
Titanium Mobile: API Reference



Titanium Module

Copyright © 2010 Appcelerator, Inc. All rights reserved.

Appcelerator, Inc. 444 Castro Street, Suite 818, Mountain View, California 94041

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Appcelerator, Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Appcelerator's copyright notice.

The Appcelerator name and logo are registered trademarks of Appcelerator, Inc. Appcelerator Titanium is a trademark of Appcelerator, Inc. All other trademarks are the property of their respective owners.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Appcelerator retains all intellectual property rights associated with the technology described in this document.

Every effort has been made to ensure that the information in this document is accurate. Appcelerator is not responsible for typographical or technical errors. Even though Appcelerator has reviewed this document, APPCELERATOR MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY. IN NO EVENT WILL APPCELERATOR BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages. THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Appcelerator dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty. Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Contents

Titanium	Titanium.UI.Window
Titanium.API	Titanium.UI.addEventListener
Titanium.Accelerometer	Titanium.UI.create2DMatrix
Titanium.Analytics	Titanium.UI.create3DMatrix
Titanium.App	Titanium.UI.createActivityIndicator
Titanium.Contacts	Titanium.UI.createAlertDialog
Titanium.Database	Titanium.UI.createAnimation
Titanium.Facebook	Titanium.UI.createButton
Titanium.Filesystem	Titanium.UI.createButtonBar
Titanium.Geolocation	Titanium.UI.createCoverFlowView
Titanium.Gesture	Titanium.UI.createDashboardItem
Titanium.Map	Titanium.UI.createDashboardView
Titanium.Media	Titanium.UI.createEmailDialog
Titanium.Network	Titanium.UI.createImageview
Titanium.Platform	Titanium.UI.createLabel
Titanium.UI	Titanium.UI.createOptionDialog
Titanium.UI.2DMatrix	Titanium.UI.createPicker
Titanium.UI.3DMatrix	Titanium.UI.createPickerColumn
Titanium.UI.ActivityIndicator	Titanium.UI.createPickerRow
Titanium.UI.AlertDialog	Titanium.UI.createProgressBar
Titanium.UI.Android	Titanium.UI.createScrollView
Titanium.UI.Animation	Titanium.UI.createScollableView
Titanium.UI.Button	Titanium.UI.createSearchBar
Titanium.UI.ButtonBar	Titanium.UI.createSlider
Titanium.UI.CoverFlowView	Titanium.UI.createSwitch
Titanium.UI.DashboardItem	Titanium.UI.createTab
Titanium.UI.DashboardView	Titanium.UI.createTabGroup
Titanium.UI.EmailDialog	Titanium.UI.createTabbedBar
Titanium.UI.ImageView	Titanium.UI.createTableView
Titanium.UI.Label	Titanium.UI.createTableViewRow
Titanium.UI.OptionDialog	Titanium.UI.createTableViewSection
Titanium.UI.Picker	Titanium.UI.createTextArea
Titanium.UI.PickerColumn	Titanium.UI.createTextField
Titanium.UI.PickerRow	Titanium.UI.createToolbar
Titanium.UI.ProgressBar	Titanium.UI.createView
Titanium.UI.ScrollView	Titanium.UI.createWebView
Titanium.UI.ScrollableView	Titanium.UI.createWindow
Titanium.UI.SearchBar	Titanium.UI.fireEvent
Titanium.UI.Slider	Titanium.UI.iOS
Titanium.UI.Switch	Titanium.UI.iPad
Titanium.UI.Tab	Titanium.UI.iPhone
Titanium.UI.TabGroup	Titanium.UI.removeEventListener
Titanium.UI.TabbedBar	Titanium.Utils
Titanium.UI.TableView	Titanium.XML
Titanium.UI.TableViewRow	Titanium.Yahoo
Titanium.UI.TableViewSection	Titanium.addEventListener
Titanium.UI.TextArea	Titanium.fireEvent
Titanium.UI.TextField	Titanium.include
Titanium.UI.Toolbar	Titanium.removeEventListener
Titanium.UI.View	Titanium Mobile 1.4.0 - 7/26/2010
Titanium.UI.WebView	

Introduction

This experimental document contains the API documentation available on our Web site, presented as a PDF file.

Please let us know how useful this is. We know there are some formatting issues, and we're working to improve them.

Titanium

module

The top level Titanium module.

Methods

Name	Description
<code>addEventListener</code>	add an event listener for the instance to receive view triggered events
<code>fireEvent</code>	fire a synthesized event to the views listener
<code>include</code>	one or more filenames to include as if the Javascript code was written in place. This is similar to a C <code>#include</code> function.
<code>removeEventListener</code>	remove a previously added event listener

Properties

Name	Type	Description
<code>userAgent</code>	string	the user-agent string used by Titanium
<code>version</code>	string	the version of Titanium that is executing

Events

This module has no events

Notes

Titanium provides a number of built-in objects in the Javascript that are not part of the Titanium namespace. Specifically, the following are available.

JSON

Titanium has a built-in JSON parser with two main functions: `parse` and `stringify`. `JSON.parse` will safely evaluate a string encoded as Javascript into a Javascript object. `JSON.stringify` will encode a Javascript object into a string.

Timers

Titanium has built-in support for one-shot and repeating timers with two main functions: `setTimeout` and `setInterval`. `setTimeout` takes 2 arguments: function and timeout in milliseconds after which the function should be executed and returns a timer handle that can be used to cancel a pending timer with `clearTimeout`. `setInterval` takes 2 arguments: function and timeout in milliseconds for how often the function should be executed until cancelled and returns a timer handle that can be used to cancel a timer with `clearInterval`.

Alert

Titanium has a built-in convenience function `alert` which can be used as a shortcut to `Titanium.UI.createAlertDialog` for creating a message box. Note that unlike a web browser-based version of `alert`, the method is asynchronous.

Titanium.API

submodule of Titanium

The top level API module. The API module is mainly used for logging.

Methods

Name	Description
<code>addEventListener</code>	add an event listener for the instance to receive view triggered events
<code>debug</code>	function for logging debug messages
<code>error</code>	function for logging error messages
<code>fireEvent</code>	fire a synthesized event to the views listener
<code>info</code>	function for logging informational messages
<code>log</code>	function for logging custom severity messages
<code>removeEventListener</code>	remove a previously added event listener
<code>warn</code>	function for logging warning messages

Properties

This module has no properties

Events

This module has no events

Titanium.API.addEventListener

function of Titanium.API

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
<code>name</code>	string	name of the event
<code>callback</code>	function	callback function to invoke when the event is fired

Return Type

`void`

Titanium.API.debug

function of Titanium.API

function for logging debug messages

Arguments

Name	Type	Description
message	string	the message to log

Return Type

void

Titanium.API.error

function of Titanium.API

function for logging error messages

Arguments

Name	Type	Description
message	string	the message to log

Return Type

void

Titanium.API.fireEvent

function of Titanium.API

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

function for logging informational messages

Arguments

Name	Type	Description
message	string	the message to log

Return Type

void

Titanium.API.log

function of Titanium.API

function for logging custom severity messages

Arguments

Name	Type	Description
level	string	the log level
message	string	the message to log

Return Type

void

Titanium.API.removeEventListener

function of Titanium.API

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.API.warn

function of Titanium.API

function for logging warning messages

Arguments

Name	Type	Description
message	string	the message to log

Return Type

void

Titanium.Accelerometer

submodule of Titanium

The top level Accelerometer module. The Accelerometer modules contains methods and properties for using the device accelerometer.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener

Properties

This module has no properties

Events

Name	Description												
update	<p>fired when the accelerometer changes</p> <p>Event properties</p> <table border="1"> <tbody> <tr> <td>source</td> <td>the source object that fired the event</td> </tr> <tr> <td>timestamp</td> <td>reference timestamp since the previous change. this is not a valid timestamp and should simply be used to determine the time between events and not an exact timestamp.</td> </tr> <tr> <td>type</td> <td>the name of the event fired</td> </tr> <tr> <td>x</td> <td>the x axis of the device</td> </tr> <tr> <td>y</td> <td>the y axis of the device</td> </tr> <tr> <td>z</td> <td>the z axis of the device</td> </tr> </tbody> </table>	source	the source object that fired the event	timestamp	reference timestamp since the previous change. this is not a valid timestamp and should simply be used to determine the time between events and not an exact timestamp.	type	the name of the event fired	x	the x axis of the device	y	the y axis of the device	z	the z axis of the device
source	the source object that fired the event												
timestamp	reference timestamp since the previous change. this is not a valid timestamp and should simply be used to determine the time between events and not an exact timestamp.												
type	the name of the event fired												
x	the x axis of the device												
y	the y axis of the device												
z	the z axis of the device												

Notes

You'll want to be selective about turning on and off the device accelerometer. It's recommended you turn off the accelerometer when you're not using it.

You can turn off the accelerometer by simply removing your event listener function. You can turn it back on by re-adding the same function with `addEventListener`.

```
Titanium.Accelerometer.removeEventListener('update',myFunction);
```

Code Examples

Basic Accelerometer Event

Adds an accelerometer update event listener which prints out the x, y and z axis as the device is moved. You should generally remove the event when not used to conserve device resources. If you do not have an active event listener, the accelerometer is turned off to conserve device resources.

```
Titanium.Accelerometer.addEventListener('update',function(e)
{
    Ti.API.debug("accelerometer - x:"+e.x+,y:"+e.y+,z:"+e.z);
});
```

Titanium.Accelerometer.addEventListener

function of Titanium.Accelerometer

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Accelerometer.fireEvent

function of Titanium.Accelerometer

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Accelerometer.removeEventListener

function of Titanium.Accelerometer
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Analytics

submodule of Titanium

The top level Analytics module. The Analytics module is used for transmitting developer-defined Analytics event for your application to the Appcelerator Analytics product. It can be used to augment additional context or application-specific information which can then be accessed during analysis using Analytics.

Methods

Name	Description
addEvent	send a generic event for the application session
addEventListener	add an event listener for the instance to receive view triggered events
featureEvent	send an analytics feature event for the application session
fireEvent	fire a synthesized event to the views listener
navEvent	send an analytics nav event for the application session
removeEventListener	remove a previously added event listener
settingsEvent	send a analytics settings event for the application session
timedEvent	send an analytics timed event for the application session
userEvent	

Properties

This module has no properties

Events

This module has no events

Code Examples

Custom Feature Event

This examples shows how to send a featureEvent during an application session to indicate some feature was triggered/used by the user that you would like to track.

```
Titanium.Analytics.featureEvent('app.feature.blah',{product:'killer'});
```

Titanium.Analytics.addEvent

function of Titanium.Analytics

send a generic event for the application session

Arguments

Name	Type	Description
type	string	the event type
name	string	the event name
data	object	event data or null if not specified. the object must be serializable as JSON

Return Type

void

Titanium.Analytics.addEventListener

function of Titanium.Analytics

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Analytics.featureEvent

function of Titanium.Analytics

send an analytics feature event for the application session

Arguments

Name	Type	Description
name	string	the event name
data	object	event data or null if not specified. the object must be serializable as JSON

Return Type

void

Titanium.Analytics.fireEvent

function of Titanium.Analytics

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Analytics.navEvent

function of Titanium.Analytics

send an analytics nav event for the application session

Arguments

Name	Type	Description
from	string	the <code>from</code> location in the nav event
to	string	the <code>to</code> location in the nav event
name	string	the event name
data	object	event data or null if not specified. the object must be serializable as JSON

Return Type

void

Titanium.Analytics.removeEventListener

function of Titanium.Analytics
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Analytics.settingsEvent

function of Titanium.Analytics

send a analytics settings event for the application session

Arguments

Name	Type	Description
name	string	the event name
data	object	event data or null if not specified. the object must be serializable as JSON

Return Type

void

Titanium.Analytics.timedEvent

function of Titanium.Analytics

send an analytics timed event for the application session

Arguments

Name	Type	Description
name	string	the event name
start	date	the event start as a Date object
stop	date	the event end as a Date object

duration	float	the event duration
data	object	event data or null if not specified. the object must be serializable as JSON

Return Type

void

Titanium.Analytics.userEvent

function of Titanium.Analytics

send an analytics user event for the application session

Arguments

Name	Type	Description
name	string	the event name
data	object	event data or null if not specified. the object must be serializable as JSON

Return Type

void

Titanium.App

submodule of Titanium

The top level App module. The App module is mainly used for accessing information about the application at runtime.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a cross-context application event. listeners in any Javascript context can receive these events if they have added themselves as a listener with the event name. NOTE: you can only pass JSON serializable data in the data payload of the event object since the data must be transportable between contexts.
getArguments	return the arguments passed to the application on startup as a dictionary
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
copyright	string	the application's copyright

description	string	the application's description
guid	string	the application's globally unique id (this is system generated and consistent through all versions)
id	string	the application's app id as specified in Titanium Developer
idleTimerDisabled	boolean	property for controlling whether the phone screen will be locked on idle time. Can be set to true to disable the timer
name	string	the application's name
proximityDetection	boolean	a boolean to indicate whether proximity detection is enabled
proximityState	int	the state of the device's proximity detector
publisher	string	the application's publisher
url	string	the application url
version	string	the application's version

Events

Name	Description						
proximity	<p>fired when a proximity state changes</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td> <td>the source object that fired the event</td> </tr> <tr> <td>state</td> <td>the proximity state value</td> </tr> <tr> <td>type</td> <td>the name of the event fired</td> </tr> </table>	source	the source object that fired the event	state	the proximity state value	type	the name of the event fired
source	the source object that fired the event						
state	the proximity state value						
type	the name of the event fired						

Titanium.App.Properties

submodule of Titanium.App

The App Properties module is used for storing application related property/value pairs which persist beyond application sessions.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
getBool	return a boolean value
getDouble	return a double value
getInt	return an integer value
getList	return a value as an array
getString	return a string value
hasProperty	returns true if the property exists
listProperties	return an array of property properties
removeEventListener	remove a previously added event listener
removeProperty	remove an existing property
setBool	set a property as a boolean value
setDouble	set a property as a double value
setInt	set a property as an integer value
setList	set a property as an array value
setString	set a property as a string value

Properties

This module has no properties

Events

This module has no events

Code Examples

store a property

In this example, we store a string property:

```
Titanium.App.Properties.setString("my_prop", "cool");
```

enumerate over saved properties

In this example, we print out all the saved properties to the console:

```
var props = Titanium.App.Properties.listProperties();
for (var c=0;c<props.length;c++)
{
    var value = Titanium.App.Properties.getString(props[c]);
    Titanium.API.info(props[c] + " = " + value);
}
```

Titanium.App.Properties.addEventListener

function of Titanium.App.Properties

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.App.Properties.fireEvent

function of Titanium.App.Properties

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.App.Properties.setBool

Titanium.App.Properties.getBool

function of Titanium.App.Properties

return a boolean value

Arguments

Name	Type	Description
property	string	return a boolean value for property
default	boolean	optional default value if property is not found

Return Type

boolean

Titanium.App.Properties.getDouble

function of Titanium.App.Properties

return a double value

Arguments

Name	Type	Description
property	string	return a double value for property
default	double	optional default value if property is not found

Return Type

double

Titanium.App.Properties.getInt

function of Titanium.App.Properties

return an integer value

Arguments

Name	Type	Description
property	string	return a integer value for property
default	int	optional default value if property is not found

Return Type

int

Titanium.App.Properties.getList

function of Titanium.App.Properties

return a value as an array

Arguments

Name	Type	Description
property	string	return an array value for property
default	array	optional default value if property is not found

Return Type

array

Titanium.App.Properties.getString

function of Titanium.App.Properties

return a string value

Arguments

Name	Type	Description
property	string	return a string value for property
default	string	optional default value if property is not found

Return Type

string

Titanium.App.Properties.hasProperty

function of Titanium.App.Properties

returns true if the property exists

Arguments

Name	Type	Description
property	string	property name to check

Return Type

boolean

Titanium App Properties listProperties

Titanium.App.Properties.listProperties

function of Titanium.App.Properties

return an array of property properties

Arguments

This function takes no arguments.

Return Type

array

Titanium.App.Properties.removeEventListener

function of Titanium.App.Properties

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.App.Properties.removeProperty

function of Titanium.App.Properties

remove an existing property

Arguments

Name	Type	Description
property	string	property name to remove

Return Type

void

Titanium.App.Properties.setBool

function of Titanium.App.Properties

set a property as a boolean value

Arguments

Name	Type	Description
property	string	property name
value	boolean	value

Return Type

void

Titanium.App.Properties.setDouble

function of Titanium.App.Properties

set a property as a double value

Arguments

Name	Type	Description
property	string	property name
value	double	value

Return Type

void

Titanium.App.Properties.setInt

function of Titanium.App.Properties

set a property as an integer value

Arguments

Name	Type	Description
property	string	property name
value	int	value
property	string	property name
value	int	value

Return Type

void

Titanium.App.Properties.setList

function of Titanium.App.Properties

set a property as an array value

Arguments

Name	Type	Description
property	string	property name
value	array	value

Return Type

void

Titanium.App.Properties.setString

function of Titanium.App.Properties

set a property as a string value

Arguments

Name	Type	Description
property	string	property name
value	string	value

Return Type

void

Titanium.App.addEventListener

function of Titanium.App

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium App fireEvent

Titanium.App.fireEvent

function of Titanium.App

fire a cross-context application event. listeners in any Javascript context can receive these events if they have added themselves as a listener with the event name. NOTE: you can only pass JSON serializable data in the data payload of the event object since the data must be transportable between contexts.

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object
name	string	the event name
data	object	optional data payload for the event. NOTE: you can only pass JSON serializable data since the data must be transportable between contexts.

Return Type

void

Titanium.App.getArguments

function of Titanium.App

return the arguments passed to the application on startup as a dictionary

Arguments

This function takes no arguments.

Return Type

object

Titanium.App.removeEventListener

function of Titanium.App

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Contacts

submodule of Titanium

The top level Contacts module. The Contacts module is used accessing the device Address Book.

Objects

Name	Description
Titanium.Contacts.Group	An object which represents a group in the contacts database.
Titanium.Contacts.Person	An object which represents a person in the contacts database.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
createGroup	create and return an instance of Titanium.Contacts.Group
createPerson	create and return an instance of Titanium.Contacts.Person
fireEvent	fire a synthesized event to the views listener
getAllGroups	Returns an array of all Titanium.Contacts.Group objects in the contacts database
getAllPeople	Returns an array of all Titanium.Contacts.Person objects in the contacts database
getGroupByID	Returns a Titanium.Contacts.Group object with the given ID
getPeopleWithName	Returns an array of Titanium.Contacts.Person objects who have a name (first, last, middle, composite) which matches the given name
getPersonByID	Returns a Titanium.Contacts.Person object with the given ID
removeEventListener	remove a previously added event listener
removeGroup	Removes a group from the contacts database
removePerson	Removes a person from the contacts database
revert	Reverts all changes made to the previous save of the database
save	

Saves all changes to contacts to the database

showContacts

Displays the contact picker

Properties

Name	Type	Description
CONTACTS_KIND_ORGANIZATION	int	constant for 'organization' kind property of Person object
CONTACTS_KIND_PERSON	int	constant for 'person' kind property of Person object
CONTACTS_SORT_FIRST_NAME	int	constant for sorting group members by first name
CONTACTS_SORT_LAST_NAME	int	constant for sorting group members by last name

Events

This module has no events

Notes

Titanium.Contacts.Person objects which have been removed from the database are invalidated after a save operation, whether from creating a new contact or explicitly calling Titanium.Contacts.save(). Using them may result in unpredictable behavior, including crashes.

Titanium.Contacts.Group

object of Titanium.Contacts

An object which represents a group in the contacts database.

Methods

Name	Description
add	Adds a person to the group
members	The complete list of members of the group
remove	Removes a person from the group
sortedMembers	A list of sorted members

Properties

Name	Type	Description
name	string	The name of the group

Events

This object has no events

Titanium.Contacts.Group.add

function of Titanium.Contacts.Group

Adds a person to the group

Arguments

Name	Type	Description
person	object	Titanium.Contacts.Person object to remove from the group

Return Type

void

Titanium.Contacts.Group.members

function of Titanium.Contacts.Group

The complete list of members of the group

Arguments

This function takes no arguments.

Return Type

void

Titanium.Contacts.Group.remove

function of Titanium.Contacts.Group

Removes a person from the group

Arguments

Name	Type	Description
person	object	Titanium.Contacts.Person object to remove from the group

Return Type

void

Titanium.Contacts.Group.sortedMembers

function of Titanium.Contacts.Group

A list of sorted members

Arguments

Name	Type	Description
sortBy	int	Method for sorting. Must be one of Titanium.Contacts.CONTACTS_SORT_FIRST_NAME or Titanium.Contacts.CONTACTS_SORT_LAST_NAME

Return Type

void

Titanium.Contacts.Person

object of Titanium.Contacts

An object which represents a person in the contacts database.

Methods

This object has no methods

Properties

Name	Type	Description
URL	object	URLs of webpages associated with the person. Multi-value, valid labels are: <code>home</code> , <code>work</code> , <code>other</code> , <code>homepage</code> . Values are strings.
address	object	The addresses for the person. Multi-value, valid labels are: <code>home</code> , <code>work</code> , <code>other</code> . Values are dictionaries.
birthday	string	The birthday of the person. Single value, format is "yyyy-MM-dd'T'HH:mm:ss'.SSS+0000"
created	string	The date the person was created in the database. Single value, format is "yyyy-MM-dd'T'HH:mm:ss'.SSS+0000", read-only
date	object	Dates associated with the person. Multi-value, valid labels are: <code>anniversary</code> . Values are strings of format "yyyy-MM-dd'T'HH:mm:ss'.SSS+0000"
department	string	The department of the person. Single value

email	object	The email addresses for the person. Multi-value, valid labels are: <code>home</code> , <code>work</code> , <code>other</code> . Values are strings.
firstName	string	The first name of the person. Single value
firstPhonetic	string	The phonetic first name of the person. Single value
fullName	string	The localized full name of the person. Single value, read-only
image	object	A blob object representing the image for the person. Set to <code>null</code> to remove the image. Single value
instantMessage	object	The instant messenger names for the person. Multi-value, valid labels are: <code>aim</code> , <code>icq</code> , <code>jabber</code> , <code>msn</code> , <code>yahoo</code> . Values are dictionaries.
jobTitle	string	The job title of the person. Single value
kind	int	The kind of person. Single value, one of Titanium.Contacts.CONACTS_KIND_PERSON or Titanium.Contacts.CONACTS_KIND_ORGANIZATION
lastName	string	The last name of the person. Single value
lastPhonetic	string	The phonetic last name of the person. Single value
middleName	string	The middle name of the person. Single value
middlePhonetic	string	The phonetic middle name of the person. Single value
modified	string	The most recent date the person was modified. Single value, format is "yyyy-MM-dd'T'HH:mm:ss'.SSS+0000", read-only
nickname	string	The nickname of the person. Single value
note	string	Notes for the person. Single value
organization	string	The organization the person belongs to. Single value
phone	object	The phone numbers for the person. Multi-value, valid labels are: <code>home</code> , <code>work</code> , <code>other</code> , <code>mobile</code> , <code>pager</code> , <code>workFax</code> , <code>homeFax</code> , <code>main</code> , <code>iPhone</code> . Values are strings.

prefix	string	The prefix for the person. Single value
relatedNames	object	The names of people the person is related to. Multi-value, valid labels are: mother , father , parent , sister , brother , child , friend , spouse , partner , manager , assistant . Values are strings.
suffix	string	The suffix for the person. Single value

Events

This object has no events

Notes

There are two kinds of properties: single value and multivalue. Single value properties are returned as a direct value, while multivalue properties are returned as dictionary objects with keys which represent labels of the property, with values that are arrays of all values in the property which correspond to that label (e.g. {"home" : [address1, address2]})

Titanium.Contacts.addEventListener

function of Titanium.Contacts

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Contacts.createGroup

function of Titanium.Contacts

create and return an instance of Titanium.Contacts.Group

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Contacts.Group

Return Type

object

Titanium.Contacts.createPerson

function of Titanium.Contacts

create and return an instance of Titanium.Contacts.Person

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Contacts.Person

Return Type

object

Titanium.Contacts.fireEvent

function of Titanium.Contacts

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Contacts.getAllGroups

function of Titanium.Contacts

Returns an array of all Titanium.Contacts.Group objects in the contacts database

Arguments

This function takes no arguments.

Return Type

void

Titanium.Contacts.getAllPeople

function of Titanium.Contacts

Returns an array of all Titanium.Contacts.Person objects in the contacts database

Arguments

This function takes no arguments.

Return Type

void

Titanium.Contacts.getGroupByID

function of Titanium.Contacts

Returns a Titanium.Contacts.Group object with the given ID

Arguments

Name	Type	Description
id	int	The database ID of the group to retrieve

Return Type

object

Titanium.Contacts.getPeopleWithName

function of Titanium.Contacts

Returns an array of Titanium.Contacts.Person objects who have a name (first, last, middle, composite) which matches the given name

Arguments

Name	Type	Description
name	string	The name to match in the database

Return Type

array

Titanium.Contacts.getPersonByID

function of Titanium.Contacts

Returns a Titanium.Contacts.Person object with the given ID

Arguments

Name	Type	Description
id	int	The database ID of the person to retrieve

Return Type

object

Titanium.Contacts.removeEventListener

function of Titanium.Contacts
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Contacts.removeGroup

function of Titanium.Contacts

Removes a group from the contacts database

Arguments

Name	Type	Description
group	object	The Titanium.Contacts.Group object to remove from the database.

Return Type

void

Titanium.Contacts.removePerson

function of Titanium.Contacts

Removes a person from the contacts database

Arguments

Name	Type	Description
person	object	The Titanium.Contacts.Person object to remove from the database.

Return Type

void

Titanium.Contacts.revert

function of Titanium.Contacts

Reverts all changes made to the previous save of the database

Arguments

This function takes no arguments.

Return Type

void

Titanium.Contacts.save

function of Titanium.Contacts

Saves all changes to contacts to the database

Arguments

This function takes no arguments.

Return Type

void

Titanium.Contacts.showContacts

function of Titanium.Contacts

Displays the contact picker

Arguments

Name	Type	Description
cancel	function	The function to call when selection is cancelled
selectedPerson	function	The function to call when a person is selected. Mutually exclusive with selectedProperty
selectedProperty	function	The function to call when a property is selected. Mutually exclusive with selectedPerson
animated	boolean	Whether or not to animate the show/hide of the contacts picker
fields	array	A list of field names to show when selecting properties, default is to show all available

Return Type

void

Titanium.Database

The top level Database module. The Database module is used for creating and accessing the in-application Database.

Objects

Name	Description
Titanium.Database.DB	The Database instance returned by Titanium.Database.open or Titanium.Database.install.
Titanium.Database.ResultSet	The ResultSet instance returned by invoking a database SQL execute .

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
install	install a database from the application Resources folder (at build time) and return a reference to the opened database. it is safe to call this method multiple times since this method will only install once if it doesn't already exist on the device.
open	open a database. if it doesn't yet exist, create it.
removeEventListener	remove a previously added event listener

Properties

This module has no properties

Events

This module has no events

Titanium.Database.DB

object of Titanium.Database

The Database instance returned by Titanium.Database.open or Titanium.Database.install.

Methods

Name	Description
close	close the database and release resources from memory. once closed, this instance is no longer valid and must no longer be used.
execute	execute a SQL statement against the database and returns a ResultSet

remove

remove the database files for this instance from disk. WARNING: this is a destructive operation and cannot be reversed. All data in the database will be lost upon calling this function. Use with caution.

Properties

Name	Type	Description
lastInsertRowId	int	the last row identifier by the last INSERT query
name	string	the name of the database
rowsAffected	int	the number of rows affected by the last query

Events

This object has no events

Titanium.Database.DB.close

function of Titanium.Database.DB

close the database and release resources from memory. once closed, this instance is no longer valid and must no longer be used.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Database.DB.execute

function of Titanium.Database.DB

execute a SQL statement against the database and returns a ResultSet

Arguments

Name	Type	Description
sql	string	the SQL to execute
vararg	array,...	one or more optional variable arguments passed to this function or an array of objects to be replaced in the query using ? substitution.

Return Type

object

Titanium.Database.DB.remove

function of Titanium.Database.DB

remove the database files for this instance from disk. WARNING: this is a destructive operation and cannot be reversed. All data in the database will be lost upon calling this function. Use with caution.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Database.ResultSet

object of Titanium.Database

The ResultSet instance returned by invoking a database SQL `execute`.

Methods

Name	Description
<code>close</code>	close the result set and release resources. once closed, this result set must no longer be used
<code>field</code>	retrieve a row value by field index
<code>fieldByName</code>	retrieve a row value by field name
<code>fieldCount</code>	return the number of columns in the result set
<code>fieldName</code>	return the field name for field index
<code>isValidRow</code>	return true if the row is a valid row

Properties

Name	Type	Description
<code>rowCount</code>	int	the number of rows in the result set
<code>validRow</code>	boolean	returns true if the current row is still valid

Events

This object has no events

Titanium.Database.ResultSet.close

function of Titanium.Database.ResultSet

close the result set and release resources. once closed, this result set must no longer be used

Arguments

This function takes no arguments.

Return Type

void

Titanium.Database.ResultSet.field

function of Titanium.Database.ResultSet

retrieve a row value by field index

Arguments

Name	Type	Description
index	int	column index (which is zero based)

Return Type

object

Titanium.Database.ResultSet.fieldByName

function of Titanium.Database.ResultSet

retrieve a row value by field name

Arguments

Name	Type	Description
name	string	column name from SQL query

Return Type

object

Titanium.Database.ResultSet.fieldCount

Titanium.Database.ResultSet.fieldCount

function of Titanium.Database.ResultSet

return the number of columns in the result set

Arguments

This function takes no arguments.

Return Type

int

Titanium.Database.ResultSet.fieldName

function of Titanium.Database.ResultSet

return the field name for field index

Arguments

Name	Type	Description
index	int	field name column index (which is zero based)

Return Type

string

Titanium.Database.ResultSet.isValidRow

function of Titanium.Database.ResultSet

return true if the row is a valid row

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Database.addEventListener

function of Titanium.Database

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
fn	function	the function to be called when the event is fired

callback

function | callback function to invoke when the event is fired

Return Type

void

Titanium.Database.fireEvent

function of Titanium.Database

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Database.install

function of Titanium.Database

install a database from the application Resources folder (at build time) and return a reference to the opened database. it is safe to call this method multiple times since this method will only install once if it doesn't already exist on the device.

Arguments

Name	Type	Description
path	string	the path (relative to the main application Resources folder at build time) to the db to install. this file must be in the SQLite 3 file format.
name	string	the name of the database

Return Type

object

Titanium.Database.open

function of Titanium.Database

open a database. if it doesn't yet exist, create it.

Arguments

Name	Type	Description
name	string	the name of the database

Return Type

object

Titanium.Database.removeListener

function of Titanium.Database
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Facebook

submodule of Titanium

The top level Facebook module. The Facebook module is used for connecting your application with Facebook through Facebook Connect.

Objects

Name	Description
Titanium.Facebook.LoginButton	The Login Button created by Titanium.Facebook.createLoginButton.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
createLoginButton	create and return an instance of Titanium.Facebook.LoginButton
execute	execute a FB API execute request
fireEvent	fire a synthesized event to the views listener
hasPermission	checks the existing permission and returns true if the user has granted the requested permission
isLoggedIn	return true if the user has logged in

publishStream	execute a stream request to FB
query	execute a FQL query against the FB API
removeEventListener	remove a previously added event listener
requestPermission	request a special permission from the user

Properties

Name	Type	Description
loggedIn	boolean	return true if the user has logged in
permissions	object	return a dictionary of permissions with the keys being the name of the permission and the value being a boolean of true if granted, false if not granted
session	object	return the special properties of the session
userId	long	the unique user id returned from Facebook. returns 0 if not logged in

Events

Name	Description										
login	<p>fired at session login</p> <p>Event properties</p> <table border="1"> <tr> <td>cancel</td> <td>true if the user cancelled the request by closing the dialog</td> </tr> <tr> <td>error</td> <td>error message if success was false</td> </tr> <tr> <td>source</td> <td>the source object that fired the event</td> </tr> <tr> <td>success</td> <td>true if the login was successful</td> </tr> <tr> <td>type</td> <td>the name of the event fired</td> </tr> </table>	cancel	true if the user cancelled the request by closing the dialog	error	error message if success was false	source	the source object that fired the event	success	true if the login was successful	type	the name of the event fired
cancel	true if the user cancelled the request by closing the dialog										
error	error message if success was false										
source	the source object that fired the event										
success	true if the login was successful										
type	the name of the event fired										
logout	<p>fired at session logout</p> <p>Event properties</p>										

cancel	true if the user cancelled the request by closing the dialog
error	error message if success was false
source	the source object that fired the event
success	true if the logout was successful
type	the name of the event fired

Titanium.Facebook.LoginButton

object of Titanium.Facebook

The Login Button created by Titanium.Facebook.createLoginButton.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListenner	add an event listenner for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hiearchy
removeEventListenner	remove a previously added event listenner
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)

backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.

float value or a string of the width.

touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto` , a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
cancel	<p>fired when the login is cancelled</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired						
source	the source object that fired the event										
type	the name of the event fired										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
doubletap	<p>fired when the device detects a double tap against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates								
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										

source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

login

fired at session login

Event properties

cancel	true if the user cancelled the request by closing the dialog
error	error message if success was false
source	the source object that fired the event
success	true if the login was successful
type	the name of the event fired

logout

fired at session logout

Event properties

cancel	true if the user cancelled the request by closing the dialog
error	error message if success was false
source	the source object that fired the event
success	true if the logout was successful
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart	fired as soon as the device detects a gesture
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
twofingertap	fired when the device detects a two-finger tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Titanium.Facebook.LoginButton.add

function of Titanium.Facebook.LoginButton
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.Facebook.LoginButton.addEventListener

function of Titanium.Facebook.LoginButton
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event

callback	function	callback function to invoke when the event is fired
-----------------	----------	---

Return Type

void

Titanium.Facebook.LoginButton.animate

function of Titanium.Facebook.LoginButton

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.Facebook.LoginButton.fireEvent

function of Titanium.Facebook.LoginButton

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Facebook.LoginButton.hide

function of Titanium.Facebook.LoginButton

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.Facebook.LoginButton.remove

function of Titanium.Facebook.LoginButton
remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.Facebook.LoginButton.removeEventListener

function of Titanium.Facebook.LoginButton
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Facebook.LoginButton.show

function of Titanium.Facebook.LoginButton
make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.Facebook.LoginButton.toImage

function of Titanium.Facebook.LoginButton
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.Facebook.addEventListener

function of Titanium.Facebook

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Facebook.createLoginButton

function of Titanium.Facebook

create and return an instance of Titanium.Facebook.LoginButton

Arguments

Name	Type	Description
params	object	the parameters for the request. the following keys are valid: apikey (the application API key), secret (the application API secret or null if using a session proxy), sessionProxy (the URL to the application session proxy), style (the button style - which is either normal or wide).
parameters	object	(optional) a dictionary object properties defined in Titanium.Facebook.LoginButton

Return Type

object

Titanium.Facebook.execute

function of Titanium.Facebook

execute a FB API execute request

Arguments

Arguments

Name	Type	Description
method	string	method to execute
params	object	JSON serializable object or null (if no parameters) to send with the request
callback	function	the callback function to execute upon receiving a response. the result object will contain a success boolean to indicate the result. if success is false, the error property will give the error message. the data property will contain the result if successfully executed.
data	object	the data payload for the request. Must either null, a string or a Blob object.

Return Type

void

Titanium.Facebook.fireEvent

function of Titanium.Facebook

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Facebook.hasPermission

function of Titanium.Facebook

checks the existing permission and returns true if the user has granted the requested permission

Arguments

Name	Type	Description
permission	string	the permission to check

Return Type

boolean

Titanium.Facebook.isLoggedIn

function of Titanium.Facebook

return true if the user has logged in

Arguments

This function takes no arguments.

Return Type

void

Titanium.Facebook.publishStream

function of Titanium.Facebook

execute a stream request to FB

Arguments

Name	Type	Description
title	string	the title of the stream post
data	object	the data to include in the post. Must be JSON serializable or null.
target	string	the target user id to publish the stream or null if the logged in users account
callback	function	the callback function to execute upon receiving a response. the result object will contain a success boolean to indicate the result. if success is false, the error property will give the error message. the data property will contain the result if successfully executed. the cancel property will be set to true if the user cancelled the dialog.

Return Type

void

Titanium.Facebook.query

function of Titanium.Facebook

execute a FQL query against the FB API

Arguments

Name	Type	Description
fql	string	the FQL query to execute
callback	function	the callback to execute with results once the query is completed. the result object will contain a success boolean to indicate the result. if success is false, the error property will give the error message. the data property will contain the result if successfully executed.

Return Type

void

Titanium.Facebook.removeListener

function of Titanium.Facebook
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Facebook.requestPermission

function of Titanium.Facebook

request a special permission from the user

Arguments

Name	Type	Description
permission	string	name of the permission
callback	function	the callback function to execute upon receiving a response. the result object will contain a success boolean to indicate the result. if success is false, the error property will give the error message. the data property will contain the result if successfully executed. the cancel property will be set to true if the user cancelled the dialog.

Return Type

void

Titanium.Filesystem

submodule of Titanium

The top level Filesystem module. The Filesystem module is used for reading and saving files and directories on the device.

