

Themes Cheat Sheet for Graphics & Apps

Setting & Getting the Theme

```
theme light
theme dark
theme auto
theme(fig,...)
t = theme(fig)
```

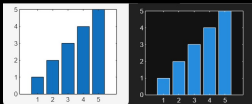
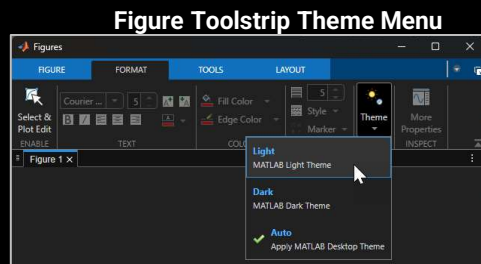


Figure Property

```
fig.Theme = 'light'
t = fig.Theme
```

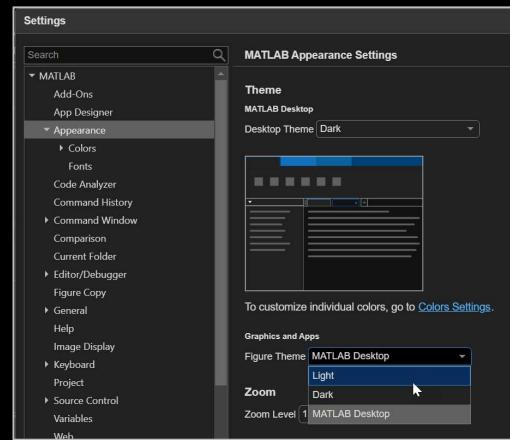
GraphicsTheme Object Properties

```
Name: 'Dark Theme'
BaseColorStyle: 'dark'
```



Automatic Theme Behavior

Theme Preferences



Automatic Figure Theme

```
fig.ThemeMode = 'auto';
```

ThemeMode is auto by default. The theme is chosen by the system according to your theme preferences.

Manual Figure Theme

```
fig.ThemeMode = 'manual';
```

Setting the theme toggles ThemeMode to manual. The theme will not change regardless of preferences.

Adjusting Colors & Icons

Generate lighter/darker colors

```
newcolor = fliplightness(color);
newCData = fliplightness(cdata);
```



Theme Changed Callback

```
fig.ThemeChangedFcn = @(o,e)themeChanged(o,e);

function themeChanged(fig, event)
prevTheme = event.PreviousTheme.BaseColorStyle;
newTheme = event.Theme.BaseColorStyle;
end
```

ChartContainer & ComponentContainer

PROTECTED BASECLASS METHOD

```
currentTheme = getTheme(obj);
```

The update() method will execute when the object's theme changes; leverage getTheme() to react to the current theme in update().

Other Ways to Customize Colors

Axes ColorOrder

```
bar(x,y,'SeriesIndex',2)
plot(x,y,'SeriesIndex','none')
```

Use the SeriesIndex property to specify which index in the ColorOrder a plot should use; 'none' is a neutral color.

```
colororder('meadow')
colororder('reef')
colororder('glow')
...and more!
```

Standard Icons for Button, StateButton, ToggleButton and Dialogs

```
b = uibutton(fig,Icon = 'success')
```

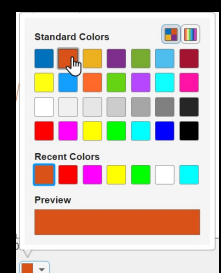


UIStyle for Table, Tree, ListBox and DropDown

```
redFont = uistyle('FontColor','red');
[row,col] = find(uiTable.Data < 0);
addStyle(uiTable,redFont,...
'cell',[row,col])
```

Color Picker Component

```
c = uicolorpicker
```



Themes Cheat Sheet for Graphics & Apps

Setting & Getting the Theme

```
theme light
theme dark
theme auto
theme(fig,...)
t = theme(fig)
```

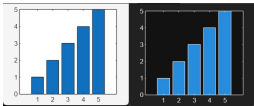


Figure Toolstrip Theme Menu

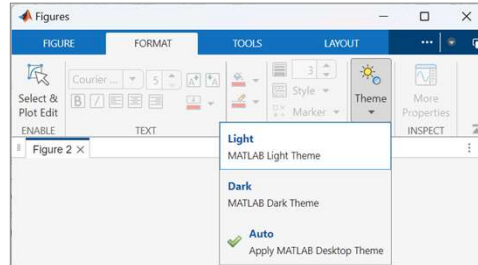


Figure Property

```
fig.Theme = 'light'
t = fig.Theme
```

GraphicsTheme Object Properties

```
Name: 'Dark Theme'
BaseColorStyle: 'dark'
```

Adjusting Colors & Icons

Generate lighter/darker colors

```
newcolor = fliplightness(color);
newCData = fliplightness(cdata);
```



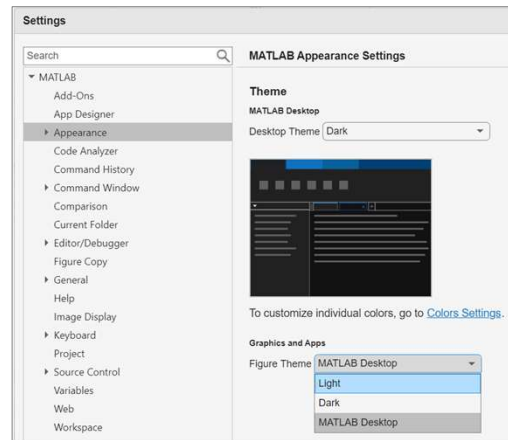
Theme Changed Callback

```
fig.ThemeChangedFcn = @(o,e)themeChanged(o,e);

function themeChanged(fig, event)
prevTheme = event.PreviousTheme.BaseColorStyle;
newTheme = event.Theme.BaseColorStyle;
end
```

Automatic Theme Behavior

Theme Preferences



Automatic Figure Theme

```
fig.ThemeMode = 'auto';
```

ThemeMode is auto by default. The theme is chosen by the system according to your theme preferences.

Manual Figure Theme

```
fig.ThemeMode = 'manual';
```

Setting the theme toggles ThemeMode to manual.

The theme will not change regardless of preferences.

ChartContainer & ComponentContainer

PROTECTED BASECLASS METHOD

```
currentTheme = getTheme(obj);
```

The update() method will execute when the object's theme changes; leverage getTheme() to react to the current theme in update().

Other Ways to Customize Colors

Axes ColorOrder

```
bar(x,y,'SeriesIndex',2)
plot(x,y,'SeriesIndex','none')
```

Use the SeriesIndex property to specify which index in the ColorOrder a plot should use; 'none' is a neutral color.

```
colororder('meadow')
colororder('reef')
colororder('glow')
...and more!
```

Standard Icons for Button, StateButton, ToggleButton and Dialogs

```
b = uibutton(fig,Icon = 'success')
```



UIStyle for Table, Tree, ListBox and DropDown

```
redFont = uistyle('FontColor','red');
[row,col] = find(uiTable.Data < 0);
addStyle(uiTable,redFont,...
'cell',[row,col])
```

Color Picker Component

```
c = uicolorpicker
```

