Capstone: Airport Locker Rental

Please write an application that allows users to manage locker rentals at an airport. The application should provide users with a menu of five options to view, rent, or end a locker rental, list all locker contents, or quit the application.

Requirements

Please add the following features to your code:

- Use a string array of size 100 to store locker data.
- Locker numbers should be in the range of 1 to 100
  - Remember that array indexes start at zero but humans count from 1, so when asking the user for a locker number, subtract 1 to get the index.
- Use a while loop to display a menu with five options.
- When prompting the user for any information, be sure to use a do...while loop and validate the information. Use TryParse() with friendly error messages.
  - This means do not allow a menu option outside of 1-5 or locker numbers outside of 1-100. Also, when renting a locker do not allow an empty string for the contents.

Menu Options

Before displaying the menu, always clear the console window. The menu output should look like this:

```
Airport Locker Rental Menu
=============================
1. View a locker
2. Rent a locker
3. End a locker rental
4. List all locker contents
5. Quit

Enter your choice (1-5):
```

At the bottom of the while loop, put the following code snippet so that it waits to clear the screen until the user hits a key:

```
Console.WriteLine("\nPress any key to continue...");
Console.ReadKey();
```

View a Locker

1. Prompt the user for a locker number.
2. Display the string value in the corresponding array index.
3. If the element is null, display "EMPTY".

Example output:

```
Enter locker number (1-100): 5
Locker 5 is EMPTY.
```

```
Enter locker number (1-100): 10
Locker 5 contents: Laptop.
```

Rent a Locker

1. Prompt the user for a locker number.
2. Prompt the user for a string value representing what is being stored in the locker.
3. Assign the string value to the corresponding array element.
4. Do not allow rental of a locker that is not empty (null element).

Example output:

```
Enter locker number (1-100): 5
Enter the item you want to store in the locker: Backpack
Locker 5 has been rented for Backpack storage.
```

```
Enter locker number (1-100): 10
Sorry, but locker 10 has already been rented!
```

End a Locker Rental

1. Prompt the user for a locker number.
2. Assign a null value to the corresponding array element.
3. Provide an error message if the locker has not been rented (null element).

Example output:

```
Enter locker number (1-100): 5
Locker 5 rental has ended, please take your Backpack.
```

```
Enter locker number (1-100): 3
Locker 3 is not currently rented.
```

List all Locker Contents:

1. Iterate through the locker array.
2. Output the locker number and string element only if the element is not null.

Example output:

```
Locker 5: Laptop
Locker 10: Backpack
Locker 53: Suitcase
```

Quit

1. Break the while loop to exit the application.

Hints

- Remember to subtract 1 from the locker number to get the correct array index.
- Use int.TryParse() to convert the user's input to an integer value in an if statement.
- Use Console.ReadLine() and Console.WriteLine() for user input and output, respectively.