Objects

Name	Description
Titanium.Filesystem.File	The File object which support various filesystem based operations.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events

createFile	create and return an instance of Titanium.Filesystem.File
createTempDirectory	create temp file and return a Titanium.Filesystem.File
createTempFile	create a temp file and return a Titanium.Filesystem.File
fireEvent	fire a synthesized event to the views listener
getFile	return a fully formed file path as a Titanium.Filesystem.File object
isExternalStoragePresent	returns true if the device supports external storage such as an SD card
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
MODE_APPEND	int	constant for append mode for file operations
MODE_READ	int	constant for read mode for file operations
MODE_WRITE	int	constant for write mode for file operations
applicationDataDirectory	string	readonly constant where your application data directory is located. this directory should be used to place applications-specific files. on iPhone, this directory is also backed up.
applicationDirectory	string	readonly constant where your application is located
lineEnding	string	readonly platform specific line ending constant
resourcesDirectory	string	readonly constant where your application resources are located
separator	string	readonly path separator constant
tempDirectory	string	readonly constant where your application can place temporary files

Events

This module has no events

Titanium.Filesystem.File

object of Titanium.Filesystem

The File object which support various filesystem based operations.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
createDirectory	create a directory at the path for the file object
createFile	create a file path at the path for the file object
createTimestamp	return the created timestamp for the file
deleteDirectory	delete the file directory path
deleteFile	delete the file
executable	returns true if the file is executable
exists	returns true if the file or directory exists on the device
extension	return the file extension
fireEvent	fire a synthesized event to the views listener
getDirectoryListing	return an array of paths in the directory of the file object
getParent	return the file parent
hidden	returns true if the file is hidden
modificationTimestamp	return the last modification timestamp for the file
move	move the file to another path
name	the name of the file

nativePath	returns the fully resolved native path
read	return the contents of file as blob
readonly	returns true if the file is readonly
removeEventListerner	remove a previously added event listener
rename	rename the file
setExecutable	make the file executable
setHidden	make the file hidden
setReadonly	make the file readonly
spaceAvailable	return boolean to indicate if the path has space available for storage
symbolicLink	returns true if the file points to a symbolic link
write	write the contents to file
writeable	returns true if the file is writeable

Properties

This object has no properties

Events

This object has no events

Titanium.Filesystem.File.addEventListener

function of Titanium.Filesystem.File

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description

name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Filesystem.File.createDirectory

function of Titanium.Filesystem.File

create a directory at the path for the file object

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Filesystem.File.createFile

function of Titanium.Filesystem.File

create a file path at the path for the file object

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Filesystem.File.createTimestamp

function of Titanium.Filesystem.File

return the created timestamp for the file

Arguments

This function takes no arguments.

Return Type

long

Titanium.Filesystem.File.deleteDirectory

function of Titanium.Filesystem.File

Arguments

Name	Type	Description
recursive	boolean	pass true to recursively delete any contents. defaults to false

Return Type

boolean

Titanium.Filesystem.File.deleteFile

function of Titanium.Filesystem.File

delete the file

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Filesystem.File.executable

function of Titanium.Filesystem.File

returns true if the file is executable

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Filesystem.File.exists

function of Titanium.Filesystem.File

returns true if the file or directory exists on the device

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Filesystem.File.extension

function of Titanium.Filesystem.File

return the file extension

Arguments

This function takes no arguments.

Return Type

string

Titanium.Filesystem.File.fireEvent

function of Titanium.Filesystem.File

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Filesystem.File.getDirectoryListing

function of Titanium.Filesystem.File

return an array of paths in the directory of the file object

Arguments

This function takes no arguments.

Return Type

array

Titanium.Filesystem.File.getParent

function of Titanium.Filesystem.File

return the file parent

Arguments

This function takes no arguments

This function takes no arguments.

Return Type

Titanium.Filesystem.File

Titanium.Filesystem.File.hidden

function of Titanium.Filesystem.File

returns true if the file is hidden

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Filesystem.File.modificationTimestamp

function of Titanium.Filesystem.File

return the last modification timestamp for the file

Arguments

This function takes no arguments.

Return Type

long

Titanium.Filesystem.File.move

function of Titanium.Filesystem.File

move the file to another path

Arguments

Name	Type	Description
newpath	string object	new location

Return Type

boolean

Titanium.Filesystem.File.name

function of Titanium.Filesystem.File

Arguments

This function takes no arguments.

Return Type

string

Titanium.Filesystem.File.nativePath

function of Titanium.Filesystem.File

returns the fully resolved native path

Arguments

This function takes no arguments.

Return Type

string

Titanium.Filesystem.File.read

function of Titanium.Filesystem.File

return the contents of file as blob

Arguments

This function takes no arguments.

Return Type

Titanium.Blob

Titanium.Filesystem.File.readonly

function of Titanium.Filesystem.File

returns true if the file is readonly

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Filesystem.File.removeEventListener

function of Titanium.Filesystem.File
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Filesystem.File.rename

function of Titanium.Filesystem.File

rename the file

Arguments

Name	Type	Description
newname	string	new name

Return Type

boolean

Titanium.Filesystem.File.setExecutable

function of Titanium.Filesystem.File

make the file executable

Arguments

This function takes no arguments.

Return Type

void

Titanium.Filesystem.File.setHidden

function of Titanium.Filesystem.File

make the file hidden

Arguments

This function takes no arguments

This function takes no arguments.

Return Type

void

Titanium.Filesystem.File.setReadonly

function of Titanium.Filesystem.File

make the file readonly

Arguments

This function takes no arguments.

Return Type

void

Titanium.Filesystem.File.spaceAvailable

function of Titanium.Filesystem.File

return boolean to indicate if the path has space available for storage

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Filesystem.File.symbolicLink

function of Titanium.Filesystem.File

returns true if the file points to a symbolic link

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Filesystem.File.write

function of Titanium.Filesystem.File

write the contents to file

Arguments

Name	Type	Description
contents	stringobject	write the contents of string, blob or [[Titanium.Filesystem.File]] to file

Return Type

boolean

Titanium.Filesystem.File.writeable

function of Titanium.Filesystem.File

returns true if the file is writeable

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Filesystem.addEventListener

function of Titanium.Filesystem

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Filesystem.createFile

function of Titanium.Filesystem

create and return an instance of Titanium.Filesystem.File

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Filesystem.File

Return Type

Titanium.Filesystem.createTempDirectory

function of Titanium.Filesystem

create temp file and return a Titanium.Filesystem.File

Arguments

This function takes no arguments.

Return Type

Titanium.Filesystem.File

Titanium.Filesystem.createTempFile

function of Titanium.Filesystem

create a temp file and return a Titanium.Filesystem.File

Arguments

This function takes no arguments.

Return Type

Titanium.Filesystem.File

Titanium.Filesystem.fireEvent

function of Titanium.Filesystem

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Filesystem.getFile

function of Titanium.Filesystem

return a fully formed file path as a Titanium.Filesystem.File object

Arguments

Name	Type	Description
path	string	one or more path arguments to form the full path joined together with the platform specific path separator. if a relative path is passed, the full path will be relative to the application resource folder.

Return Type

Titanium.Filesystem.File

Titanium.Filesystem.isExternalStoragePresent

function of Titanium.Filesystem

returns true if the device supports external storage such as an SD card

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Filesystem.removeEventListener

function of Titanium.Filesystem

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Geolocation

submodule of Titanium

The top level Geolocation module. The Geolocation module is used for accessing device location based information.

Methods

Name	Description
addEventListener	add an event Listener for the instance to receive when triggered events

addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener
setShowCalibration	configure the calibration UI. set the false to disable the calibration display.

Properties

Name	Type	Description
ACCURACY_BEST	int	accuracy constant
ACCURACY_HUNDRED_METERS	int	accuracy constant
ACCURACY_KILOMETER	int	accuracy constant
ACCURACY_NEAREST_TEN_METERS	int	accuracy constant
ACCURACY_THREE_KILOMETERS	int	accuracy constant
PROVIDER_GPS	string	The GPS location provider
PROVIDER_NETWORK	string	The Network location provider
locationServicesEnabled	boolean	returns true if the user has enabled or disable location services for the device (not the application).
preferredProvider	string	allows setting of the preferred location provider. Returns undefined when the preferred provider is auto-detected
purpose	string	(iOS only). This property informs the end-user why location services are being requested by the application. This string will be display in the permission dialog. This property is REQUIRED starting in 4.0.
showCalibration	boolean	returns true if the calibration UI can show

Events

Name	Description
calibration	fired only on iPhone/iPad when the device detects interface and requires calibration. when this event is fired, the calibration UI is displayed in the application.

Event properties

source	the source object that fired the event
type	the name of the event fired

heading

fired when a heading event is received

Event properties

error	if success is false, returns a string of the error description
heading	heading results dictionary with the following sub-properties: <code>magneticHeading</code> , <code>trueHeading</code> , <code>accuracy</code> , <code>x</code> , <code>y</code> , <code>z</code> , <code>timestamp</code> .
source	the source object that fired the event
success	boolean to indicate if the heading event was successfully received or an error occurred
type	the name of the event fired

location

fired when a location event is received

Event properties

coords	location coordinates dictionary with the following sub-properties: <code>latitude</code> , <code>longitude</code> , <code>altitude</code> , <code>accuracy</code> , <code>altitudeAccuracy</code> , <code>heading</code> , <code>speed</code> , <code>timestamp</code> .
error	if success is false, returns a string of the error description
source	the source object that fired the event
success	boolean to indicate if the location event was successfully received or an error occurred
type	the name of the event fired

Titanium.Geolocation.addEventListener

function of Titanium.Geolocation

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Geolocation.fireEvent

function of Titanium.Geolocation

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Geolocation.removeEventListener

function of Titanium.Geolocation

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Geolocation.setShowCalibration

function of Titanium.Geolocation

configure the calibration UI. set the false to disable the calibration display.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Gesture

submodule of Titanium

The top level Gestures module. The Gesture module is responsible for high level device gestures that are device-wide.

Methods

Name	Description
<code>addEventListener</code>	add an event listener for the instance to receive view triggered events
<code>fireEvent</code>	fire a synthesized event to the views listener
<code>removeEventListener</code>	remove a previously added event listener

Properties

This module has no properties

Events

Name	Description						
<code>orientationchange</code>	fired when the device orientation changes Event properties <table border="1"><tr><td>orientation</td><td>the orientation constant</td></tr><tr><td>source</td><td>the source object that fired the event</td></tr><tr><td>type</td><td>the name of the event fired</td></tr></table>	orientation	the orientation constant	source	the source object that fired the event	type	the name of the event fired
orientation	the orientation constant						
source	the source object that fired the event						
type	the name of the event fired						
<code>shake</code>	fired when the device is shaken Event properties <table border="1"><tr><td>source</td><td>the source object that fired the event</td></tr><tr><td>timestamp</td><td>timestamp reference since previous shake</td></tr><tr><td>type</td><td>the name of the event fired</td></tr></table>	source	the source object that fired the event	timestamp	timestamp reference since previous shake	type	the name of the event fired
source	the source object that fired the event						
timestamp	timestamp reference since previous shake						
type	the name of the event fired						

Titanium.Gesture.addEventListener

function of Titanium.Gesture

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Gesture.fireEvent

function of Titanium.Gesture

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Gesture.removeEventListener

function of Titanium.Gesture

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Map

submodule of Titanium

The top level Map module. The Map module is used for creating in-application native maps.

Objects

Name	Description
Titanium.Map.Annotation	An Annotation object that is created by the method Titanium.Map.createAnnotation. This object gives you low level control over annotations that can be added to a Map View.
Titanium.Map.MapView	The MapView is an object created by Titanium.Map.createView and is used for embedding native mapping capabilities as a view in your application. With native maps, you can control the mapping location, the type of map, the zoom level and you can add custom annotations directly to the map.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
createAnnotation	create and return an instance of Titanium.Map.Annotation
createMapView	create and return an instance of Titanium.Map.MapView
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
HYBRID_TYPE	int	Displays a satellite image of the area with road and road name information layered on top.
SATELLITE_TYPE	int	Displays satellite imagery of the area.
STANDARD_TYPE	int	Displays a street map that shows the position of all roads and some road names.

Events

This module has no events

Code Examples

Map Example

This is a basic map example that uses a custom annotation on the map.

```
var mountainView = Titanium.Map.createAnnotation({  
    latitude:37.390749,  
    longitude:-122.081651,  
    title:"Appcelerator Headquarters",  
    subtitle:'Mountain View, CA',  
    pincolor:Titanium.Map.ANNOTATION_RED,
```

```

animate:true,
leftButton: '../images/appcelerator_small.png',
myid:1 // CUSTOM ATTRIBUTE THAT IS PASSED INTO EVENT OBJECTS
});

var mapview = Titanium.Map.createView({
mapType: Titanium.Map.STANDARD_TYPE,
region: {latitude:33.74511, longitude:-84.38993,
latitudeDelta:0.01, longitudeDelta:0.01},
animate:true,
regionFit:true,
userLocation:true,
annotations:[mountainView]
});

win.add(mapview);

```

Titanium.Map.Annotation

object of Titanium.Map

An Annotation object that is created by the method Titanium.Map.createAnnotation. This object gives you low level control over annotations that can be added to a Map View.

Methods

Name	Description
addEventListerner	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
removeEventListerner	remove a previously added event listener

Properties

Name	Type	Description
animate	boolean	boolean to indicate whether the pin should animate when dropped
image	string	image view for the pin instead of default image. currently only supported in iphone
leftButton	int,string	the left button image on the annotation. must either be a button type constant or url
leftView	object	a left view that is displayed on the annotation
pincolor	int	the pin color as one of Titanium.Map.ANNOTATION_RED, Titanium.Map.ANNOTATION_GREEN or Titanium.Map.ANNOTATION_PURPLE.
rightButton	int,string	the right button image on the annotation. must either be a button type constant or url
rightView	object	

rightview	object	a right view that is displayed on the annotation
subtitle	string	the secondary title of the annotation view
title	string	the primary title of the annotation view

Events

This object has no events

Titanium.Map.Annotation.addEventListener

function of Titanium.Map.Annotation

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Map.Annotation.fireEvent

function of Titanium.Map.Annotation

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Map.Annotation.removeEventListener

function of Titanium.Map.Annotation

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Map.MapView

object of Titanium.Map

The MapView is an object created by Titanium.Map.createView and is used for embedding native mapping capabilities as a view in your application. With native maps, you can control the mapping location, the type of map, the zoom level and you can add custom annotations directly to the map.

Methods

Name	Description
add	add a child to the view hierarchy
addAnnotation	add a new annotation to the map
addAnnotations	add one or more new annotation to the map
addEventListener	add an event listener for the instance to receive view triggered events
addRoute	add a route. currently only supported on iphone
animate	animate the view
deselectAnnotation	cause the annotation to be deselected (hidden).
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeAllAnnotations	removes all annotations added to the map
removeAnnotation	remove an existing annotation from the map
removeAnnotations	remove one or more existing annotations from the map

removeEventListener	remove a previously added event listener
removeRoute	remove a previously added route. currently only supported on iphone
selectAnnotation	cause the annotation to be selected (shown).
show	make the view visible
toImage	return a Blob image of the rendered view
zoom	zoom in or out of the map

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animate	boolean	boolean is mapping actions should be animated
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
annotations	array	an array of annotations to add to the map
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)

backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
location	object	a dictionary that specifies the following properties specifying the region location to set the map: <code>latitudeDelta</code> , <code>longitudeDelta</code> , <code>latitude</code> , <code>longitude</code> .
mapType	int	the map type constant of either Titanium.Map.STANDARD_TYPE, Titanium.Map.SATELLITE_TYPE or Titanium.Map.HYBRID_TYPE.
opacity	float	the opacity from 0.0-1.0
region	object	a dictionary that specifies the following properties specifying the region location to set the map: <code>latitudeDelta</code> , <code>longitudeDelta</code> , <code>latitude</code> , <code>longitude</code> .
regionFit	boolean	boolean to indicate if the map should attempt to fit the map view into the region in the visible view
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to

		peers (false)
transform	object	the transformation matrix to apply to the view
userLocation	boolean	boolean to indicate if the map should show the user's current device location as a pin on the map
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto` , a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description																				
click	<p>fired when a map view or annotation is touched</p> <p>Event properties</p> <table border="1"> <tr> <td>annotation</td><td>the announce source object</td></tr> <tr> <td>clicksource</td><td>the source of either annotation, leftButton or rightButton to indicate to location of the click</td></tr> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>index</td><td>the annotation index</td></tr> <tr> <td>map</td><td>the map view instance</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>title</td><td>the annotation title</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	annotation	the announce source object	clicksource	the source of either annotation, leftButton or rightButton to indicate to location of the click	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	index	the annotation index	map	the map view instance	source	the source object that fired the event	title	the annotation title	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
annotation	the announce source object																				
clicksource	the source of either annotation, leftButton or rightButton to indicate to location of the click																				
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates																				
index	the annotation index																				
map	the map view instance																				
source	the source object that fired the event																				
title	the annotation title																				
type	the name of the event fired																				
x	the x point of the event in receiving view coordinates																				
y	the y point of the event, in receiving view coordinates																				
complete	<p>fired when the map completes loading</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired																
source	the source object that fired the event																				
type	the name of the event fired																				

dblclick

fired when the device detects a double click against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap

fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

error

fired when the map receives a mapping error

Event properties

message	the error message
source	the source object that fired the event
type	the name of the event fired

loading

fired when the map begins loading

Event properties

source	the source object that fired the event
type	the name of the event fired

regionChanged

fired when the mapping region changes

Event properties

latitude	the new latitude
-----------------	------------------

latitudeDelta	the new latitude delta
longitude	the new longitude
longitudeDelta	the new longitude delta
source	the source object that fired the event
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart

fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Notes

For Android, you will need to obtain a map key from Google to use maps in your application. On iPhone, you cannot call any methods on a map view until it has been added to a view.

Titanium.Map.MapView.add

function of Titanium.MapMapView
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.Map.MapView.addAnnotation

function of Titanium.MapMapView

add a new annotation to the map

Arguments

Name	Type	Description
annotation	object	either a dictionary of properties for the annotation or a Titanium.Map.Annotation instance.

Return Type

void

Titanium.Map.MapView.addAnnotations

function of Titanium.MapMapView

add one or more new annotation to the map

Arguments

Name	Type	Description
annotations	array	an array of either a dictionary of properties for the annotation or a

Return Type

void

Titanium.Map.MapView.addEventListener

function of Titanium.Map.MapView

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Map.MapView.addRoute

function of Titanium.Map.MapView

add a route. currently only supported on iphone

Arguments

Name	Type	Description
route	object	dictionary with the properties: name route name, points dictionary of values with longitude and latitude keys, color for the line color and width for the line width.

Return Type

void

Titanium.Map.MapView.animate

function of Titanium.Map.MapView

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.Map.MapView.deselectAnnotation

function of Titanium.Map.MapView

cause the annotation to be deselected (hidden).

Arguments

Name	Type	Description
annotation	string,object	either a string of the annotation title or a Titanium.Map.Annotation reference.

Return Type

void

Titanium.Map.MapView.fireEvent

function of Titanium.Map.MapView

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Map.MapView.hide

function of Titanium.Map.MapView

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.Map.MapView.remove

function of Titanium.MapMapView

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.Map.MapView.removeAllAnnotations

function of Titanium.MapMapView

removes all annotations added to the map

Arguments

This function takes no arguments.

Return Type

void

Titanium.Map.MapView.removeAnnotation

function of Titanium.MapMapView

remove an existing annotation from the map

Arguments

Name	Type	Description
annotation	string,object	either a string of the annotation title or a Titanium.Map.Annotation reference.

Return Type

void

Titanium.Map.MapView.removeAnnotations

function of Titanium.MapMapView

remove one or more existing annotations from the map

Arguments

Name	Type	Description
annotation	array	an array of either a string of the annotation title or a Titanium.Map.Annotation reference.

Return Type

void

Titanium.Map.MapView.removeEventListener

function of Titanium.Map.MapView

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Map.MapView.removeRoute

function of Titanium.Map.MapView

remove a previously added route. currently only supported on iphone

Arguments

Name	Type	Description
route	object	same route object used during addRoute

Return Type

void

Titanium.Map.MapView.selectAnnotation

function of Titanium.Map.MapView

cause the annotation to be selected (shown).

Arguments

Name	Type	Description
annotation	string object	either a string of the annotation title or a Titanium.Map.Annotation reference

annotation string, object either a string or the annotation title or a Titanium.Map.Annotation reference.

Return Type

void

Titanium.Map.MapView.show

function of Titanium.Map.MapView

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.Map.MapView.toImage

function of Titanium.Map.MapView

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.Map.MapView.zoom

function of Titanium.Map.MapView

zoom in or out of the map

Arguments

Name	Type	Description
level	double	zoom level (can be positive or negative)

Return Type

void

Titanium.Map.addEventListener

Titanium.Map.addEventListener

function of Titanium.Map

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Map.createAnnotation

function of Titanium.Map

create and return an instance of Titanium.Map.Annotation

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Map.Annotation

Return Type

object

Titanium.Map.createMapView

function of Titanium.Map

create and return an instance of Titanium.Map.MapView

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Map.MapView

Return Type

object

Titanium.Map.fireEvent

function of Titanium.Map

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Map.removeEventListener

function of Titanium.Map

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Media

submodule of Titanium

The top level Media module. The Media module is used accessing the device's media related functionality such as playing audio or recording video.

Objects

Name	Description
Titanium.Media.AudioPlayer	The AudioPlayer object is returned by Titanium.Media.createAudioPlayer and is used for streaming audio to the device and low-level control of the audio playback.
Titanium.Media.AudioRecorder	The AudioRecorder object is returned by Titanium.Media.createAudioRecorder and is used for recording audio from the device microphone.
Titanium.Media.Item	A representation of a media item returned by the music picker as part of the <code>items</code> array in the dictionary passed to its <code>success</code> function.
Titanium.Media.MusicPlayer	The MusicPlayer instance returned from Titanium.Media.createMusicPlayer. This object represents a music controller.

Titanium.Media.Sound	The Sound object is returned by Titanium.Media.createSound and is useful for playing basic sounds. The Sound object loads the entire media resource in memory before playing. If you need to support streaming, use the Titanium.Media.createAudioPlayer API.
Titanium.Media.VideoPlayer	The VideoPlayer object is returned by Titanium.Media.createVideoPlayer and is useful for playing videos.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
beep	play a device beep notification
createAudioPlayer	create and return an instance of Titanium.Media.AudioPlayer
createAudioRecorder	create and return an instance of Titanium.Media.AudioRecorder
createItem	create and return an instance of Titanium.Media.Item
createMusicPlayer	create and return an instance of Titanium.Media.MusicPlayer
createSound	create and return an instance of Titanium.Media.Sound
createVideoPlayer	create and return an instance of Titanium.Media.VideoPlayer
fireEvent	fire a synthesized event to the views listener
hideCamera	hide the device camera UI. this must be called after calling <code>showCamera</code> and only when <code>autohide</code> is set to false. this method will cause the media capture device be hidden.
hideMusicLibrary	hide the music library. This must be called after calling <code>showMusicLibrary</code> and only when <code>autohide</code> is set to false. iPhone and iPad only.
isMediaTypeSupported	return boolean to indicate if the media type is supported
openPhotoGallery	open the photo gallery picker
removeEventListener	remove a previously added event listener
saveToPhotoGallery	save media to photo gallery / camera roll
showCamera	show the camera
showMusicLibrary	show the music library. iPhone and iPad only.

startMicrophoneMonitor	start the monitoring of microphone sound level
stopMicrophoneMonitor	stop the monitoring of microphone sound level
takePicture	use the device camera to capture a photo. this must be called after calling <code>showCamera</code> and only when <code>autohide</code> is set to false. this method will cause the media capture device to capture a photo and call the <code>success</code> callback.
takeScreenshot	take a screen shot of the visible UI on the device
vibrate	play a device vibration

Properties

Name	Type	Description
AUDIO_FILEFORMAT_3GP2	int	audio file format 3GPP-2
AUDIO_FILEFORMAT_3GPP	int	audio file format 3GPP
AUDIO_FILEFORMAT_AIFF	int	audio file format AIFF
AUDIO_FILEFORMAT_AMR	int	audio file format AMR
AUDIO_FILEFORMAT_CAF	int	audio file format Apple Compressed Audio Format (CAF)
AUDIO_FILEFORMAT_MP3	int	audio file format MP3
AUDIO_FILEFORMAT_MP4	int	audio file format MP4
AUDIO_FILEFORMAT_MP4A	int	audio file format MP4A
AUDIO_FILEFORMAT_WAVE	int	audio file format WAVE
AUDIO_FORMAT_AAC	int	audio format MPEG4 AAC encoding
AUDIO_FORMAT_ALAW	int	audio format 8-bit aLaw encoding

AUDIO_FORMAT_APPLE_LOSSLESS	int	audio format apple lossless encoding
AUDIO_FORMAT_ILBC	int	audio format iLBC encoding
AUDIO_FORMAT_IMA4	int	audio format Apple IMA4 encoding
AUDIO_FORMAT_LINEAR_PCM	int	audio format Linear 16-bit, PCM encoding
AUDIO_FORMAT_ULAW	int	audio format 8-bit muLaw encoding
AUDIO_HEADPHONES	int	constant for line type headphones
AUDIO_HEADPHONES_AND_MIC	int	constant line type headphones and microphone
AUDIO_HEADSET_INOUT	int	constant line type headset in/out
AUDIO_LINEOUT	int	constant line type line out
AUDIO_MICROPHONE	int	constant line type microphone
AUDIO_MUTED	int	constant line type muted switch is on
AUDIO_RECEIVER_AND_MIC	int	constant line type receiver and microphone
AUDIO_SESSION_MODE_AMBIENT	int	For long-duration sounds such as rain, car engine noise, and so on. It is also for 'play along' style applications, such a virtual piano that a user plays over iPod audio.
AUDIO_SESSION_MODE_PLAYBACK	int	For playing recorded music or other sounds that are central to the successful use of your application. When using this mode, your application audio continues with the Ring/Silent switch set to silent or when the screen locks. This property normally disallows iPod audio to mix with application audio
AUDIO_SESSION_MODE_PLAY_AND_RECORD	int	Allows recording (input) and playback (output) of audio, such as for a VOIP (voice over IP) application. This category is appropriate for simultaneous recording and playback, and also for applications that record and play back but not simultaneously. If you want to ensure that sounds such as Messages alerts do not play while your application is recording, use Titanium Media AUDIO_SESSION_MODE_RECORD instead. This

		Titanium.Media.AUDIO_SESSION_MODE_RECORD instead. This category normally disallows iPod audio to mix with application audio.
AUDIO_SESSION_MODE_RECORD	int	For recording audio; it silences playback audio.
AUDIO_SESSION_MODE_SOLO_AMBIENT	int	The solo ambient sound category is for long-duration sounds such as rain, car engine noise, and so on. When you use this category, audio from built-in applications, such as the iPod, is silenced. Your audio is silenced when the Ring/Silent switch is set to silent or when the screen locks.
AUDIO_SPEAKER	int	constant line type speaker
AUDIO_UNAVAILABLE	int	constant line type unavailable
AUDIO_UNKNOWN	int	constant line type unknown or not determined
DEVICE_BUSY	int	constant for media device busy error
MEDIA_TYPE_PHOTO	int	media type constant to signify photo
MEDIA_TYPE_VIDEO	int	media type constant to signify video
MUSIC_MEDIA_TYPE_ALL	int	media containing any content
MUSIC_MEDIA_TYPE_ANY_AUDIO	int	media containing any audio content
MUSIC_MEDIA_TYPE_AUDIOBOOK	int	media containing audiobook content
MUSIC_MEDIA_TYPE_MUSIC	int	media containing music content
MUSIC_MEDIA_TYPE_PODCAST	int	media containing podcast content
MUSIC_PLAYER_REPEAT_ALL	int	constant for repeating all setting
MUSIC_PLAYER_REPEAT_DEFAULT	int	constant for user default repeat setting
MUSIC_PLAYER_REPEAT_NONE	int	constant for no repeat setting

		constant for no repeat setting
MUSIC_PLAYER_REPEAT_ONE	int	constant for repeating one item setting
MUSIC_PLAYER_SHUFFLE_ALBUMS	int	constant for shuffling complete albums setting
MUSIC_PLAYER_SHUFFLE_DEFAULT	int	constant for user default shuffle setting
MUSIC_PLAYER_SHUFFLE_NONE	int	constant for no shuffle setting
MUSIC_PLAYER_SHUFFLE_SONGS	int	constant for shuffling songs setting
MUSIC_PLAYER_STATE_INTERRUPTED	int	constant for interrupted state
MUSIC_PLAYER_STATE_PAUSED	int	constant for paused state
MUSIC_PLAYER_STATE_PLAYING	int	constant for playing state
MUSIC_PLAYER_STATE_SEEK_BACKWARD	int	constant for backward seek state
MUSIC_PLAYER_STATE_SEEK_FORWARD	int	constant for forward seek state
MUSIC_PLAYER_STATE_STOPPED	int	constant for stopped state
NO_CAMERA	int	constant for media no camera error
NO_VIDEO	int	constant for media no video error
QUALITY_HIGH	int	media type constant to use high-quality video recording. Recorded files are suitable for on-device playback and for wired transfer to the Desktop using Image Capture; they are likely to be too large for transfer using Wi-Fi.
QUALITY_LOW	int	media type constant to use low-quality video recording. Recorded files can usually be transferred over the cellular network.
QUALITY_MEDIUM	int	media type constant to use medium-quality video recording. Recorded files can usually be transferred using Wi-Fi. This is the default video quality setting.

			quality setting.
UNKNOWN_ERROR	int	constant for unknown media error	
VIDEO_CONTROL_DEFAULT	int	constant for video controls default	
VIDEO_CONTROL_EMBEDDED	int	Controls for an embedded view. Used in conjunction with movieControlStyle property of Titanium.Media.VideoPlayer in iPhone 3.2+	
VIDEO_CONTROL_FULLSCREEN	int	Controls for fullscreen. Used in conjunction with movieControlStyle property of Titanium.Media.VideoPlayer in iPhone 3.2+	
VIDEO_CONTROL_HIDDEN	int	constant for video controls hidden	
VIDEO_CONTROL_NONE	int	No controls. Used in conjunction with movieControlStyle property of Titanium.Media.VideoPlayer in iPhone 3.2+	
VIDEO_CONTROL_VOLUME_ONLY	int	constant for video controls volume only	
VIDEO_FINISH_REASON_PLAYBACK_ENDED	int	the video playback ended normally	
VIDEO_FINISH_REASON_PLAYBACK_ERROR	int	the video playback ended abnormally	
VIDEO_FINISH_REASON_USER_EXITED	int	the video playback ended by user action (such as clicking the Done button)	
VIDEO_LOAD_STATE_PLAYABLE	int	the current media is playable	
VIDEO_LOAD_STATE_PLAYTHROUGH_OK	int	playback will be automatically started in this state when <code>autoplay</code> is true	
VIDEO_LOAD_STATE_STALLED	int	playback will be automatically paused in this state, if started	
VIDEO_LOAD_STATE_UNKNOWN	int	the current load state is not known	
VIDEO_MEDIA_TYPE_AUDIO	int	A audio type of media in the movie returned by Titanium.Media.VideoPlayer <code>mediaTypes</code> property	
VIDEO_MEDIA_TYPE_NONE	int	An unknown type of media in the movie returned by	

		An unknown type of media in the movie returned by Titanium.Media.VideoPlayer <code>mediaTypes</code> property.
VIDEO_MEDIA_TYPE_VIDEO	int	A video type of media in the movie returned by Titanium.Media.VideoPlayer <code>mediaTypes</code> property
VIDEO_PLAYBACK_STATE_INTERRUPTED	int	playback has been interrupted
VIDEO_PLAYBACK_STATE_PAUSED	int	playback is paused
VIDEO_PLAYBACK_STATE_PLAYING	int	playback is playing
VIDEO_PLAYBACK_STATE_SEEKING_BACKWARD	int	playback is rewinding
VIDEO_PLAYBACK_STATE_SEEKING_FORWARD	int	playback is seeking forward
VIDEO_PLAYBACK_STATE_STOPPED	int	playback has stopped
VIDEO_REPEAT_MODE_NONE	int	the video will not repeat
VIDEO_REPEAT_MODE_ONE	int	the video will repeat once
VIDEO_SCALING_ASPECT_FILL	int	constant for video aspect where the movie will be scaled until the movie fills the entire screen. Content at the edges of the larger of the two dimensions is clipped so that the other dimension fits the screen exactly. The aspect ratio of the movie is preserved.
VIDEO_SCALING_ASPECT_FIT	int	constant for video aspect fit where the movie will be scaled until one dimension fits on the screen exactly. In the other dimension, the region between the edge of the movie and the edge of the screen is filled with a black bar. The aspect ratio of the movie is preserved.
VIDEO_SCALING_MODE_FILL	int	constant for video aspect where the movie will be scaled until both dimensions fit the screen exactly. The aspect ratio of the movie is not preserved.
VIDEO_SCALING_NONE	int	constant for video scaling where the scaling is turn off. The movie will not be scaled.
VIDEO_SOURCE_TYPE_FILE	int	the video source type is a file. Related to the <code>sourceType</code> property of Titanium.Media.VideoPlayer

VIDEO_SOURCE_TYPE_STREAMING	int	the video source type is a remote stream. Related to the <code>sourceType</code> property of <code>Titanium.Media.VideoPlayer</code>
VIDEO_SOURCE_TYPE_UNKNOWN	int	the video source type is unknown. Related to the <code>sourceType</code> property of <code>Titanium.Media.VideoPlayer</code>
VIDEO_TIME_OPTION_EXACT	int	use the exact time
VIDEO_TIME_OPTION_NEAREST_KEYFRAME	int	use the closest keyframe in the time
appMusicPlayer	object	an instance of <code>Titanium.Media.MusicPlayer</code> representing the app-specific music player. iPhone/iPad only
audioLineType	int	returns the line type constant for the current line type
audioPlaying	boolean	returns true if the device is playing audio
availableCameraMediaTypes	array	return an array of media type constants supported for the camera
availablePhotoGalleryMediaTypes	array	return an array of media type constants supported for saving to the photo gallery
availablePhotoMediaTypes	array	return an array of media type constants supported for the photo
averageMicrophonePower	float	return the current average microphone level in dB or -1 if microphone monitoring is disabled
canRecord	boolean	returns true if the device has recording input device available
defaultAudioSessionMode	int	a constant for the default audio session mode to be used. One of <code>Titanium.Media.AUDIO_SESSION_MODE_AMBIENT</code> , <code>Titanium.Media.AUDIO_SESSION_MODE_SOLO_AMBIENT</code> <code>Titanium.Media.AUDIO_SESSION_MODE_PLAYBACK</code> <code>Titanium.Media.AUDIO_SESSION_MODE_RECORD</code> , <code>Titanium.Media.AUDIO_SESSION_MODE_PLAY_AND_RECORD</code> .
isCameraSupported	boolean	returns true if the device has camera support
peakMicrophonePower	float	return the current microphone level peak power in dB or -1 if

systemMusicPlayer	object	an instance of Titanium.Media.MusicPlayer representing the system-wide music player. iPhone/iPad only
volume	float	the current volume of the playback device

Events

Name	Description						
linechange	fired when a audio line type change is detected Event properties <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired		
source	the source object that fired the event						
type	the name of the event fired						
recordinginput	fired when availability of recording input changes Event properties <table border="1"> <tr> <td>available</td><td>bool indicating availability of recording device</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	available	bool indicating availability of recording device	source	the source object that fired the event	type	the name of the event fired
available	bool indicating availability of recording device						
source	the source object that fired the event						
type	the name of the event fired						
volume	fired when the volume output changes Event properties <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>volume</td><td>float of the new volume in dB</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired	volume	float of the new volume in dB
source	the source object that fired the event						
type	the name of the event fired						
volume	float of the new volume in dB						

Titanium.Media.AudioPlayer

object of Titanium.Media

The AudioPlayer object is returned by Titanium.Media.createAudioPlayer and is used for streaming audio to the device and low-level control of the audio playback.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
pause	pause playback
removeEventListener	remove a previously added event listener
setPaused	control the playback of the audio
setUrl	change the url of the audio playback
start	start playback
stateDescription	convert a state into a textual description suitable for display
stop	stop playback

Properties

Name	Type	Description
STATE_BUFFERING	int	current playback is in the buffering from the network state
STATE_INITIALIZED	int	current playback is in the initialization state
STATE_PAUSED	int	current playback is in the paused state
STATE_PLAYING	int	current playback is in the playing state
STATE_STARTING	int	current playback is in the starting playback state
STATE_STOPPED	int	current playback is in the stopped state
STATE_STOPPING	int	current playback is in the stopping state
STATE_WAITING_FOR_DATA	int	current playback is in the waiting for audio data from the network state
STATE_WAITING_FOR_QUEUE	int	current playback is in the waiting for queue state

STATE_WAITING_FOR_QUEUE	int	current playback is in the waiting for audio data to fill the queue state
allowBackground	boolean	boolean to indicate if audio should continue playing even if Activity is paused (Android only as of 1.3.0)
audioSessionMode	int	the default audio session mode to be used for this player. One of Titanium.Media.AUDIO_SESSION_MODE_AMBIENT, Titanium.Media.AUDIO_SESSION_MODE_SOLO_AMBIENT Titanium.Media.AUDIO_SESSION_MODE_PLAYBACK Titanium.Media.AUDIO_SESSION_MODE_RECORD, Titanium.Media.AUDIO_SESSION_MODE_PLAY_AND_RECORD.
bitRate	double	bit rate of the current playback stream
idle	boolean	returns boolean indicating if the playback is idle
paused	boolean	returns boolean indicating if the playback is paused
playing	boolean	returns boolean indicating if the playback is streaming audio
progress	double	returns the current playback progress. Will return zero if sampleRate has not yet been detected
state	int	returns int for the current state of playback
url	string	returns the url for the current playback
waiting	boolean	returns boolean indicating if the playback is waiting for audio data from the network

Events

Name	Description						
change	fired when the state of the playback changes Event properties <table border="1"> <tr> <td>description</td><td>textual description of the state of playback</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>state</td><td>current state of playback</td></tr> </table>	description	textual description of the state of playback	source	the source object that fired the event	state	current state of playback
description	textual description of the state of playback						
source	the source object that fired the event						
state	current state of playback						

	type	the name of the event fired
progress	fired once per second with the current progress during playback	
Event properties		
	progress	current progress value
	source	the source object that fired the event
	type	the name of the event fired

Titanium.Media.AudioPlayer.addEventListener

function of Titanium.Media.AudioPlayer

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Media.AudioPlayer.fireEvent

function of Titanium.Media.AudioPlayer

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Media.AudioPlayer.pause

pause playback

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.AudioPlayer.removeEventListener

function of Titanium.Media.AudioPlayer

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Media.AudioPlayer.setPaused

function of Titanium.Media.AudioPlayer

control the playback of the audio

Arguments

Name	Type	Description
paused	boolean	pass true to pause the current playback temporarily, false to unpause it

Return Type

void

Titanium.Media.AudioPlayer.setUrl

function of Titanium.Media.AudioPlayer

change the url of the audio playback

Arguments

Name	Type	Description
url	string	the new url

Return Type

void

Titanium.Media.AudioPlayer.start

function of Titanium.Media.AudioPlayer

start playback

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.AudioPlayer.stateDescription

function of Titanium.Media.AudioPlayer

convert a state into a textual description suitable for display

Arguments

This function takes no arguments.

