

---

Subject: Is there a way to get all character supported in a font with Font?

Posted by [Mountacir](#) on Fri, 15 Apr 2022 19:54:05 GMT

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

---

Hi,

I've been trying to get all code points in a font so i can preview them, but i couldn't do it with the Font Class.

I tried something like this but no luck.

```
char c;
String s;
for(int i = 1; i<1000;i++){
    if(fnt.HasChar(i)){
```

```
        c = i;
        s << c;
```

```
    }else {
        break;
    }
```

```
    }
    LOG(s);
```

In freetype.h this solutions was given

```
FT_ULong  charcode;
FT_UInt   gindex;
```

```
charcode = FT_Get_First_Char( face, &gindex );
while ( gindex != 0 )
```

```
{
    //... do something with (charcode,gindex) pair ...
```

```
    charcode = FT_Get_Next_Char( face, charcode, &gindex );
}
```

Is it possible to add FT\_Get\_Next\_Char and FT\_Get\_First\_Char to Font class?

Thank you.

---

---

Subject: Re: Is there a way to get all character supported in a font with Font?

Posted by [Mountacir](#) on Tue, 19 Apr 2022 23:04:59 GMT

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

---

Hi,

I found a solution in Ultimatepp's github (uppbbox), and i made a simple Char Map with it (Package attached). (I had to upgrade from 16098 to latest)

Fonts like ms serif (raster typeface) don't work.

---

### File Attachments

1) [CharMap.zip](#), downloaded 80 times

---

---

Subject: Re: Is there a way to get all character supported in a font with Font?

Posted by [Mountacir](#) on Tue, 19 Apr 2022 23:15:39 GMT

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

---

Added this

```
if(char_list.GetCount()<=0){  
    return;  
}
```

---

### File Attachments

1) [CharMap2.zip](#), downloaded 78 times

---

---

Subject: Re: Is there a way to get all character supported in a font with Font?

Posted by [deep](#) on Wed, 20 Apr 2022 14:49:08 GMT

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

---

I was checking your CharMap code.

In Indian Unicode scripts there is Extended char set. Which Unicode values are beyond 0xFFFF.

I am enclosing one font from Noto ttf. This has Glyphs from many langs.

Extended chars from 0x10000 to 0x10663

Can you give some hints on how to display Extended chars.

---

### File Attachments

---

- 1) [Screenshot\\_2022-04-20\\_20-16-27.png](#), downloaded 217 times
  - 2) [NotoSans-Regular.ttf](#), downloaded 63 times
- 

---

Subject: Re: Is there a way to get all character supported in a font with Font?

Posted by [Mountacir](#) on Wed, 20 Apr 2022 22:13:46 GMT

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

---

Hi deep,

The solution I used in my code "TypeReader.cpp/h" is from Uppbox (Emoji example), I shamelessly copied it as it is and still learning how it works.

As for the font you attached, I tried it with multiple CharMap viewers on Windows and online, it doesn't seem to go beyond 0xFFFF. But using Photoshop's Glyphs viewer i noticed that it shows Glyphs beyond 0xFFFF, but they don't have a Unicode or a name just a GID (GlyphID?) number. So I think a solution to preview non-Unicode glyphs is needed.

I also tried to loop over 0x10000 to 0x10663, but all characters got replaced.

In the other hand "NotoSansEgyptianHieroglyphs-Regular.ttf" does go beyond 0xFFFF:

---

### File Attachments

- 1) [egy.png](#), downloaded 244 times
- 

---

Subject: Re: Is there a way to get all character supported in a font with Font?

Posted by [mirek](#) on Wed, 18 May 2022 08:44:31 GMT

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

---

[quote title=Mountacir wrote on Fri, 15 April 2022 21:54]Hi,

I've been trying to get all code points in a font so i can preview them, but i couldn't do it with the Font Class.

I tried something like this but no luck.

```
char c;
String s;
for(int i = 1; i<1000;i++){
    if(fnt.HasChar(i)){
```

```
    c = i;
    s << c;
```

```
}else {
```

```
break;  
}  
  
}  
LOG(s);
```

Have you tried to remove break? :)

Other than that, I think what you really need is `IsNormal` - that means that unicode codepoint is really in given font. `HasChar` will return true even if codepoint is rendered using multiple glyphs from the font or even being replaced by some glyph from some other font.

---