[1210] String Parsing

C#'s String class has many built in methods that can help you break data apart into the parts that you are interested in.

The process of breaking data down into smaller, easier to use items is called "parsing." We've seen parsing before when converting strings into other data types like "Integer.TryParse()"

There are two important String methods that we'll need to parse the log files: String.Split() and String.Replace()

String.Split()

The version of String.Split() that we are interested in takes a character as a parameter and returns an array of smaller strings that are sometimes called "tokens." The tokens are the smaller strings that exist between instances of the character used to split the original string. For example:

```
string myString = "One|Two|Three|Four";
string[] tokens = myString.split('|');
foreach (string s in tokens)
    Console.WriteLine(s);
```

Output:

```
One
Two
Three
Four
```

Datafiles and logfile often contain multiple fields on a single line that are separated by special characters so that they can be used easily with String.Split().

String.Replace()

String.Replace() lets you replace a substring inside of a larger string with a different substring. Here is an example:

```
string myString = "One|Two,Three,Four|Five";
string[] tokens = myString.Replace(",", "|").Split('|');
foreach (string s in tokens)
     Console.WriteLine(s);
```

Output:

```
One
Two
Three
Four
Five
```

As you can see, String.Replace() is very helpful when you need to change one or more parts of a logfile so that String.Split() can work better.

In-Class Exercise

Write a program that breaks the following line into into a collection of words using String.Split(). Anything that is not a letter needs to be used as a separator or replaced:

```
Now-Is The*Time*For All Good^Programmers to#come#to the aid-of-their,school!
```

Expected output:

```
Now
Is
The
Time
For
All
Good
Programmers
to
come
to
the
aid
of
their
school
```