Return Type

string

Titanium.Media.AudioPlayer.stop

function of Titanium.Media.AudioPlayer

stop playback

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.AudioRecorder

object of Titanium.Media

The AudioRecorder object is returned by Titanium.Media.createAudioRecorder and is used for recording audio from the device microphone.

Methods

Name	Description
addEventListenerr	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
pause	called to temporarily pause recording
removeEventListenerr	remove a previously added event listener
resume	called to resume audio recording
start	called to start recording audio
stop	called to stop recording audio

Properties

Name	Type	Description
audioSessionMode	int	the default audio session mode to be used for this recorder. One of Titanium.Media.AUDIO_SESSION_MODE_AMBIENT, Titanium.Media.AUDIO_SESSION_MODE_SOLO_AMBIENT Titanium.Media.AUDIO_SESSION_MODE_PLAYBACK Titanium.Media.AUDIO_SESSION_MODE_RECORD, Titanium.Media.AUDIO_SESSION_MODE_PLAY_AND_RECORD.
compression	int	audio compression constant to be used for the recording
format	int	audio format constant for used for the recording
paused	boolean	readonly property to indicate if paused
recording	boolean	readonly property to indicate if recording
stopped	boolean	readonly property to indicate if stopped

Events

This object has no events

Titanium.Media.AudioRecorder.addEventListener

function of Titanium.Media.AudioRecorder

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Media.AudioRecorder.fireEvent

function of Titanium.Media.AudioRecorder

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Media.AudioRecorder.pause

function of Titanium.Media.AudioRecorder

called to temporarily pause recording

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.AudioRecorder.removeEventListener

function of Titanium.Media.AudioRecorder

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Media.AudioRecorder.resume

function of Titanium.Media.AudioRecorder

called to resume audio recording

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.AudioRecorder.start

function of Titanium.Media.AudioRecorder

called to start recording audio

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.AudioRecorder.stop

function of Titanium.Media.AudioRecorder

called to stop recording audio

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.Item

object of Titanium.Media

A representation of a media item returned by the music picker as part of the `items` array in the dictionary passed to its `success` function.

Methods

This object has no methods

Properties

Name	Type	Description
<code>albumArtist</code>	string	the artist for the album of the item
<code>albumTitle</code>	string	the album title of the item
<code>albumTrackCount</code>	int	the number of tracks for the album of the item
<code>albumTrackNumber</code>	int	the track number of the item
<code>artist</code>	string	the artist of the item
<code>artwork</code>	object	a blob object containing the image for the item's artwork, or null if none
<code>composer</code>	string	the composer of the item
<code>discCount</code>	int	the total number of discs of the item
<code>discNumber</code>	int	the disc number of the item
<code>genre</code>	string	the genre of the item
<code>isCompilation</code>	boolean	true if the item is part of a compilation album
<code>lyrics</code>	string	the lyrics of the item
<code>mediaType</code>	int	the type of the item
<code>playCount</code>	int	the number of times the item has been played

playbackDuration	double	the length (in seconds) of the item
podcastTitle	string	the title of a podcast item. Only for media types of Titanium.Media.MUSIC_MEDIA_TYPE_PODCAST.
rating	int	the rating of the item
skipCount	int	the number of times the item has been skipped
title	string	the title of the item

Events

This object has no events

Titanium.Media.MusicPlayer

object of Titanium.Media

The MusicPlayer instance returned from Titanium.Media.createMusicPlayer. This object represents a music controller.

Methods

Name	Description
pause	pause playback
play	begin playback
seekBackward	seek backward in the currently playing media
seekForward	seek forward in the currently playing media
setQueue	set the media queue
skipToBeginning	skip to the beginning of the currently playing media
skipToNext	skip to the next media in the queue
skipToPrevious	

skip to the previous media in the queue

stop

stop playback

stopSeeking

end a seek operation

Properties

Name	Type	Description
currentPlaybackTime	double	the current point in song playback
nowPlaying	object	an Item object which indicates the currently playing media
playbackState	int	the playback state; one of Titanium.Media.MUSIC_PLAYER_STATE_STOPPED, Titanium.Media.MUSIC_PLAYER_STATE_PLAYING, Titanium.Media.MUSIC_PLAYER_STATE_PAUSED, Titanium.Media.MUSIC_PLAYER_STATE_INTERRUPTED, Titanium.Media.MUSIC_PLAYER_STATE_SEEK_FORWARD, Titanium.Media.MUSIC_PLAYER_STATE_SEEK_BACKWARD
repeatMode	int	the repeat setting; one of Titanium.Media.MUSIC_PLAYER_REPEAT_DEFAULT, Titanium.Media.MUSIC_PLAYER_REPEAT_NONE, Titanium.Media.MUSIC_PLAYER_REPEAT_ONE, Titanium.Media.MUSIC_PLAYER_REPEAT_ALL
shuffleMode	int	the shuffle setting; one of Titanium.Media.MUSIC_PLAYER_SHUFFLE_DEFAULT, Titanium.Media.MUSIC_PLAYER_SHUFFLE_NONE, Titanium.Media.MUSIC_PLAYER_SHUFFLE_SONGS, Titanium.Media.MUSIC_PLAYER_SHUFFLE_ALBUMS
volume	float	a value between 0.0 and 1.0 indicating the volume level for the music player

Events

Name	Description				
playingChange	the currently playing media changed Event properties <table border="1"><tr><td>source</td><td>the source object that fired the event</td></tr><tr><td>type</td><td>the name of the event fired</td></tr></table>	source	the source object that fired the event	type	the name of the event fired
source	the source object that fired the event				
type	the name of the event fired				
stateChange					

the playback state changed

Event properties

source	the source object that fired the event
type	the name of the event fired

volumeChange

the volume changed

Event properties

source	the source object that fired the event
type	the name of the event fired

Titanium.Media.MusicPlayer.pause

function of Titanium.Media.MusicPlayer

pause playback

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.MusicPlayer.play

function of Titanium.Media.MusicPlayer

begin playback

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.MusicPlayer.seekBackward

function of Titanium.Media.MusicPlayer

seek backward in the currently playing media

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.MusicPlayer.seekForward

function of Titanium.Media.MusicPlayer

seek forward in the currently playing media

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.MusicPlayer.setQueue

function of Titanium.Media.MusicPlayer

set the media queue

Arguments

Name	Type	Description
queue	object	a queue representation to set the player queue to. Can be any of: A dictionary with an <code>items</code> key that is an array of Titanium.Media.Item objects, an array of Titanium.Media.Item objects, or a single Titanium.Media.Item object.

Return Type

void

Titanium.Media.MusicPlayer.skipToBeginning

function of Titanium.Media.MusicPlayer

skip to the beginning of the currently playing media

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.MusicPlayer.skipToNext

function of Titanium.Media.MusicPlayer

skip to the next media in the queue

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.MusicPlayer.skipToPrevious

function of Titanium.Media.MusicPlayer

skip to the previous media in the queue

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.MusicPlayer.stop

function of Titanium.Media.MusicPlayer

stop playback

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.MusicPlayer.stopSeeking

function of Titanium.Media.MusicPlayer

end a seek operation

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.Sound

object of Titanium.Media

The Sound object is returned by Titanium.Media.createSound and is useful for playing basic sounds. The Sound object loads the entire media resource in memory before playing. If you need to support streaming, use the Titanium.Media.createAudioPlayer API.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
getTime	return the current time position of the audio
getVolume	return the value of the audio
isLooping	returns true if the audio will loop
isPaused	returns true if the audio is paused
isPlaying	returns true if the audio is playing
pause	temporarily pause the audio. to resume, invoke <code>play</code> .
play	starting playing the source. if paused, will resume.
release	release all internal resources. this is typically unnecessary but can be useful if you load a large audio file in <code>app.js</code> and play it only once and you would like to release all releases after your final play to reduce memory.
removeEventListener	remove a previously added event listener
reset	reset the audio to the beginning.
setLooping	control whether the audio should loop
setPaused	control whether the audio is paused
setTime	set the time position of the audio
setVolume	set the volume of the audio

stop	stop playing the audio and reset it to the beginning.
-------------	---

Properties

Name	Type	Description
allowBackground	boolean	boolean to indicate if audio should continue playing even if Activity is paused (Android only as of 1.3.0)
audioSessionMode	int	the default audio session mode to be used for this player. One of Titanium.Media.AUDIO_SESSION_MODE_AMBIENT, Titanium.Media.AUDIO_SESSION_MODE_SOLO_AMBIENT Titanium.Media.AUDIO_SESSION_MODE_PLAYBACK Titanium.Media.AUDIO_SESSION_MODE_RECORD, Titanium.Media.AUDIO_SESSION_MODE_PLAY_AND_RECORD.
duration	float	the duration of the audio.
looping	boolean	boolean to indicate if the audio should loop upon completion
paused	boolean	boolean to indicate if the audio is paused
playing	boolean	boolean to indicate if the audio is playing
time	float	the time position of the audio.
url	string	url to the audio
volume	float	the volume of the audio. this volume only affects the media, not the device audio.

Events

Name	Description						
complete	fired when the audio has completed Event properties <table border="1"> <tr> <td>source</td> <td>the source object that fired the event</td> </tr> <tr> <td>success</td> <td>boolean to indicate if the sound was successfully played</td> </tr> <tr> <td>type</td> <td>the name of the event fired</td> </tr> </table>	source	the source object that fired the event	success	boolean to indicate if the sound was successfully played	type	the name of the event fired
source	the source object that fired the event						
success	boolean to indicate if the sound was successfully played						
type	the name of the event fired						

error

called when an error is received playing the audio.

Event properties

message	error message
source	the source object that fired the event
type	the name of the event fired

interrupted

called when the audio is interrupted by the device. this is typically called during an interruption due to an incoming phone call.

Event properties

source	the source object that fired the event
type	the name of the event fired

resume

called when the audio is resumed after an interruption.

Event properties

interruption	boolean to indicate if the resume was from an interruption
source	the source object that fired the event
type	the name of the event fired

Code Examples**Simple Example**

Simple example of playing a WAVE file from the Resources directory.

```
var player = Ti.UI.createSound({url:"sound.wav"});  
player.play();
```

Notes

You can pass the property `preload` in the constructor as an object property to cause the audio to be preloaded before `play` is invoked. For larger audio files, this is recommended if you want the audio to begin immediately when invoking `play`.

Titanium.Media.Sound.addEventListener

function of Titanium.Media.Sound

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Media.Sound.fireEvent

function of Titanium.Media.Sound

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Media.Sound.getTime

function of Titanium.Media.Sound

return the current time position of the audio

Arguments

This function takes no arguments.

Return Type

double

Titanium.Media.Sound.getVolume

function of Titanium.Media.Sound

return the value of the audio

Arguments

This function takes no arguments.

Return Type

float

Titanium.Media.Sound.isLooping

function of Titanium.Media.Sound

returns true if the audio will loop

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Media.Sound.isPaused

function of Titanium.Media.Sound

returns true if the audio is paused

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Media.Sound.isPlaying

function of Titanium.Media.Sound

returns true if the audio is playing

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.Media.Sound.pause

function of Titanium.Media.Sound

temporarily pause the audio. to resume, invoke play.

Arguments

This function takes no arguments.

Return Type

Titanium.Media.Sound.play

function of Titanium.Media.Sound

starting playing the source. if paused, will resume.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.Sound.release

function of Titanium.Media.Sound

release all internal resources. this is typically unnecessary but can be useful if you load a large audio file in `app.js` and play it only once and you would like to release all releases after your final play to reduce memory.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.Sound.removeEventListener

function of Titanium.Media.Sound

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Media.Sound.reset

function of Titanium.Media.Sound

reset the audio to the beginning.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.Sound.setLooping

function of Titanium.Media.Sound

control whether the audio should loop

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.Sound.setPaused

function of Titanium.Media.Sound

control whether the audio is paused

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.Sound.setTime

function of Titanium.Media.Sound

set the time position of the audio

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.Sound.setVolume

function of Titanium.Media.Sound

set the volume of the audio

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.Sound.stop

function of Titanium.Media.Sound

stop playing the audio and reset it to the beginning.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.VideoPlayer

object of Titanium.Media

The VideoPlayer object is returned by Titanium.Media.createVideoPlayer and is useful for playing videos.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
cancelAllThumbnailImageRequests	Cancels all pending asynchronous thumbnail requests.
fireEvent	fire a synthesized event to the views listener
hide	hide the view
pause	pause playing the video. On iOS only available under 3.2 and later
play	start playing the video
release	release the internal video resources immediately. this is not usually necessary but can help if you no longer need to use the player after it is used to help conserve memory

no longer need to use the player after it is used to help converse memory.

remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
requestThumbnailImagesAtTimes	Asynchronously request thumbnails for one or more times, provided as an array of numbers (double). Fires a <code>thumbnail</code> event on completion. Optionally invokes the callback function passed in the method.
setBackgroundView	A view for customization which is always displayed behind movie content.
setMedia	the a non-url based media to play, either a File or Blob.
setUrl	the url to play
show	make the view visible
stop	stop playing the video
thumbnailImageAtTime	Returns a thumbnail at the given time as a blob.
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
autoplay	boolean	indicates if a movie should automatically start playback when it is likely to finish uninterrupted based on e.g. network conditions. Defaults to true.
backgroundColor	string	background color for the view. deprecated, set the background color on the view.
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.

backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
contentURL	string	deprecated. use url property instead.
duration	double	The duration of the movie, or 0.0 if not known.
endPlaybackTime	double	The end time of movie playback. Defaults to NaN, which indicates natural end time of the movie.
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
fullscreen	boolean	Determines if the movie is presented in the entire screen (obscuring all other application content). Default is false. Setting this property to true before the movie player's view is visible will have no effect.
height	float,string	property for the view height. can be either float value or a string of the width.
initialPlaybackTime	double	The start time of movie playback. Defaults to NaN, indicating the natural start time of the movie.

left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
loadState	int	Returns the network load state of the movie player.
media	object	the media object, either a File or Blob.
mediaControlStyle	int	The style of the playback controls. Defaults to Titanium.Media.VIDEO_CONTROL_DEFAULT
mediaTypes	int	The types of media in the movie, or Titanium.Media.VIDEO_MEDIA_TYPE_NONE if not known.
movieControlMode	int	deprecated in 3.2+ of the iPhone SDK - use mediaControlStyle . Provides the ability to set the control mode of the movie player. Defaults to Titanium.Media.VIDEO_CONTROL_DEFAULT.
naturalSize	object	returns a dictionary with properties width and height . Returns the natural size of the movie or 0 for both properties if not known or applicable.
opacity	float	the opacity from 0.0-1.0
playableDuration	double	The currently playable duration of the movie, for progressively downloaded network content.
playbackState	int	Returns the current playback state of the music player
playing	boolean	Boolean to indicate if the player has started playing.
repeatMode	int	Determines how the movie player repeats when reaching the end of playback. Defaults to Titanium.Media.VIDEO_REPEAT_MODE_NONE.
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
scalingMode	int	Determines how the content scales to fit the view. Defaults to Titanium.Media.VIDEO_SCALING_ASPECT_FIT.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
sourceType	int	

The playback type of the movie. Defaults to Titanium.Media.VIDEO_SOURCE_TYPE_UNKNOWN. Specifying a playback type before playing the movie can result in faster load times.

top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
url	string	url of the media.
useApplicationAudioSession	boolean	Indicates if the movie player should inherit the application's audio session instead of creating a new session (which would interrupt the application's session). Defaults to true. Setting this property during playback will not take effect until playback is stopped and started again.
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
complete	<p>fired when movie playback ends or a user exits playback</p> <p>Event properties</p> <table border="1"> <tr> <td>reason</td><td>the completion reason. One of Titanium.Media.VIDEO_FINISH_REASON_PLAYBACK_ENDED, Titanium.Media.VIDEO_FINISH_REASON_PLAYBACK_ERROR or Titanium.Media.VIDEO_FINISH_REASON_USER_EXITED.</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> </table>	reason	the completion reason. One of Titanium.Media.VIDEO_FINISH_REASON_PLAYBACK_ENDED, Titanium.Media.VIDEO_FINISH_REASON_PLAYBACK_ERROR or Titanium.Media.VIDEO_FINISH_REASON_USER_EXITED.	source	the source object that fired the event						
reason	the completion reason. One of Titanium.Media.VIDEO_FINISH_REASON_PLAYBACK_ENDED, Titanium.Media.VIDEO_FINISH_REASON_PLAYBACK_ERROR or Titanium.Media.VIDEO_FINISH_REASON_USER_EXITED.										
source	the source object that fired the event										

type	the name of the event fired
-------------	-----------------------------

dblclick

fired when the device detects a double click against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap

fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

durationAvailable

fired when the duration is available

Event properties

duration	duration
source	the source object that fired the event
type	the name of the event fired

error

fired when movie playback encounters an error

Event properties

message	reason for error as string
source	the source object that fired the event
type	the name of the event fired

fullscreen

fired when a movie changes from fullscreen view

Event properties

entering	true to indicate entering fullscreen, false to indicate leaving fullscreen
source	the source object that fired the event
type	the name of the event fired

load

fired when the movie play loads

Event properties

source	the source object that fired the event
type	the name of the event fired

loadstate

fired when the network loadState changes

Event properties

loadState	loadState
source	the source object that fired the event
type	the name of the event fired

mediaTypesAvailable

fired when the media types is available

Event properties

mediaTypes	media types
source	the source object that fired the event
type	the name of the event fired

naturalSizeAvailable

fired when the natural size is available

Event properties

naturalSize	natural size as a dictionary. see naturalSize property.
source	the source object that fired the event
type	the name of the event fired

playbackState

fired when the playbackState changes

Event properties

playbackState	playbackState
source	the source object that fired the event
type	the name of the event fired

playing

fired when the currently playing movie changes

Event properties

source	the source object that fired the event
type	the name of the event fired
url	url of the media

preload

fired when the movie has preloaded and is ready to play

Event properties

source	the source object that fired the event
type	the name of the event fired

resize

fired when the movie player is resized

Event properties

source	the source object that fired the event
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

sourceChange

fired when the movieSourceType is Titanium.Media.VIDEO_SOURCE_TYPE_UNKNOWN when preparing for playback

playback.

Event properties

source	the source object that fired the event
sourceChange	the new sourceType
type	the name of the event fired

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

thumbnail

fired when a movie thumbnail is available from `requestThumbnailImagesAtTimes`

Event properties

error	if success is <code>false</code> , error message as string for reason
image	blob of the thumbnail image
source	the source object that fired the event
success	boolean to indicate success or failure
time	time
type	the name of the event fired

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event

source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend	fired when a touch event is completed										
	<p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td> <td>a dictionary with properties x and y describing the point of the event in screen coordinates</td> </tr> <tr> <td>source</td> <td>the source object that fired the event</td> </tr> <tr> <td>type</td> <td>the name of the event fired</td> </tr> <tr> <td>x</td> <td>the x point of the event in receiving view coordinates</td> </tr> <tr> <td>y</td> <td>the y point of the event, in receiving view coordinates</td> </tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred										
	<p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td> <td>a dictionary with properties x and y describing the point of the event in screen coordinates</td> </tr> <tr> <td>source</td> <td>the source object that fired the event</td> </tr> <tr> <td>type</td> <td>the name of the event fired</td> </tr> <tr> <td>x</td> <td>the x point of the event in receiving view coordinates</td> </tr> <tr> <td>y</td> <td>the y point of the event, in receiving view coordinates</td> </tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
touchstart	fired as soon as the device detects a gesture										
	<p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td> <td>a dictionary with properties x and y describing the point of the event in screen coordinates</td> </tr> <tr> <td>source</td> <td>the source object that fired the event</td> </tr> <tr> <td>type</td> <td>the name of the event fired</td> </tr> <tr> <td>x</td> <td>the x point of the event in receiving view coordinates</td> </tr> <tr> <td>y</td> <td>the y point of the event, in receiving view coordinates</td> </tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
twofingertap	fired when the device detects a two-finger tap against the view										
	<p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td> <td>a dictionary with properties x and y describing the point of the event in screen coordinates</td> </tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates								
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										

source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Notes

Starting in iPhone 3.2 SDK, the video player can be embedded in any normal view. However, if you create multiple video players at one time, only the last one to have its URL set will display video content or behave as if it is a view for a video player. Prior to 3.2, only fullscreen views were available on iPhone.

Titanium.Media.VideoPlayer.add

function of Titanium.Media.VideoPlayer
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.Media.VideoPlayer.addEventListener

function of Titanium.Media.VideoPlayer
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Media.VideoPlayer.animate

function of Titanium.Media.VideoPlayer
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.Media.VideoPlayer.cancelAllThumbnailImageRequests

function of Titanium.Media.VideoPlayer

Cancels all pending asynchronous thumbnail requests.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.VideoPlayer.fireEvent

function of Titanium.Media.VideoPlayer

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Media.VideoPlayer.hide

function of Titanium.Media.VideoPlayer

hide the view

Arguments

This function takes no arguments.

Return Type

Titanium.Media.VideoPlayer.pause

function of Titanium.Media.VideoPlayer

pause playing the video. On iOS only available under 3.2 and later

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.VideoPlayer.play

function of Titanium.Media.VideoPlayer

start playing the video

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.VideoPlayer.release

function of Titanium.Media.VideoPlayer

release the internal video resources immediately. this is not usually necessary but can help if you no longer need to use the player after it is used to help conserve memory.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.VideoPlayer.remove

function of Titanium.Media.VideoPlayer

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
------	------	-------------

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.Media.VideoPlayer.removeEventListener

function of Titanium.Media.VideoPlayer
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Media.VideoPlayer.requestThumbnailImagesAtTimes

function of Titanium.Media.VideoPlayer

Asynchronously request thumbnails for one or more times, provided as an array of numbers (double). Fires a `thumbnail` event on completion. Optionally invokes the callback function passed in the method.

Arguments

Name	Type	Description
times	array	array of doubles for each time to request
option	int	either [[Titanium.Media.VIDEO_TIME_OPTION_NEAREST_KEYFRAME]] or [[Titanium.Media.VIDEO_TIME_OPTION_EXACT]].

Return Type

void

Titanium.Media.VideoPlayer.setBackgroundView

function of Titanium.Media.VideoPlayer

A view for customization which is always displayed behind movie content.

Arguments

Name	Type	Description

view

object view to set

Return Type

void

Titanium.Media.VideoPlayer.setMedia

function of Titanium.Media.VideoPlayer

the a non-url based media to play, either a File or Blob.

Arguments

Name	Type	Description
media	object	media to play

Return Type

void

Titanium.Media.VideoPlayer.setUrl

function of Titanium.Media.VideoPlayer

the url to play

Arguments

Name	Type	Description
url	string	the url to play

Return Type

void

Titanium.Media.VideoPlayer.show

function of Titanium.Media.VideoPlayer

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.VideoPlayer.stop

function of Titanium.Media.VideoPlayer

stop playing the video

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.VideoPlayer.thumbnailImageAtTime

function of Titanium.Media.VideoPlayer

Returns a thumbnail at the given time as a blob.

Arguments

Name	Type	Description
time	double	playback time
option	int	either [[Titanium.Media.VIDEO_TIME_OPTION_NEAREST_KEYFRAME]] or [[Titanium.Media.VIDEO_TIME_OPTION_EXACT]]

Return Type

object

Titanium.Media.VideoPlayer.toImage

function of Titanium.Media.VideoPlayer

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.Media.addEventListener

function of Titanium.Media

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Media.beep

function of Titanium.Media

play a device beep notification

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.createAudioPlayer

function of Titanium.Media

create and return an instance of Titanium.Media.AudioPlayer

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Media.AudioPlayer

Return Type

object

Titanium.Media.createAudioRecorder

function of Titanium.Media

create and return an instance of Titanium.Media.AudioRecorder

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Media.AudioRecorder

Return Type

Titanium.Media.createItem

function of Titanium.Media

create and return an instance of Titanium.Media.Item

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Media.Item

Return Type

object

Titanium.Media.createMusicPlayer

function of Titanium.Media

create and return an instance of Titanium.Media.MusicPlayer

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Media.MusicPlayer

Return Type

object

Titanium.Media.createSound

function of Titanium.Media

create and return an instance of Titanium.Media.Sound

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Media.Sound

Return Type

object

Titanium.Media.createVideoPlayer

function of Titanium.Media

create and return an instance of Titanium.Media.VideoPlayer

create and return an instance of Titanium.Media.VideoPlayer

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Media.VideoPlayer

Return Type

object

Titanium.Media.fireEvent

function of Titanium.Media

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Media.hideCamera

function of Titanium.Media

hide the device camera UI. this must be called after calling `showCamera` and only when `autohide` is set to false. this method will cause the media capture device be hidden.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.hideMusicLibrary

function of Titanium.Media

hide the music library. This must be called after calling `showMusicLibrary` and only when `autohide` is set to false. iPhone and iPad only.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.isMediaTypeSupported

function of Titanium.Media

return boolean to indicate if the media type is supported

Arguments

Name	Type	Description
media	string	media type as a string of either camera, photo or photogallery.
type	string	the type of media to check

Return Type

boolean

Titanium.Media.openPhotoGallery

function of Titanium.Media

open the photo gallery picker

Arguments

Name	Type	Description
options	object	pass a dictionary with the following supported keys: success a function that will be called when the camera is completed, error a function that will be called upon receiving an error, cancel a function that will be called if the user presses the cancel button, autohide boolean if the camera should auto hide after the media capture is completed (defaults to true), animated boolean if the dialog should be animated (defaults to true) upon showing and hiding, saveToPhotoGallery boolean if the media should be saved to the photo gallery upon successful capture, allowEditing boolean if the media should be editable after capture in the UI interface, mediaTypes an array of media type constants supported by the capture device UI, showControls boolean to indicate if the built-in UI controls should be displayed, overlay view which is added as an overlay to the UI (on top), transform an transformation matrix that applies to the UI transform. For iPad, popoverView can be provided to position the photo gallery popover a specific view and arrowDirection can be provided to control the type of arrow and position of the gallery.

Return Type

void

Titanium.Media.removeEventListener

function of Titanium.Media

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Media.saveToPhotoGallery

function of Titanium.Media

save media to photo gallery / camera roll

Arguments

Name	Type	Description
media	object	save the media passed to the cameras photo roll/media gallery. must be one of Blob object or File object or an error will be generated.

Return Type

void

Titanium.Media.showCamera

function of Titanium.Media

show the camera

Arguments

Name	Type	Description
options	object	pass a dictionary with the following supported keys: success a function that will be called when the camera is completed, error a function that will be called upon receiving an error, cancel a function that will be called if the user presses the cancel button, autohide boolean if the camera should auto hide after the media capture is completed (defaults to true), animated boolean if the dialog should be animated (defaults to true) upon showing and hiding, saveToPhotoGallery boolean if the media should be saved to the photo gallery upon successful capture, allowEditing boolean if the media should be editable after capture in the UI interface, mediaTypes an array of media type constants supported by the capture device UI, videoMaximumDuration float duration on how long in milliseconds to allow capture before completing, videoQuality constant to indicate the video quality during capture, showControls boolean to indicate if the built-in UI controls should be displayed, overlay view which is added as an overlay to the camera UI (on top), transform an transformation matrix that applies to the camera UI transform.

Return Type

void

Titanium.Media.showMusicLibrary

function of Titanium.Media

show the music library. iPhone and iPad only.

Arguments

Name	Type	Description
options	object	pass a dictionary with the following supported keys: success a function that will be called when the camera is completed, error a function that will be called upon receiving an error, cancel a function that will be called if the user presses the cancel button, autohide boolean if the library listing should auto hide after selection is completed (defaults to true), animated boolean if the dialog should be animated (defaults to true) upon showing and hiding, mediaTypes an array of media type constants defining selectable media (see MUSIC_MEDIA_TYPE_* properties below) as either an array or a bitwise-or single value, allowMultipleSelections boolean whether or not more than one media selection is allowed

Return Type

void

Titanium.Media.startMicrophoneMonitor

function of Titanium.Media

start the monitoring of microphone sound level

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.stopMicrophoneMonitor

function of Titanium.Media

stop the monitoring of microphone sound level

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.takePicture

use the device camera to capture a photo. this must be called after calling `showCamera` and only when `autohide` is set to false. this method will cause the media capture device to capture a photo and call the `success` callback.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Media.takeScreenshot

function of Titanium.Media

take a screen shot of the visible UI on the device

Arguments

Name	Type	Description
<code>callback</code>	function	function that will be called upon capture. the event property <code>media</code> will contain an image Blob object of the screenshot

Return Type

void

Titanium.Media.vibrate

function of Titanium.Media

play a device vibration

Arguments

This function takes no arguments.

Return Type

void

Titanium.Network

submodule of Titanium

The top level Network module. The Network module is used accessing Networking related functionality.

Objects

Name	Description
------	-------------

Titanium.Network.BonjourBrowser	The BonjourBrowser instance returned from Titanium.Network.createBonjourBrowser. This object is a browser for the discovery and retrieval of Bonjour services available on the network.
Titanium.Network.BonjourService	The BonjourService instance returned either from Titanium.Network.createBonjourService or via the service list from a BonjourBrowser <code>updatedServices</code> event. This object describes a service on the network which is published by Bonjour.
Titanium.Network.HTTPClient	The HttpClient instance returned from Titanium.Network.createHTTPClient. This object (mostly) implements the XMLHttpRequest specification.
Titanium.Network.TCPSocket	The TCPSocket instance returned from Titanium.Network.createTCPSocket. This object represents a socket which either listens locally on the device for connections, or connects to a remote machine.

Methods

Name	Description
addConnectivityListener	adds a connectivity listener to listen for network changes. This method has been deprecated in favor of listening for a <code>change</code> event.
addEventListener	add an event listener for the instance to receive view triggered events
createBonjourBrowser	create and return an instance of Titanium.Network.BonjourBrowser
createBonjourService	create and return an instance of Titanium.Network.BonjourService
createHTTPClient	create and return an instance of Titanium.Network.HTTPClient
createTCPSocket	create and return an instance of Titanium.Network.TCPSocket
decodeURIComponent	decode a URI component part using URI encoding
encodeURIComponent	encode a URI component part using URI encoding
fireEvent	fire a synthesized event to the views listener
registerForPushNotifications	register for push notifications with the Apple Push Notification Service. Only available on iPhone.
removeConnectivityListener	removes a connectivity listener. This method has been deprecated in favor of listening for a <code>change</code> event.
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
------	------	-------------

INADDR_ANY	string	constant value representing the ability for sockets to listen on any locally available network device
NETWORK_LAN	int	constant value to indicate that the network is LAN
NETWORK_MOBILE	int	constant value to indicate that the network is MOBILE
NETWORK_NONE	int	constant value to indicate that the network is not available
NETWORK_UNKNOWN	int	constant value to indicate that the network is not known
NETWORK_WIFI	int	constant value to indicate that the network is WIFI
NOTIFICATION_TYPE_ALERT	int	constant value for the push notification alert type
NOTIFICATION_TYPE_BADGE	int	constant value for the push notification badge type
NOTIFICATION_TYPE_SOUND	int	constant value for the push notification sound type
READ_MODE	int	constant value specifying read-only mode for sockets
READ_WRITE_MODE	int	constant value specifying read-write mode for sockets
WRITE_MODE	int	constant value specifying write-only mode for sockets
networkType	int	the network type value as a constant.
networkTypeName	string	the network type name constant. Returns one of NONE, WIFI, LAN or MOBILE.
online	boolean	readonly boolean value that indicates if the network is reachable to the Internet either via WIFI or Carrier network
remoteDeviceUUID	string	the remote device UUID if the device was registered with the Apple Push Notification Service or null if not set. Only available on iPhone.
remoteNotificationTypes	array	returns an array of network type constants enabled for the application. Only available on iPhone.

remoteNotificationsEnabled	boolean	returns true if remote notifications have been enabled. Only available on iPhone.
-----------------------------------	---------	---

Events

Name	Description	
change	fired upon a network connectivity change	
Event properties		
networkType	the new network type	
networkTypeName	the new network type as a string	
online	boolean to indicate if the network is online	
source	the source object that fired the event	
type	the name of the event fired	

Titanium.Network.BonjourBrowser

object of Titanium.Network

The BonjourBrowser instance returned from Titanium.Network.createBonjourBrowser. This object is a browser for the discovery and retrieval of Bonjour services available on the network.

Methods

Name	Description
search	Conduct a search for Bonjour services matching the type and domain specified during creation
stopSearch	Halt an ongoing search

Properties

Name	Type	Description
domain	string	The domain the browser is searching in
isSearching	boolean	Whether or not the browser is currently searching

serviceType	string	The type of the service the browser searches for
--------------------	--------	--

Events

Name	Description				
-event	<p>updatedServices</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired
source	the source object that fired the event				
type	the name of the event fired				
services	An array of BonjourService objects corresponding to currently available services. If you cache this value, including using it as table data, be aware that it could become out of date at any time due to the asynchronous nature of Bonjour service discovery.				
updatedServices	<p>Fired when the discovered services list is updated</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired
source	the source object that fired the event				
type	the name of the event fired				

Notes

If your application publishes Bonjour services itself, that service will be discovered by the browser if necessary; be prepared to perform a check if you do not want to list local services as available. Bonjour service browsing is an asynchronous operation, meaning that you should be extremely careful when caching values from the 'services' property returned by the updatedServices event. In particular, if you maintain a local copy of available services and a user tries to connect to one, you should be prepared to handle failures gracefully; the next updatedServices event should provide the new services list, but you should not rely on it being delivered before user input. When a window which uses Bonjour browsing is closed, if you do not want to continue searching, you must call the stop() method.

Titanium.Network.BonjourBrowser.search

function of Titanium.Network.BonjourBrowser

Conduct a search for Bonjour services matching the type and domain specified during creation

Arguments

This function takes no arguments.

Return Type

void

Titanium.Network.BonjourBrowser.stopSearch

function of Titanium.Network.BonjourBrowser

Halt an ongoing search

Arguments

This function takes no arguments.

Return Type

void

Titanium.Network.BonjourService

object of Titanium.Network

The BonjourService instance returned either from Titanium.Network.createBonjourService or via the service list from a BonjourBrowser updatedServices event. This object describes a service on the network which is published by Bonjour.

Methods

Name	Description
publish	Publish a Bonjour service to the network. Only works if isLocal is TRUE
resolve	Resolve a Bonjour service from the network. Must be done before attempting to access the service's socket information, if a remote service. You cannot resolve a locally published service.
stop	Halts publication of a service.

Properties

Name	Type	Description
domain	string	the domain of the service
isLocal	boolean	whether or not the service is local to the device
name	string	the name of the service
socket	object	the TCP Socket object that is used to connect to the service

type	string	the type of the service
-------------	--------	-------------------------

Events

This object has no events

Notes

You can only publish Bonjour services attached to a socket which is currently listening; you cannot publish a service for a remotely connected socket. If you stop the Bonjour service and wish to close the socket it uses, it is strongly recommended that you stop the service first. When a window which publishes a Bonjour service is closed, you must stop the service if the associated socket is also to be closed, or if it is no longer necessary to publish. Unlike other network operations, Bonjour service resolution and publishing is synchronous, so be aware that your code may block while resolution is going on. In particular, you may wish to display UI elements indicating background activity before beginning resolution.

Titanium.Network.BonjourService.publish

function of Titanium.Network.BonjourService

Publish a Bonjour service to the network. Only works if isLocal is TRUE

Arguments

Name	Type	Description
socket	object	a TCPSocket object to associate with the Bonjour service.

Return Type

void

Titanium.Network.BonjourService.resolve

function of Titanium.Network.BonjourService

Resolve a Bonjour service from the network. Must be done before attempting to access the service's socket information, if a remote service. You cannot resolve a locally published service.

Arguments

Name	Type	Description
timeout	double	the timeout for service resolution, in seconds. Optional, default is 120s.

Return Type

void

Titanium.Network.BonjourService.stop

function of Titanium.Network.BonjourService

Halts publication of a service.

Arguments

This function takes no arguments.

Return Type

void

Titanium.Network.HTTPClient

object of Titanium.Network

The HttpClient instance returned from Titanium.Network.createHttpClient. This object (mostly) implements the XMLHttpRequest specification.

Methods

Name	Description
<code>abort</code>	abort a pending request
<code>getResponseHeader</code>	return the response header.
<code>open</code>	open the request and ready the connection
<code>send</code>	send the request
<code>setRequestHeader</code>	set the request header. Must be called after <code>open</code> but before <code>send</code> .
<code>setTimeout</code>	set the request timeout

Properties

Name	Type	Description
<code>DONE</code>	int	the DONE readyState constant
<code>HEADERS_RECEIVED</code>	int	the HEADERS_RECEIVED readyState constant
<code>LOADING</code>	int	the LOADING readyState constant
<code>OPENED</code>	int	the OPENED readyState constant

UNSENT	int	the UNSET readyState constant
connected	boolean	boolean to indicate that the response was successful
connectionType	string	the connection type, normally either GET or POST.
location	string	the absolute URL of the request
ondatastream	function	set this to a function before calling open to cause the function to be called at regular intervals as the request data is being received. the progress property of the event will contain a value from 0.0-1.0 with the progress.
onerror	function	set this to a function before calling open to cause the function to be called upon a error response
onload	function	set this to a function before calling open to cause the function to be called upon a successful response
onreadystatechange	function	set this to a function before calling open to cause the function to be called for each readyState change
onsendstream	function	set this to a function before calling open to cause the function to be called at regular intervals as the request data is being transmitted. the progress property of the event will contain a value from 0.0-1.0 with the progress.
readyState	int	the readyState value
responseData	object	the response data as a Blob object.
responseText	string	the response as text or null if an error was received or no data was returned
responseXML	object	the response object as an XML DOMDocument object. returns null if the content type returned by the server was not XML or the content could not be parsed
status	int	the response HTTP status code
timeout	int	timeout is milliseconds when the connection should be aborted
validatesSecureCertificate	boolean	set this to control how SSL certification validation is performed on connection. defaults to false if

Events

This object has no events

Titanium.Network.HTTPClient.abort

function of Titanium.Network.HTTPClient

abort a pending request

Arguments

This function takes no arguments.

