
Subject: Re: [BUG] ToUpper, ToLower, and ToAscii (char*, int) causes AssertFailed
Posted by [omari](#) on Mon, 06 Jun 2016 10:56:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

Yes, there is tow,

a - the first can be avoided by specifying a charset,

a possible solution to this is to test if charset is equal to UTF8 (like others ToUpper functions), and do not call s_cset(charset) in this case.

replace : (in file CharSet.cpp, line ~ 2660)

```
void ToUpper(char *t, const char *s, int len, byte charset)
{
    charset = ResolveCharset(charset);
    CharSetData& cs = s_cset(charset);
    const char *lim = s + len;
    while(s < lim)
        *t++ = cs.FromUnicode(UPP::ToUpper(cs.ToUnicode(*s++)));
}
```

by:

```
void ToUpper(char *t, const char *s, int len, byte charset)
{
    charset = ResolveCharset(charset);
    if(charset == CHARSET_UTF8)
    {
        String u = ToUtf8(ToUpper(FromUtf8(s, len)));
        memcpy(t, ~u, len);
        return;
    }
```

```
    CharSetData& cs = s_cset(charset);
    const char *lim = s + len;
    while(s < lim)
        *t++ = cs.FromUnicode(UPP::ToUpper(cs.ToUnicode(*s++)));
}
```

b - here a test case for the second BUG:
the code below produce EXCEPTION_ACCESS_VIOLATION

```
CONSOLE_APP_MAIN
{
    char* s = "abcd";
```

```
ToUpper(s, s, 4, CHARSET_WIN1250);  
}
```

but the code below work as expected:

```
CONSOLE_APP_MAIN  
{  
    char* s = "abcd";  
    StringBuffer r(4);  
    ToUpper(r, s, 4, CHARSET_WIN1250);  
}
```

then, the bug can be resolved by replacing : (in file CharSet.cpp, line ~ 2630)

```
void ToUpper(char *s, int len, byte charset)  
{  
    ToUpper(s, s, len, charset);  
}
```

by

```
void ToUpper(char *s, int len, byte charset)  
{  
    StringBuffer r(len);  
    ToUpper(r, s, len, charset);  
    memcpy(s, r, len);  
}
```