

Using Custom Fonts within your BT Project

Custom fonts for your BT Project

Adding a custom font isn't difficult and can add much to your apps appearance with little technical requirements or graphic experience.

This tutorial is based <u>on a thread in the Buzztouch User Forum</u> and subsequent <u>link to an older tutorial</u>. The information isn't mine; I just updated it a bit.

•		
Carrier 🗢	11:10 PM Weather	7 🗅
10]°-;	.
	ruce up your App ppearance with a custom font!	
	(©



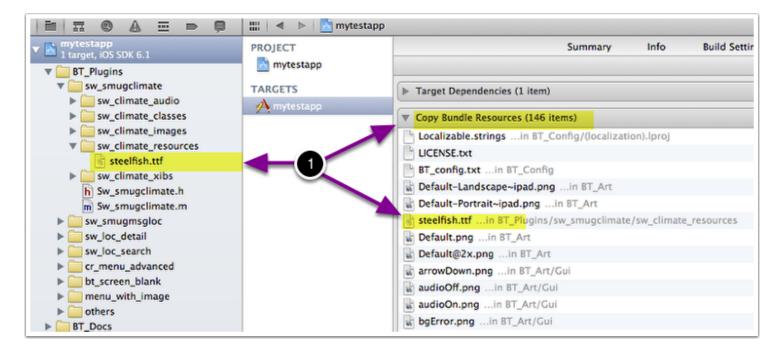
Step 1: Add Fonts

1) Select a font you like, and 'add' it to the BT Project the same way you would documents, images, audio, etcetera. Same procedure.



Step 2: Ensure bundling

1) Ensure the new custom font is properly added to your project, and also be certain that it is added to your 'Copy Bundle Resources' list.





Step 3: Adding Fonts to the Plist

In your BT_info.plist, either 'right-click', or 'control-click' and select "Add Row"

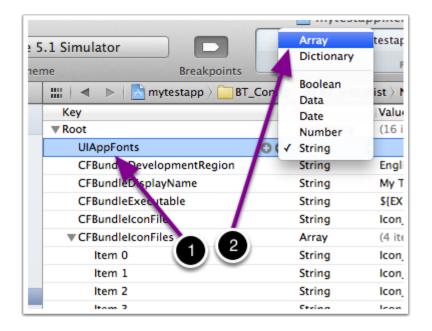
💦 mytestapp	Key		Туре	Valu	e
1 target, iOS SDK 6.1	▼ Root	C .	D ¹ ···	A /2	'ems)
BT_Plugins	CFBundleDevelo	Cut			h
BT_Docs	CFBundleDisplay	Сору			st App
▶ 📴 BT_Images	CFBundleExecut	Paste			CUTABLE_NAME}
BT_Sound	CFBundleIconFile	Chift Dow Dight			57.png
BT_Video	▼ CFBundleIconFile	Shift Row Right			ns)
BT_Config	Item 0	Shift Row Left			57.png
LICENSE.txt README.txt	Item 1	Value Type		Þ	72.png
BT_config.txt	Item 2	value Type			114.png
BT_info.plist	Item 3	Add Row			144.png
h mytestapp appDelegate.h	► CFBundleIcons				n)
m mytestapp_appDelegate.m	CFBundleIdentifi	Show Raw Keys	Values		narianasgps.testapp
Localizable.strings	CFBundleInfoDic	Property List Ty	20	•	
▶ 📴 BT_Layout	CFBundleName	Property List Ty	he		tapp
▶ 🛅 BT_Art	CFBundlePackag	Property List Ed	itor Help	•	
▶ 🧰 BT_Core	CFBundleShortVer		String	1.0	
Frameworks	CFBundleSignature	2	String	7777	
Products	CFBundleVersion		String	1.0	
	▶ UISupportedInterf	UISupportedInterfaceOrientations		(5 items)	
	▶ UISupportedInterf	aceOrientations~ipad	Array	(4 ite	ems)



Step 4: Create a Font Array within the plist

1) Name your new Row "UIAppFonts"

2) Choose "Array" as the item type



Step 5: Add a font to your Font Array

1) Ensure the 'tree view' triangle is pointing down, showing you a (currently) empty array

2) Press the "plus" button to add a new font to your array

Key		Туре	Value
Root		Dictionary	(16 items)
VIAppFonts	00	Array	🛊 (0 items)
C. IndleDevelopmentRegion		String	English
CFBundleDisplayName	-	String	My Test App
CFBundleExecutable	1	String	<pre>\${EXECUTABLE_NAME}</pre>
CFBundleIconFile	1	String	lcon_57.png
▶ CFBundleIconFiles	0	Array	(4 items)
▶ CFBundleIcons	2	Dictionary	(1 item)
CFBundleIdentifier	-	String	com.marianasgps.testapp



Step 5A: Adding a font to your font array

The item will automatically be named 'item 0' (and subsequent items will be incremented accordingly) just enter a (string) value equal to your font filename.

Key	Туре	Value
Root	Dictionary	(16 items)
▼UIAppFonts	Array	(1 item)
ltem 0 🗧	🔵 🗢 String	🛊 steelfish.ttf
CFBundleDevelopmentRegion	String	English
CFBundleDisplayName	String	My Test App
CER U.E. 11	String	\${EXECUTABLE_NAME}
ust add your font filename	String	lcon_57.png
dot add your font monant	Array	(4 items)
▶ CFBundlelcons	Dictionary	(1 item)
CFBundleldentifier	String	com.marianasgps.testapp
CFBundleInfoDictionaryVersion	String	6.0
CFBundleName	String	mytestapp
CFBundlePackageType	String	APPL
CFBundlePackageType CFBundleShortVersionString	String String	APPL 1.0
•	-	
CFBundleShortVersionString	String	1.0
CFBundleShortVersionString CFBundleSignature	String String	1.0 ????

Just use your font by name

Now all you have to do is use your font's correct name in the code to select your custom font!

But wait... 'font's correct name'. This may NOT be the same as the file name.





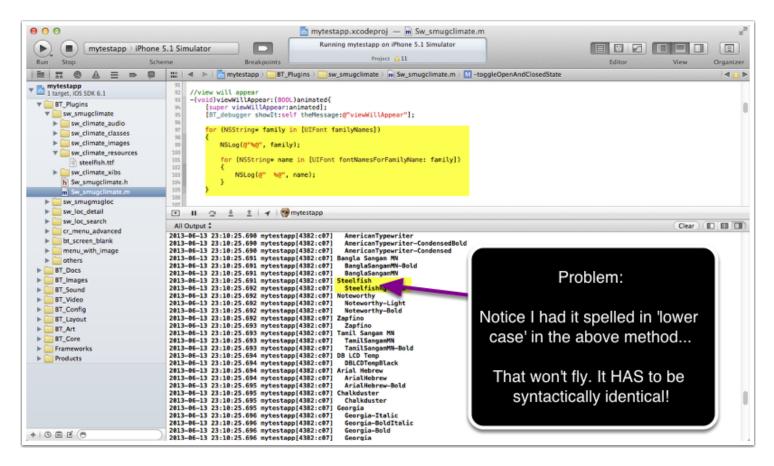
Finding your font's correct name

}

Add this code to your 'view will appear' method, and it will log all fonts available to your app...

```
for (NSString* family in [UIFont familyNames])
{
    NSLog(@"%@", family);
    for (NSString* name in [UIFont fontNamesForFamilyName: family])
    {
        NSLog(@" %@", name);
    }
```

As you know by now, computers are quite literal, and you must have your font name (not font filename) exactly correct in order for it to work.



And that should be about it. Thanks to Mr. David, Kittsy, and the "code with chris" blog.