Return Type

void

Titanium.Network.HTTPClient.getResponseHeader

function of Titanium.Network.HTTPClient

return the response header.

Arguments

Name	Type	Description
name	string	the header name

Return Type

string

Titanium.Network.HTTPClient.open

function of Titanium.Network.HTTPClient

open the request and ready the connection

Arguments

Name	Type	Description
method	string	the HTTP method
url	string	the URL for the request
async	boolean	optional property to indicate if asynchronous (default) or not

Return Type

Titanium.Network.HTTPClient.send

function of Titanium.Network.HTTPClient

send the request

Arguments

Name	Type	Description
data	object	the data to send in the request. can either be null, dictionary, string, File object or Blob.

Return Type

void

Titanium.Network.HTTPClient.setRequestHeader

function of Titanium.Network.HTTPClient

set the request header. Must be called after `open` but before `send`.

Arguments

Name	Type	Description
name	string	name of the header
value	string	value of the header. May be null to clearing out a property, such as X-Requested-With

Return Type

void

Titanium.Network.HTTPClient.setTimeout

function of Titanium.Network.HTTPClient

set the request timeout

Arguments

Name	Type	Description
timeout	double	the timeout in milliseconds

Return Type

void

Titanium.Network.TCPSocket

object of Titanium.Network

The TCPSocket instance returned from Titanium.Network.createTCPSocket. This object represents a socket which either listens locally on the device for connections, or connects to a remote machine.

Methods

Name	Description
close	close the socket
connect	connect the socket to a TCP server
listen	set up the socket to receive connections
write	write data to the socket, if the mode is WRITE_MODE or READ_WRITE_MODE

Properties

Name	Type	Description
hostName	string	the host name to connect to. Must be Titanium.Network.INADDR_ANY or an identifier for the local device in order to listen
isValid	boolean	whether or not the socket is valid
mode	int	the socket's mode
port	int	the port to connect/listen on
stripTerminator	boolean	strip terminating null character when sending string data; default is false

Events

Name	Description		
read	new data was read off the socket Event properties <table border="1"><tr><td>data[object]</td><td>a blob representing the data read, can be interpreted via toString</td></tr></table>	data[object]	a blob representing the data read, can be interpreted via toString
data[object]	a blob representing the data read, can be interpreted via toString		

from[int]	the reference for the socket that data was retrieved from
source	the source object that fired the event
type	the name of the event fired

readError

an error occurred when reading

Event properties

code[int]	the BSD socket error code
error[string]	a description of the error
source	the source object that fired the event
type	the name of the event fired

writeError

an error occurred when writing

Event properties

code[int]	the BSD socket error code
error[string]	a description of the error
source	the source object that fired the event
type	the name of the event fired

Notes

Sockets are nontrivial; it is recommended that anyone using them be familiar with the basics of BSD sockets. All sockets use TCP connections, and are asynchronous for read operations, so your program should be ready to receive 'read' events at any point. Socket references cannot be transferred to socket objects, and vice-versa - socket references are an internal mechanism which is used only to determine which sockets to send data to and read data from. For listening sockets, it is highly recommended that you use the Titanium.Network.INADDR_ANY constant as the host name. If a window containing a socket is closed, the socket MUST be closed also unless you intend to continue to receive data, otherwise the socket will consume resources (and potentially cause conflicts with opening the window again, if a listener) until the program is restarted. Be aware of the differences between the listen() and connect() functions; attempting to use one when you mean the other may result in errors, unpredictable behavior, or both.

Titanium.Network.TCPSocket.close

function of Titanium.Network.TCPSocket

close the socket

Arguments

This function takes no arguments.

Return Type

void

Titanium.Network.TCPSocket.connect

function of Titanium.Network.TCPSocket

connect the socket to a TCP server

Arguments

This function takes no arguments.

Return Type

void

Titanium.Network.TCPSocket.listen

function of Titanium.Network.TCPSocket

set up the socket to receive connections

Arguments

This function takes no arguments.

Return Type

void

Titanium.Network.TCPSocket.write

function of Titanium.Network.TCPSocket

write data to the socket, if the mode is WRITE_MODE or READ_WRITE_MODE

Arguments

Name	Type	Description
data	object,string	either a string or blob object representing the data to be transferred
sendTo	int	the socket reference to send the data to. Default is to send to all connected sockets

Return Type

void

Titanium.Network.addConnectivityListener

function of Titanium.Network

adds a connectivity listener to listen for network changes. This method has been deprecated in favor of listening for a `change` event.

Arguments

Name	Type	Description
callback	function	callback function to invoke upon network connectivity changes

Return Type

void

Titanium.Network.addEventListenert

function of Titanium.Network

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Network.createBonjourBrowser

function of Titanium.Network

create and return an instance of Titanium.Network.BonjourBrowser

Arguments

Name	Type	Description
serviceType	string	service to search for, must include the protocol type suffix (_tcp)
domain	string	the Bonjour service domain to conduct the search in. Default value is 'local.'
parameters	object	(optional) a dictionary object properties defined in Titanium.Network.BonjourBrowser

Return Type

object

Titanium.Network.createBonjourService

function of Titanium.Network

create and return an instance of Titanium.Network.BonjourService

Arguments

Name	Type	Description
name	string	the name of the service. Must be a unique identifier for this service type and domain.
type	string	the type of service. Must include the protocol identifier (.tcp)
domain	string	the domain to publish the service in. Default value is 'local.'
parameters	object	(optional) a dictionary object properties defined in Titanium.Network.BonjourService

Return Type

object

Titanium.Network.createHTTPClient

function of Titanium.Network

create and return an instance of Titanium.Network.HTTPClient

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.Network.HTTPClient

Return Type

object

Titanium.Network.createTCPSocket

function of Titanium.Network

create and return an instance of Titanium.Network.TCPSocket

Arguments

Name	Type	Description
hostName	string	the host name to connect to/listen on
port	int	the port for the socket
mode	int	the socket's mode; one of Titanium.Network.READ_MODE, Titanium.Network.WRITE_MODE, Titanium.Network.READ_WRITE_MODE
parameters	object	(optional) a dictionary object properties defined in Titanium.Network.TCPSocket

Return Type

object

Titanium.Network.decodeURIComponent

function of Titanium.Network

decode a URI component part using URI encoding

Arguments

Name	Type	Description
value	string	input value to be decoded

Return Type

string

Titanium.Network.encodeURIComponent

function of Titanium.Network

encode a URI component part using URI encoding

Arguments

Name	Type	Description
value	string	input value to be encoded

Return Type

string

Titanium.Network.fireEvent

function of Titanium.Network

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Network.registerForPushNotifications

function of Titanium.Network

register for push notifications with the Apple Push Notification Service. Only available on iPhone.

Arguments

Name	Type	Description
config	object	dictionary of the following: <code>types</code> is an array of type constants that the application would like to receive, <code>success</code> is a callback function that is called when the push registration is successfully completed, <code>error</code> is a callback function that is called when an error is received during registration and <code>callback</code> is a callback function that is invoked upon receiving a new push notification. This method should be called at application startup.

Return Type

void

Titanium.Network.removeConnectivityListener

function of Titanium.Network

removes a connectivity listener. This method has been deprecated in favor of listening for a `change` event.

Arguments

Name	Type	Description
callback	function	callback function to remove

Return Type

void

Titanium.Network.removeEventListener

function of Titanium.Network

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Platform

submodule of Titanium

The top level Platform module. The Platform module is used accessing the device's platform related functionality.

Objects

Name	Description
Titanium.Platform.DisplayCaps	The Display Caps object returned by the Titanium.Platform.displayCaps property.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
createUUID	create a globally unique identifier
fireEvent	fire a synthesized event to the views listener
openURL	open a URL in the system default manner
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
BATTERY_STATE_CHARGING	int	the device is plugged in and currently being charged
BATTERY_STATE_FULL	int	the battery is fully charged
BATTERY_STATE_UNKNOWN	int	the battery state is unknown or not monitoring is not enabled
BATTERY_STATE_UNPLUGGED	int	the device is unplugged
address	string	the ip address that the device reports (only applicable on WIFI network)
architecture	string	the processor architecture that the device reports
availableMemory	double	return the amount of memory available on the device in bytes
batteryLevel	float	the current device battery level, this property is only accessible if

the current device battery level. this property is only accessible if **batteryMonitoring** is enabled. on iPhone, this level only changes at 5% intervals.

batteryMonitoring	boolean	boolean to indicate if battery monitoring is enabled
batteryState	int	constant that represents the state of the battery. this property is only accessible if batteryMonitoring is enabled
displayCaps	object	return the DisplayCaps object for platform
id	string	the unique id of the device
locale	string	the primary language of the device that the user has enabled
macaddress	string	this property will return a unique identifier for the device
model	string	the model of the phone that the device reports
name	string	the name of the platform returned by the device
netmask	string	the network mask that the device reports (only applicable on WIFI network)
osname	string	the shortname of the operating system. for example, on an iPhone, will return iphone , iPad will return ipad and Android will return android .
ostype	string	the OS architecture, such as 32 bit
processorCount	int	the number of processors the device reports
username	string	the username of the device, if set
version	string	the version of the platform returned by the device

Events

Name	Description
battery	fired when the battery state changes. the battery state changes are only tracked on iPhone/iPad at 5% increments.

Event properties

level	the new battery level
source	the source object that fired the event
state	the new battery state
type	the name of the event fired

Titanium.Platform.DisplayCaps

object of Titanium.Platform

The Display Caps object returned by the Titanium.Platform.displayCaps property.

Methods

This object has no methods

Properties

Name	Type	Description
density	string	returns the density property of the display device.
dpi	int	the DPI of the display device.
platformHeight	float	the height of the device screen
platformWidth	float	the width of the device screen

Events

This object has no events

Notes

On iPhone, the **density** property will return **low** and the **dpi** property will return **160**. For iPad, the **density** property will return **high** and the **dpi** property will return **130**. For Android, these values are device specific.

Titanium.Platform.addEventListener

function of Titanium.Platform

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Platform.createUUID

function of Titanium.Platform

create a globally unique identifier

Arguments

This function takes no arguments.

Return Type

string

Titanium.Platform.fireEvent

function of Titanium.Platform

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Platform.openURL

function of Titanium.Platform

open a URL in the system default manner

Arguments

Name	Type	Description
url	string	the url to open

Return Type

void

Titanium.Platform.removeEventListener

function of Titanium.Platform
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI

submodule of Titanium

The main Titanium.UI module. The UI module is responsible for native user-interface components and interaction inside Titanium. The goal of the UI module is to provide a native experience along with native performance by compiling Javascript code into their native counterparts as part of the build process.

Objects

Name	Description
Titanium.UI.2DMatrix	The 2DMatrix is created by Titanium.UI.create2DMatrix. The 2D Matrix is an object for holding values for an affine transformation matrix. A 2D matrix is used to rotate, scale, translate, or skew the objects in a two-dimensional space. A 2D matrix is represented by a 3 by 3 matrix. Because the third column is always (0,0,1), the data structure contains values for only the first two columns.
Titanium.UI.3DMatrix	The 2DMatrix is created by Titanium.UI.create2DMatrix. The 2D Matrix is an object for holding values for an affine transformation matrix. A 2D matrix is used to rotate, scale, translate, or skew the objects in a three-dimensional space. A 3D matrix is represented by a 4 by 4 matrix. Because the forth column is always (0,0,1), the data structure contains values for only the first three columns.
Titanium.UI.ActivityIndicator	An Activity Indicator is created by the method Titanium.UI.createActivityIndicator. An activity indicator can be used to show the progress of an operation in the UI to let the user know some action is taking place.
Titanium.UI.AlertDialog	The Alert Dialog is created by Titanium.UI.createAlertDialog and allows you to show a modal application dialog.

Titanium.UI.Animation	The Animation object is used for specifying lower-level animation properties and more low-level control of events during an animation. The Animation is created by the method Titanium.UI.createAnimation.
Titanium.UI.Button	A Button is created by the method Titanium.UI.createButton.
Titanium.UI.ButtonBar	A Button Bar is created by the method Titanium.UI.createButtonBar.
Titanium.UI.CoverFlowView	The Cover Flow view is container for showing animated, three dimensional images in a nice UI. The Cover Flow view is created by the method Titanium.UI.createCoverFlowView.
Titanium.UI.DashboardItem	An item that represents a visual icon in the Titanium.UI.DashboardView. The Dashboard Item is created by the method Titanium.UI.createDashboardItem.
Titanium.UI.DashboardView	The Dashboard View provides a view that supports the ability to have Springboard-like view of icons which can be reordered by dragging and can contain multiple pages of icons in a scrollable view. The Dashboard View is created by the method Titanium.UI.createDashboardView.
Titanium.UI.EmailDialog	The Email Dialog is created by Titanium.UI.createEmailDialog and allows you to send in application emails on behalf of the application user.
Titanium.UI.ImageView	An Image View is used to display an image or a series of images in an animation. The Image View is created by the method Titanium.UI.createImageView.
Titanium.UI.Label	A Label is created by the method Titanium.UI.createLabel.
Titanium.UI.OptionDialog	The Option Dialog is created by Titanium.UI.createOptionDialog and allows you to show a modal dialog of one or more options to the user.
Titanium.UI.Picker	A Picker is created by the method Titanium.UI.createPicker. A Picker can be used to select one or more fixed values.
Titanium.UI.PickerColumn	The picker row object created by Titanium.UI.createPickerColumn.
Titanium.UI.PickerRow	The picker row object created by Titanium.UI.createPickerRow.
Titanium.UI.ProgressBar	A Progress Bar is created by the method Titanium.UI.createProgressBar.
Titanium.UIScrollView	A Scroll View is used to create a scrollable region of content. Views added to the Scroll View will be scrolled based on the content size of the Scroll View. The Scroll View is created by the method Titanium.UI.createScrollView.

Titanium.UI.createScrollView. Note: In Android, Scroll Views can only scroll in one direction, either vertical or horizontal, but not both at the same time. See the Titanium.UI.ScrollView.scrollType property

Titanium.UI.SccrollView	The Scrollable View provides a view that supports horizontal scrolling on one or more views in a gesture motion. The Scrollable View also optionally supports a visual paging control to indicate the page that the view is visible. The Scrollable View is created by the method Titanium.UI.createScrollView.
Titanium.UI.SearchBar	A Search Bar is created by the method Titanium.UI.createSearchBar.
Titanium.UI.Slider	A Slider is created by the method Titanium.UI.createSlider.
Titanium.UI.Switch	A Switch is created by the method Titanium.UI.createSwitch.
Titanium.UI.Tab	A TabGroup Tab instance. Each Tab instance maintains a stack of tab windows. Only one window within in the Tab can be visible at a time. When a window is closed, either by the user or by code, the window is removed from the stack, make the previous window visible. The root tab window cannot be removed. The Tab Group is created by the method Titanium.UI.createTab.
Titanium.UI.TabGroup	The Tab Group allows you to manage a tabbed UI of one or more windows. The Tab Group is created by the method Titanium.UI.createTabGroup.
Titanium.UI.TabbedPane	A Tabbed Bar is created by the method Titanium.UI.createTabbedPane. The difference between the Tabbed Bar and the Button Bar is that the tabbed bar visually maintains a state (visually distinguished as a pressed or selected look).
Titanium.UI.TableView	A Table View allows you to create a scrollable table of content in a list-based fashion. The Table View is created by the method Titanium.UI.createTableView.
Titanium.UI.TableViewRow	A TableView row object created by the method Titanium.UI.createTableViewRow.
Titanium.UI.TableViewSection	A TableView section object created by the method Titanium.UI.createTableViewSection.
Titanium.UI.TextArea	A Text Area is created by the method Titanium.UI.createTextArea. The Text Area is a multiline field.
Titanium.UI.TextField	A Text Area is created by the method Titanium.UI.createTextField. The Text Field is a single line field.
Titanium.UI.Toolbar	A Toolbar is created by the method Titanium.UI.createToolbar. A Toolbar can be placed at the bottom of a window and contain buttons.

Titanium.UI.View

The View is an empty drawing surface or container. The View is created by the method Titanium.UI.createView.

Titanium.UI.WebView

The Web View allows you to open an HTML5 based view which can load either local or remote content. The content can be any valid web content such as HTML, PDF, SVG or other WebKit supported content types. The Web View is created by the method Titanium.UI.createWebView.

Titanium.UI.Window

The Window is an empty drawing surface or container. The Window is created by the method Titanium.UI.createWindow. Unlike Views, Windows can be opened and closed and can have special display properties such as `fullscreen` or `modal`.

Methods

Name	Description
<code>addEventListener</code>	add an event listener for the instance to receive view triggered events
<code>create2DMatrix</code>	create and return an instance of Titanium.UI.2DMatrix
<code>create3DMatrix</code>	create and return an instance of Titanium.UI.3DMatrix
<code>createActivityIndicator</code>	create and return an instance of Titanium.UI.ActivityIndicator
<code>createAlertDialog</code>	create and return an instance of Titanium.UI.AlertDialog
<code>createAnimation</code>	create and return an instance of Titanium.UI.Animation
<code>createButton</code>	create and return an instance of Titanium.UI.Button
<code>createButtonBar</code>	create and return an instance of Titanium.UI.ButtonBar
<code>createCoverFlowView</code>	create and return an instance of Titanium.UI.CoverFlowView
<code>createDashboardItem</code>	create and return an instance of Titanium.UI.DashboardItem
<code>createDashboard View</code>	create and return an instance of Titanium.UI.DashboardView
<code>createEmailDialog</code>	create and return an instance of Titanium.UI.EmailDialog
<code>createImageView</code>	create and return an instance of Titanium.UI.ImageView
<code>createLabel</code>	create and return an instance of Titanium.UI.Label
<code>createOptionDialog</code>	create and return an instance of Titanium.UI.OptionDialog
<code>createPicker</code>	create and return an instance of Titanium.UI.Picker
<code>createPickerColumn</code>	create and return an instance of Titanium.UI.PickerColumn
<code>createPickerRow</code>	create and return an instance of Titanium.UI.PickerRow
<code>createProgressBar</code>	create and return an instance of Titanium.UI.ProgressBar
<code>createScrollView</code>	create and return an instance of Titanium.UI.ScrollView
<code>createScrollView</code>	create and return an instance of Titanium.UI.ScrollableView
<code>createearchBar</code>	create and return an instance of Titanium.UI.SearchBar
<code>createSlider</code>	create and return an instance of Titanium UI Slider

createSlider	create and return an instance of Titanium.UI.Slider
createSwitch	create and return an instance of Titanium.UI.Switch
createTab	create and return an instance of Titanium.UI.Tab
createTabGroup	create and return an instance of Titanium.UI.TabGroup
createTabbedBar	create and return an instance of Titanium.UI.TabbedBar
createTableView	create and return an instance of Titanium.UI.TableView
createTableViewRow	create and return an instance of Titanium.UI.TableViewRow
createTableViewSection	create and return an instance of Titanium.UI.TableViewSection
createTextArea	create and return an instance of Titanium.UI.TextArea
createTextField	create and return an instance of Titanium.UI.TextField
createToolbar	create and return an instance of Titanium.UI.Toolbar
createView	create and return an instance of Titanium.UI.View
createWebView	create and return an instance of Titanium.UI.WebView
createWindow	create and return an instance of Titanium.UI.Window
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
ANIMATION_CURVE_EASE_IN	int	animation curve constant
ANIMATION_CURVE_EASE_IN_OUT	int	animation curve constant
ANIMATION_CURVE_EASE_OUT	int	animation curve constant
ANIMATION_CURVE_LINEAR	int	animation curve constant
BLEND_MODE_CLEAR	int	image mode constant
BLEND_MODE_COLOR	int	image mode constant
BLEND_MODE_COLOR_BURN	int	image mode constant
BLEND_MODE_COLOR_DODGE	int	image mode constant

BLEND_MODE_COPY	int	image mode constant
BLEND_MODE_DARKEN	int	image mode constant
BLEND_MODE_DESTINATION_ATOP	int	image mode constant
BLEND_MODE_DESTINATION_IN	int	image mode constant
BLEND_MODE_DESTINATION_OUT	int	image mode constant
BLEND_MODE_DESTINATION_OVER	int	image mode constant
BLEND_MODE_DIFFERENCE	int	image mode constant
BLEND_MODE_EXCLUSION	int	image mode constant
BLEND_MODE_HARD_LIGHT	int	image mode constant
BLEND_MODE_HUE	int	image mode constant
BLEND_MODE_LIGHTEN	int	image mode constant
BLEND_MODE_LUMINOSITY	int	image mode constant
BLEND_MODE_MULTIPLY	int	image mode constant
BLEND_MODE_NORMAL	int	image mode constant
BLEND_MODE_OVERLAY	int	image mode constant
BLEND_MODE_PLUS_DARKER	int	image mode constant
BLEND_MODE_PLUS_LIGHTER	int	image mode constant
BLEND_MODE_SATURATION	int	image mode constant

BLEND_MODE_SCREEN	int	image mode constant
BLEND_MODE_SOFT_LIGHT	int	image mode constant
BLEND_MODE_SOURCE_ATOP	int	image mode constant
BLEND_MODE_SOURCE_IN	int	image mode constant
BLEND_MODE_SOURCE_OUT	int	image mode constant
BLEND_MODE_XOR	int	image mode constant
FACE_DOWN	int	orientation constant
FACE_UP	int	orientation constant
INPUT_BORDERSTYLE_BEZEL	int	input border style constant
INPUT_BORDERSTYLE_LINE	int	input border style constant
INPUT_BORDERSTYLE_NONE	int	input border style constant
INPUT_BORDERSTYLE_ROUNDED	int	input border style constant
INPUT_BUTTONMODE_ALWAYS	int	input button mode constant
INPUT_BUTTONMODE_NEVER	int	input button mode constant
INPUT_BUTTONMODE_ONBLUR	int	input button mode constant
INPUT_BUTTONMODE_ONFOCUS	int	input button mode constant
KEYBOARD_APPEARANCE_ALERT	int	textfield keyboard appearance constant
KEYBOARD_APPEARANCE_DEFAULT	int	textfield keyboard appearance constant

KEYBOARD_ASCII	int	textfield keyboard constant
KEYBOARD_DEFAULT	int	textfield keyboard constant
KEYBOARD_EMAIL	int	textfield keyboard constant
KEYBOARD_NAMEPHONE_PAD	int	textfield keyboard constant
KEYBOARD_NUMBERS_PUNCTUATION	int	textfield keyboard constant
KEYBOARD_NUMBER_PAD	int	textfield keyboard constant
KEYBOARD_PHONE_PAD	int	textfield keyboard constant
KEYBOARD_URL	int	textfield keyboard constant
LANDSCAPE_LEFT	int	orientation (home button on left) constant
LANDSCAPE_RIGHT	int	orientation (home button on right) constant
NOTIFICATION_DURATION_LONG	int	Duration of the notification
NOTIFICATION_DURATION_SHORT	int	Duration of the notification
PICKER_TYPE_COUNT_DOWN_TIMER	int	picker type constant
PICKER_TYPE_DATE	int	picker type constant
PICKER_TYPE_DATE_AND_TIME	int	picker type constant
PICKER_TYPE_PLAIN	int	picker type constant
PICKER_TYPE_TIME	int	picker type constant
PORTRAIT	int	orientation (home button on bottom) constant

RETURNKEY_DEFAULT	int	textfield return key constant
RETURNKEY_DONE	int	textfield return key constant
RETURNKEY_EMERGENCY_CALL	int	textfield return key constant
RETURNKEY_GO	int	textfield return key constant
RETURNKEY_GOOGLE	int	textfield return key constant
RETURNKEY_JOIN	int	textfield return key constant
RETURNKEY_NEXT	int	textfield return key constant
RETURNKEY_ROUTE	int	textfield return key constant
RETURNKEY_SEARCH	int	textfield return key constant
RETURNKEY_SEND	int	textfield return key constant
RETURNKEY_YAHOO	int	textfield return key constant
TEXT_ALIGNMENT_CENTER	int	text align constant
TEXT_ALIGNMENT_LEFT	int	text align constant
TEXT_ALIGNMENT_RIGHT	int	text align constant
TEXT_AUTOCAPITALIZATION_ALL	int	text capitalization constant
TEXT_AUTOCAPITALIZATION_NONE	int	text capitalization constant
TEXT_AUTOCAPITALIZATION_SENTENCES	int	text capitalization constant
TEXT_AUTOCAPITALIZATION_WORDS	int	text capitalization constant

TEXT_VERTICAL_ALIGNMENT_BOTTOM	int	text vertical align constant
TEXT_VERTICAL_ALIGNMENT_CENTER	int	text vertical align constant
TEXT_VERTICAL_ALIGNMENT_TOP	int	text vertical align constant
UNKNOWN	int	orientation constant
UPSIDE_PORTRAIT	int	orientation (home button on top) constant

Events

This module has no events

Notes

Design

The UI module is broken down into 3 major area:

- **Views** - Views are containers that host visual elements such as controls or other views. Views can have their properties customized, such as their border color and radius, can fire events such as swipe events or touches, and can optionally contain a hierarchy or other views as children. In Titanium, most views are specialized to perform both a visual function and set of interaction behaviors such as Table View or Overflow View. Views are always named with the suffix `View`.
- **Controls** - controls, or sometimes referred as widgets, are visual elements such as sliders, buttons and switches. They provide a visual element which has a defined behavior and typically have special configuration and special events. Controls themselves are views and also inherit view's properties, functions and events.
- **Windows** - Windows are typically top-level visual constructs that are the main part of your interface. An application will always have at least one window and windows can take different shapes and sizes, can have display and interaction properties such as fullscreen or modal and can be customized, such as changing their opacity or background color. Windows themselves are views and also inherit view's properties, functions and events. There are a few specializations of Windows such as a Tab Group which offer additional behavior beyond the basic Window.

Titanium uses the Factory Pattern for constructing objects and a general naming pattern for APIs. For example, to construct an Alert Dialog, you call the method `Titanium.UI.createAlertDialog`. To create a `TextArea`, you call the method `Titanium.UI.createTextArea`. Once an object is created, it will be available until it goes out of scope.

Optimizations

UI objects are optimized by Titanium to not be realized into the drawing context and placed into the device UI surface until needed. That means that you can create UI objects, set their properties and add them to their hierarchy without much worry about memory or performance. When the native drawing surface needs to render a specific view or control, Titanium will automatically create the view as needed. Additionally, Titanium is optimized to also release memory once the view is no longer needed, on screen or in low memory situations. However, it's a good idea to help Titanium along in certain cases where you are no longer using objects. For example, you should call `close` on a `Window` instance when you are no longer using it. You can safely call `open` on the `Window` again to re-open it.

Global Context and Threading

Be careful with the objects that are created in `app.js` but only used once. Since the `app.js` context is global and generally is not garbage collected until the application exits, you should think about the design of your application as it relates to this fact. `Window` objects that are opened up with the `url` property to another Javascript file contain a nice way to decompose your application into smaller units. The other benefit is that when a `Window` is closed, its resources can be immediately cleaned up, saving resources such as memory and CPU. Additionally, `Window` objects run in a separate Javascript context and Thread. While all UI processing is done on the main UI thread, other processing inside a `Window` or the `app.js` that does not have UI interaction will run in its own thread.

Portability

Titanium components are designed to be portable across as many platforms as it supports. However, there are cases where a device either does not support a specific feature or capability or where it supports additional functionality. For cases where the device OS supports capabilities that other platforms do not, we attempt to place those capabilities in a separate namespace, such as Titanium.UI.iPhone. However, in cases where the control is in a common namespace and supports additional features, we continue to place that functionality directly on the object.

Titanium.UI.2DMatrix

object of Titanium.UI

The 2DMatrix is created by Titanium.UI.create2DMatrix. The 2D Matrix is an object for holding values for an affine transformation matrix. A 2D matrix is used to rotate, scale, translate, or skew the objects in a two-dimensional space. A 2D matrix is represented by a 3 by 3 matrix. Because the third column is always (0,0,1), the data structure contains values for only the first two columns.

Methods

Name	Description
<code>addEventListener</code>	add an event listener for the instance to receive view triggered events
<code>fireEvent</code>	fire a synthesized event to the views listener
<code>invert</code>	Returns a matrix constructed by inverting an existing matrix
<code>multiply</code>	Returns a matrix constructed by combining two existing matrix.
<code>removeEventListener</code>	remove a previously added event listener
<code>rotate</code>	Returns a matrix constructed by rotating an existing matrix
<code>scale</code>	Returns a matrix constructed by scaling an existing matrix
<code>translate</code>	Returns a matrix constructed by translating an existing matrix

Properties

This object has no properties

Events

This object has no events

Notes

You create an `identity matrix` by creating a 2D Matrix with an empty constructor.

Titanium.UI.2DMatrix.addEventListener

function of Titanium.UI.2DMatrix

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.2DMatrix.fireEvent

function of Titanium.UI.2DMatrix

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.2DMatrix.invert

function of Titanium.UI.2DMatrix

Returns a matrix constructed by inverting an existing matrix

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.2DMatrix.multiply

function of Titanium.UI.2DMatrix

Returns a matrix constructed by combining two existing matrix.

Arguments

Name	Type	Description
t2	object	The second matrix. This matrix is concatenated to the matrix instance against which the function is invoked. The result of this function is the first matrix multiplied by the second matrix. You might perform several multiplications in order to create a single matrix that contains the cumulative effects of several transformations. Note that matrix operations are not commutative - the order in which you concatenate matrices is important. That is, the result of multiplying matrix t1 by matrix t2 does not necessarily equal the result of multiplying matrix t2 by matrix t1.

Return Type

object

Titanium.UI.2DMatrix.removeEventListener

function of Titanium.UI.2DMatrix

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.2DMatrix.rotate

function of Titanium.UI.2DMatrix

Returns a matrix constructed by rotating an existing matrix

Arguments

Name	Type	Description
angle	float	The angle, in degrees, by which to rotate the matrix. A positive value specifies counterclockwise rotation and a negative value specifies clockwise rotation.

Return Type

object

Titanium.UI.2DMatrix.scale

function of Titanium.UI.2DMatrix

Returns a matrix constructed by scaling an existing matrix

Arguments

Name	Type	Description
sx	float	The value by which to scale x values of the matrix
sy	float	The value by which to scale y values of the matrix

Return Type

object

Titanium.UI.2DMatrix.translate

function of Titanium.UI.2DMatrix

Returns a matrix constructed by translating an existing matrix

Arguments

Name	Type	Description
tx	float	The value by which to move x values with the matrix
ty	float	The value by which to move y values with the matrix

Return Type

object

Titanium.UI.3DMatrix

object of Titanium.UI

The 2DMatrix is created by Titanium.UI.create2DMatrix. The 2D Matrix is an object for holding values for an affine transformation matrix. A 2D matrix is used to rotate, scale, translate, or skew the objects in a three-dimensional space. A 3D matrix is represented by a 4 by 4 matrix. Because the forth column is always (0,0,1), the data structure contains values for only the first three columns.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
invert	Returns a matrix constructed by inverting an existing matrix
multiply	Returns a matrix constructed by combining two existing matrix.
removeEventListener	remove a previously added event listener
rotate	Returns a matrix constructed by rotating an existing matrix

scale	Returns a matrix constructed by scaling an existing matrix
translate	Returns a matrix constructed by translating an existing matrix

Properties

This object has no properties

Events

This object has no events

Notes

You create an `identity` matrix by creating a 3D Matrix with an empty constructor.

Titanium.UI.3DMatrix.addEventListener

function of Titanium.UI.3DMatrix

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.3DMatrix.fireEvent

function of Titanium.UI.3DMatrix

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.3DMatrix.invert

function of Titanium.UI.3DMatrix

Returns a matrix constructed by inverting an existing matrix

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.3DMatrix.multiply

function of Titanium.UI.3DMatrix

Returns a matrix constructed by combining two existing matrix.

Arguments

Name	Type	Description
t2	object	The second matrix. This matrix is concatenated to the matrix instance against which the function is invoked. The result of this function is the first matrix multiplied by the second matrix. You might perform several multiplications in order to create a single matrix that contains the cumulative effects of several transformations. Note that matrix operations are not commutative - the order in which you concatenate matrices is important. That is, the result of multiplying matrix t1 by matrix t2 does not necessarily equal the result of multiplying matrix t2 by matrix t1.

Return Type

object

Titanium.UI.3DMatrix.removeEventListener

function of Titanium.UI.3DMatrix

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.3DMatrix.rotate

function of Titanium.UI.3DMatrix

Rotates a matrix around the specified axis.

Returns a matrix constructed by rotating an existing matrix

Arguments

Name	Type	Description
angle	float	The angle, in degrees, by which to rotate the matrix. A positive value specifies counterclockwise rotation and a negative value specifies clockwise rotation.
x	float	The x part of the vector about which to rotate
y	float	The y part of the vector about which to rotate
z	float	The z part of the vector about which to rotate

Return Type

object

Titanium.UI.3DMatrix.scale

function of Titanium.UI.3DMatrix

Returns a matrix constructed by scaling an existing matrix

Arguments

Name	Type	Description
sx	float	The value by which to scale x values of the matrix
sy	float	The value by which to scale y values of the matrix
sz	float	The value by which to scale z values of the matrix

Return Type

object

Titanium.UI.3DMatrix.translate

function of Titanium.UI.3DMatrix

Returns a matrix constructed by translating an existing matrix

Arguments

Name	Type	Description
tx	float	The value by which to move x values with the matrix
ty	float	The value by which to move y values with the matrix
tz	float	The value by which to move z values with the matrix

Return Type

object

Titanium.UI.ActivityIndicator

object of Titanium.UI

An Activity Indicator is created by the method Titanium.UI.createActivityIndicator. An activity indicator can be used to show the progress of an operation in the UI to let the user know some action is taking place.

Methods

Name	Description
hide	call hide to make the activity indicator hidden and stop spinning
show	call show to make the activity indicator visible and start spinning

Properties

Name	Type	Description
color	string	the color of the message label
font	object	the font object for the activity message label
message	string	the activity message label text
style	int	the style constant of the activity indicator

Events

This object has no events

Code Examples

Simple Activity Indicator

In this example, we create a basic activity indicator and start it.

```
var actInd = Titanium.UI.createActivityIndicator({
    height:50,
    width:10
});
actInd.show();
```

Notes

For iPhone, the style can be set from the constants Titanium.UI.iPhone.ActivityIndicatorStyle.

Titanium.UI.ActivityIndicator.hide

function of Titanium.UI.ActivityIndicator

call hide to make the activity indicator hidden and stop spinning

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.ActivityIndicator.show

function of Titanium.UI.ActivityIndicator

call show to make the activity indicator visible and start spinning

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.AlertDialog

object of Titanium.UI

The Alert Dialog is created by Titanium.UI.createAlertDialog and allows you to show a modal application dialog.

Methods

Name	Description
addEventListerner	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
hide	cause the dialog to become hidden
removeEventListerner	remove a previously added event listener
show	cause the dialog to become visible

Properties

Name	Type	Description

buttonNames	array	array of button names as strings
cancel	int	an index to indicate which button should be the cancel button.
message	string	the message of the dialog
title	string	the title of the dialog

Events

Name	Description								
click	<p>fired when a button in the dialog is clicked</p> <p>Event properties</p> <table border="1"> <tr> <td>cancel</td><td>boolean to indicate if the cancel button was pressed</td></tr> <tr> <td>index</td><td>the button index that was pressed</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	cancel	boolean to indicate if the cancel button was pressed	index	the button index that was pressed	source	the source object that fired the event	type	the name of the event fired
cancel	boolean to indicate if the cancel button was pressed								
index	the button index that was pressed								
source	the source object that fired the event								
type	the name of the event fired								

Code Examples

Simple Alert Dialog

In this example, we show a simple alert dialog.

```
var alertDialog = Titanium.UI.createAlertDialog({
    title: 'Hello',
    message: 'You got mail',
    buttonNames: ['OK','Doh!']
});
alertDialog.show();
```

Notes

the global method `alert` is aliased to this object and can be invoked with a single message. For example:

```
alert("this is a message");
```

Note: you should be careful not to call `alert` more than once while a pending alert is visible.

On iOS 4, alert dialogs will automatically be cancelled upon application suspend.

Titanium.UI.AlertDialog.addEventListerner

function of Titanium.UI.AlertDialog

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.AlertDialog.fireEvent

function of Titanium.UI.AlertDialog

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.AlertDialog.hide

function of Titanium.UI.AlertDialog

cause the dialog to become hidden

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.AlertDialog.removeEventListener

function of Titanium.UI.AlertDialog

remove a previously added event listener

Arguments

Name	Type	Description
------	------	-------------

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.AlertDialog.show

function of Titanium.UI.AlertDialog

cause the dialog to become visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Android

submodule of Titanium.UI

The Android specific UI capabilities. All properties, methods and events in this namespace will only work on Android related devices.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
LINKIFY_ALL	int	Auto link all supported types.
LINKIFY_EMAIL_ADDRESSES	int	Auto link email addresses.
LINKIFY_MAP_ADDRESSES	int	Auto link physical addresses.
LINKIFY_PHONE_NUMBERS	int	Auto link phone numbers.

