

# A set class for characters

```
/*
C# A Beginner's Guide
By Schildt

Publisher: Osborne McGraw-Hill
ISBN: 0072133295
*/
/*
Project 7-1

A set class for characters.
*/
using System;

class Set {
char[] members; // this array holds the set
int len; // number of members

// Construct a null set.
public Set() {
len = 0;
}

// Construct an empty set of a given size.
public Set(int size) {
members = new char[size]; // allocate memory for set
len = 0; // no members when constructed
}

// Construct a set from another set.
public Set(Set s) {
members = new char[s.len]; // allocate memory for set
for(int i=0; i < s.len; i++) members[i] = s[i]; len = s.len;
// number of members } // Implement read-only Length property.
public int Length { get{ return len; } } // Implement read-
```

```

only indexer. public char this[int idx]{ get { if(idx >= 0 &
idx < len) return members[idx]; else return (char)0; } } /*
See if an element is in the set. Return the index of the
element or -1 if not found. */ int find(char ch) { int i;
for(i=0; i < len; i++) if(members[i] == ch) return i; return
-1; } // Add a unique element to a set. public static Set
operator +(Set ob, char ch) { Set newset = new Set(ob.len +
1); // make a new set one element larger // copy elements
for(int i=0; i < ob.len; i++) newset.members[i] =
ob.members[i]; // set len newset.len = ob.len; // see if
element already exists if(ob.find(ch) == -1) { // if not
found, then add // add new element to new set
newset.members[newset.len] = ch; newset.len++; } return
newset; // return updated set } // Remove an element from the
set. public static Set operator -(Set ob, char ch) { Set
newset = new Set(); int i = ob.find(ch); // i will be -1 if
element not found // copy and compress the remaining elements
for(int j=0; j < ob.len; j++) if(j != i) newset = newset +
ob.members[j]; return newset; } // Set union. public static
Set operator +(Set ob1, Set ob2) { Set newset = new Set(ob1);
// copy the first set // add unique elements from second set
for(int i=0; i < ob2.len; i++) newset = newset + ob2[i];
return newset; // return updated set } // Set difference.
public static Set operator -(Set ob1, Set ob2) { Set newset =
new Set(ob1); // copy the first set // subtract elements from
second set for(int i=0; i < ob2.len; i++) newset = newset -
ob2[i]; return newset; // return updated set } } //
Demonstrate the Set class. public class SetDemo { public
static void Main() { // construct 10-element empty Set Set s1
= new Set(); Set s2 = new Set(); Set s3 = new Set(); s1 = s1 +
'A'; s1 = s1 + 'B'; s1 = s1 + 'C'; Console.WriteLine("s1 after
adding A B C: "); for(int i=0; i

```