

```
1 using Microsoft.VisualBasic;
2
3 namespace RepasoCurso
4 {
5     public class Library
6     {
7         public class Book
8         {
9             private string _title;
10            private string _writer;
11            private int _year;
12
13            public string Title
14            {
15                get { return _title; }
16                set { _title = value; }
17            }
18            public string Writer
19            {
20                get { return _writer; }
21                set { _writer = value; }
22            }
23            public int Year
24            {
25                get { return _year; }
26                set { _year = value; }
27            }
28
29            public Book()
30            {
31            }
32
33            public Book (string title, string writer, int year)
34            {
35                _title = title;
36                _writer = writer;
37                _year = year;
38            }
39
40            public bool IsOld()
41            {
42                return _year < 1500;
43            }
44
45            public bool IsValid()
46            {
47                if (_writer == null || _title == null)
48                    return false;
49                return true;
50            }
51
52            public Book Clone()
53            {
```

```
54         Book clone = new Book();
55         clone._writer = _writer;
56         clone._title = _title;
57         clone._year = _year;
58         return clone;
59     }
60 }
61
62 private List<Book> _listBooks = new List<Book>();
63
64 public int ListCount => _listBooks.Count;
65 public bool IsEmpty => _listBooks.Count < 0;
66
67 public Book GetBookAt(int index)
68 {
69     if (index < 0 || index > _listBooks.Count)
70         return null;
71     return _listBooks[index];
72 }
73
74 public int GetIndexOf(Book book)
75 {
76     for (int i = 0; i < _listBooks.Count; i++)
77     {
78         if (_listBooks[i] == book)
79             return i;
80     }
81     return -1;
82 }
83
84 public bool ContainsBook(Book book)
85 {
86     return GetIndexOf(book) >= 0;
87 }
88
89 public int ContainsBookWithTitle(string title)
90 {
91     for (int i = 0; i < _listBooks.Count; i++)
92     {
93         if (_listBooks[i].Title == title)
94             return i;
95     }
96     return -1;
97 }
98
99 public int ContainsBookWithTitle(string title, string writer)
100 {
101     for (int i = 0; i < _listBooks.Count; i++)
102     {
103         if (_listBooks[i].Title == title && _listBooks
104             [i].Writer == writer)
105             return i;
106     }
107 }
```

```
106         return -1;
107     }
108
109     public void AddBook(Book book)
110     {
111         if (book == null || book.IsValid() == false)
112             return;
113         _listBooks.Add(book);
114     }
115
116     public int RemoveBookAt(int index)
117     {
118         if (index < 0 || index > _listBooks.Count)
119             return -1;
120         _listBooks.RemoveAt(index);
121         return 1;
122     }
123
124     public void Clear()
125     {
126         _listBooks = new List<Book>();
127     }
128
129     public List<Book> Clone()
130     {
131         List<Book> cloneLibrary = new List<Book>();
132         cloneLibrary = _listBooks;
133         return cloneLibrary;
134     }
135 }
136 }
137
```