LINKIFY_WEB_URLS	int	Auto link web urls.
SOFT_KEYBOARD_DEFAULT_ON_FOCUS	int	Use Android default soft keyboard handling. Same as not specifying a value.
SOFT_KEYBOARD_HIDE_ON_FOCUS	int	Request hide soft keyboard on focus. Note: OS can override request.
SOFT_KEYBOARD_SHOW_ON_FOCUS	int	Request show soft keyboard on focus. Note: OS can override request.

Events

This module has no events

Titanium.UI.Android.OptionMenu

submodule of Titanium.UI.Android

The menu that appears at the lower portion of the display when the device's *Menu* button is pressed.

Objects

Name	Description
Titanium.UI.Android.OptionMenu.Menu	A Menu allows you to provide a selection of options that appear when the menu button is pressed on device. The Menu is created by the method Titanium.UI.Android.OptionMenu.createMenu
Titanium.UI.Android.OptionMenu.MenuItem	A MenuItem allows you provide a selectable option with a graphic and title. The Menu Item is created by the method Titanium.UI.Android.OptionMenu.createMenuItem

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
createMenu	create and return an instance of Titanium.UI.Android.OptionMenu.Menu
createMenuItem	create and return an instance of Titanium.UI.Android.OptionMenu.MenuItem
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
menu	object	The Titanium.UI.Android.OptionMenu.Menu for this context.

Events

This module has no events

Code Examples

Menu

```
var menu = Titanium.UI.Android.OptionMenu.createMenu();

var item1 = Titanium.UI.Android.OptionMenu.createMenuItem({
    title : 'Item 1',
    icon : '/images/item1.png'
});

item1.addEventListener('click', function(){
    Ti.UI.createAlertDialog({ title : 'You clicked Item 1'}).show();
});

var item2 = Titanium.UI.Android.OptionMenu.createMenuItem({
    title : 'Refresh',
    icon : '/images/refresh.png'
});
item2.addEventListener('click', function(){
    Ti.UI.createAlertDialog({ title : 'You clicked Refresh'}).show();
});

menu.add(item1);
menu.add(item2);

// Set the menu on the current heavyweight window. A heavyweight window maps to an Android
// Activity. To create a heavyweight window, specify one or more of [**fullscreen**, **navBarHidden**,
Titanium.UI.Android.OptionMenu.setMenu(menu);
```

Titanium.UI.Android.OptionMenu.Menu

object of Titanium.UI.Android.OptionMenu

A Menu allows you to provide a selection of options that appear when the **menu** button is pressed on device. The Menu is created by the method Titanium.UI.Android.OptionMenu.createMenu

Methods

Name	Description
add	add a menu item. Items will be displayed in the menu based on insertion order.
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener

Properties

This object has no properties

Events

This object has no events

Titanium.UI.Android.OptionMenu.Menu.add

function of Titanium.UI.Android.OptionMenu.Menu

add a menu item. Items will be displayed in the menu based on insertion order.

Arguments

Name	Type	Description
menuitem	object	the menu item to add. Must be a Titanium.UI.Android.OptionMenu.MenuItem.

Return Type

void

Titanium.UI.Android.OptionMenu.Menu.addEventListener

function of Titanium.UI.Android.OptionMenu.Menu

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Android.OptionMenu.Menu.fireEvent

function of Titanium.UI.Android.OptionMenu.Menu

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

Titanium.UI.Android.OptionMenu.Menu.removeEventListener

function of Titanium.UI.Android.OptionMenu.Menu

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Android.OptionMenu.MenuItem

object of Titanium.UI.Android.OptionMenu

A MenuItem allows you provide a selectable option with a graphic and title. The Menu Item is created by the method Titanium.UI.Android.OptionMenu.createMenuItem

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
icon	string	the path to the image to display on the menu option
title	string	the text for the menu option

Events

Name	Description
click	fired when the menu option is selected

Event properties

source	the source object that fired the event
type	the name of the event fired

Titanium.UI.Android.OptionMenu.MenuItem.addEventListener

function of Titanium.UI.Android.OptionMenu.MenuItem

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Android.OptionMenu.MenuItem.fireEvent

function of Titanium.UI.Android.OptionMenu.MenuItem

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Android.OptionMenu.MenuItem.removeEventListener

function of Titanium.UI.Android.OptionMenu.MenuItem

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Android.OptionMenu.addEventListener

function of Titanium.UI.Android.OptionMenu

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Android.OptionMenu.createMenu

function of Titanium.UI.Android.OptionMenu

create and return an instance of Titanium.UI.Android.OptionMenu.Menu

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.Android.OptionMenu.Menu

Return Type

object

Titanium.UI.Android.OptionMenu.createMenuItem

function of Titanium.UI.Android.OptionMenu

create and return an instance of Titanium.UI.Android.OptionMenu.MenuItem

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.Android.OptionMenu.MenuItem

Return Type

object

Titanium.UI.Android.OptionMenu.fireEvent

function of Titanium.UI.Android.OptionMenu

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Android.OptionMenu.removeEventListener

function of Titanium.UI.Android.OptionMenu

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Android.addEventListener

function of Titanium.UI.Android

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Android.fireEvent

Titanium.UI.fireEvent

function of Titanium.UI.Android
fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Android.removeEventListener

function of Titanium.UI.Android
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Animation

object of Titanium.UI

The Animation object is used for specifying lower-level animation properties and more low-level control of events during an animation. The Animation is created by the method Titanium.UI.createAnimation.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
autoreverse	boolean	the property specifies if the animation should be replayed in reverse upon completion

backgroundColor	string	value of the backgroundColor property to change during animation
bottom	float	value of the bottom property to change during animation
center	object	value of the center property to change during animation
color	string	value of the color property to change during animation
curve	int	the curve of the animation
delay	float	the duration of time in milliseconds before starting the animation
duration	float	the duration of time in milliseconds to perform the animation
height	float	value of the height property to change during animation
left	float	value of the left property to change during animation
opacity	float	value of the opacity property to change during animation
opaque	boolean	value of the opaque property to change during animation
repeat	int	the number of times the animation should be performed
right	float	value of the right property to change during animation
top	float	value of the top property to change during animation
transform	object	value of the transform property to change during animation
transition	int	during a transition animation, this is the constant to the type of transition to use
visible	boolean	value of the visible property to change during animation
width	float	value of the width property to change during animation

zIndex	int	value of the zIndex property to change during animation
---------------	-----	---

Events

Name	Description				
complete	fired when the animation completes Event properties <table border="1"> <tr> <td>source</td> <td>the source object that fired the event</td> </tr> <tr> <td>type</td> <td>the name of the event fired</td> </tr> </table>	source	the source object that fired the event	type	the name of the event fired
source	the source object that fired the event				
type	the name of the event fired				
start	fired when the animation starts Event properties <table border="1"> <tr> <td>source</td> <td>the source object that fired the event</td> </tr> <tr> <td>type</td> <td>the name of the event fired</td> </tr> </table>	source	the source object that fired the event	type	the name of the event fired
source	the source object that fired the event				
type	the name of the event fired				

Code Examples

Animation applied to a view

Create a simple animation and apply it to the view. In this example, the view will animate from red to black to orange over 2 seconds.

```
var view = Titanium.UI.createView({
    backgroundColor:'red'
});
var animation = Titanium.UI.createAnimation();
animation.backgroundColor = 'black';
animation.duration = 1000;
animation.addEventListener('complete',function()
{
    animation.removeEventListener('complete',this);
    animation.backgroundColor = 'orange';
    view.animate(animation);
});
view.animate(animation);
```

Titanium.UI.Animation.addEventListener

function of Titanium.UI.Animation

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event

name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Animation.fireEvent

function of Titanium.UI.Animation

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Animation.removeEventListener

function of Titanium.UI.Animation

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Button

object of Titanium.UI

A Button is created by the method Titanium.UI.createButton.

Methods

Name	Description
add	add a child to the view hierarchy

addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the buttons background color
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	url to a button image that is drawn as the background of the button when the button is in the disabled state
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	url to a button image that is drawn as the background of the button
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	url to a button image that is drawn as the background of the button when the button is in the selected state
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is

backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
color	string	the foreground color of the button text
enabled	boolean	boolean that indicates if the button is enabled or not
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
font	object	the font properties of the button
height	float,string	property for the view height. can be either float value or a string of the width.
image	string	the image to display on the button to the left of the title
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
selectedColor	string	the selected color of the button text when the button is in the selected state
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
style	int	style constant for the type of button
title	string	button title

top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
click	<p>fired when the user presses the button</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
doubletap	<p>fired when the device detects a double tap against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										

singletap	fired when the device detects a single tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
swipe	fired when the device detects a swipe (left or right) against the view
Event properties	
direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
touchcancel	fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
touchend	fired when a touch event is completed
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Simple Button Example

```
var button = Titanium.UI.createButton({
    title: 'Hello'
});
button.addEventListener('click',function(e)
{
    Titanium.API.info("You clicked the button");
});
```

Titanium.UI.Button.add

function of Titanium.UI.Button
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.Button.addEventListener

function of Titanium.UI.Button

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Button.animate

function of Titanium.UI.Button

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.Button.fireEvent

function of Titanium.UI.Button

fire a synthesized event to the views listener

Arguments

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Button.hide

function of Titanium.UI.Button

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Button.remove

function of Titanium.UI.Button

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.Button.removeEventListener

function of Titanium.UI.Button

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Button.show

function of Titanium.UI.Button
make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Button.toImage

function of Titanium.UI.Button
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.ButtonBar

object of Titanium.UI

A Button Bar is created by the method Titanium.UI.createButtonBar.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener

show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the button bar
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.

center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
index	int	the selected index
labels	array	the array of labels for the button bar. each object should have the properties title , image , width and enabled .
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
style	int	the style of the button bar
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description				
click	fired when a button on the button bar is clicked Event properties <table border="1"> <tr> <td>globalPoint</td> <td>a dictionary with properties x and y describing the point of the event in screen coordinates</td> </tr> <tr> <td>index[int]</td> <td>the index of the button that was clicked</td> </tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	index[int]	the index of the button that was clicked
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates				
index[int]	the index of the button that was clicked				

source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

dblclick fired when the device detects a double click against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

singletap fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event

type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofinger tap fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Simple 3 button button bar

```
var bb1 = Titanium.UI.createButtonBar({
    labels:['One', 'Two', 'Three'],
    backgroundColor:'#336699',
    top:50,
    style:Titanium.UI.iPhone.SystemButtonStyle.BAR,
    height:25,
    width:200
});
win.add(bb1);
```

Notes

For iPhone, the style constants are available in the constants defined in Titanium.UI.iPhone.SystemButtonStyle.

Titanium.UI.ButtonBar.add

function of Titanium.UI.ButtonBar
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.ButtonBar.addEventListener

function of Titanium.UI.ButtonBar
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.ButtonBar.animate

function of Titanium.UI.ButtonBar

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.ButtonBar.fireEvent

function of Titanium.UI.ButtonBar

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.ButtonBar.hide

function of Titanium.UI.ButtonBar

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.ButtonBar.remove

function of Titanium.UI.ButtonBar

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.ButtonBar.removeEventListener

function of Titanium.UI.ButtonBar

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.ButtonBar.show

function of Titanium.UI.ButtonBar

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.ButtonBar.toImage

function of Titanium.UI.ButtonBar
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.CoverFlowLayout

object of Titanium.UI

The Cover Flow view is container for showing animated, three dimensional images in a nice UI. The Cover Flow view is created by the method Titanium.UI.createCoverFlowLayout.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
setURL	change an image for a index
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundImage	string	the full URL or local path of the image (Anchored)

backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
images	array	array of images to display in the view
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.

selected	int	index to make selected
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
change	<p>fired when the user changes the image using a gesture</p> <p>Event properties</p> <table border="1"> <tr> <td>index</td><td>the index of the image that is now visible</td></tr> <tr> <td>previous</td><td>the previous index of the image that was visible</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	index	the index of the image that is now visible	previous	the previous index of the image that was visible	source	the source object that fired the event	type	the name of the event fired		
index	the index of the image that is now visible										
previous	the previous index of the image that was visible										
source	the source object that fired the event										
type	the name of the event fired										
click	<p>fired when the user clicks on an image</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>index</td><td>the index of the image that is now visible</td></tr> <tr> <td>previous</td><td>the previous index of the image that was visible</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	index	the index of the image that is now visible	previous	the previous index of the image that was visible	source	the source object that fired the event	type	the name of the event fired
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
index	the index of the image that is now visible										
previous	the previous index of the image that was visible										
source	the source object that fired the event										
type	the name of the event fired										

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

dblclick	fired when the device detects a double click against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap	fired when the device detects a double tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

singletap	fired when the device detects a single tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe	fired when the device detects a swipe (left or right) against the view
Event properties	
direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates

y	the y point of the event, in receiving view coordinates
----------	---

touchcancel fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples**Simple 3 image cover flow example**

Create a rounded view.

```
var view = Titanium.UI.createCoverFlowLayout({
    images: ['a.png', 'b.png', 'c.png'],
    backgroundColor: '#000'
});
window.add(view);
```

Titanium.UI.CoverFlowLayout.add

function of Titanium.UI.CoverFlowLayout

add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.CoverFlowLayout.addEventListener

function of Titanium.UI.CoverFlowLayout

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.CoverFlowView.animate

function of Titanium.UI.CoverFlowView

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.CoverFlowView.fireEvent

function of Titanium.UI.CoverFlowView

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.CoverFlowView.hide

function of Titanium.UI.CoverFlowView

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.CoverFlowView.remove

function of Titanium.UI.CoverFlowView

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hiearchy

Return Type

void

Titanium.UI.CoverFlowView.removeEventListener

function of Titanium.UI.CoverFlowView

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.CoverFlowView.setURL

function of Titanium.UI.CoverFlowView

change an image for a index

Arguments

Name	Type	Description
index	int	index to change
url	string	url to the new image

Return Type

void

Titanium.UI.CoverFlowView.show

function of Titanium.UI.CoverFlowView

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.CoverFlowView.toImage

function of Titanium.UI.CoverFlowView
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.DashboardItem

object of Titanium.UI

An item that represents a visual icon in the Titanium.UI.DashboardView. The Dashboard Item is created by the method Titanium.UI.createDashboardItem.

Methods

Name	Description
addEventListenerr	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
badge	int	the badge value or 0 to remove the badge
canDelete	boolean	a boolean to indicate if this item can be deleted when it edit mode
image	string	the URL to the image

selectedImage	string	the URL to the image to display when the item is depressed (clicked)
----------------------	--------	--

Events

Name	Description								
click	<p>fired when a Titanium.UI.DashboardItem is clicked</p> <p>Event properties</p> <table border="1"> <tr> <td>item</td><td>the Titanium.UI.DashboardItem that was clicked</td></tr> <tr> <td>location</td><td>the location object with the properties <code>x</code> and <code>y</code> of where in the parent view the click was</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	item	the Titanium.UI.DashboardItem that was clicked	location	the location object with the properties <code>x</code> and <code>y</code> of where in the parent view the click was	source	the source object that fired the event	type	the name of the event fired
item	the Titanium.UI.DashboardItem that was clicked								
location	the location object with the properties <code>x</code> and <code>y</code> of where in the parent view the click was								
source	the source object that fired the event								
type	the name of the event fired								
delete	<p>fired when a Titanium.UI.DashboardItem is deleted during editings</p> <p>Event properties</p> <table border="1"> <tr> <td>item</td><td>the Titanium.UI.DashboardItem that was deleted</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	item	the Titanium.UI.DashboardItem that was deleted	source	the source object that fired the event	type	the name of the event fired		
item	the Titanium.UI.DashboardItem that was deleted								
source	the source object that fired the event								
type	the name of the event fired								
move	<p>fired when a Titanium.UI.DashboardItem is moved during editing</p> <p>Event properties</p> <table border="1"> <tr> <td>item</td><td>the Titanium.UI.DashboardItem that was moved</td></tr> <tr> <td>items</td><td>the pending data array in the new order (uncommitted)</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	item	the Titanium.UI.DashboardItem that was moved	items	the pending data array in the new order (uncommitted)	source	the source object that fired the event	type	the name of the event fired
item	the Titanium.UI.DashboardItem that was moved								
items	the pending data array in the new order (uncommitted)								
source	the source object that fired the event								
type	the name of the event fired								

Titanium.UI.DashboardItem.addEventListener

function of Titanium.UI.DashboardItem

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.DashboardItem.fireEvent

function of Titanium.UI.DashboardItem

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.DashboardItem.removeEventListerner

function of Titanium.UI.DashboardItem

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListerner

Return Type

void

Titanium.UI.DashboardView

object of Titanium.UI

The Dashboard View provides a view that supports the ability to have Springboard-like view of icons which can be reordered by dragging and can contain multiple pages of icons in a scrollable view. The Dashboard View is created by the method Titanium.UI.createDashboardView.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
startEditing	put the dashboard in edit mode
stopEditing	cancel editing
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image

backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
data	array	an array of Titanium.UI.DashboardItem objects to display in the view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
wobble	boolean	true (default) to wobble during edit, false to disable wobble effect

zIndex	int	the z index position relative to other sibling views
---------------	-----	--

Events

Name	Description														
click	<p>fired when a Titanium.UI.DashboardItem is clicked</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>item</td><td>the Titanium.UI.DashboardItem that was clicked</td></tr> <tr> <td>location</td><td>the location object with the properties x and y of where in the parent view the click was</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	item	the Titanium.UI.DashboardItem that was clicked	location	the location object with the properties x and y of where in the parent view the click was	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates														
item	the Titanium.UI.DashboardItem that was clicked														
location	the location object with the properties x and y of where in the parent view the click was														
source	the source object that fired the event														
type	the name of the event fired														
x	the x point of the event in receiving view coordinates														
y	the y point of the event, in receiving view coordinates														
commit	<p>fired when editing ends</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired										
source	the source object that fired the event														
type	the name of the event fired														
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates				
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates														
source	the source object that fired the event														
type	the name of the event fired														
x	the x point of the event in receiving view coordinates														
y	the y point of the event, in receiving view coordinates														
delete	<p>fired when a Titanium.UI.DashboardItem is deleted during editings</p> <p>Event properties</p> <table border="1"> <tr> <td>item</td><td>the Titanium.UI.DashboardItem that was deleted</td></tr> </table>	item	the Titanium.UI.DashboardItem that was deleted												
item	the Titanium.UI.DashboardItem that was deleted														

source	the source object that fired the event
type	the name of the event fired

doubletap fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

edit

fired when editing begins

Event properties

source	the source object that fired the event
type	the name of the event fired

move

fired when a Titanium.UI.DashboardItem is moved during editing

Event properties

item	the Titanium.UI.DashboardItem that was moved
items	the pending data array in the new order (uncommitted)
source	the source object that fired the event
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart

fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Titanium.UI.DashboardView.addfunction of Titanium.UI.DashboardView
add a child to the view hierarchy**Arguments**

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.DashboardView.addEventListenerfunction of Titanium.UI.DashboardView
add an event listener for the instance to receive view triggered events**Arguments**

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.DashboardView.animate

function of Titanium.UI.DashboardView

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.DashboardView.fireEvent

function of Titanium.UI.DashboardView

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.DashboardView.hide

function of Titanium.UI.DashboardView

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.DashboardView.remove

function of Titanium.UI.DashboardView

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.DashboardView.removeEventListener

function of Titanium.UI.DashboardView

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.DashboardView.show

function of Titanium.UI.DashboardView

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.DashboardView.startEditing

function of Titanium.UI.DashboardView

put the dashboard in edit mode

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.DashboardView.stopEditing

function of Titanium.UI.DashboardView

cancel editing

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.DashboardView.toImage

function of Titanium.UI.DashboardView

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.EmailDialog

object of Titanium.UI

The Email Dialog is created by Titanium.UI.createEmailDialog and allows you to send in application emails on behalf of the application user.

Methods

Name	Description
addAttachment	add an attachment to the email. the attachment can either be a Blob or File object. (Note: Android will only accept one attachment at this time.)
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener

open	open the email dialog. the email dialog itself is a modal window
removeEventListener	remove a previously added event listener

Properties

Name	Type	Description
CANCELLED	int	constant for the CANCELLED status result
FAILED	int	constant for the FAILED status result
SAVED	int	constant for the SAVED status result
SENT	int	constant for the SENT status result
barColor	string	the bar color of the email dialog window when opened
bccRecipients	array	array of email BCC: recipients
ccRecipients	array	array of email CC: recipients
html	boolean	boolean to indicate whether the email messageBody should be sent as HTML content type. defaults to false. (Android note: you should not use html messages in 1.6: after that is okay.)
messageBody	string	the email message body
subject	string	the subject line for the email
toRecipients	array	array of email recipients

Events

Name	Description		
complete	fired when the email dialog has completed sending the email		
	Event properties <table border="1"> <tr> <td>error</td> <td>string message of the error or null if successfully sent</td> </tr> </table>	error	string message of the error or null if successfully sent
error	string message of the error or null if successfully sent		

result	result status either as SENT, SAVED, CANCELLED or FAILED. (Note: Android result will be SENT even if user discards or saves the message. SAVED and CANCELLED are not supported on Android.)
source	the source object that fired the event
success	boolean to indicate if the email was successfully sent
type	the name of the event fired

Code Examples

Simple Email Dialog with Attachment

In this example, we send an email with a file attachment.

```
var emailDialog = Titanium.UI.createEmailDialog()
emailDialog.subject = "Hello from Titanium";
emailDialog.toRecipients = ['foo@yahoo.com'];
emailDialog.messageBody = 'Appcelerator Titanium Rocks!';
var f = Ti.Filesystem.getFile('cricket.wav');
emailDialog.addAttachment(f);
emailDialog.open();
```

Titanium.UI.EmailDialog.addAttachment

function of Titanium.UI.EmailDialog

add an attachment to the email. the attachment can either be a Blob or File object. (Note: Android will only accept one attachment at this time.)

Arguments

Name	Type	Description
attachment	object	attachment object as either a Blob or File object

Return Type

void

Titanium.UI.EmailDialog.addEventListener

function of Titanium.UI.EmailDialog

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.EmailDialog.fireEvent

function of Titanium.UI.EmailDialog

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.EmailDialog.open

function of Titanium.UI.EmailDialog

open the email dialog. the email dialog itself is a modal window

Arguments

Name	Type	Description
properties	object	object of animation properties. pass <code>animated</code> property (as boolean) to indicate if the dialog should be animated on open.

Return Type

void

Titanium.UI.EmailDialog.removeEventListener

function of Titanium.UI.EmailDialog

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

Titanium.UI.ImageView

object of Titanium.UI

An Image View is used to display an image or a series of images in an animation. The Image View is created by the method Titanium.UI.createImageView.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
pause	pause a started animation.
remove	remove a previously add view from the view hiearchy
removeEventListener	remove a previously added event listener
show	make the view visible
start	start the image animation. this method only works if you set multiple images
stop	stop a started animation and reset the index to the first image
toBlob	return the image as a Blob object
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
animating	boolean	readonly boolean to indicate if the animation is animating
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)

backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
canScale	boolean	allow image to scale (Android)
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
defaultImage	string	url to the default image to display while loading a remote image
duration	float	amount of time in milliseconds to animate one cycle
enableZoomControls	boolean	enable zoom controls on Android. Default is true for backward compatibility. (1.3.0)
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)

height	float	height of the image display
image	object	image to display either as string url, Blob or File
images	array	array of images (either as string url, Blob or File) to display in an animation
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
paused	boolean	readonly boolean to indicate if the animation is paused
preventDefaultImage	boolean	boolean to indicate if the default image should be displaying while loading a remote image
repeatCount	int	number of times to repeat the image animation
reverse	boolean	boolean to indicate if the animation should happen in reverse (from last to first)
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
url	string	url to the image to display (NOTE: this property is deprecated. use image instead)
visible	boolean	a boolean of the visibility of the view
width	float	width of the image display
zIndex	int	the z index position relative to other sibling views

Events

Name	Description

change

fired for each frame changed during an animation

Event properties

index	the index of the image frame being displayed
source	the source object that fired the event
type	the name of the event fired

click

fired when the device detects a click (longer than touch) against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

dblclick

fired when the device detects a double click against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap

fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

load

fired when either the initial image and/or all of the images in an animation are loaded

Event properties

source	the source object that fired the event
state	either <code>url</code> to indicate the url property is loaded or <code>images</code> when all the images are loaded
type	the name of the event fired

singletap	fired when the device detects a single tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

start	fired when the animation starts
Event properties	
source	the source object that fired the event
type	the name of the event fired

stop	fired when the animation stops
Event properties	
source	the source object that fired the event
type	the name of the event fired

swipe	fired when the device detects a swipe (left or right) against the view
Event properties	
direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel	fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.
--------------------	--

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart

fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Basic Image View

In this example, we create a simple image view:

```
var image = Titanium.UI.createImageView({url:'myimage.png'});
view.add(image);
```

Notes

If you specify a width and/or height property on the image, the image will be scaled to fit into this space if the image is larger.

Titanium.UI.ImageView.add

function of Titanium.UI.ImageView
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.ImageView.addEventListener

function of Titanium.UI.ImageView
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.ImageView.animate

function of Titanium.UI.ImageView

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.ImageView.fireEvent

function of Titanium.UI.ImageView

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.ImageView.hide

function of Titanium.UI.ImageView

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.ImageView.pause

function of Titanium.UI.ImageView

pause a started animation.

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.ImageView.remove

function of Titanium.UI.ImageView

remove a previously add view from the view hiearchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hiearchy

Return Type

void

Titanium.UI.ImageView.removeEventListener

function of Titanium.UI.ImageView

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.ImageView.show

function of Titanium.UI.ImageView

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.ImageView.start

function of Titanium.UI.ImageView

start the image animation. this method only works if you set multiple images

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.ImageView.stop

function of Titanium.UI.ImageView

stop a started animation and reset the index to the first image

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.ImageView.toBlob

function of Titanium.UI.ImageView

return the image as a Blob object

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.ImageView.toImage

function of Titanium.UI.ImageView

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.Label

object of Titanium.UI

A Label is created by the method Titanium.UI.createLabel.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
autoLink	int	One of Ti.UI.Android.LINKIFY constants. Automatically create clickable links for the specified type. (Android Only)
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type startPoint endPoint startRadius endRadius backfillStart backfillEnd colors

		type,startUnit,endUnit,startRadius,endRadius,backInStart,backInEnd,colors.
backgroundImage	string	background image for the label
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundPaddingBottom	int	the number of pixels to extend the background image past the label on the bottom
backgroundPaddingLeft	int	the number of pixels to extend the background image past the label on the left
backgroundPaddingRight	int	the number of pixels to extend the background image past the label on the right
backgroundPaddingTop	int	the number of pixels to extend the background image past the label on the top
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
color	string	the color of the label
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
font	object	the label font object properties

the label font object properties

height	float,string	property for the view height. can be either float value or a string of the width.
highlightedColor	string	the color of the label when in the highlighted state
html	string	simple html formatting. (Android Only)
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
minimumFontSize	int	the minimum size of the font when the font is sized based on the contents. Enables font scaling to fit and forces the label content to be limited to a single line
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
shadowColor	string	the text shadow color
shadowOffset	object	the shadow offset as a dictionary with the properties x and y
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
text	string	the text of the label
textAlign	string,int	the alignment constant or string value such as left , center or right
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description
click	fired when the device detects a click (longer than touch) against the view

click fired when the device detects a click (longer than touch) against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

dblclick

fired when the device detects a double click against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap

fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Basic Label

Create a label with a nice text shadow, 48px font that's aligned `center` and height `auto`.

```
var l2 = Titanium.UI.createLabel({
  text:'Appcelerator',
  height:'auto',
  width:'auto',
  shadowColor:'#aaa',
  shadowOffset:{x:5,y:5},
  color:'#900',
  font:{fontSize:48},
  textAlign:'center'
});
```

Titanium.UI.Label.add

function of `Titanium.UI.Label`
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.Label.addEventListener

function of Titanium.UI.Label

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Label.animate

function of Titanium.UI.Label

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.Label.fireEvent

function of Titanium.UI.Label

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Label.hide

Titanium.UI.Label

function of Titanium.UI.Label
hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Label.remove

function of Titanium.UI.Label
remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.Label.removeEventListener

function of Titanium.UI.Label
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Label.show

function of Titanium.UI.Label
make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Label.toImage

function of Titanium.UI.Label
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.OptionDialog

object of Titanium.UI

The Option Dialog is created by Titanium.UI.createOptionDialog and allows you to show a modal dialog of one or more options to the user.

Methods

Name	Description
show	cause the dialog to become visible

Properties

Name	Type	Description
cancel	int	an index to indicate which button should be the cancel button
destructive	int	the destructive button (indicated by a visual clue in the UI)
options	array	array of button names as strings
title	string	the title of the dialog

Events

Name	Description

click

fired when a button in the dialog is clicked

Event properties

cancel	boolean to indicate if the cancel button was pressed
destructive	boolean to indicate if the destructive button was pressed
index	the button index that was pressed
source	the source object that fired the event
type	the name of the event fired

Code Examples

Simple Options Dialog with 2 Options

In this example, we show a simple option dialog.

```
var dialog = Titanium.UI.createOptionDialog({
    title: 'Hello',
    options: ['Option 1','Option 2'],
    cancel:1
});
dialog.show();
```

Titanium.UI.OptionDialog.show

function of Titanium.UI.OptionDialog

cause the dialog to become visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Picker

object of Titanium.UI

A Picker is created by the method Titanium.UI.createPicker. A Picker can be used to select one or more fixed values.

Methods

Name	Description

add	add an array of rows, a single row or a column to the picker
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
getSelectedRow	get the selected row object for column
reloadColumn	causes the picker to reload the values from the new column
removeEventListener	remove a previously added event listener
setSelectedRow	set the column's row to the selected state

Properties

Name	Type	Description
columns	array	array of column values
countDownDuration	double	the duration value in milliseconds for count down timer pickers
locale	string	the locale used for displaying Date/Time pickers values
minDate	date	the minimum Date/Time for value for date pickers
minuteInterval	int	property to set the interval displayed by the minutes wheel (for example, 15 minutes). The interval value must be evenly divided into 60; if it is not, the default value is used. The default and minimum values are 1; the maximum value is 30.
selectionIndicator	boolean	for basic picker, boolean value to indicate whether the visual selection style is shown. On the iPhone, this is a blue selected bar.
type	int	the type constant for the picker. One of Titanium.UI.PICKER_TYPE_PLAIN (default), Titanium.UI.PICKER_TYPE_DATE_AND_TIME, Titanium.UI.PICKER_TYPE_DATE, Titanium.UI.PICKER_TYPE_TIME or Titanium.UI.PICKER_TYPE_COUNT_DOWN_TIMER.
value	date	the Date/Time value for date pickers

Events

Name	Description
change	

fired when the value of a picker row and/or column changes

Event properties

column	the column object
columnIndex	the selected column index
row	the row object
rowIndex	the selected row index
selectedValue	(plain picker only) the array of selected values, one element per column in the picker.
source	the source object that fired the event
type	the name of the event fired
value	(date/time pickers only) the selected date/time value.

Code Examples

Basic Single Column Picker

In this basic picker example, we create a one column picker with 4 rows.

```
var picker = Titanium.UI.createPicker();
var data = [];
data[0]=Titanium.UI.createPickerRow({title:'Bananas'});
data[1]=Titanium.UI.createPickerRow({title:'Strawberries'});
data[2]=Titanium.UI.createPickerRow({title:'Mangos'});
data[3]=Titanium.UI.createPickerRow({title:'Grapes'});
picker.add(data);
```

Custom View for Row

In this example, we use a custom label for each row in a column.

```
var picker = Titanium.UI.createPicker();
var row = Titanium.UI.createPickerRow();
var label = Titanium.UI.createLabel({
  text:text,
  font:{fontSize:24,fontWeight:'bold'},
  color:text,
  width:'auto',
  height:'auto'
});
row.add(label);
picker.add(row);
```

Notes

Android does not support the DateTime or Count Down Timer picker type.

Titanium.UI.Picker.add

function of Titanium.UI.Picker

add an array of rows, a single row or a column to the picker

Arguments

Name	Type	Description
data	array,object	add an array of rows, a single row or a column to the picker

Return Type

void

Titanium.UI.Picker.addEventListener

function of Titanium.UI.Picker

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Picker.fireEvent

function of Titanium.UI.Picker

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Picker.getSelectedRow

function of Titanium.UI.Picker

get the selected row object for column

Arguments

Name	Type	Description
index	int	for the column index, return the row object or nil if not found

Return Type

object

Titanium.UI.Picker.reloadColumn

function of Titanium.UI.Picker

causes the picker to reload the values from the new column

Arguments

Name	Type	Description
column	object	new column to load

Return Type

void

Titanium.UI.Picker.removeEventListener

function of Titanium.UI.Picker

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Picker.setSelectedRow

function of Titanium.UI.Picker

set the column's row to the selected state

Arguments

Name	Type	Description
column	int	the column index
row	int	the row index
animated	boolean	boolean to indicate if the selection should be animated (default)

Return Type

void

Titanium.UI.PickerColumn

object of Titanium.UI

The picker row object created by Titanium.UI.createPickerColumn.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListenert	add an event listener for the instance to receive view triggered events
addRow[object]	a Titanium.UI.PickerRow object to add to the column
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
removeRow[object]	a Titanium.UI.PickerRow object to remove
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view

backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
rowCount	int	

		number of rows in the column (readonly)
rows	array	an array of rows
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
doubletap	<p>fired when the device detects a double tap against the view</p> <p>Event properties</p>										

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

singletap fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Titanium.UI.PickerColumn.add

function of Titanium.UI.PickerColumn
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.PickerColumn.addEventListener

function of Titanium.UI.PickerColumn

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.PickerColumn.addRow[object]

function of Titanium.UI.PickerColumn

a Titanium.UI.PickerRow object to add to the column

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.PickerColumn.animate

function of Titanium.UI.PickerColumn

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object

callback	function	function to be invoked upon completion of the animation
-----------------	----------	---

Return Type

void

Titanium.UI.PickerColumn.fireEvent

function of Titanium.UI.PickerColumn

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.PickerColumn.hide

function of Titanium.UI.PickerColumn

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.PickerColumn.remove

function of Titanium.UI.PickerColumn

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.PickerColumn.removeEventListener

function of Titanium.UI.PickerColumn
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.PickerColumn.removeRow[object]

function of Titanium.UI.PickerColumn

a Titanium.UI.PickerRow object to remove

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.PickerColumn.show

function of Titanium.UI.PickerColumn

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.PickerColumn.toImage

function of Titanium.UI.PickerColumn

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion, if non-null, this method will be performed asynchronously, if null, it will be

		function	function to be invoked upon completion. If non-null, this method will be performed asynchronously. If null, it will be performed immediately
--	--	----------	--

Return Type

object

Titanium.UI.PickerRow

object of Titanium.UI

The picker row object created by Titanium.UI.createPickerRow.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view

backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
fontSize	int	the font size when displaying the text. ignored when using a custom view
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
selected	boolean	when used in the constructor, set the row to selected on initial display
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
title	string	the display text

top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
doubletap	<p>fired when the device detects a double tap against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart

fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples**Custom Views for a picker row**

Since the row object is itself a view, you can add views and widgets to it to customize the the rows display. In the example below, we create a custom label for the row.

```
var row = Ti.UI.createPickerRow();
var label = Ti.UI.createLabel({
    text:text,
    font:{fontSize:24,fontWeight:'bold'},
    color:text,
    width:'auto',
    height:'auto'
});
row.add(label);
picker.add(row);
```

Titanium.UI.PickerRow.add

function of Titanium.UI.PickerRow

add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.PickerRow.addEventListener

function of Titanium.UI.PickerRow

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.PickerRow.animate

function of Titanium.UI.PickerRow

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.PickerRow.fireEvent

function of Titanium.UI.PickerRow
fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.PickerRow.hide

function of Titanium.UI.PickerRow
hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.PickerRow.remove

function of Titanium.UI.PickerRow
remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.PickerRow.removeEventListener

function of Titanium.UI.PickerRow
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event

Name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.PickerRow.show

function of Titanium.UI.PickerRow

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.PickerRow.toImage

function of Titanium.UI.PickerRow

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.ProgressBar

object of Titanium.UI

A Progress Bar is created by the method Titanium.UI.createProgressBar.

Methods

This object has no methods

Properties

Name	Type	Description
color	string	the color of the progress bar text

font	object	the font object for the progress bar text
max	float	the maximum value of the progress bar
message	string	the progress bar message
min	float	the minimum value of the progress bar
style	int	the style of the progress bar
value	float	the current value of the progress bar

Events

This object has no events

Code Examples

Simple Progress Bar

In this example we create a progress bar with the min value of 0 and the max value of 10 and the current value of 0. To change the value of the progress bar to cause it to move, we would set the **value** property to a value between **min** and **max**.

```
var pb=Titanium.UI.createProgressBar({
    width:250,
    min:0,
    max:10,
    value:0,
    color:'#fff',
    message:'Downloading 0 of 10',
    font:{fontSize:14, fontWeight:'bold'},
    style:Titanium.UI.iPhone.ProgressBarStyle.PLAIN,
});
```

Notes

For iPhone, progress bar styles are constants defined in Titanium.UI.iPhone.ProgressBarStyle.

