
Subject: Re: Split/Join/Merge
Posted by [Oblivion](#) on Mon, 21 Apr 2014 20:18:06 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Mirek.

Thanks for these handy functions, they are really useful!
But latest SplitMerge.cpp fails to compile on gcc 4.2+;
Also, api doc for Split/Merge/Join stays as a no-name entry in Topic++. Maybe it should go under String utilities topic?
Below errors basically say "no match with function, etc..."

/Ultimate++/uppsrc/Core/SplitMerge.cpp: In function 'Upp::Vector<Upp::String> Upp::Split(int, const char*, int (*)(int), bool)':
Ultimate++/uppsrc/Core/SplitMerge.cpp:38:61: hata: 'SplitGeneric(int&, Upp::Split(int, const char*,

```
    return SplitGeneric<String>(maxcount, delim, s, ignoreempty);  
                                   ^
```

/Ultimate++/uppsrc/Core/SplitMerge.cpp:38:61: bilgi: candidate is:
/Ultimate++/uppsrc/Core/SplitMerge.cpp:6:11: bilgi: template<class S, class Char, class F>
Upp::Vector<T> Upp::SplitGeneric(int, const F&, const Char*, bool)
Vector<S> SplitGeneric(int maxcount, const F& delim, const Char *s, bool ignoreempty = true)
 ^

/Ultimate++/uppsrc/Core/SplitMerge.cpp:6:11: bilgi: template argument deduction/substitution failed:
/Ultimate++/uppsrc/Core/SplitMerge.cpp: In substitution of 'template<class S, class Char, class F> Upp::Vector<T> Upp::SplitGeneric(int, const F&, const Char*, bool) [with S = Upp::String; Char = char; F = Upp::Split(int, const char*, int (*)(int), bool)::<anonymous struct>]':
Ultimate++/uppsrc/Core/SplitMerge.cpp:38:61: required from here
Ultimate++/uppsrc/Core/SplitMerge.cpp:38:61: hata: 'Upp::Split(int, const char*, int (*)(int),

```
    return SplitGeneric<String>(maxcount, delim, s, ignoreempty);  
                                   ^
```

/Ultimate++/uppsrc/Core/SplitMerge.cpp:38:61: hata: 'template<class S, class Char, class F> Upp::Vector<T> Upp::SplitGeneric(int, const F&, const Char*, bool)' örneklenmeye

/Ultimate++/uppsrc/Core/SplitMerge.cpp: In function 'Upp::Vector<Upp::String> Upp::Split(int, const char*, int, bool)':
Ultimate++/uppsrc/Core/SplitMerge.cpp:48:61: hata: 'SplitGeneric(int&, Upp::Split(int, const char*,

yok
 return SplitGeneric<String>(maxcount, delim, s, ignoreempty);
 ^

/Ultimate++/uppsrc/Core/SplitMerge.cpp:48:61: bilgi: candidate is:
/Ultimate++/uppsrc/Core/SplitMerge.cpp:6:11: bilgi: template<class S, class Char, class F> Upp::Vector<T> Upp::SplitGeneric(int, const F&, const Char*, bool)

```
Vector<S> SplitGeneric(int maxcount, const F& delim, const Char *s, bool ignoreempty = true)
```

^

/Ultimate++/uppsrc/Core/SplitMerge.cpp:6:11: bilgi: template argument deduction/substitution failed:

/Ultimate++/uppsrc/Core/SplitMerge.cpp: In substitution of 'template<class S, class Char, class F> Upp::Vector<T> Upp::SplitGeneric(int, const F&, const Char*, bool) [with S = Upp::String; Char = char; F = Upp::Split(int, const char*, int, bool)::<anonymous struct>]':

F = Upp::Split(int, const char*, int, bool)::<anonymous struct>]:

/Ultimate++/uppsrc/Core/SplitMerge.cpp:48:61: required from here

/Ultimate++/uppsrc/Core/SplitMerge.cpp:48:61: hata: 'Upp::Split(int, const char*, int,

```
    return SplitGeneric<String>(maxcount, delim, s, ignoreempty);
```

^

...

Regards.
