

NAME

icalBuddyConfig — documentation for the icalBuddy configuration file

SYNOPSIS

The icalBuddy configuration file should be saved in `~/icalBuddyConfig.plist` and is thus per-user. The contents of this file are expected to be a valid property list, so it's recommended that you use the Property List Editor application distributed with the OS X Developer Tools to edit it. If you want to edit this file manually, please consult the *plist(5)* and *plutil(1)* man pages. The example at the end of this man page might also be of help.

DESCRIPTION**Opening the Configuration File for Editing Using icalBuddy**

icalBuddy offers convenience arguments for quickly opening the configuration file for editing (and automatically creating it if it doesn't exist.) Run:

```
icalBuddy editConfig
```

to open the configuration file in a GUI editor (the Property List Editor application, or if you don't have that (it's installed as part of the OS X Developer Tools package,) whichever application is associated with the property list file type) or:

```
icalBuddy editConfigCLI
```

to open it in a command-line editor (either the app that your EDITOR environment variable points to or one of these standard editors (whichever is found first:) vim, vi, nano, pico, emacs, ed.)

Structure of the Configuration File

The root object of the configuration file should be a dictionary. This dictionary may have the following keys:

constantArguments

This key can be used to set constant values for any of icalBuddy's arguments (see the *icalBuddy(1)* man page for documentation for the arguments,) i.e. argument values that will always be set when you run icalBuddy. Using this key will save you the trouble of always having to explicitly specify some arguments when running icalBuddy. The value for this key should be another dictionary, and the keys in this dictionary should be the long names of arguments (as specified in the *icalBuddy(1)* man page.) The types of the values set for each argument key depend on the kind of argument: for arguments that require a textual value to be specified the type should be a String, for arguments that require a numeric value the type should be a Number and for arguments that require no value at all (i.e. if on the command line you would simply set the argument but no value after it) the type should be a Boolean. So for example if you would like to set the `-f` (or `--formatOutput`) argument to always true, you would add the key `formatOutput` with the Boolean value `true` into the `constantArguments` dictionary.

formatting

Specifies the visual formatting for different elements in the output when using ANSI escape sequences formatting (i.e. when the `-f` or `--formatOutput` argument is set.) The value for this key should be another dictionary with any of the keys listed below, with their values corresponding to the syntax described in section **Formatting Parameters** further below. If some of these keys are missing in the configuration file, the default formatting will be used for them. The settings in this dictionary will only be used if the `-f` or `--formatOutput` argument is set when running icalBuddy.

sectionTitle	The title of a section and the section separator
bullet	The normal bullet point
alertBullet	The "alert" bullet point

firstItemLine	The first full line printed for an item (an event or a task)
datetimeName	The name of the datetime property
datetimeValue	A value for the datetime property
titleValue	The title of an item (an event or a task)
calendarNameInTitle	The calendar name, when printed along with an item's title
notesName	The name of the notes property
notesValue	A value for the notes property
urlName	The name of the url property
urlValue	A value for the url property
locationName	The name of the location property
locationValue	A value for the location property
dueDateName	The name of the due date property
dueDateValue	A value for the due date property
priorityName	The name of the priority property
priorityValue	A numeric value for the priority property
priorityValueHigh	The value "high" for the priority property
priorityValueMedium	The value "medium" for the priority property
priorityValueLow	The value "low" for the priority property

formattedKeywords

This key can be used to affect the formatting of all occurrences of any string in the output of icalBuddy. The value for this key should also be a dictionary, its keys the keywords you want formatted in some way and the values corresponding to those keys the parameters that specify the formatting you want. So for example you could specify a more prominent formatting for all instances of the word "today" wherever it may occur in the output by adding formatting parameters (see section **Formatting Parameters** below for the syntax) as the value for the key "today" in this dictionary. The settings in this dictionary will only be used if the `-f` or `--formatOutput` argument is set when running icalBuddy.

Formatting Parameters

In the **formatting** and **formattedKeywords** sections of the configuration file (see section **Structure of the Configuration File** above) you can specify visual formatting parameters for different parts of the output. For example you could make something blue and bold with a white background by specifying a value like "blue, bg:white, bold". Like in this example, you always need to separate different formatting parameters with commas. The parameters you may use are listed below:

black	Black foreground color
white	White foreground color
red	Red foreground color
green	Green foreground color
yellow	Yellow foreground color
blue	Blue foreground color
cyan	Cyan foreground color

magenta	Magenta foreground color
bg:black	Black background color
bg:white	White background color
bg:red	Red background color
bg:green	Green background color
bg:yellow	Yellow background color
bg:blue	Blue background color
bg:cyan	Cyan background color
bg:magenta	Magenta background color
bold	Bold text
underlined	Underlined text
double-underlined	Double underlined text (note that this is not supported by Apple's Terminal application)

EXAMPLES

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>constantArguments</key>
  <dict>
    <key>bullet</key>
    <string>- </string>
    <key>alertBullet</key>
    <string>!! </string>
    <key>noCalendarNames</key>
    <true/>
    <key>formatOutput</key>
    <true/>
    <key>includeCals</key>
    <string>Home,Work</string>
  </dict>
  <key>formatting</key>
  <dict>
    <key>datetimeName</key>
    <string>cyan</string>
    <key>datetimeValue</key>
    <string>yellow</string>
    <key>titleValue</key>
    <string>magenta,bg:black</string>
    <key>notesName</key>
    <string>cyan</string>
    <key>notesValue</key>
    <string>white</string>
    <key>urlName</key>
    <string>cyan</string>
    <key>urlValue</key>
    <string>white</string>
    <key>locationName</key>
    <string>cyan</string>
    <key>locationValue</key>
    <string>white</string>
    <key>dueDateName</key>
    <string>cyan</string>
    <key>dueDateValue</key>
    <string>yellow</string>
    <key>priorityName</key>
    <string>cyan</string>
    <key>priorityValue</key>
    <string>black,bg:white</string>
    <key>priorityValueHigh</key>
    <string>red,bg:white</string>
    <key>priorityValueMedium</key>
    <string>yellow,bg:white</string>
    <key>priorityValueLow</key>
    <string>green,bg:white</string>
    <key>sectionTitle</key>
    <string>blue,underlined</string>
    <key>bullet</key>
  </dict>
</dict>
</plist>

```

```
        <string>white</string>
        <key>alertBullet</key>
        <string>red</string>
        <key>firstItemLine</key>
        <string>bold</string>
    </dict>
    <key>formattedKeywords</key>
    <dict>
        <key>today</key>
        <string>red,bold</string>
    </dict>
</dict>
</plist>
```

SEE ALSO

icalBuddy (1), *plist* (5), *plutil* (1), *icalBuddyLocalization* (1)

AUTHORS

This man page has been written by Ali Rantakari (<http://hasseg.org>)