Titanium.UIScrollView

object of Titanium.UI

A Scroll View is used to create a scrollable region of content. Views added to the Scroll View will be scrolled based on the content size of the Scroll View. The Scroll View is created by the method Titanium.UI.createScrollView. Note: In Android, Scroll Views can only scroll in one direction, either vertical or horizontal, but not both at the same time. See the Titanium.UIScrollView.scrollType property

Methods

Name	Description
add	add a child to the view hierarchy

addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
scrollTo	scrollTo a particular point
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image

		interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
canCancelEvents	boolean	(iPhone only) boolean to indicate if the scroll view can cancel subview touches in order to scroll instead. Default of true
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
contentHeight	float	the height of the scrollable area
contentOffset	object	an object (with x and y properties) to indicate the offset of the content area
contentWidth	float	the width of the scrollable area
disableBounce	boolean	boolean to control bounce during scrolling
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
horizontalBounce	boolean	boolean to control the horizontal bounce during scrolling
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
maxZoomScale	float	the maximum scale of the content
minZoomScale	float	the minimum scale of the content
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
scrollType	string	(Android only) the type of ScrollView: "vertical" or "horizontal"

showHorizontalScrollIndicator	boolean	boolean to indicate whether the horizontal scroll indicator is visible
showVerticalScrollIndicator	boolean	boolean to indicate whether the vertical scroll indicator is visible
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
verticalBounce	boolean	boolean to control the vertical bounce during scrolling
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views
zoomScale	float	set the zoom scale for the current content area

Events

Name	Description										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event						
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										

type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

scale

fired when the zoom scale factor changes

Event properties

scale	the new scale as a float
source	the source object that fired the event
type	the name of the event fired

scroll

fired when the view is scrolled

Event properties

decelerating	boolean to indicate if the scroll is decelerating
dragging	boolean to indicate if the scroll is based on a dragging gesture
source	the source object that fired the event
type	the name of the event fired
x	the new x location
y	the new y location

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Simple Scroll View

Create a scroll view with content.

```
var scrollView = Titanium.UI.createScrollView({
    contentWidth:'auto',
    contentHeight:'auto',
    top:0,
    showVerticalScrollIndicator:true,
    showHorizontalScrollIndicator:true
});
var view = Ti.UI.createView({
    backgroundColor:'#336699',
    borderRadius:10,
    width:300,
    height:2000,
    top:10
});
scrollView.add(view);
Titanium.UI.currentWindow.add(scrollView);
```

Notes

Notes

If your scrollable region fits within the size of the content area the scroll view will not scroll.

Titanium.UIScrollView.add

function of Titanium.UIScrollView
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UIScrollView.addEventListener

function of Titanium.UIScrollView
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UIScrollView.animate

function of Titanium.UIScrollView
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UIScrollView.fireEvent

function of Titanium.UIScrollView

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UIScrollView.hide

function of Titanium.UIScrollView

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UIScrollView.remove

function of Titanium.UIScrollView

remove a previously add view from the view hiearchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hiearchy

Return Type

void

Titanium.UIScrollView.removeEventListener

function of Titanium.UIScrollView

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UIScrollView.scrollTo

function of Titanium.UIScrollView

scrollTo a particular point

Arguments

Name	Type	Description
x	float	the x point within the view
y	float	the y point within the view

Return Type

void

Titanium.UIScrollView.show

function of Titanium.UIScrollView

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UIScrollView.toImage

function of Titanium.UIScrollView

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.ScrollableView

object of Titanium.UI

The Scrollable View provides a view that supports horizontal scrolling on one or more views in a gesture motion. The Scrollable View also optionally supports a visual paging control to indicate the page that the view is visible. The Scrollable View is created by the method Titanium.UI.createScrollView.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListenner	add an event listenner for the instance to receive view triggered events
addView	add a new view to the Scrollable View
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hiearchy
removeEventListenner	remove a previously added event listenner
removeView	remove an existing view from the Scrollable View
scrollToView	scroll to a specific view
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view, focusable must be true for normal views. (Android)

backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
currentPage	int	the current page visible in the view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
maxZoomScale	float	the maximum zoom scale for the view
minZoomScale	float	the minimum zoom scale for the view
opacity	float	the opacity from 0.0-1.0

pagingControlColor	string	the color of the paging control. defaults to black.
pagingControlHeight	float	the height in pixels of the paging control, if visible. defaults to 20
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
showPagingControl	boolean	boolean to indicate whether the paging control UI is visible
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
views	array	array of view objects to place in the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description														
click	fired when the page control is touched <p>Event properties</p> <table border="1"> <tr> <td>currentPage</td><td>the current page index</td></tr> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>view</td><td>the current page view</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	currentPage	the current page index	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	view	the current page view	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
currentPage	the current page index														
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates														
source	the source object that fired the event														
type	the name of the event fired														
view	the current page view														
x	the x point of the event in receiving view coordinates														
y	the y point of the event, in receiving view coordinates														

dblclick	fired when the device detects a double click against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
doubletap	fired when the device detects a double tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
scroll	fired when the scroll view is scrolled
Event properties	
currentPage	the current page index
source	the source object that fired the event
type	the name of the event fired
view	the current page view
singletap	fired when the device detects a single tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart	fired as soon as the device detects a gesture
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap	fired when the device detects a two-finger tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Simple Scrollable View with 3 Views

Create three views and place each into a scrollable view.

```
var view1 = Titanium.UI.createView({backgroundColor:'#123'});
var view2 = Titanium.UI.createView({backgroundColor:'#123'});
var view3 = Titanium.UI.createView({backgroundColor:'#123'});
var scrollView = Titanium.UI.createScrollView({
    views:[view1,view2,view3],
    showPagingControl:true
});
win.add(scrollView);
```

Titanium.UI.ScrollableView.add

function of Titanium.UI.ScrollableView
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.ScrollableView.addEventListener

function of Titanium.UI.ScrollableView

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.ScrollableView.addView

function of Titanium.UI.ScrollableView

add a new view to the Scrollable View

Arguments

Name	Type	Description
view	object	the view to add

Return Type

void

Titanium.UI.ScrollableView.animate

function of Titanium.UI.ScrollableView

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.ScrollableView.fireEvent

function of Titanium.UI.ScrollableView

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.ScrollableView.hide

function of Titanium.UI.ScrollableView

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.ScrollableView.remove

function of Titanium.UI.ScrollableView

remove a previously add view from the view hiearchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hiearchy

Return Type

void

Titanium.UI.ScrollableView.removeEventListener

function of Titanium.UI.ScrollableView

remove a previously added event listener

Arguments

Name	Type	Description
------	------	-------------

	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UIScrollView.removeView

function of Titanium.UIScrollView

remove an existing view from the Scrollable View

Arguments

Name	Type	Description
view	object	the view to remove

Return Type

void

Titanium.UIScrollView.scrollToView

function of Titanium.UIScrollView

scroll to a specific view

Arguments

Name	Type	Description
view	int,object	either an integer index or the view object to bring into view as the currentPage

Return Type

void

Titanium.UIScrollView.show

function of Titanium.UIScrollView

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UIScrollView.toImage

function of Titanium.UIScrollView
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.SearchBar

object of Titanium.UI

A Search Bar is created by the method Titanium.UI.createSearchBar.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
blur	called to force the search bar to lose focus
fireEvent	fire a synthesized event to the views listener
focus	called to force the search bar to focus
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and v to indicate the anchor point value. anchor specifies the

anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
autocapitalization	boolean	boolean to indicate if the text in the field should be autocapitalized as typed
autocomplete	boolean	boolean to indicate if the text in the field should be autocorrected as typed
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
barColor	string	the bar color of the search bar view
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the

		parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
hintText	string	the text to show when the search bar field is not focused
keyboardType	int	the keyboard type constant to use when the field is focused
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
prompt	string	a single line of text displayed at the top of the search bar
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
showCancel	boolean	boolean indicates whether the cancel button is displayed
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
value	string	the value of the search bar
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description
blur	fired when the search bar loses focus

Event properties

source	the source object that fired the event
---------------	--

type	the name of the event fired
-------------	-----------------------------

cancel	fired when the cancel button is pressed	
Event properties		
source	the source object that fired the event	
type	the name of the event fired	
change	fired when the value of the search bar changes	
Event properties		
source	the source object that fired the event	
type	the name of the event fired	
click	fired when the device detects a click (longer than touch) against the view	
Event properties		
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	
source	the source object that fired the event	
type	the name of the event fired	
x	the x point of the event in receiving view coordinates	
y	the y point of the event, in receiving view coordinates	
dblclick	fired when the device detects a double click against the view	
Event properties		
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	
source	the source object that fired the event	
type	the name of the event fired	
x	the x point of the event in receiving view coordinates	
y	the y point of the event, in receiving view coordinates	
doubletap	fired when the device detects a double tap against the view	
Event properties		
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	
source	the source object that fired the event	
type	the name of the event fired	

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

focus

fired when the search bar gains focus

Event properties

source	the source object that fired the event
type	the name of the event fired

return

fired when keyboard search button is pressed

Event properties

source	the source object that fired the event
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
--------------------	--

source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend	fired when a touch event is completed
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart	fired as soon as the device detects a gesture
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap	fired when the device detects a two-finger tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Simple Search Bar

```
var search = Titanium.UI.createSearchBar({
    barColor:'#000',
    showCancel:true,
    height:43,
    top:0,
});
```

Titanium.UI.SearchBar.add

function of Titanium.UI.SearchBar
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.SearchBar.addEventListener

function of Titanium.UI.SearchBar
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.SearchBar.animate

function of Titanium.UI.SearchBar
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.SearchBar.blur

function of Titanium.UI.SearchBar

called to force the search bar to lose focus

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.SearchBar.fireEvent

function of Titanium.UI.SearchBar

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.SearchBar.focus

function of Titanium.UI.SearchBar

called to force the search bar to focus

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.SearchBar.hide

function of Titanium.UI.SearchBar

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.SearchBar.remove

function of Titanium.UI.SearchBar

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.SearchBar.removeEventListener

function of Titanium.UI.SearchBar

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.SearchBar.show

function of Titanium.UI.SearchBar

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.SearchBar.toImage

function of Titanium.UI.SearchBar

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.Slider

object of Titanium.UI

A Slider is created by the method Titanium.UI.createSlider.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
------	------	-------------

anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
disabledLeftTrackImage	string	the image url of the slider left track when in the disabled state
disabledRightTrackImage	string	the image url of the slider right track when in the disabled state

disabledThumbImage	string	the image url of the slider thumb when in the disabled state
enabled	boolean	boolean to indicate the enabled state of the slider
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
highlightedLeftTrackImage	string	the image url of the slider left track when in the highlighted state
highlightedRightTrackImage	string	the image url of the slider right track when in the highlighted state
highlightedThumbImage	string	the image url of the slider thumb when in the highlighted state
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
leftTrackImage	string	the image url of the slider left track
max	float	the maximum slider value
min	float	the minimum slider value
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
rightTrackImage	string	the image url of the slider right track
selectedLeftTrackImage	string	the image url of the slider left track when in the selected state
selectedRightTrackImage	string	the image url of the slider right track when in the selected state
selectedThumbImage	string	the image url of the slider thumb when in the selected state
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)

thumbImage	string	the image url to the slider thumb
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
value	string	the value of the slider
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
change	<p>fired when the value of the slider changes</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>value[string]</td><td>the new value of the slider</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired	value[string]	the new value of the slider				
source	the source object that fired the event										
type	the name of the event fired										
value[string]	the new value of the slider										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired				
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										

type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap	fired when the device detects a double tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

singletap	fired when the device detects a single tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe	fired when the device detects a swipe (left or right) against the view
Event properties	
direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel	fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend	fired when a touch event is completed
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
touchstart	fired as soon as the device detects a gesture
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
twofingertap	fired when the device detects a two-finger tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Titanium.UI.Slider.add

function of Titanium.UI.Slider
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.Slider.addEventListener

function of Titanium.UI.Slider
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Slider.animate

function of Titanium.UI.Slider
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.Slider.fireEvent

function of Titanium.UI.Slider

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Slider.hide

function of Titanium.UI.Slider

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Slider.remove

function of Titanium.UI.Slider

remove a previously add view from the view hiearchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hiearchy

Return Type

void

Titanium.UI.Slider.removeEventListener

function of Titanium.UI.Slider

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Slider.show

function of Titanium.UI.Slider
make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Slider.toImage

function of Titanium.UI.Slider
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.Switch

object of Titanium.UI

A Switch is created by the method Titanium.UI.createSwitch.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view

fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view

borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
enabled	boolean	boolean for the state of the switch
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
value	boolean	boolean value of the switch where true is the switch is on and false the switch is off
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description						
change	fired when the switch value is changed Event properties <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>value[boolean]</td><td>the current value of the switch</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired	value[boolean]	the current value of the switch
source	the source object that fired the event						
type	the name of the event fired						
value[boolean]	the current value of the switch						

click	fired when the device detects a click (longer than touch) against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
dblclick	fired when the device detects a double click against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
doubletap	fired when the device detects a double tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
singletap	fired when the device detects a single tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
swipe	fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart

fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples**Simple Switch Example**

The following is a simple example of a switch and receiving change events.

```
var basicSwitch = Titanium.UI.createSwitch({
    value:false
});
basicSwitch.addEventListener('change',function(e)
{
    Titanium.API.info('Basic Switch value = ' + e.value + ' act val ' + basicSwitch.value);
});
```

Titanium.UI.Switch.addfunction of Titanium.UI.Switch
add a child to the view hierarchy**Arguments**

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.Switch.addEventListener

function of Titanium.UI.Switch

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Switch.animate

function of Titanium.UI.Switch

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.Switch.fireEvent

function of Titanium.UI.Switch

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Switch.hide

Titanium.UI.Switch.hide

function of Titanium.UI.Switch
hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Switch.remove

function of Titanium.UI.Switch
remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.Switch.removeEventListener

function of Titanium.UI.Switch
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Switch.show

function of Titanium.UI.Switch
make the view visible

Arguments

This function takes no arguments.

Return Type

Return Type

void

Titanium.UI.Switch.toImage

function of Titanium.UI.Switch
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.Tab

object of Titanium.UI

A TabGroup Tab instance. Each Tab instance maintains a stack of tab windows. Only one window within in the Tab can be visible at a time. When a window is closed, either by the user or by code, the window is removed from the stack, make the previous window visible. The root tab window cannot be removed. The Tab Group is created by the method Titanium.UI.createTab.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5

animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
badge	string	the badge value for the tab group for this tab. null indicates no badge is value
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
icon	string	the icon url for the tab group for this tab

left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
title	string	the title for the tab group for this tab
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
window	object	the root level tab window. all tabs must have at least one root level tab window.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event						
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										

type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

singletap fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired

type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend	fired when a touch event is completed
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart	fired as soon as the device detects a gesture
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap	fired when the device detects a two-finger tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates

Code Examples

Simple Tab Example

In this example, we create a simple tab and add it to a tab group.

```
var tab = Titanium.UI.createTab({  
    window:mywin,  
    title:'Hello',  
    icon:'myicon.png'  
});  
tabGroup.addTab(tab);
```

Titanium.UI.Tab.add

function of Titanium.UI.Tab
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.Tab.addEventListener

function of Titanium.UI.Tab
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Tab.animate

function of Titanium.UI.Tab
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.Tab.fireEvent

function of Titanium.UI.Tab

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Tab.hide

function of Titanium.UI.Tab

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Tab.remove

function of Titanium.UI.Tab

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.Tab.removeEventListener

function of Titanium.UI.Tab

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Tab.show

function of Titanium.UI.Tab

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Tab.toImage

function of Titanium.UI.Tab

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.TabGroup

↳ Titanium.UI

The Tab Group allows you to manage a tabbed UI of one or more windows. The Tab Group is created by the method `Titanium.UI.createTabGroup`.

Methods

Name	Description
<code>add</code>	add a child to the view hierarchy
<code>addEventListenerr</code>	add an event listener for the instance to receive view triggered events
<code>addTab</code>	add a tab to the tab group
<code>animate</code>	animate the view
<code>close</code>	close the tab group and remove it from the UI
<code>fireEvent</code>	fire a synthesized event to the views listener
<code>hide</code>	hide the view
<code>open</code>	open the tab group and make it visible
<code>remove</code>	remove a previously add view from the view hierarchy
<code>removeEventListener</code>	remove a previously added event listener
<code>removeTab</code>	remove a tab from the tab group
<code>setActiveTab</code>	select the currently active tab in a tab group
<code>show</code>	make the view visible
<code>toImage</code>	return a Blob image of the rendered view

Properties

Name	Type	Description
<code>activeTab</code>	object	the active tab
<code>allowUserCustomization</code>	boolean	whether or not the user can configure the tab group via the 'More' tab's edit functionality. iPhone/iPad only
<code>anchorPoint</code>	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
<code>animatedCenterPoint</code>	object	read-only object with x and y properties of where the view is during animation

backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
barColor	string	the bar color
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
editButtonTitle	string	the title for the 'More' tab edit button. iPhone/iPad only
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.

left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
tabs	array	array of tab objects that are managed by the tab group
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description												
blur	<p>fired when the tab group loses focus</p> <p>Event properties</p> <table border="1"> <tr> <td>index</td><td>the tab index</td></tr> <tr> <td>previousIndex</td><td>the previous tab index</td></tr> <tr> <td>previousTab</td><td>the previous tab object</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>tab</td><td>the tab object</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	index	the tab index	previousIndex	the previous tab index	previousTab	the previous tab object	source	the source object that fired the event	tab	the tab object	type	the name of the event fired
index	the tab index												
previousIndex	the previous tab index												
previousTab	the previous tab object												
source	the source object that fired the event												
tab	the tab object												
type	the name of the event fired												
click	fired when the device detects a click (longer than touch) against the view												

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

close

fired when the tab group is closed

Event properties

source	the source object that fired the event
type	the name of the event fired

dblclick

fired when the device detects a double click against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap

fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

focus

fired when the tab group gains focus

Event properties

index	the tab index
previousIndex	the previous tab index

previousTab	the previous tab object
source	the source object that fired the event
tab	the tab object
type	the name of the event fired

open

fired when the tab group is opened

Event properties

source	the source object that fired the event
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event

type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend	fired when a touch event is completed
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart	fired as soon as the device detects a gesture
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap	fired when the device detects a two-finger tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Titanium.UI.TabGroup.add

function of Titanium.UI.TabGroup
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.TabGroup.addEventListener

function of Titanium.UI.TabGroup
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.TabGroup.addTab

function of Titanium.UI.TabGroup

add a tab to the tab group

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TabGroup.animate

function of Titanium.UI.TabGroup
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.TabGroup.close

function of Titanium.UI.TabGroup

close the tab group and remove it from the UI

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TabGroup.fireEvent

function of Titanium.UI.TabGroup
fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.TabGroup.hide

function of Titanium.UI.TabGroup
hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TabGroup.open

function of Titanium.UI.TabGroup

open the tab group and make it visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TabGroup.remove

function of Titanium.UI.TabGroup

remove a previously add view from the view hiearchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hiearchy

Return Type

void

Titanium.UI.TabGroup.removeEventListener

function of Titanium.UI.TabGroup

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.TabGroup.removeTab

function of Titanium.UI.TabGroup

remove a tab from the tab group

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TabGroup.setActiveTab

function of Titanium.UI.TabGroup

select the currently active tab in a tab group

Arguments

Name	Type	Description
indexOrObject	object	an int representing the desired tab index or a reference to the tab object you'd like to switch to

Return Type

void

Titanium.UI.TabGroup.show

function of Titanium.UI.TabGroup

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TabGroup.toImage

function of Titanium.UI.TabGroup

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.TabbedBar

object of Titanium.UI

A Tabbed Bar is created by the method Titanium.UI.createTabbedBar. The difference between the Tabbed Bar and the Button Bar is that the tabbed bar visually maintains a state (visually distinguished as a pressed or selected look).

Methods

Name	Description
add	add a child to the view hierarchy
addEventListenner	add an event listenner for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hiearchy
removeEventListenner	remove a previously added event listenner
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the tabbed bar
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view

backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
index	int	the selected index
labels	array	the array of labels for the tabbed bar. each object should have the properties <code>title</code> , <code>image</code> , <code>width</code> and <code>enabled</code> .
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
style	int	

style	int	the style of the tabbed bar
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description												
click	<p>fired when a button on the tabbed bar is clicked</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>index[int]</td><td>the index of the button that was clicked</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	index[int]	the index of the button that was clicked	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates												
index[int]	the index of the button that was clicked												
source	the source object that fired the event												
type	the name of the event fired												
x	the x point of the event in receiving view coordinates												
y	the y point of the event, in receiving view coordinates												
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates		
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates												
source	the source object that fired the event												
type	the name of the event fired												
x	the x point of the event in receiving view coordinates												
y	the y point of the event, in receiving view coordinates												
doubletap	<p>fired when the device detects a double tap against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event								
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates												
source	the source object that fired the event												

type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event

type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Simple Tabbed Bar with 3 items

```
var bb1 = Titanium.UI.createTabbedBar({
  labels:['One', 'Two', 'Three'],
  backgroundColor:'#336699',
  top:50,
  style:Titanium.UI.iPhone.SystemButtonStyle.BAR,
  height:25,
  width:200
```

```
});  
win.add(bb1);
```

Notes

For iPhone, the style constants are available in the constants defined in Titanium.UI.iPhone.SystemButtonStyle.

Titanium.UI.TabbedBar.add

function of Titanium.UI.TabbedBar
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.TabbedBar.addEventListener

function of Titanium.UI.TabbedBar
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.TabbedBar.animate

function of Titanium.UI.TabbedBar
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

Return Type

void

Titanium.UI.TabbedBar.fireEvent

function of Titanium.UI.TabbedBar

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.TabbedBar.hide

function of Titanium.UI.TabbedBar

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TabbedBar.remove

function of Titanium.UI.TabbedBar

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.TabbedBar.removeEventListener

function of Titanium.UI.TabbedBar

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.TabbedBar.show

function of Titanium.UI.TabbedBar
make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TabbedBar.toImage

function of Titanium.UI.TabbedBar
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.TableView

object of Titanium.UI

A Table View allows you to create a scrollable table of content in a list-based fashion. The Table View is created by the method Titanium.UI.createTableView.

Methods

Name	Description
------	-------------

add	add a child to the view hierarchy	
addEventListener	add an event listener for the instance to receive view triggered events	
animate	animate the view	
appendRow	append a row to the table, optionally with animation	
deleteRow	delete an existing row, optionally with animation	
deselectRow	programmatically deselect a row	
fireEvent	fire a synthesized event to the views listener	
hide	hide the view	
insertRowAfter	insert a row before another row, optionally with animation	
insertRowBefore	insert a row after another row, optionally with animation	
remove	remove a previously add view from the view hierarchy	
removeEventListener	remove a previously added event listener	
scrollToIndex	scroll to a specific row index and ensure that that row is on screen	
scrollToTop	scroll the table to a specific top position where 0 is the topmost y position in the table view	
selectRow	programmatically select a row	
setData	set the data in the table, optionally with animation	
show	make the view visible	
toImage	return a Blob image of the rendered view	
updateRow	update an existing row, optionally with animation	

Properties

Name	Type	Description
allowsSelection	boolean	true if the rows can be selected
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5

animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the table view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image to render in the background of the table view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
data	array	the data array of objects to be used for the rows of the table view
editable	boolean	allow the table view to be editable (this must be true for swipe-to-delete)

editing	boolean	boolean to control the editing state of the table view
filterAttribute	string	the filter attribute to be used when searching. this property maps to your data object or a property on the row object
filterCaseInsensitive	boolean	boolean to indicate if the search should be case sensitive or case insensitive (default)
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
footerTitle	string	the table view footer title
footerView	object	the table view footer as a view that will be rendered instead of a label
headerTitle	string	the table view header title
headerView	object	the table view header as a view that will be rendered instead of a label
height	float,string	property for the view height. can be either float value or a string of the width.
index	array	an array of objects (with title and index properties) to control the table view index
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
maxRowHeight	float	max row height for table view rows
minRowHeight	float	min row height for table view rows
moving	boolean	boolean to control the moveable state of the table view
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
rowHeight	float	default row height for table view rows
scrollable	boolean	true (default) if tableview can be scrolled
search	object	the search field to use for the table view

searchHidden	boolean	boolean to control the visibility of the search field
separatorColor	string	the separator color color as a hex or named value
separatorStyle	int	the separator style constant. For iPhone, Titanium.UI.iPhone.TableViewSeparatorStyle
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
style	int	iPhone only. the style of the table view. constant from Titanium.UI.iPhone.TableViewStyle
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description												
click	<p>fired when a table row is clicked</p> <p>Event properties</p> <table border="1"> <tr> <td>detail</td><td>boolean to indicate if the right area was clicked</td></tr> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>index</td><td>table view row index</td></tr> <tr> <td>row</td><td>table view row object</td></tr> <tr> <td>rowData</td><td>table view row data object</td></tr> <tr> <td>searchMode</td><td>boolean to indicate if the table is in search mode</td></tr> </table>	detail	boolean to indicate if the right area was clicked	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	index	table view row index	row	table view row object	rowData	table view row data object	searchMode	boolean to indicate if the table is in search mode
detail	boolean to indicate if the right area was clicked												
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates												
index	table view row index												
row	table view row object												
rowData	table view row data object												
searchMode	boolean to indicate if the table is in search mode												

	<table border="1"> <tr> <td>section</td><td>table view section object</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	section	table view section object	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates						
section	table view section object																
source	the source object that fired the event																
type	the name of the event fired																
x	the x point of the event in receiving view coordinates																
y	the y point of the event, in receiving view coordinates																
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates						
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates																
source	the source object that fired the event																
type	the name of the event fired																
x	the x point of the event in receiving view coordinates																
y	the y point of the event, in receiving view coordinates																
delete	<p>fired when a table row is delete by the user</p> <p>Event properties</p> <table border="1"> <tr> <td>detail</td><td>boolean to indicate if the right area was clicked</td></tr> <tr> <td>index</td><td>table view row index</td></tr> <tr> <td>row</td><td>table view row object</td></tr> <tr> <td>rowData</td><td>table view row data object</td></tr> <tr> <td>searchMode</td><td>boolean to indicate if the table is in search mode</td></tr> <tr> <td>section</td><td>table view section object</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	detail	boolean to indicate if the right area was clicked	index	table view row index	row	table view row object	rowData	table view row data object	searchMode	boolean to indicate if the table is in search mode	section	table view section object	source	the source object that fired the event	type	the name of the event fired
detail	boolean to indicate if the right area was clicked																
index	table view row index																
row	table view row object																
rowData	table view row data object																
searchMode	boolean to indicate if the table is in search mode																
section	table view section object																
source	the source object that fired the event																
type	the name of the event fired																
doubletap	<p>fired when the device detects a double tap against the view</p> <p>Event properties</p>																

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

move

fired when a table row is moved by the user

Event properties

detail	boolean to indicate if the right area was clicked
index	table view row index
row	table view row object
rowData	table view row data object
searchMode	boolean to indicate if the table is in search mode
section	table view section object
source	the source object that fired the event
type	the name of the event fired

scroll

fired when the table view is scrolled (currently, iphone only)

Event properties

contentOffset	dictionary with x and y properties containing the content offset
contentSize	dictionary with width and height properties containing the size of the content (regardless of the display size in the case of scrolling)
size	dictionary with width and height properties containing the size of the visible table view
source	the source object that fired the event
type	the name of the event fired

scrollEnd

fired when the table view stops scrolling (currently, iphone only)

Event properties

contentOffset	dictionary with x and y properties containing the content offset
contentSize	dictionary with width and height properties containing the size of the content (regardless of the display size in the case of scrolling)
size	dictionary with width and height properties containing the size of the visible table view
source	the source object that fired the event
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
--------------------	--

source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend	fired when a touch event is completed
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart	fired as soon as the device detects a gesture
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap	fired when the device detects a two-finger tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Simple Table View with basic rows

The most basic example of a table view.

```
var data = [{title:"Row 1"}, {title:"Row 2"}];
var table = Titanium.UI.createTableView({data:data});
win.add(table);
```

Titanium.UI.TableView.add

function of Titanium.UI.TableView
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.TableView.addEventListener

function of Titanium.UI.TableView
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.TableView.animate

function of Titanium.UI.TableView
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.TableView.appendRow

function of Titanium.UI.TableView

append a row to the table, optionally with animation

Arguments

Name	Type	Description
row	object	row to append
properties	object	animation properties

Return Type

void

Titanium.UI.TableView.deleteRow

function of Titanium.UI.TableView

delete an existing row, optionally with animation

Arguments

Name	Type	Description
row	int	index of row to delete
properties	object	animation properties

Return Type

void

Titanium.UI.TableView.deselectRow

function of Titanium.UI.TableView

programmatically deselect a row

Arguments

Name	Type	Description
row	int	row index to deselect

Return Type

void

Titanium.UI.TableView.fireEvent

function of Titanium.UI.TableView

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.TableView.hide

function of Titanium.UI.TableView

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TableView.insertRowAfter

function of Titanium.UI.TableView

insert a row before another row, optionally with animation

Arguments

Name	Type	Description
index	int	index
row	object	row to insert
options	object	animation options

Return Type

void

Titanium.UI.TableView.insertRowBefore

function of Titanium.UI.TableView

insert a row after another row, optionally with animation

Arguments

Name	Type	Description
index	int	index
row	object	row to insert
properties	object	animation properties

Return Type

void

Titanium.UI.TableView.remove

function of Titanium.UI.TableView

remove a previously add view from the view hiearchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hiearchy

Return Type

void

Titanium.UI.TableView.removeEventListener

function of Titanium.UI.TableView

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.TableView.scrollToIndex

function of Titanium.UI.TableView

scroll to a specific row index and ensure that that row is on screen

Arguments

Name	Type	Description
index	int	index
properties	object	animation properties. <code>position</code> property controls the position constant to use for position (on iPhone, use constants from <code>Titanium.UI.iPhone.TableViewScrollPosition</code>).

Return Type

void

Titanium.UI.TableView.scrollToTop

function of Titanium.UI.TableView

scroll the table to a specific top position where 0 is the topmost y position in the table view

Arguments

Name	Type	Description
top	float	y position
properties	object	optional dictionary with the key <code>animated</code> (default, true) as boolean to indicate if the scroll should be animated or immediate

Return Type

void

Titanium.UI.TableView.selectRow

function of Titanium.UI.TableView

programmatically select a row

Arguments

Name	Type	Description

Return Type

void

Titanium.UI.TableView.setData

function of Titanium.UI.TableView

set the data in the table, optionally with animation

Arguments

Name	Type	Description
data	array	data array of rows either as objects or row objects
properties	object	animation properties

Return Type

void

Titanium.UI.TableView.show

function of Titanium.UI.TableView

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TableView.toImage

function of Titanium.UI.TableView

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.TableView.updateRow

function of Titanium.UI.TableView

update an existing row, optionally with animation

Arguments

Name	Type	Description
row	object	row data to update
properties	object	animation properties

Return Type

void

Titanium.UI.TableViewRow

object of Titanium.UI

A TableView row object created by the method Titanium.UI.createTableViewRow.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background cell color

backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background cell image
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
className	string	the class name of the table. each table view cell must have a unique class name if the cell layout is different. however, use the same name for rows that have the same structural layout (even if the content is different) to provide maximum rendering performance.
color	string	default color of the row when not selected
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)

hasCheck	boolean	render a system provided check mark in the right image area of the row cell
hasChild	boolean	render a system provided right arrow in the right image area of the row cell
hasDetail	boolean	render a system provided blue indicator icon in the right image area of the row cell
height	float	the height of the row. specify auto to calculate the row height based on the size of the child views of the row
indentLevel	int	the indentation level for the cell (defaults to 0)
layout	string	the layout algorithm to use for the layout. either absolute (default) or vertical.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
leftImage	string	image url to render in the left image area of the row cell
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
rightImage	string	image url to render in the right image area of the row cell
selectedBackgroundColor	string	the background color to render when the row cell is selected
selectedBackgroundImage	string	the background image to render when the row cell is selected
selectedColor	string	color of the row during selection
selectionStyle	int	a selection style constant to control the selection color. For iPhone, use the constants from Titanium.UI.iPhone.TableViewCellStyle
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
title	string	the title cell value. do not specify if using views as children of the row
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width

		float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description																						
click	<p>fired when the row is clicked. row events automatically propagate to the section and table view if an event listener is not added directly to the cell.</p> <p>Event properties</p> <table border="1"> <tr> <td>detail</td><td>boolean to indicate if the right area was clicked</td></tr> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>index</td><td>table view row index</td></tr> <tr> <td>row</td><td>table view row object</td></tr> <tr> <td>rowData</td><td>table view row data object</td></tr> <tr> <td>searchMode</td><td>boolean to indicate if the table is in search mode</td></tr> <tr> <td>section</td><td>table view section object</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	detail	boolean to indicate if the right area was clicked	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	index	table view row index	row	table view row object	rowData	table view row data object	searchMode	boolean to indicate if the table is in search mode	section	table view section object	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
detail	boolean to indicate if the right area was clicked																						
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates																						
index	table view row index																						
row	table view row object																						
rowData	table view row data object																						
searchMode	boolean to indicate if the table is in search mode																						
section	table view section object																						
source	the source object that fired the event																						
type	the name of the event fired																						
x	the x point of the event in receiving view coordinates																						
y	the y point of the event, in receiving view coordinates																						
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event																		
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates																						
source	the source object that fired the event																						

type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap	fired when the device detects a double tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
singletap	fired when the device detects a single tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
swipe	fired when the device detects a swipe (left or right) against the view
Event properties	
direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
touchcancel	fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event

type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend	fired when a touch event is completed
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
touchstart	fired as soon as the device detects a gesture
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
twofingertap	fired when the device detects a two-finger tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates

Code Examples

Simple Table View Row example

In this simple example, we create a table view row with a red square in the cell.

```
var row = Titanium.UI.createTableViewRow();
var view = Titanium.UI.createView({backgroundColor:'red',width:20,height:20});
row.height = 'auto';
row.add(view);
```

Notes

Make sure you set the tableClass on each row instance if using more than one type of row layout. You can use the same value for each instance of a row where the layout is the same - even if the value of the elements inside the row have different values. For example, if the text is the only thing different between two cells but the layout is the same, both row instances should have the same value for tableClass.

You can listen for table row events on all rows by adding an event listener to the table view instance. Events automatically propagate to parent views.

Titanium.UI.TableViewRow.add

function of Titanium.UI.TableViewRow
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.TableViewRow.addEventListener

function of Titanium.UI.TableViewRow
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.TableViewRow.animate

function of Titanium.UI.TableViewRow
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.TableViewRow.fireEvent

function of Titanium.UI.TableViewRow
fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.TableViewRow.hide

function of Titanium.UI.TableViewRow
hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TableViewRow.remove

function of Titanium.UI.TableViewRow
remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.TableViewRow.removeEventListener

function of Titanium.UI.TableViewRow
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.TableViewRow.show

function of Titanium.UI.TableViewRow
make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TableViewRow.toImage

function of Titanium.UI.TableViewRow
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

Titanium.UI.TableViewSection

object of Titanium.UI

A TableView section object created by the method Titanium.UI.createTableViewSection.

Methods

Name	Description
add	add a row to the section
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a remove from the section
removeEventListener	remove a previously added event listener
rowAtIndex	retrieve the row object at a specific index
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.

backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
footerTitle	string	the title of the section footer
footerView	object	a view to use instead of the default label when rendering the section footer
headerTitle	string	the title of the section header
headerView	object	a view to use instead of the default label when rendering the section header
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.

rowCount	int	the (readonly) number of rows in the section
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
doubletap	<p>fired when the device detects a double tap against the view</p> <p>Event properties</p>										

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

singletap fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend fired when a touch event is completed

Event properties

	<table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred										
	<p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
touchstart	fired as soon as the device detects a gesture										
	<p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
twofingertap	fired when the device detects a two-finger tap against the view										
	<p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										

Code Examples

Simple Table View Section Example

In this example, we create a section with the header Hello and add two rows to it.

```
var section = Titanium.UI.createTableViewSection();
section.headerTitle = "Hello";
var row1 = Titanium.UI.createTableViewRow({title:"Hello 1"});
var row2 = Titanium.UI.createTableViewRow({title:"Hello 2"});
section.add(row1);
section.add(row2);
tableview.add(section);
```

Titanium.UI.TableViewSection.add

function of Titanium.UI.TableViewSection

add a row to the section

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy
row	object	the row object to add to the section

Return Type

void

Titanium.UI.TableViewSection.addEventListener

function of Titanium.UI.TableViewSection

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.TableViewSection.animate

function of Titanium.UI.TableViewSection

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object

callback	function	function to be invoked upon completion of the animation
-----------------	----------	---

Return Type

void

Titanium.UI.TableViewSection.fireEvent

function of Titanium.UI.TableViewSection
fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.TableViewSection.hide

function of Titanium.UI.TableViewSection
hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TableViewSection.remove

function of Titanium.UI.TableViewSection

remove a remove from the section

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy
row	object	the row object to remove from the section

Return Type

void

Titanium.UI.TableViewSection.removeEventListener

function of Titanium.UI.TableViewSection

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.TableViewSection.rowAtIndex

function of Titanium.UI.TableViewSection

retrieve the row object at a specific index

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TableViewSection.show

function of Titanium.UI.TableViewSection

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TableViewSection.toImage

function of Titanium.UI.TableViewSection

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.TextArea

object of Titanium.UI

A Text Area is created by the method Titanium.UI.createTextArea. The Text Area is a multiline field.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
blur	force the field to lose focus
fireEvent	fire a synthesized event to the views listener
focus	force the field to gain focus
hasText	return boolean (true) if the field has text
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	value of the background color of the field

backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
editable	boolean	boolean indicating if the field is editable
enabled	boolean	boolean indicating the enabled state of the field
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
keyboardToolbar	array	array of toolbar button objects to be used when the keyboard is displayed

keyboardToolbarColor	string	the color of the keyboard toolbar
keyboardToolbarHeight	float	the height of the keyboard toolbar
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
suppressReturn	boolean	boolean to indicate if the return key should be suppressed during entry
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
value	string	value of the field
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description						
blur	<p>fired when the field loses focus</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>value</td><td>the value of the field upon blur</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired	value	the value of the field upon blur
source	the source object that fired the event						
type	the name of the event fired						
value	the value of the field upon blur						
change							

fired when the field value changes

Event properties

source	the source object that fired the event
type	the name of the event fired
value	the value of the field upon change

click

fired when the device detects a click (longer than touch) against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

dblclick

fired when the device detects a double click against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap

fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

focus

fired when the field gains focus

Event properties

source	the source object that fired the event
---------------	--

type	the name of the event fired
value	the value of the field upon focus

return

fired when the field return key is pressed on the keyboard

Event properties

source	the source object that fired the event
type	the name of the event fired
value	the value of the field upon return

selected

fired when the text in the field is selected

Event properties

range	the range of text. range is an object with the properties location and length .
source	the source object that fired the event
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates

y

the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart

fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofinger tap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Basic Text Area with Customizations

This example created a highly customized text area.

```
var tal = Titanium.UI.createTextArea({  
    value:'I am a textarea',  
    height:70,  
    width:300,  
    top:60,  
    font:{fontSize:20,fontFamily:'Marker Felt', fontWeight:'bold'},  
    color:'#888',  
    textAlign:'left',  
    appearance:Titanium.UI.KEYBOARD_APPEARANCE_ALERT,  
    keyboardType:Titanium.UI.KEYBOARD_NUMBERS_PUNCTUATION,  
    returnKeyType:Titanium.UI.RETURNKEY_EMERGENCY_CALL,  
    borderWidth:2,  
    borderColor:'#bbb',  
    borderRadius:5  
});
```

Notes

Both Text Areas and Text Fields can control the buttons displayed in a button bar above the keyboard when it's visible.

