

VELOCITY

C String Manipulation Library

Copyright (c) David H. Kristensen

Introduction

The C standard library has many, rich features for string manipulation. But with those rich features comes lacks too - to do a simple insertion of a string, you'd have to pull it apart, add the new substring and reassemble. That is where VELOCITY comes in. Velocity has fast and *easy-to-use* functions for **tokenizing, getting substrings, deleting substrings, inserting substrings and replacing all characters of a kind.**

Functions

```
VTokenizer VTokenize( char * string, char delimiter );
```

This function returns a *VTokenizer* structure (a part of the Velocity library). Here's an example usage of the VTokenizer:

```
VTokenizer temp      = VTokenize("A string to be", ' ');
int count = temp.tokenCount;
char *token_3 = temp.tokens[ 3 ];
```

```
char * VSubString( char * string, unsigned from, unsigned len );
```

This function returns the substring in *string*, starting at *from*, *len* characters long.

```
char * VInsert( char * string, char * new, unsigned where );
```

Inserts the string *new* into *string* at the position *where*

```
char * VDelete( char * string, unsigned from, unsigned len );
```

Deletes *len* characters from *string* starting at *from*

```
char * VReplace( char * string, char this, char with );
```

Replaces all characters, that matches *this* in the string *string* with *with*

As you've probably realized by now, the functions are way easier to use than a rather unwieldy combination of Standard Library functions.

I am constantly looking to expand the VELOCITY repository of string processing functions - if you want to contribute, be sure to e-mail me at: northen@gmail.com

Linkage

http://www.departure.dk/proj_velocity.html