Subject: Re: Unicode Question
Posted by mitch grunes on Fri, 06 Jan 2006 21:16:39 GMT
View Forum Message <> Reply to Message

First, you might look at

  http://www.unicode.org

to see what unicode codes are.

Don't forget that some people write the ASCII subset in 8 bits, others
include a null byte to make it 16.

Open and read an 8 bit code to the file in the usual way:
  a=string(0b) & b=a
  openr,1,'yourfilename'
  readu,1,a

Then if a is 0, drop it. If not, and it is a legit ASCII char, rather
than one of the Unicode prefixes, as you can determine from its range,
the character is string(a). else
  readu,1,b
and the character will be in string
  string([a,b])
Of course, that string is two bytes long - which is right for Unicode.

I haven't checked this out, as I don't have a licensed IDL where I am
now, but it should work. I'll let you figure out the prefix codes.

Subject: Re: Unicode Question
Posted by David Fanning on Fri, 06 Jan 2006 21:30:47 GMT
View Forum Message <> Reply to Message

grunes@yahoo.com writes:

> First, you might look at
>
>   http://www.unicode.org
>
> to see what unicode codes are.
>
> Don't forget that some people write the ASCII subset in 8 bits, others
> include a null byte to make it 16.
>
> Open and read an 8 bit code to the file in the usual way:
>   a=string(0b) & b=a
>   openr,1,'yourfilename'

> readu,1,a
>
> Then if a is 0, drop it. If not, and it is a legit ASCII char, rather
> than one of the Unicode prefixes, as you can determine from its range,
> the character is string(a). else
> readu,1,b
> and the character will be in string
> string([a,b])
> Of course, that string is two bytes long - which is right for Unicode.
>
> I haven't checked this out, as I don't have a licensed IDL where I am
> now, but it should work.

Yeah, that's kinda what I thought, too. But I'm not so sure
it is as simple as this anymore. :-)

But I am handicapped by not having the actual file, too.
I really was just wondering if anyone had any experience
with this. My suggestions are still resulting in a lot of
*&%^$ type of nonsense.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: Unicode Question
Posted by R.Bauer on Sat, 07 Jan 2006 22:57:34 GMT
View Forum Message <> Reply to Message

Hi all

I think it's time to add a feature request to rsi

in python it's done this way

```
import codecs
f = codecs.open("file.txt","rb","utf8").read()
```

Are there plans known about when utf-8 is added to idl?

cheers
Reimar

David Fanning wrote:

> grunes@yahoo.com writes:
>
>> First, you might look at
>>
>> http://www.unicode.org
>>
>> to see what unicode codes are.
>>
>> Don't forget that some people write the ASCII subset in 8 bits, others
>> include a null byte to make it 16.
>>
>> Open and read an 8 bit code to the file in the usual way:
>> a=string(0b) & b=a
>> openr,1,'yourfilename'
>> readu,1,a
>>
>> Then if a is 0, drop it. If not, and it is a legit ASCII char, rather
>> than one of the Unicode prefixes, as you can determine from its range,
>> the character is string(a). else
>> readu,1,b
>> and the character will be in string
>> string([a,b])
>> Of course, that string is two bytes long - which is right for Unicode.
>>
>> I haven't checked this out, as I don't have a licensed IDL where I am
>> now, but it should work.
>
> Yeah, that's kinda what I thought, too. But I'm not so sure
> it is as simple as this anymore. :-)
>
> But I am handicapped by not having the actual file, too.
> I really was just wondering if anyone had any experience
> with this. My suggestions are still resulting in a lot of
> *&%^$ type of nonsense.
>
> Cheers,
>
> David
>

Subject: Re: Unicode Question

Posted by mitch grunes on Sun, 08 Jan 2006 05:40:55 GMT
View Forum Message <> Reply to Message

Oops. Based on

http://www.unicode.org/versions/Unicode4.0.0

There are more characters than 16 bits accounts for. Can everything I said. Used to be true.