Example using a custom keyboard toolbar:

```
var textfield = Titanium.UI.createTextField({  
    color:'#336699',  
    value:'Focus to see keyboard w/ toolbar',  
    height:35,  
    width:300,  
    top:10,  
    borderStyle:Titanium.UI.INPUT_BORDERSTYLE_ROUNDED,  
    keyboardToolbar:[flexSpace,camera, flexSpace,tf,flexSpace, send,flexSpace],  
    keyboardToolbarColor: '#999',  
    keyboardToolbarHeight: 40,  
});
```

Titanium.UI.TextArea.add

function of Titanium.UI.TextArea
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.TextArea.addEventListener

function of Titanium.UI.TextArea

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.TextArea.animate

function of Titanium.UI.TextArea

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.TextArea.blur

function of Titanium.UI.TextArea

force the field to lose focus

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TextArea.fireEvent

function of Titanium.UI.TextArea

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.TextArea.focus

function of Titanium.UI.TextArea

force the field to gain focus

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TextArea.hasText

function of Titanium.UI.TextArea

return boolean (true) if the field has text

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TextArea.hide

function of Titanium.UI.TextArea

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TextArea.remove

function of Titanium.UI.TextArea

remove a previously add view from the view hiearchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hiearchy

Return Type

void

Titanium.UI.TextArea.removeEventListener

function of Titanium.UI.TextArea

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.TextArea.show

function of Titanium.UI.TextArea

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TextArea.toImage

function of Titanium.UI.TextArea

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.TextField

object of Titanium.UI

A Text Area is created by the method Titanium.UI.createTextField. The Text Field is a single line field.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
blur	force the field to lose focus
fireEvent	fire a synthesized event to the views listener
focus	force the field to gain focus
hasText	return boolean (true) if the field has text
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	value of the background color of the field
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the image url to the background image of the field when in the disabled state
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the image url to the background image of the field
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderStyle	int	the border style constant for the field
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width

		either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
clearButtonMode	int	the mode constant for how to handle displaying the clear button
clearOnEdit	boolean	boolean that indicates if the value of the field is cleared upon editing
editable	boolean	boolean indicating if the field is editable
enabled	boolean	boolean indicating the enabled state of the field
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
hintText	string	the hint text to display when the field is unfocused
keyboardToolbar	array	array of toolbar button objects to be used when the keyboard is displayed
keyboardToolbarColor	string	the color of the keyboard toolbar
keyboardToolbarHeight	float	the height of the keyboard toolbar
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
leftButton	object	the left button view
leftButtonMode	int	the mode of the left button view
leftButtonPadding	float	the left padding of the space between the button and the edge of the field
minimumFontSize	int	the minimum size of the font when the font is sized based on the contents. Enables font scaling to fit
opacity	float	the opacity from 0.0-1.0
paddingLeft	float	the left padding of the text field
paddingRight	float	the right padding of the text field

right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
rightButton	object	the right button view
rightButtonMode	int	the mode of the right button view
rightButtonPadding	float	the right padding of the space between the button and the edge of the field
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
suppressReturn	boolean	boolean to indicate if the return key should be suppressed during entry
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
value	string	value of the field
verticalAlign	int,string	the constant or string value for the fields vertical alignment.
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description						
blur	<p>fired when the field loses focus</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>value</td><td>the value of the field upon blur</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired	value	the value of the field upon blur
source	the source object that fired the event						
type	the name of the event fired						
value	the value of the field upon blur						

change

fired when the field value changes

Event properties

source	the source object that fired the event
type	the name of the event fired
value	the value of the field upon change

click

fired when the device detects a click (longer than touch) against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

dblclick

fired when the device detects a double click against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap

fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

focus

fired when the field gains focus

Event properties

source	the source object that fired the event
type	the name of the event fired
value	the value of the field upon focus

return

fired when the field return key is pressed on the keyboard

Event properties

source	the source object that fired the event
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates

y	the y point of the event, in receiving view coordinates
----------	---

touchend	fired when a touch event is completed
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart	fired as soon as the device detects a gesture
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap	fired when the device detects a two-finger tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Basic Text Field with rounded border

Create a simple text field with a round border style.

```
var tf1 = Titanium.UI.createTextField({
    color:'#336699',
    height:35,
    top:10,
    left:10,
    width:250,
    borderStyle:Titanium.UI.INPUT_BORDERSTYLE_ROUNDED
});
```

Notes

On iPhone, the borderStyle can be set with a constant from Titanium.UI.

Both Text Areas and Text Fields can control the buttons displayed in a button bar above the keyboard when it's visible.

Example using a custom keyboard toolbar:

```
var textfield = Titanium.UI.createTextField({
    color:'#336699',
    value:'Focus to see keyboard w/ toolbar',
    height:35,
    width:300,
    top:10,
    borderStyle:Titanium.UI.INPUT_BORDERSTYLE_ROUNDED,
    keyboardToolbar:[flexSpace,camera, flexSpace,tf,flexSpace, send,flexSpace],
    keyboardToolbarColor: '#999',
    keyboardToolbarHeight: 40,
});
```

Titanium.UI.TextField.add

function of Titanium.UI.TextField
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.TextField.addEventListener

function of Titanium.UI.TextField
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.TextField.animate

function of Titanium.UI.TextField

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.TextField.blur

function of Titanium.UI.TextField

force the field to lose focus

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TextField.fireEvent

function of Titanium.UI.TextField

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.TextField.focus

function of Titanium.UI.TextField

force the field to gain focus

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TextField.hasText

function of Titanium.UI.TextField

return boolean (true) if the field has text

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TextField.hide

function of Titanium.UI.TextField

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TextField.remove

function of Titanium.UI.TextField

remove a previously add view from the view hierarchy

Arguments



Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.TextField.removeEventListener

function of Titanium.UI.TextField
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.TextField.show

function of Titanium.UI.TextField
make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.TextField.toImage

function of Titanium.UI.TextField
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.Toolbar

object of Titanium.UI

A Toolbar is created by the method Titanium.UI.createToolbar. A Toolbar can be placed at the bottom of a window and contain buttons.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListenert	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that

		value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
doubletap	<p>fired when the device detects a double tap against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
singletap	<p>fired when the device detects a single tap against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
swipe	fired when the device detects a swipe (left or right) against the view										

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart

fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofinger tap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Titanium.UI.Toolbar.add

function of Titanium.UI.Toolbar
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.Toolbar.addEventListener

function of Titanium.UI.Toolbar
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Toolbar.animate

function of Titanium.UI.Toolbar
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.Toolbar.fireEvent

function of Titanium.UI.Toolbar
fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Toolbar.hide

function of Titanium.UI.Toolbar
hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Toolbar.remove

function of Titanium.UI.Toolbar

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hiearchy

Return Type

void

Titanium.UI.Toolbar.removeEventListener

function of Titanium.UI.Toolbar

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Toolbar.show

function of Titanium.UI.Toolbar

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Toolbar.toImage

function of Titanium.UI.Toolbar

return a Blob image of the rendered view

Arguments

Name	Type	Description
------	------	-------------

f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately
----------	----------	--

Return Type

object

Titanium.UI.View

object of Titanium.UI

The View is an empty drawing surface or container. The View is created by the method Titanium.UI.createView.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListenerr	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view

backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view

width	float,string	property for the view width. can either be `auto` , a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
click	fired when the device detects a click (longer than touch) against the view Event properties <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	fired when the device detects a double click against the view Event properties <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
doubletap	fired when the device detects a double tap against the view Event properties <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
singletap	fired when the device detects a single tap against the view Event properties <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired				
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Round View Example

Create a rounded view.

```
var view = Titanium.UI.createView({
    borderRadius:10,
    backgroundColor:'red',
    width:50,
    height:50
});
window.add(view);
```

Titanium.UI.View.add

function of Titanium.UI.View
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.View.addEventListener

function of Titanium.UI.View

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.View.animate

function of Titanium.UI.View

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.View.fireEvent

function of Titanium.UI.View

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

Titanium.UI.View.hide

function of Titanium.UI.View
hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.View.remove

function of Titanium.UI.View
remove a previously add view from the view hiearchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hiearchy

Return Type

void

Titanium.UI.View.removeEventListener

function of Titanium.UI.View
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.View.show

function of Titanium.UI.View
make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.View.toImage

function of Titanium.UI.View

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.WebView

object of Titanium.UI

The Web View allows you to open an HTML5 based view which can load either local or remote content. The content can be any valid web content such as HTML, PDF, SVG or other WebKit supported content types. The Web View is created by the method Titanium.UI.createWebView.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
canGoBack	returns true if the webview can go back in history
canGoForward	returns true if the webview can go forward in history
evalJS	invoke JavaScript inside the context of the webview and optionally, return a result
fireEvent	fire a synthesized event to the views listener
goBack	

	go back one entry in history to the previous page
goForward	go forward one entry in history to the page before the current page
hide	hide the view
reload	reload the current webpage
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
repaint	force the webview to repaint its contents
setBasicAuthentication	set the basic authentication for the webview instance to be used on subsequent url requests
show	make the view visible
stopLoading	stop loading a currently loading page
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color for the webview
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that

		by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
data	object	a data blob or file that is used to load the web document
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
html	string	the html content of the web document
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
loading	boolean	boolean indicating if the webview is loading content
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
scalesPageToFit	boolean	whether the webview should scale it's contents or not
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.

touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
url	string	the url to the web document. this property will change as the content of the webview changes (such as from internal hyperlinks, etc)
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
beforeload	<p>fired before the webview starts loading its content</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>url</td><td>the url of the web document</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired	url	the url of the web document				
source	the source object that fired the event										
type	the name of the event fired										
url	the url of the web document										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										

doubletap	fired when the device detects a double tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
error	fired when the webview cannot load the content
Event properties	
message	the error message
source	the source object that fired the event
type	the name of the event fired
url	the url of the web document
load	fired when the webview is loaded
Event properties	
source	the source object that fired the event
type	the name of the event fired
url	the url of the web document
singletap	fired when the device detects a single tap against the view
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
swipe	fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart

fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Basic Web View to external URL example

Create a web view to a remote URL and open the window as modal.

```
var webview = Titanium.UI.createWebView({url:'http://www.appcelerator.com'});
var window = Titanium.UI.createWindow();
window.add(webview);
window.open({modal:true});
```

Notes

Creating webviews are more expensive than creating pure native views because of the requirement to load the HTML browser into memory.

Since a webview internally wants to handle its own events, scrolling and other related touch events against its own view surface, you cannot have both Titanium style events against the webview instance and internal Javascript events in the DOM. You must choose between one or the other.

Titanium.UI.WebView.add

function of Titanium.UI.WebView
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this view's hierarchy

Return Type

Titanium.UI.WebView.addEventListener

function of Titanium.UI.WebView

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.WebView.animate

function of Titanium.UI.WebView

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.WebView.canGoBack

function of Titanium.UI.WebView

returns true if the webview can go back in history

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.UI.WebView.canGoForward

Titanium.UI.WebView.goBack

function of Titanium.UI.WebView

returns true if the webview can go forward in history

Arguments

This function takes no arguments.

Return Type

boolean

Titanium.UI.WebView.evalJS

function of Titanium.UI.WebView

invoke JavaScript inside the context of the webview and optionally, return a result

Arguments

Name	Type	Description
content	string	JavaScript code as a string. The code will be evaluated inside the webview context.

Return Type

string

Titanium.UI.WebView.fireEvent

function of Titanium.UI.WebView

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.WebView.goBack

function of Titanium.UI.WebView

go back one entry in history to the previous page

Arguments

This function takes no arguments.

This function takes no arguments.

Return Type

void

Titanium.UI.WebView.goForward

function of Titanium.UI.WebView

go forward one entry in history to the page before the current page

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.WebView.hide

function of Titanium.UI.WebView

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.WebView.reload

function of Titanium.UI.WebView

reload the current webpage

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.WebView.remove

function of Titanium.UI.WebView

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.WebView.removeEventListener

function of Titanium.UI.WebView

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.WebView.repaint

function of Titanium.UI.WebView

force the webview to repaint its contents

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.WebView.setBasicAuthentication

function of Titanium.UI.WebView

set the basic authentication for the webview instance to be used on subsequent url requests

Arguments

Name	Type	Description
username	string	the username
password	string	the password

Return Type

void

Titanium.UI.WebView.show

function of Titanium.UI.WebView

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.WebView.stopLoading

function of Titanium.UI.WebView

stop loading a currently loading page

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.WebView.toImage

function of Titanium.UI.WebView

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.Window

object of Titanium.UI

The Window is an empty drawing surface or container. The Window is created by the method Titanium.UI.createWindow. Unlike Views, Windows can be opened and closed and can have special display properties such as `fullscreen` or `modal`.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
close	close the window
fireEvent	fire a synthesized event to the views listener
hide	hide the view
open	open the window
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backButtonTitle	string	title for the back button. only available in iPhone. this is only valid when the window is a child of a tab.
backButtonTitleImage	string	url to an image to show as the back button. only available in iPhone. this is only valid when the window is a child of a tab.
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.

backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
barColor	string	web named color or hex value for the color of the nav bar. only available in iPhone.
barImage	string	url to a local image to place as the background of the nav bar. only available in iPhone.
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
exitOnClose	boolean	(Android only.) Boolean indicates if the application should exit when the Android back button is pressed while the window is being shown. You can only set this as a createWindow({...}) option. Setting it after window creation will no effect.
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
fullscreen	boolean	boolean indicates if the window is fullscreen (no device chrome)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
leftNavButton	object	view to show in the left nav bar area. only available in iPhone.

modal	boolean	boolean to indicate if the window should be opened modal in front of other windows
navBarHidden	boolean	for modal windows, hide the nav bar (true) or show the nav bar (false, default).
opacity	float	the opacity from 0.0-1.0
orientationModes	array	array of orientation mode constants defined in [Titanium.UI]
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
rightNavBar	object	view to show in the right nav bar area. only available in iPhone.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
tabBarHidden	boolean	boolean to indicate if the tab bar should be hidden. this is only valid when the window is a child of a tab.
title	string	title of the window.
titleControl	object	view to show in the title area. only available in iPhone.
titleImage	string	url to a image that show in the title area. only available in iPhone.
titlePrompt	string	title prompt for the window. only available in iPhone.
toolbar	array	array of button objects to show in the toolbar of the window. only available in iPhone. this is only valid when the window is a child of a tab.
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
translucent	boolean	boolean to indicate if the nav bar is translucent. only available in iPhone.
url	string	url to a JavaScript file with the windows instructions. this window will create a new JavaScript sub-context that will run in its own thread and global variable space.

visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description				
android:back	<p>fired when the back button is released. Setting a listener is interpreted as the handler for that button. To restore default behavior, remove the listener. It is recommended that you only have one handler per heavyweight window (Android Only)</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired
source	the source object that fired the event				
type	the name of the event fired				
android:camera	<p>fired when the camera button is released. Setting a listener is interpreted as the handler for that button. To restore default behavior, remove the listener. It is recommended that you only have one handler per heavyweight window (Android Only)</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired
source	the source object that fired the event				
type	the name of the event fired				
android:focus	<p>fired when the camera button is half-pressed then released. Setting a listener is interpreted as the handler for that button. To restore default behavior, remove the listener. It is recommended that you only have one handler per heavyweight window (Android Only)</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired
source	the source object that fired the event				
type	the name of the event fired				
android:search	<p>fired when the search button is released. Setting a listener is interpreted as the handler for that button. To restore default behavior, remove the listener. It is recommended that you only have one handler per heavyweight window (Android Only)</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired
source	the source object that fired the event				
type	the name of the event fired				
android:voldown	<p>fired when the volume down button is released. Setting a listener is interpreted as the handler for that button. To restore default behavior, remove the listener. It is recommended that you only have one handler per heavyweight window (Android Only)</p>				

Event properties

source	the source object that fired the event
type	the name of the event fired

android:volup

fired when the volume up button is released. Setting a listener is interpreted as the handler for that button. To restore default behavior, remove the listener. It is recommended that you only have one handler per heavyweight window (Android Only)

Event properties

source	the source object that fired the event
type	the name of the event fired

blur

fired when the window loses focus

Event properties

source	the source object that fired the event
type	the name of the event fired

click

fired when the device detects a click (longer than touch) against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

close

fired when the window is closed

Event properties

source	the source object that fired the event
type	the name of the event fired

dblclick

fired when the device detects a double click against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
--------------------	--

source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap

fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

focus

fired when the window gains focus

Event properties

source	the source object that fired the event
type	the name of the event fired

open

fired when the window is opened

Event properties

source	the source object that fired the event
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates

y

the y point of the event, in receiving view coordinates

touchstart

fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples**Full Screen Window example**

Create a fullscreen window with a red background.

```
var window = Titanium.UI.createWindow({  
    backgroundColor:'red'  
});  
window.open({fullscreen:true});
```

Notes**Sub-contexts**

Windows can be loaded from another Javascript file by specifying the property `url` and reference a file relative to your application `Resources` folder. It's important to note that Titanium will refuse to load Javascript files from a remote URL. Loading remote Javascript from a URL and providing it with the full capabilities of the Titanium API would be very dangerous.

When you Window is loaded from a separate Javascript file, the code will be executed in a separate Javascript context (called a "sub-context") than your `app.js` global context. It will also execute in its own separate thread.

The special property `Titanium.UI.currentWindow` is available inside a sub-context that points to the Javascript instance by reference in the global context.

Passing Data

By default, sub-context variables cannot access Javascript references in the global context. However, the global context can allow its sub-contexts the ability to have access to properties and functions in the global context by reference assignment. An example best illustrates how to do this.

In `app.js`, you could define a function and a property.

```
var a = 1;
function b()
{
    return "hello";
}
```

Now, you can create a new Window - let's call it `foo.js` - in your `app.js`.

```
var w = Titanium.UI.createWindow({
    url:'foo.js'
});
```

To give your new window access to both `a` and `b`, you would need to assign them to the new window reference `w`.

```
w.a = a;
w.b = b;
```

Of course, you could give them a different name than their original names. However, what name you give them in your `app.js` is how you will reference them in `foo.js`. Now, let's look at the code for `foo.js`.

```
alert("b() = "+Titanium.UI.currentWindow.b());
```

In the above code, the `foo.js` will execute the function `b` in the global context and return the result. Since `b` is a reference (vs. a copy by value), any changes made to these Javascript references will be immediately available in all sub-contexts.

Events

If you'd like to send events to a window from the global context and vice versa, you could use the built-in event mechanism. For example, you could define a custom event called `foo`. The window could listen for this event and then respond with some action. For example, in your sub-context you might define:

```
Titanium.UI.currentWindow.addEventListener('foo',function(e)
{
    Titanium.API.info("foo event received = "+JSON.stringify(e));
});
```

You could now fire the event from `app.js` like this:

```
var window = Titanium.UI.createWindow({
    url:'bar.js'
});
window.open();
window.fireEvent('foo',{a:'b'});
```

It's worth noting two important limitations of the example above:

- You must open the event before you can send events to it. You also may have to fire the event after a specified amount of time if you intend to immediately send data to the window. This is because windows are opened asynchronously and on a separate thread than the caller thread.
- You can only send JSON-serializable data in a `fireEvent`. If you attempt to send objects that have function references, they will be null.

Animations

Windows can be animated like any normal View. To transition between 2 windows, you can use the `transition` property on an animation. For example, to flip right-to-left between two windows, you could do the following:

```
var window2 = Titanium.UI.createWindow({url:'foo.js'});
var t = Ti.UI.iPhone.AnimationStyle.FLIP_FROM_LEFT;
window1.animate({view:window2,transition:t});
```

In the above example, the `window2` view will be animated from the right-to-left over `window1`.

Windows can be opened or closed with animation. In the example below, we create a window that will open from small to large with a bounce

effect. This is done by applying a transformation at initialization time that scales the original size of the window to 0. When the window is opened, a new 2D transformation is applied that will scale the window size from 0 to 110% of it's original size and then, after 1/20th of a second, will scale it back to it's original size at 100%. This gives the bounce effect during animation.

```
var t = Titanium.UI.create2DMatrix().scale(0);

// create a window with the initial transform scaled to 0
var w = Titanium.UI.createWindow({
    backgroundColor:'#336699',
    borderWidth:8,
    borderColor:'#999',
    height:400,
    width:300,
    borderRadius:10,
    opacity:0.92,
    transform:t
});

// create first transform to go beyond normal size
var t1 = Titanium.UI.create2DMatrix().scale(1.1);

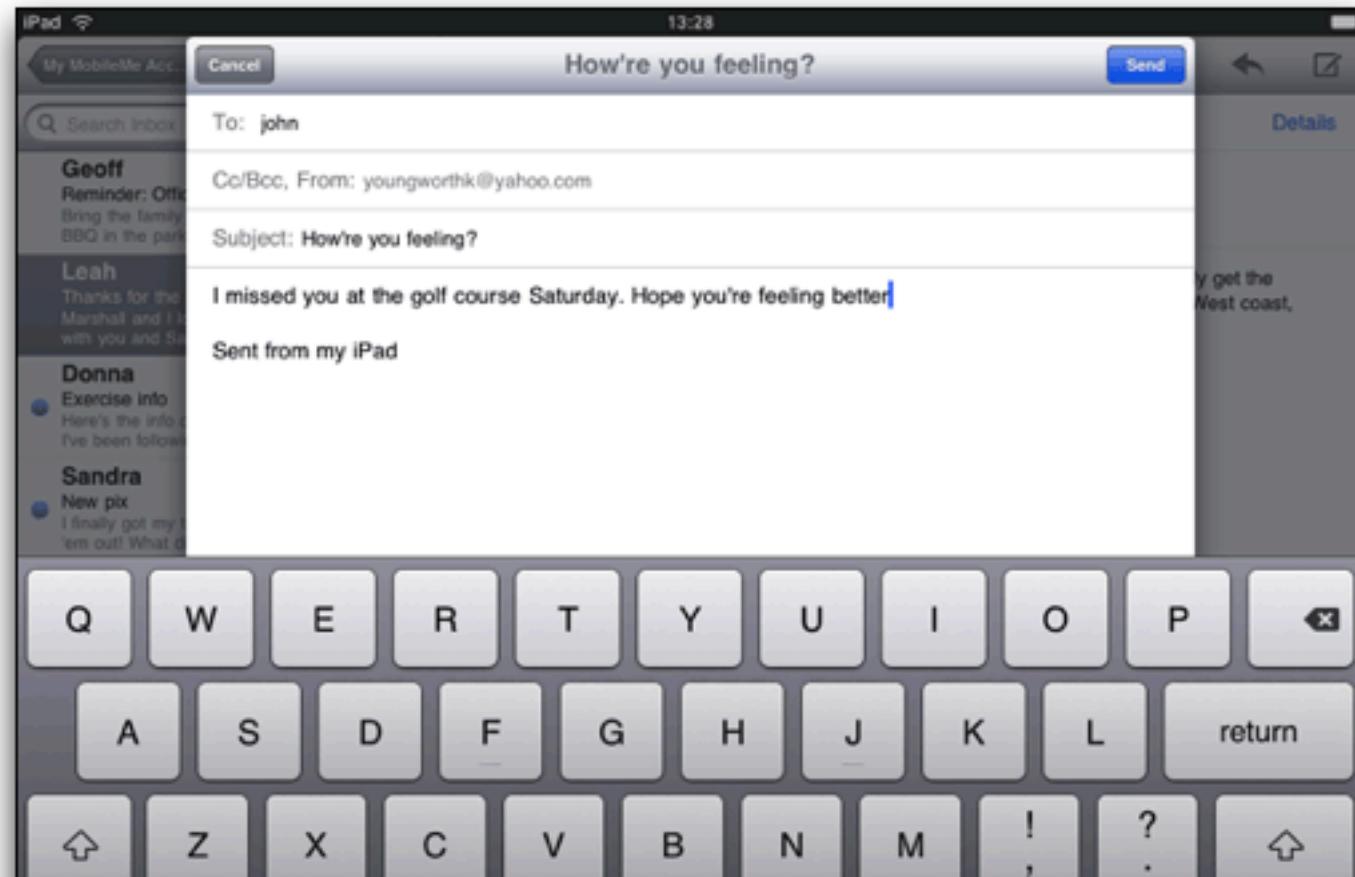
var a = Titanium.UI.createAnimation();
a.transform = t1;
a.duration = 200;

// when this animation completes, scale to normal size
a.addEventListener('complete', function()
{
    // we can use the identity transform to take it back to it's real size
    var t2 = Titanium.UI.create2DMatrix();
    w.animate({transform:t2, duration:200});
});
```

iPad Modal Windows

For iPad, iPhone SDK 3.2 and Titanium 1.2 introduced several new ways of presenting modal windows.

The example below is a modal window.





You can create this type of modal window on iPad with the following code snippet:

```
var window = Titanium.UI.createWindow();
window.open({
    modal:true,
    modalTransitionStyle: Ti.UI.iPhone.MODAL_TRANSITION_STYLE_FLIP_HORIZONTAL,
    modalStyle: Ti.UI.iPhone.MODAL_PRESENTATION_FORMSHEET
})
```

Android "root" Windows

In Android, you may wish to specify that a window which you create (such as the first window) should be considered the root window and that the application should exit when the back button is pressed from that window. This is particularly useful if your application is not using a Tab Group and therefore the splash screen window is appearing whenever you press the back button from your lowest window on the stack.

To indicate that a particular window should cause an application to exit when the back button is pressed, pass `exitOnClose: true` as one of the creation arguments, as shown here:

```
var win = Titanium.UI.createWindow({
    title: 'My Root Window',
    exitOnClose: true
});
```

Titanium.UI.Window.add

function of Titanium.UI.Window
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.Window.addEventListener

function of Titanium.UI.Window
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.Window.animate

function of Titanium.UI.Window

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.Window.close

function of Titanium.UI.Window

close the window

Arguments

Name	Type	Description
options	object	close the window with optional animation or display properties

Return Type

void

Titanium.UI.Window.fireEvent

function of Titanium.UI.Window

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.Window.hide

function of Titanium.UI.Window
hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Window.open

function of Titanium.UI.Window

open the window

Arguments

Name	Type	Description
options	object	open the window with optional animation or display properties

Return Type

void

Titanium.UI.Window.remove

function of Titanium.UI.Window
remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.Window.removeEventListener

function of Titanium.UI.Window
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.Window.show

function of Titanium.UI.Window

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.Window.toImage

function of Titanium.UI.Window

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.addEventListener

function of Titanium.UI

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.create2DMatrix

function of Titanium.UI

create and return an instance of Titanium.UI.2DMatrix

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.2DMatrix

Return Type

object

Titanium.UI.create3DMatrix

function of Titanium.UI

create and return an instance of Titanium.UI.3DMatrix

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.3DMatrix

Return Type

object

Titanium.UI.createActivityIndicator

function of Titanium.UI

create and return an instance of Titanium.UI.ActivityIndicator

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.ActivityIndicator

Return Type

object

Titanium.UI.createAlertDialog

function of Titanium.UI

create and return an instance of Titanium.UI.AlertDialog

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.AlertDialog

Return Type

object

Titanium.UI.createAnimation

function of Titanium.UI

create and return an instance of Titanium.UI.Animation

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.Animation

Return Type

object

Titanium.UI.createButton

function of Titanium.UI

create and return an instance of Titanium.UI.Button

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.Button

Return Type

object

Titanium.UI.createButtonBox

Titanium.UI.createButtonBar

function of Titanium.UI

create and return an instance of Titanium.UI.ButtonBar

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.ButtonBar

Return Type

object

Titanium.UI.createCoverFlowLayout

function of Titanium.UI

create and return an instance of Titanium.UI.CoverFlowLayout

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.CoverFlowLayout

Return Type

object

Titanium.UI.createDashboardItem

function of Titanium.UI

create and return an instance of Titanium.UI.DashboardItem

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.DashboardItem

Return Type

object

Titanium.UI.createDashboardView

function of Titanium.UI

create and return an instance of Titanium.UI.DashboardView

Arguments

Name	Type	Description
------	------	-------------

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.DashboardView

Return Type

object

Titanium.UI.createEmailDialog

function of Titanium.UI

create and return an instance of Titanium.UI.EmailDialog

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.EmailDialog

Return Type

object

Titanium.UI.createImageView

function of Titanium.UI

create and return an instance of Titanium.UI.ImageView

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.ImageView

Return Type

object

Titanium.UI.createLabel

function of Titanium.UI

create and return an instance of Titanium.UI.Label

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.Label

Return Type

object

Titanium.UI.createOptionDialog

function of Titanium.UI

create and return an instance of Titanium.UI.OptionDialog

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.OptionDialog

Return Type

object

Titanium.UI.createPicker

function of Titanium.UI

create and return an instance of Titanium.UI.Picker

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.Picker

Return Type

object

Titanium.UI.createPickerColumn

function of Titanium.UI

create and return an instance of Titanium.UI.PickerColumn

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.PickerColumn

Return Type

object

Titanium.UI.createPickerRow

function of Titanium.UI

create and return an instance of Titanium.UI.PickerRow

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.PickerRow

Return Type

object

Titanium.UI.createProgressBar

function of Titanium.UI

create and return an instance of Titanium.UI.ProgressBar

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.ProgressBar

Return Type

object

Titanium.UI.createScrollView

function of Titanium.UI

create and return an instance of Titanium.UI.ScrollView

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.ScrollView

Return Type

object

Titanium.UI.createScrollView

function of Titanium.UI

create and return an instance of Titanium.UI.ScrollableView

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.ScrollableView

Return Type

Titanium.UI.createSearchBar

function of Titanium.UI

create and return an instance of Titanium.UI.SearchBar

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.SearchBar

Return Type

object

Titanium.UI.createSlider

function of Titanium.UI

create and return an instance of Titanium.UI.Slider

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.Slider

Return Type

object

Titanium.UI.createSwitch

function of Titanium.UI

create and return an instance of Titanium.UI.Switch

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.Switch

Return Type

object

Titanium.UI.createTab

function of Titanium.UI

create and return an instance of Titanium.UI.Tab

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.Tab

Return Type

object

Titanium.UI.createTabGroup

function of Titanium.UI

create and return an instance of Titanium.UI.TabGroup

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.TabGroup

Return Type

object

Titanium.UI.createTabbedBar

function of Titanium.UI

create and return an instance of Titanium.UI.TabbedBar

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.TabbedBar

Return Type

object

Titanium.UI.createTableView

function of Titanium.UI

create and return an instance of Titanium.UI.TableView

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.TableView

parameters object (optional) a dictionary object properties defined in Titanium.UI.TableView

Return Type

object

Titanium.UI.createTableViewRow

function of Titanium.UI

create and return an instance of Titanium.UI.TableViewRow

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.TableViewRow

Return Type

object

Titanium.UI.createTableViewSection

function of Titanium.UI

create and return an instance of Titanium.UI.TableViewSection

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.TableViewSection

Return Type

object

Titanium.UI.createTextArea

function of Titanium.UI

create and return an instance of Titanium.UI.TextArea

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.TextArea

Return Type

object

Titanium.UI.createTextField

function of Titanium.UI

create and return an instance of Titanium.UI.TextField

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.TextField

Return Type

object

Titanium.UI.createToolbar

function of Titanium.UI

create and return an instance of Titanium.UI.Toolbar

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.Toolbar

Return Type

object

Titanium.UI.createView

function of Titanium.UI

create and return an instance of Titanium.UI.View

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.View

Return Type

object

Titanium.UI.createWebView

function of Titanium.UI

create and return an instance of Titanium.UI.WebView

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.WebView

Return Type

object

Titanium.UI.createWindow

function of Titanium.UI

create and return an instance of Titanium.UI.Window

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.Window

Return Type

object

Titanium.UI.fireEvent

function of Titanium.UI

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.iOS

submodule of Titanium.UI

The Apple iOS specific UI capabilities. All properties, methods and events in this namespace will only work on the Apple iOS related devices.

Objects

Name	Description
Titanium.UI.iOS.AdView	The AdView is a view for display Apple iAds. The View is created by the method

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
createAdView	create and return an instance of Titanium.UI.iOS.AdView
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener

Properties

This module has no properties

Events

This module has no events

Titanium.UI.iOS.AdView

object of Titanium.UI.iOS

The AdView is a view for display Apple iAds. The View is created by the method Titanium.UI.iOS.createAdView.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
cancelAction	A banner view action can cover your application's user interface. However, your application continues to run, and receives events normally. If your application receives an event that requires the user's attention, it can programmatically cancel the action and uncover its interface by calling cancelAction. Canceling actions frequently can cause a loss of revenue for your application.
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously add view from the view hiearchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
SIZE_320x50	string	constant for 320x50 ad sizes
SIZE_480x32	string	constant for 480x32 ad sizes
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the

		parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
action	<p>called when a banner action is performed</p> <p>Event properties</p> <table border="1"> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	source	the source object that fired the event	type	the name of the event fired						
source	the source object that fired the event										
type	the name of the event fired										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										

dblclick

fired when the device detects a double click against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap

fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

error

called when a banner could not be loaded

Event properties

message	the error message
source	the source object that fired the event
type	the name of the event fired

load

called when a banner is loaded and displayed

Event properties

source	the source object that fired the event
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe	fired when the device detects a swipe (left or right) against the view
Event properties	
direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel	fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend	fired when a touch event is completed
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Titanium.UI.iOS.AdView.add

function of Titanium.UI.iOS.AdView
add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.iOS.AdView.addEventListener

function of Titanium.UI.iOS.AdView
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.iOS.AdView.animate

function of Titanium.UI.iOS.AdView

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.iOS.AdView.cancelAction

function of Titanium.UI.iOS.AdView

A banner view action can cover your application's user interface. However, your application continues to run, and receives events normally. If your application receives an event that requires the user's attention, it can programmatically cancel the action and uncover its interface by calling cancelAction. Canceling actions frequently can cause a loss of revenue for your application.

