein,1230000  
Birthday,Joe,120000**  
List of events  
Hip Hop|Wright Brothers|300000  
Jaqulos|Einstein|1230000  
Birthday|Joe|120000

**Sort Hall List**

In Hall management application, there are many halls. When someone wants to see them it will be a mess. So we want to arrange them in particular orders. Thus get the input of item type details from the user and sort it before displaying. So write a program to get the halls and display it in sorted order (by hall onwer name).  
  
**[Note:  Strictly adhere to the object-oriented specifications given as a part of the problem statement.  
Follow the naming conventions as mentioned. Create separate classes in separate files.]**

Consider the class **Hall** with the following private attributes,

|  |  |
| --- | --- |
| **Data Type** | **Attributes** |
| string | \_name |
| double | \_costPerDay |
| string | \_owner |

The methods for **getters**, **setters** and **constructors** are given in the template code.  
Override the method CompareTo to sort the hall details by owner name using lambda expression.

Consider the class **Program**. It includes the method Main.  
In the Main method,

* read the hall details from the user, write a lambda expression to sort the hall list by owner name(ascending order) and display the hall details in the given format.

**Input Format :**

The first line of input is an integer which corresponds to the number of halls 'n'.  
The next 'n' lines of inputs are string which corresponds to the hall details separated by comma (Name,CostPerDay,Owner) .  
  
**Output Format :**

The output consists of hall details sorted by owner name in **ascending order**.  
Use the below format to display hall details  
Console.WriteLine("{0,-15}{1,-10}{2,-10}","Name","Cost","Owner")

Refer sample input and output for formatting specifications.

**[All text in bold corresponds to the input and the rest corresponds to output.]**  
  
**Sample Input and Output :**

Enter the number of halls:  
**3**  
Enter the hall details:  
**Party hall,4000,Jarviz  
Disco hall,5000,Starc  
Dining hall,3000,Chris**  
Hall list:  
Name           Cost      Owner  
Dining hall    3000      Chris  
Party hall     4000      Jarviz  
Disco hall     5000      Starc