

```

static void Method(ref int i)
{
    i = 44;
}
static void Main() {
    int val = 0;...
public readonly ref struct
Span<T>
{
    internal readonly ref T
    _reference;
    private readonly int _l
length;

    // Omitted for brevity.
..
}

var bc = new BookCollectio
bc.ListBooks();

ref var book = ref bc.GetB
if (book != null)
    book = new Book { Titl
bc.ListBooks();
// The example displays th
//      Call of the Wild,
//      Tale of Two Citie
//
//      Republic, The, by
//      Tale of Two Citie
{
    public string Author;
    public string Title;
}

public class BookCollectio
{
    private Book[] books =
        ne
    };

```

```

        private Book nobook =

        public ref Book GetBoo
        {
            for (int ctr = 0;
            {
                if (title == b
                    return ref
            }
            return ref nobook;
        }

        public void ListBooks(
        {
            foreach (var book
            {
                Console.Writel
            }
            Console.WriteLine(
        }
    }
}

```

```

        ref decimal estValue = ref

    {
        for (int i = 0; i < mat
            rix.GetLength(0); i++)
            for (int j = 0; j <
                matrix.GetLength(1); j++)
                    if (predicate(m
                        atrix[i, j]))
                        return ref
                    matrix[i, j];
            throw new InvalidOperat
                ionException("Not found");
    }
}

```

```

        ref VeryLargeStruct refloc
        public ref decimal GetCurr

```

```

class Product
{
    public Product(string
    {
        ItemName = name;
        ItemID = newID;
    }

    public string ItemName
    public int ItemID { ge
}

private static void Change
{
    // Change the address
    itemRef = new Product(

    // You can change the
    // itemRef. The change
    itemRef.ItemID = 12345
}

private static void Modify
{
    // Declare an instance
    Product item = new Pro
    System.Console.WriteLi
        item.ItemName, ite

    // Pass the product in
    ChangeByReference(ref
    System.Console.WriteLi
        item.ItemName, ite
}

// This method displays th
// Original values in Main
// Back in Main. Name: St

```

```
return ref DecimalArray[0]
```

```
public static ref int Find  
{
```

```
    for (int i = 0; i < ma
```

```
        for (int j = 0; j
```

```
            if (predicate(
```

```
                return ref
```

```
            throw new InvalidOpera
```

```
}
```

```
class RefOverloadExample
```

```
{
```

```
    public void SampleMeth
```

```
    public void SampleMeth
```

```
}
```

```
class CS0663_Example
```

```
{
```

```
    // Compiler error CS06
```

```
    // methods that differ
```

```
    public void SampleMeth
```

```
    public void SampleMeth
```

```
}
```

```
void Method(ref int re
```

```
{
```

```
    refArgument = refA
```

```
}
```

```
int number = 1;
```

```
Method(ref number);
```

```
Console.WriteLine(numbe
```

```
// Output: 45
```