Subject: String.Mid() and Unicode
Posted by jeremy_c on Fri, 10 Sep 2010 08:36:52 GMT
View Forum Message <> Reply to Message

I know a given string is at position 10 (example) and is 3 characters long. I don't know if that's 3 Unicode characters or 3 ASCII characters.

It seems that String.Mid(pos, len) is returning 3 bytes of data, not 3 characters, is that correct? If so, how can I retrieve 3 characters from a string at position XYZ?

Jeremy

Subject: Re: String.Mid() and Unicode
Posted by dolik.rce on Fri, 10 Sep 2010 09:23:34 GMT
View Forum Message <> Reply to Message

jeremy_c wrote on Fri, 10 September 2010 10:36I know a given string is at position 10 (example) and is 3 characters long. I don't know if that's 3 Unicode characters or 3 ASCII characters.

It seems that String.Mid(pos, len) is returning 3 bytes of data, not 3 characters, is that correct? If so, how can I retrieve 3 characters from a string at position XYZ?

Jeremy
Hi Jeremy,
I am by far not an expert on the topic, but I believe you need WString for this. String really works with 8bit characters only, while WString encodes the characters to 16 bits. Little example: String

```
 WString w(s);
 for(int i=0;i<s.GetCount();i++){
  Cout()<<IntStr(s[i])<<" ";
 }
 Cout()<<"\n";
 for(int i=0;i<w.GetCount();i++){
  Cout()<<IntStr(w[i])<<" ";
 }
```

(Well, the example was more to assure myself that I'am not talking nonsense)

Honza

Subject: Re: String.Mid() and Unicode
Posted by cbpporter on Fri, 10 Sep 2010 11:55:00 GMT
View Forum Message <> Reply to Message

All subset and indexing functions work on code points (the smallest binary unit to represent part of the abstract character when stored in memory, byte for String, word for WString) and not code

units (the abstract character). You can usually get the right behavior if you use WString, but it is more of a hack/convenience.