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.iOS.AdView.fireEvent

function of Titanium.UI.iOS.AdView

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event

name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.iOS.AdView.hide

function of Titanium.UI.iOS.AdView

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.iOS.AdView.remove

function of Titanium.UI.iOS.AdView

remove a previously add view from the view hiearchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hiearchy

Return Type

void

Titanium.UI.iOS.AdView.removeEventListener

function of Titanium.UI.iOS.AdView

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.iOS.AdView.show

function of Titanium.UI.iOS.AdView

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.iOS.AdView.toImage

function of Titanium.UI.iOS.AdView

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.iOS.addEventListener

function of Titanium.UI.iOS

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.iOS.createAdView

function of Titanium.UI.iOS

create and return an instance of Titanium.UI.iOS.AdView

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.iOS.AdView

Return Type

object

Titanium.UI.iOS.fireEvent

function of Titanium.UI.iOS

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.iOS.removeEventListener

function of Titanium.UI.iOS

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

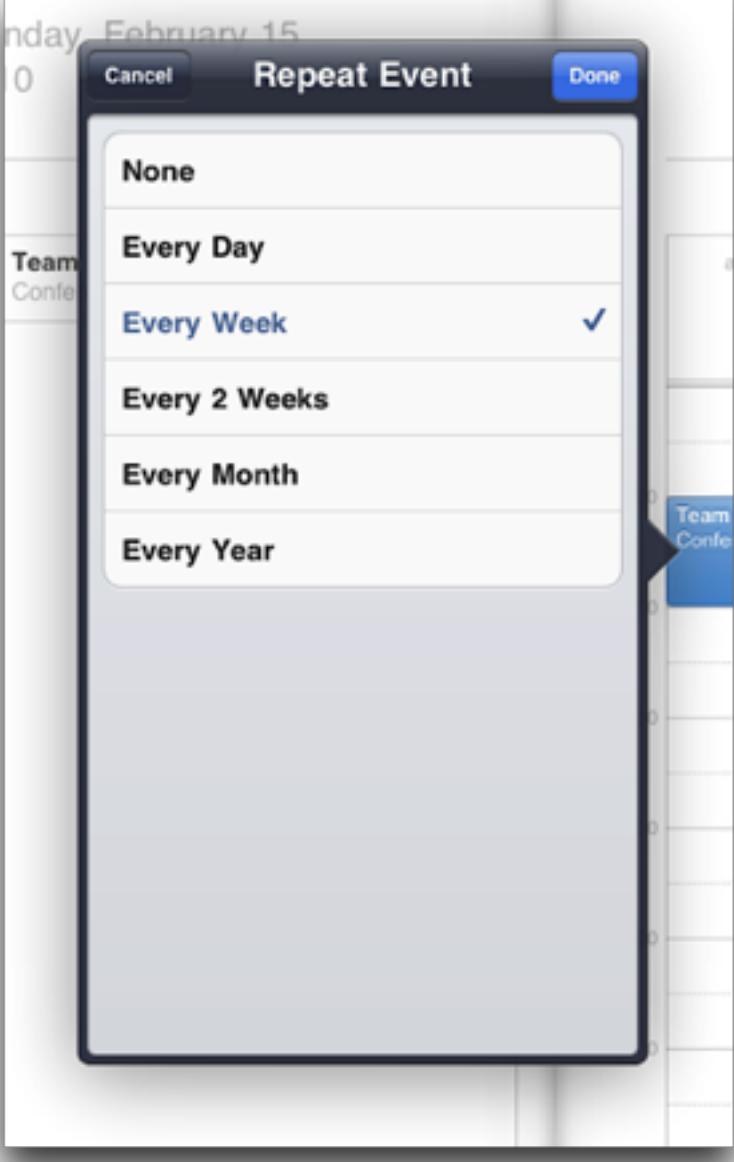
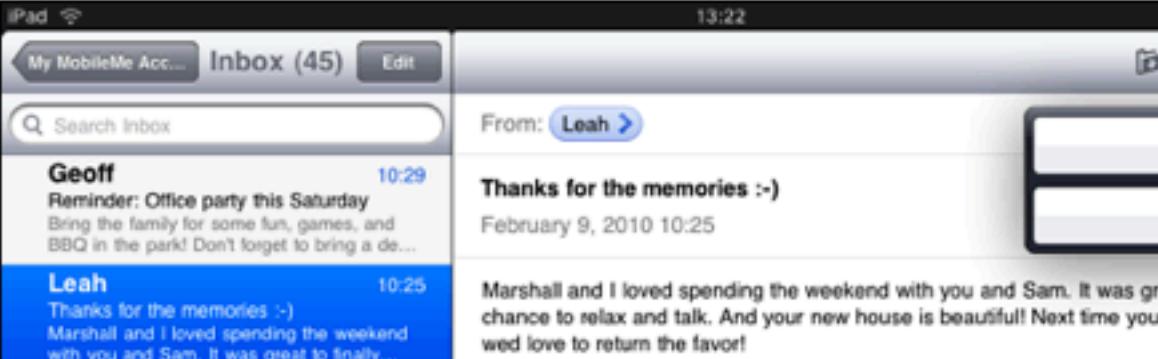
Titanium.UI.iPad

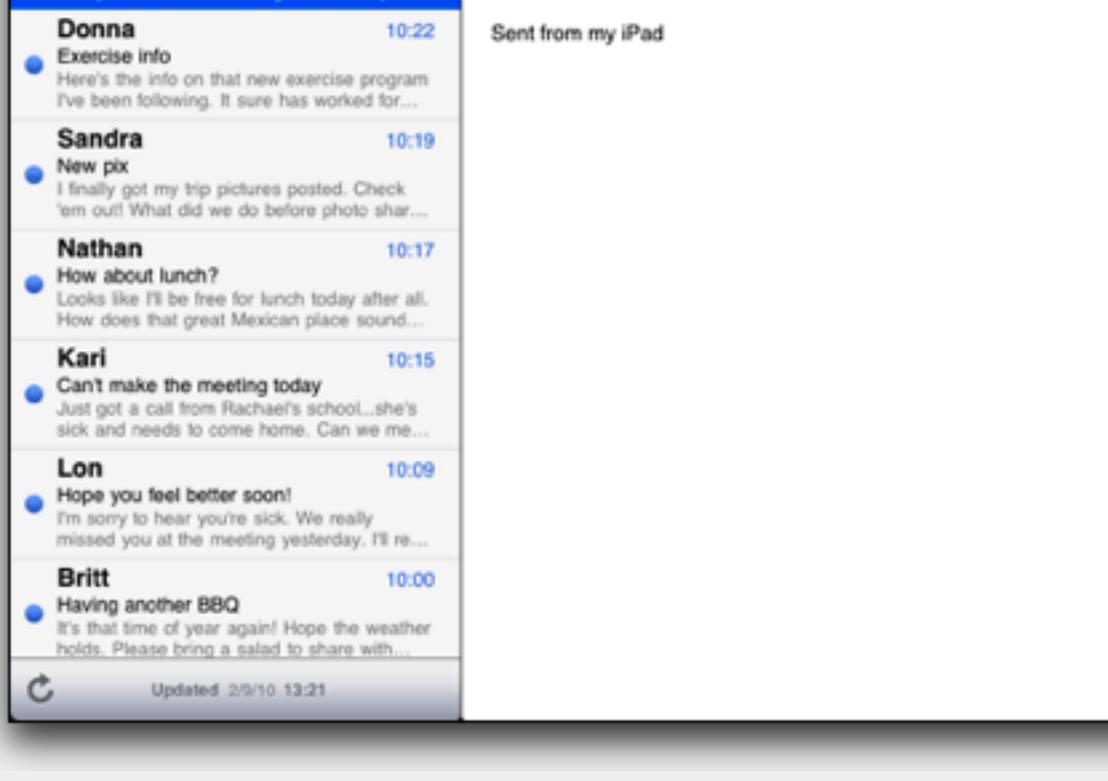
submodule of Titanium.UI

The iPad specific UI capabilities. All properties, methods and events in this namespace will only work on the Apple iPad related devices. To develop for iPad, you will need the Apple iPhone SDK 3.2 or later and Titanium Mobile SDK 1.2 or later.

For iPad UI programming guidelines, please review the iPad Human Interface Guidelines.

Objects

Name	Description
Titanium.UI.iPad.Popover	<p>A Popover is used to manage the presentation of content in a popover. You use popovers to present information that does not take over the entire screen like a modal view does. The popover content is layered on top of your current type of window. The popover remains visible until the user taps outside of the popover window or you explicitly close it by calling the method <code>Titanium.UI.iPad.createPopover</code>.</p> 
Titanium.UI.iPad.SplitWindow	<p>A SplitWindow is a window that manages the presentation of two side-by-side view controllers. You use this controller to create a detail interface, in which the left-side view presents a list of items and the right-side presents details of the selected item. It is designed for use exclusively on iPad devices. The SplitWindow is created by the method <code>Titanium.UI.iPad.createSplitWindow</code>.</p> 



Methods

Name	Description
<code>addEventListener</code>	add an event listener for the instance to receive view triggered events
<code>createPopover</code>	create and return an instance of Titanium.UI.iPad.Popover
<code>createSplitWindow</code>	create and return an instance of Titanium.UI.iPad.SplitWindow
<code>fireEvent</code>	fire a synthesized event to the views listener
<code>removeEventListener</code>	remove a previously added event listener

Properties

Name	Type	Description
<code>POPOVER_ARROW_DIRECTION_ANY</code>	int	An arrow that points in any direction.
<code>POPOVER_ARROW_DIRECTION_DOWN</code>	int	An arrow that points downward.
<code>POPOVER_ARROW_DIRECTION_LEFT</code>	int	An arrow that points toward the left.
<code>POPOVER_ARROW_DIRECTION_RIGHT</code>	int	An arrow that points toward the right.
<code>POPOVER_ARROW_DIRECTION_UNKNOWN</code>	int	The status of the arrow is currently unknown.
<code>POPOVER_ARROW_DIRECTION_UP</code>	int	An arrow that points upward.

Events

This module has no events

Notes

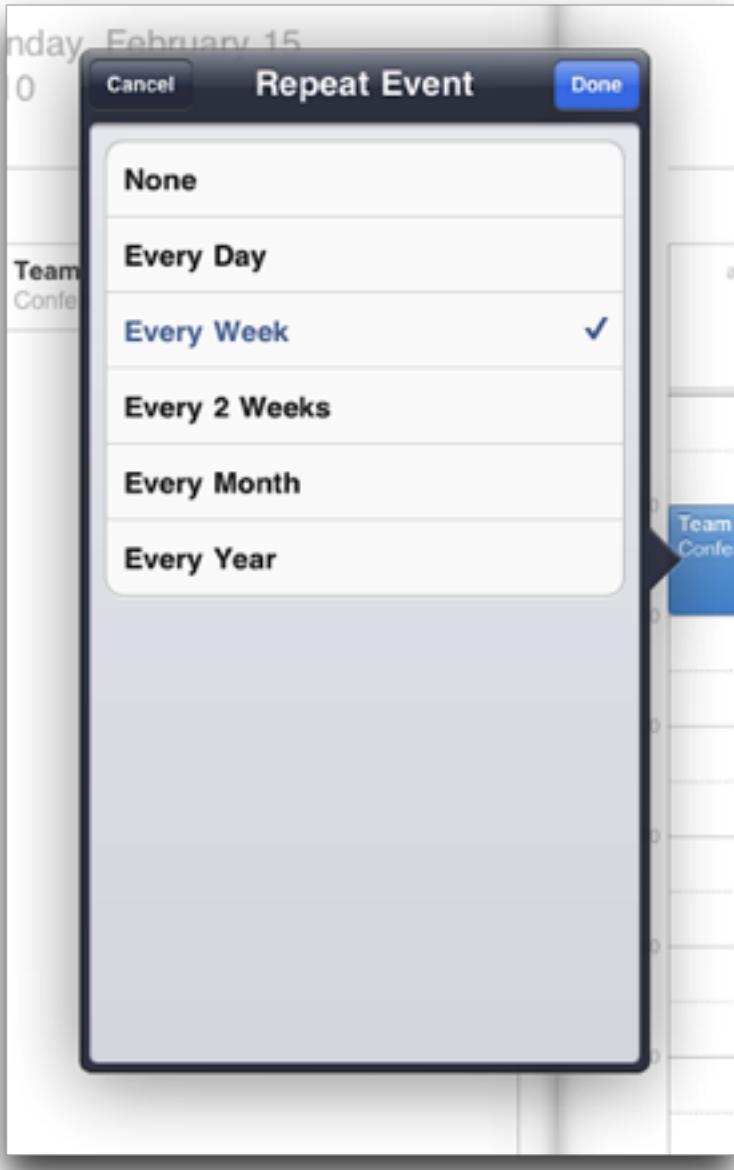
Custom Fonts

The iPad supports the ability to embed your own fonts in your iPad application. To add your own fonts, you should add the

Titanium.UI.iPad.Popover

object of Titanium.UI.iPad

A Popover is used to manage the presentation of content in a popover. You use popovers to present information temporarily but in a way that does not take over the entire screen like a modal view does. The popover content is layered on top of your existing content in a special type of window. The popover remains visible until the user taps outside of the popover window or you explicitly dismiss it. The Popover is created by the method Titanium.UI.iPad.createPopover.



Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the popover
remove	remove a previously add view from the view hiearchy
removeEventListener	remove a previously added event listener
setHeight	change the height of the popover
setWidth	change the width of the popover
show	show the popover
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
arrowDirection	int	return the arrow direction of the popover
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle

		original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float	height of the popover
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
leftNavButton	object	the left button in the nav area of the popover
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
title	string	the title of the nav area of the popover
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)

transform	object	the transformation matrix to apply to the view
visible	boolean	boolean to indicate if the popover is visible
width	float	width of the popover
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired				
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										

x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

doubletap fired when the device detects a double tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

hide

fired when the popover is hidden

Event properties

source	the source object that fired the event
type	the name of the event fired

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe	fired when the device detects a swipe (left or right) against the view
Event properties	
direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
touchcancel	fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
touchend	fired when a touch event is completed
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates
touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred
Event properties	
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates

y

the y point of the event, in receiving view coordinates

touchstart

fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples

Simple Popover

In this example, we create a simple popover and position it near the button

```
var popover = Ti.UI.iPad.createPopover({height:100,width:100});  
popover.add(view);  
popover.show({view:button});
```

Popover with title and nav button

In this example, we create a popover with a title and right navigation button.

```
var popover = Ti.UI.iPad.createPopover({  
    width:250,  
    height:100,  
    title: "Yo Yo",  
    rightNavButton:button  
});
```

Titanium.UI.iPad.Popover.add

function of Titanium.UI.iPad.Popover

add a child to the view hierarchy

Arguments

Name	Type	Description
view	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.iPad.Popover.addEventListener

function of Titanium.UI.iPad.Popover

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.iPad.Popover.animate

function of Titanium.UI.iPad.Popover

animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.iPad.Popover.fireEvent

function of Titanium.UI.iPad.Popover
fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.iPad.Popover.hide

function of Titanium.UI.iPad.Popover

hide the popover

Arguments

Name	Type	Description
options	object	dictionary with optional boolean property <code>animated</code> which indicates if the popover should be hidden with animation (defaults to true) or not

Return Type

void

Titanium.UI.iPad.Popover.remove

function of Titanium.UI.iPad.Popover
remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.iPad.Popover.removeEventListerner

function of Titanium.UI.iPad.Popover
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.iPad.Popover.setHeight

function of Titanium.UI.iPad.Popover

change the height of the popover

Arguments

Name	Type	Description
height	int,string	height of the popover

Return Type

void

Titanium.UI.iPad.Popover.setWidth

function of Titanium.UI.iPad.Popover

change the width of the popover

Arguments

Name	Type	Description

width

int,string width of the popover

Return Type

void

Titanium.UI.iPad.Popover.show

function of Titanium.UI.iPad.Popover

show the popover

Arguments

Name	Type	Description
options	object	dictionary of the following properties: <code>rect</code> , <code>animated</code> and <code>view</code> . The <code>view</code> property should specify the view relative to where the popover should display. <code>animated</code> is a boolean indicating if the showing of the popover should be animated, defaults to true. The property <code>rect</code> is a dictionary with the following properties: <code>x</code> , <code>y</code> , <code>width</code> and <code>height</code> , where the rectangle in <code>view</code> at which to anchor the popover.

Return Type

void

Titanium.UI.iPad.Popover.toImage

function of Titanium.UI.iPad.Popover

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

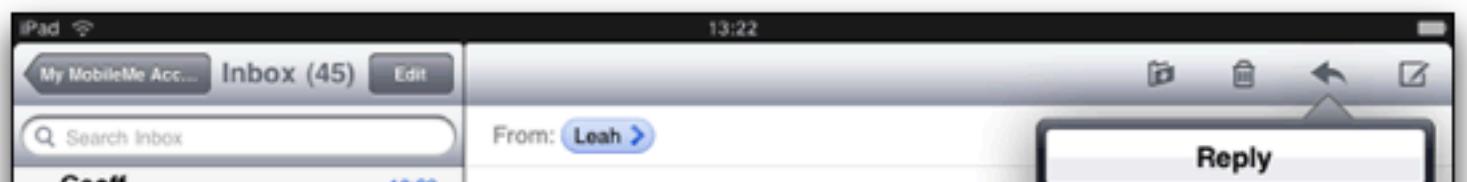
Return Type

object

Titanium.UI.iPad.SplitWindow

object of Titanium.UI.iPad

A SplitWindow is a window that manages the presentation of two side-by-side view controllers. You use this class to implement a master-detail interface, in which the left-side view presents a list of items and the right-side presents details of the selected item. The SplitView is for use exclusively on iPad devices. The SplitWindow is created by the method `Titanium.UI.iPad.createSplitWindow`.



Geoff 10:29
Reminder: Office party this Saturday
Bring the family for some fun, games, and
BBQ in the park! Don't forget to bring a de...

Leah 10:25
Thanks for the memories :-)

Marshall and I loved spending the weekend
with you and Sam. It was great to finally...

Donna 10:22
Exercise info

Here's the info on that new exercise program
I've been following. It sure has worked for...

Sandra 10:19
New pix

I finally got my trip pictures posted. Check
'em out! What did we do before photo shar...

Nathan 10:17
How about lunch?

Looks like I'll be free for lunch today after all.
How does that great Mexican place sound...

Kari 10:15
Can't make the meeting today

Just got a call from Rachael's school...she's
sick and needs to come home. Can we me...

Lon 10:09
Hope you feel better soon!

I'm sorry to hear you're sick. We really
missed you at the meeting yesterday. I'll re...

Britt 10:00
Having another BBQ

It's that time of year again! Hope the weather
holds. Please bring a salad to share with...



Updated: 2/9/10 13:21

Thanks for the memories :-)

February 9, 2010 10:25

Forward

Marshall and I loved spending the weekend with you and Sam. It was great to finally get the chance to relax and talk. And your new house is beautiful! Next time you're on the West coast, we'd love to return the favor!

Sent from my iPad

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
fireEvent	fire a synthesized event to the views listener
hide	hide the view
remove	remove a previously added view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation

backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
detailView	object	view for the detail view section of the SplitWindow
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
masterView	object	view for the master view section of the SplitWindow

opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS, Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)
top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"> <tr> <td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr> <tr> <td>source</td><td>the source object that fired the event</td></tr> <tr> <td>type</td><td>the name of the event fired</td></tr> <tr> <td>x</td><td>the x point of the event in receiving view coordinates</td></tr> <tr> <td>y</td><td>the y point of the event, in receiving view coordinates</td></tr> </table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
doubletap	fired when the device detects a double tap against the view										

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchmove

fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart

fired as soon as the device detects a gesture

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap

fired when the device detects a two-finger tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

visible

fired when the masterView or detailView becomes visible.

Event properties

button	for <code>view</code> view type, the button that is automatically wired to control the master view popover
popover	for either <code>popover</code> or <code>detail</code> view types, the popover instance
source	the source object that fired the event
type	the name of the event fired
view	the type of view becoming visible. either <code>master</code> , <code>popover</code> or <code>detail</code> .

Code Examples

Split Window Example

This is an example of a Split Window.

```
var win = Ti.UI.createWindow();

var nav = Ti.UI.iPhone.createNavigationGroup({
    window: win
});

var splitwin = Ti.UI.iPad.createSplitWindow({
    detailView:nav,
    masterView:masterView
});

splitwin.addEventListener('visible',function(e)
{
    if (e.view == 'detail')
    {
        e.button.title = "Master";
        win.leftNavButton = e.button;
    }
    else if (e.view == 'master')
    {
        e.button.title = "Detail";
        win.rightNavButton = e.button;
    }
});
```

```
        else if (e.view == master )  
    {  
        win.leftNavButton = null;  
    }  
});  
  
splitwin.open();
```

Notes

The SplitWindow inherits its methods and properties from Titanium.UI.Window.

The `masterView` and `detailView` properties are required in the constructor of the SplitWindow and cannot be changed once set.

The SplitWindow is a top-level window and cannot be contained within another window or view.

Titanium.UI.iPad.SplitWindow.add

function of Titanium.UI.iPad.SplitWindow
add a child to the view hierarchy

Arguments

Name	Type	Description
<code>view</code>	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.iPad.SplitWindow.addEventListener

function of Titanium.UI.iPad.SplitWindow
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
<code>name</code>	string	name of the event
<code>callback</code>	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.iPad.SplitWindow.animate

function of Titanium.UI.iPad.SplitWindow
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.iPad.SplitWindow.fireEvent

function of Titanium.UI.iPad.SplitWindow
fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.iPad.SplitWindow.hide

function of Titanium.UI.iPad.SplitWindow
hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.iPad.SplitWindow.remove

function of Titanium.UI.iPad.SplitWindow
remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

Titanium.UI.iPad.SplitWindow.removeEventListener

function of Titanium.UI.iPad.SplitWindow
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.iPad.SplitWindow.show

function of Titanium.UI.iPad.SplitWindow
make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.iPad.SplitWindow.toImage

function of Titanium.UI.iPad.SplitWindow
return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.iPad.addEventListener

function of Titanium.UI.iPad
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.iPad.createPopover

function of Titanium.UI.iPad

create and return an instance of Titanium.UI.iPad.Popover

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.iPad.Popover

Return Type

object

Titanium.UI.iPad.createSplitWindow

function of Titanium.UI.iPad

create and return an instance of Titanium.UI.iPad.SplitWindow

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.iPad.SplitWindow

Return Type

object

Titanium.UI.iPad.fireEvent

function of Titanium.UI.iPad

fire a synthesized event to the views listener

Arguments

Name	Type	Description
------	------	-------------

name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.iPad.removeEventListener

function of Titanium.UI.iPad
remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.iPhone

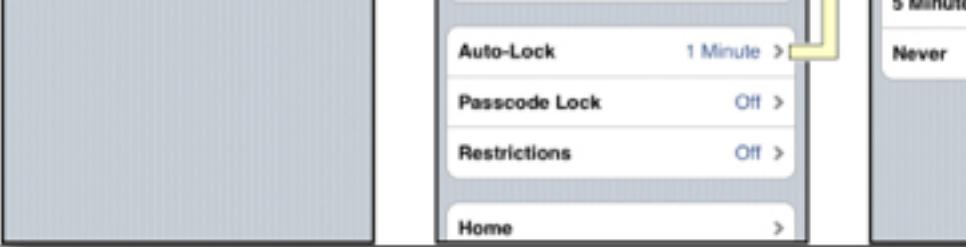
submodule of Titanium.UI

The iPhone/iPad specific UI capabilities. All properties, methods and events in this namespace will only work on the Apple iPhone or iPad related devices.

Objects

Name	Description
Titanium.UI.iPhone.ActivityIndicatorStyle	A set of constants for the styles available for Titanium.UI.ActivityIndicator objects.
Titanium.UI.iPhone.AnimationStyle	A set of constants for the Animation Styles used for transitions.
Titanium.UI.iPhone.NavigationGroup	A Navigation Group implements a specialized view that manages the navigation of hierarchical content. A Group is created by the method Titanium.UI.iPhone.NavigationGroup.





A Navigation Group is very similar to Tab Bars with the exception that they do not maintain an interface bar at the bottom.

Titanium.UI.iPhone.ProgressBarStyle

A set of constants for the bar styles used on the `style` property of Titanium.UI.ProgressBar objects.

Titanium.UI.iPhone.RowAnimationStyle

A set of constants for the Animation Styles used for transition on table view rows.

Titanium.UI.iPhone.ScrollIndicatorStyle

A set of constants for the styles available for Titanium.UI.ActivityIndicator objects.

Titanium.UI.iPhone.StatusBar

A set of constants for the status bar style.

Titanium.UI.iPhone.SystemButtonStyle

A set of constants for the system button styles that can be used for the button `style` property.

Titanium.UI.iPhone.SystemIcon

A set of constants for the system icon styles that can be used on a tab group tab.

Titanium.UI.iPhone.TableViewCellStyle

A set of constants for the style that can be used for the `selectionStyle` property of Titanium.UI.TableViewCell objects.

Titanium.UI.iPhone.TableViewScrollIndicatorPosition

A set of constants for the position value that can be used for the `position` property of Titanium.UI.TableViewScrollIndicator objects when invoking `scrollToIndex`.

Titanium.UI.iPhone.TableViewSeparatorStyle

A set of constants for the style that can be used for the `separatorStyle` property of Titanium.UI.TableView objects.

Titanium.UI.iPhone.TableViewStyle

A set of constants for the style that can be used for the button `style` property of Titanium.UI.TableView objects.

Methods

Name	Description
<code>addEventListener</code>	add an event listener for the instance to receive view triggered events
<code>createNavigationGroup</code>	create and return an instance of Titanium.UI.iPhone.NavigationGroup
<code>fireEvent</code>	fire a synthesized event to the views listener
<code>hideStatusBar</code>	convenience method to hide the status bar

<code>removeEventListener</code>	remove a previously added event listener
<code>showStatusBar</code>	convenience method to show the status bar

Properties

Name	Type	Description
<code>MODAL_PRESENTATION_CURRENT_CONTEXT</code>	int	The view is presented using the same style as its parent window. This is currently only available currently on iPhone/iPad and SDK 3.2+.
<code>MODAL_PRESENTATION_FORMSHEET</code>	int	The width and height of the presented window are smaller than those of the screen and the view is centered on the screen. If the device is in a landscape orientation and the keyboard is visible, the position of the view is adjusted upward so that the view remains visible. All uncovered areas are dimmed to prevent the user from interacting with them. This is currently only available currently on iPhone/iPad and SDK 3.2+.
<code>MODAL_PRESENTATION_FULLSCREEN</code>	int	The presented window covers the screen. This is currently only available currently on iPhone/iPad and SDK 3.2+.
<code>MODAL_PRESENTATION_PAGESHEET</code>	int	The height of the presented window is set to the height of the screen and the view's width is set to the width of the screen in a portrait orientation. Any uncovered areas are dimmed to prevent the user from interacting with them. (In portrait orientations, this option is essentially the same as <code>Titanium.UI.iPhone.MODAL_PRESENTATION_FULLSCREEN</code>). This is currently only available currently on iPhone/iPad and SDK 3.2+.
<code>MODAL_TRANSITION_STYLE_COVER_VERTICAL</code>	int	When the window is presented, its view slides up from the bottom of the screen. On dismissal, the view slides back down. This is the default transition style. This is currently only available currently on iPhone/iPad.
<code>MODAL_TRANSITION_STYLE_CROSSDISSOLVE</code>	int	When the window is presented, the current view fades out while the new view fades in at the same time. On dismissal, a similar type of cross-fade is used to return to the original view. This is currently only available currently on iPhone/iPad.
<code>MODAL_TRANSITION_STYLE_FLIP_HORIZONTAL</code>	int	When the window is presented, the current view initiates a horizontal 3D flip from right-to-left, resulting in the revealing of the new view as if it were on the back of the previous view. On dismissal, the flip occurs from left-to-right, returning to the original view. This is currently only available currently on iPhone/iPad.
<code>MODAL_TRANSITION_STYLE_PARTIAL_CURL</code>	int	When the window is presented, a portion of the new view is revealed as it curves around the edge of the previous view. On dismissal, the view returns to its original state. This is currently only available currently on iPhone/iPad.

		<p>When the window is presented, one corner of the current view curls up to reveal the modal view underneath. On dismissal, the curled up page unfurls itself back on top of the modal view. A modal view presented using this transition is itself prevented from presenting any additional modal views. This transition style is supported only if the window is presenting a Titanium.UI.iPhone.MODAL_PRESENTATION_FULLSCREEN modal presentation style. This is currently only available currently on iPhone/iPad and SDK 3.2+.</p>
appBadge	string	set the application badge for the application's icon in the springboard
appSupportsShakeToEdit	boolean	control whether the shake to edit system wide capability is enabled
statusBarHidden	boolean	control the status bar visibility
statusBarStyle	int	constant that controls the status bar color style

Events

This module has no events

Titanium.UI.iPhone.ActivityIndicatorStyle

object of Titanium.UI.iPhone

A set of constants for the styles available for Titanium.UI.ActivityIndicator objects.

Methods

This object has no methods

Properties

Name	Type	Description
BIG	int	The large white style of indicator.
DARK	int	The standard gray style of indicator.
PLAIN	int	The standard white style of indicator (the default).

Events

This object has no events

Titanium.UI.iPhone.AnimationStyle

object of Titanium.UI.iPhone

A set of constants for the Animation Styles used for transitions.

Methods

This object has no methods

Properties

Name	Type	Description
CURL_DOWN	int	Curl downwards during a transition animation
CURL_UP	int	Curl upwards during a transition animation
FLIP_FROM_LEFT	int	Flip from left to right during a transition animation
FLIP_FROM_RIGHT	int	Flip from right to left during a transition animation
NONE	int	No animation

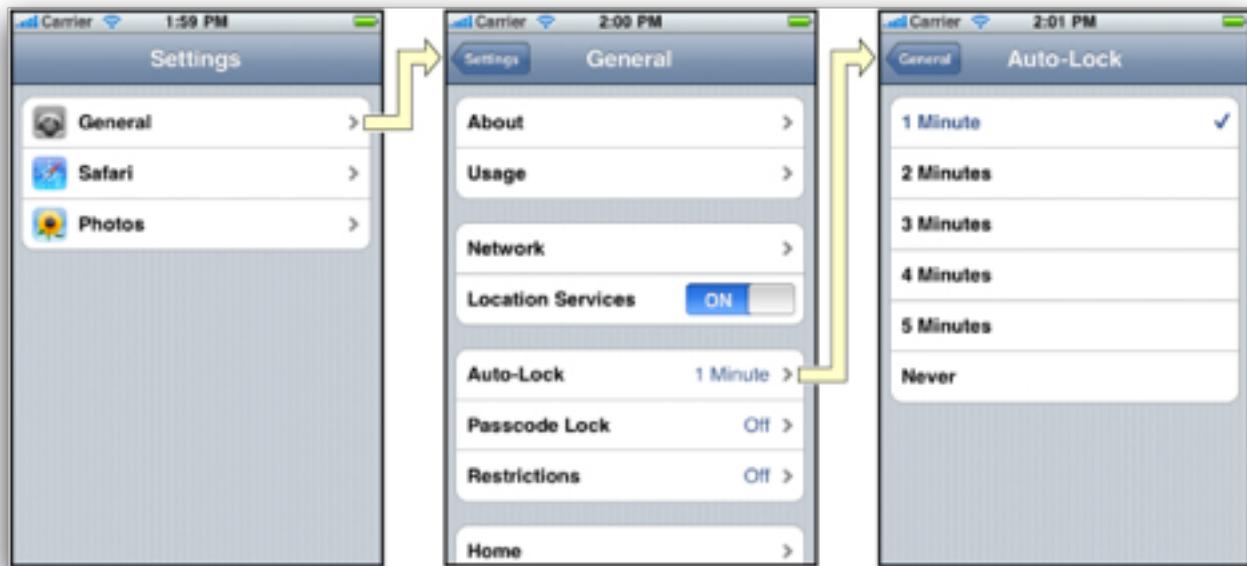
Events

This object has no events

Titanium.UI.iPhone.NavigationGroup

object of Titanium.UI.iPhone

A Navigation Group implements a specialized view that manages the navigation of hierarchical content. The Navigation Group is created by the method Titanium.UI.iPhone.NavigationGroup.



A Navigation Group is very similar to Tab Bars with the exception that they do not maintain a group of windows with a interface bar at the bottom.

Methods

Name	Description
add	add a child to the view hierarchy
addEventListener	add an event listener for the instance to receive view triggered events
animate	animate the view
close	close a window and remove it from the navigation group
fireEvent	fire a synthesized event to the views listener
hide	hide the view
open	open a window within the navigation group
remove	remove a previously add view from the view hierarchy
removeEventListener	remove a previously added event listener
show	make the view visible
toImage	return a Blob image of the rendered view

Properties

Name	Type	Description
anchorPoint	object	a dictionary with properties x and y to indicate the anchor point value. anchor specifies the position by which animation should occur. center is 0.5, 0.5
animatedCenterPoint	object	read-only object with x and y properties of where the view is during animation
backgroundColor	string	the background color of the view
backgroundDisabledColor	string	the disabled background color of the view. (Android)
backgroundDisabledImage	string	the disabled background image url of the view. (Android)
backgroundFocusedColor	string	the focused background color of the view. focusable must be true for normal views. (Android)
backgroundFocusedImage	string	the focused background image url of the view. focusable must be true for normal views. (Android)
backgroundGradient	object	a background gradient for the view with the properties: type,startPoint,endPoint,startRadius,endRadius,backfillStart,backfillEnd,colors.
backgroundImage	string	the background image url of the view
backgroundLeftCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the

		middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the left end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The right end cap is therefore computed by adding the size of the left end cap and the middle portion together and then subtracting that value from the width of the image
backgroundSelectedColor	string	the selected background color of the view. focusable must be true for normal views. (Android)
backgroundSelectedImage	string	the selected background image url of the view. focusable must be true for normal views. (Android)
backgroundTopCap	float	End caps specify the portion of an image that should not be resized when an image is stretched. This technique is used to implement buttons and other resizable image-based interface elements. When a button with end caps is resized, the resizing occurs only in the middle of the button, in the region between the end caps. The end caps themselves keep their original size and appearance. This property specifies the size of the top end cap. The middle (stretchable) portion is assumed to be 1 pixel wide. The bottom end cap is therefore computed by adding the size of the top end cap and the middle portion together and then subtracting that value from the height of the image
borderColor	string	the border color of the view
borderRadius	float	the border radius of the view
borderWidth	float	the border width of the view
bottom	float,string	property for the view bottom position. this position is relative to the views parent. can be either a float value or a string of the width.
center	object	a dictionary with properties x and y to indicate the center of the views position relative to the parent view
focusable	boolean	Set true if you want a view to be focusable when navigating with the trackball or D-Pad. Default: false. (Android Only)
height	float,string	property for the view height. can be either float value or a string of the width.
left	float,string	property for the view left position. this position is relative to the views parent. can be either a float value or a string of the width.
opacity	float	the opacity from 0.0-1.0
right	float,string	property for the view right position. this position is relative to the views parent. can be either a float value or a string of the width.
size	object	the size of the view as a dictionary of width and height properties
softKeyboardOnFocus	int	One of Titanium.UI.Android.SOFT_KEYBOARD_DEFAULT_ON_FOCUS,

Titanium.UI.Android.SOFT_KEYBOARD_HIDE_ON_FOCUS, or
Titanium.UI.Android.SOFT_KEYBOARD_SHOW_ON_FOCUS. (Android only)

top	float,string	property for the view top position. this position is relative to the views parent. can be either a float value or a string of the width.
touchEnabled	boolean	a boolean indicating if the view should receive touch events (true, default) or forward them to peers (false)
transform	object	the transformation matrix to apply to the view
visible	boolean	a boolean of the visibility of the view
width	float,string	property for the view width. can either be `auto`, a float value or a string of the width.
zIndex	int	the z index position relative to other sibling views

Events

Name	Description										
click	<p>fired when the device detects a click (longer than touch) against the view</p> <p>Event properties</p> <table border="1"><tr><td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr><tr><td>source</td><td>the source object that fired the event</td></tr><tr><td>type</td><td>the name of the event fired</td></tr><tr><td>x</td><td>the x point of the event in receiving view coordinates</td></tr><tr><td>y</td><td>the y point of the event, in receiving view coordinates</td></tr></table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
dblclick	<p>fired when the device detects a double click against the view</p> <p>Event properties</p> <table border="1"><tr><td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr><tr><td>source</td><td>the source object that fired the event</td></tr><tr><td>type</td><td>the name of the event fired</td></tr><tr><td>x</td><td>the x point of the event in receiving view coordinates</td></tr><tr><td>y</td><td>the y point of the event, in receiving view coordinates</td></tr></table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates	y	the y point of the event, in receiving view coordinates
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										
y	the y point of the event, in receiving view coordinates										
doubletap	<p>fired when the device detects a double tap against the view</p> <p>Event properties</p> <table border="1"><tr><td>globalPoint</td><td>a dictionary with properties x and y describing the point of the event in screen coordinates</td></tr><tr><td>source</td><td>the source object that fired the event</td></tr><tr><td>type</td><td>the name of the event fired</td></tr><tr><td>x</td><td>the x point of the event in receiving view coordinates</td></tr></table>	globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates	source	the source object that fired the event	type	the name of the event fired	x	the x point of the event in receiving view coordinates		
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates										
source	the source object that fired the event										
type	the name of the event fired										
x	the x point of the event in receiving view coordinates										

y

the y point of the event, in receiving view coordinates

singletap

fired when the device detects a single tap against the view

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

swipe

fired when the device detects a swipe (left or right) against the view

Event properties

direction	direction of the swipe - either left or right
globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchcancel

fired when a touch event is interrupted by the device. this happens in circumstances such as an incoming call to allow the UI to clean up state.

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchend

fired when a touch event is completed

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates

y

the y point of the event, in receiving view coordinates

touchmove	fired as soon as the device detects movement of a touch. Event coordinates are always relative to the view in which the initial touch occurred
------------------	--

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

touchstart	fired as soon as the device detects a gesture
-------------------	---

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

twofingertap	fired when the device detects a two-finger tap against the view
---------------------	---

Event properties

globalPoint	a dictionary with properties x and y describing the point of the event in screen coordinates
source	the source object that fired the event
type	the name of the event fired
x	the x point of the event in receiving view coordinates
y	the y point of the event, in receiving view coordinates

Code Examples**Simple Navigation Group**

In this example, we open 2 windows in a navigation group. When you run this, the initial window should be blue. When you click the back button, you should see a window that should be red.

```
var win = Titanium.UI.createWindow();
var win1 = Titanium.UI.createWindow({
    backgroundColor:"red",
    title:"Red Window"
});
var nav = Titanium.UI.iPhone.createNavigationGroup({
    window: win1
});
```

```
win.add(nav);
win.open();
var win2 = Titanium.UI.createWindow({
    backgroundColor:"blue",
    title:"Blue Window"
});
nav.open(win2,{animated:true});
```

Notes

The `window` property must be set initially in the constructor when creating a navigation group to the root level window. All Navigation Groups must have at least one root window that cannot be removed.

Titanium.UI.iPhone.NavigationGroup.add

function of Titanium.UI.iPhone.NavigationGroup
add a child to the view hierarchy

Arguments

Name	Type	Description
<code>view</code>	object	the view to add to this views hierarchy

Return Type

void

Titanium.UI.iPhone.NavigationGroup.addEventListener

function of Titanium.UI.iPhone.NavigationGroup
add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
<code>name</code>	string	name of the event
<code>callback</code>	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.iPhone.NavigationGroup.animate

function of Titanium.UI.iPhone.NavigationGroup
animate the view

Arguments

Name	Type	Description
obj	object	either a dictionary of animation properties or an Animation object
callback	function	function to be invoked upon completion of the animation

Return Type

void

Titanium.UI.iPhone.NavigationGroup.close

function of Titanium.UI.iPhone.NavigationGroup

close a window and remove it from the navigation group

Arguments

Name	Type	Description
window	object	window to close
properties	object	optional dictionary. the only current property supported is animated which is a boolean to indicate if the window should be closed animated (default) or not.

Return Type

void

Titanium.UI.iPhone.NavigationGroup.fireEvent

function of Titanium.UI.iPhone.NavigationGroup

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.iPhone.NavigationGroup.hide

function of Titanium.UI.iPhone.NavigationGroup

hide the view

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.iPhone.NavigationGroup.open

function of Titanium.UI.iPhone.NavigationGroup

open a window within the navigation group

Arguments

Name	Type	Description
window	object	window to open within the tab group
properties	object	optional dictionary. the only current property supported is animated which is a boolean to indicate if the window should be opened animated (default) or not.

Return Type

void

Titanium.UI.iPhone.NavigationGroup.remove

function of Titanium.UI.iPhone.NavigationGroup

remove a previously add view from the view hierarchy

Arguments

Name	Type	Description
view	object	the view to remove from this views hierarchy

Return Type

void

Titanium.UI.iPhone.NavigationGroup.removeEventListener

function of Titanium.UI.iPhone.NavigationGroup

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.UI.iPhone.NavigationGroup.show

function of Titanium.UI.iPhone.NavigationGroup

make the view visible

Arguments

This function takes no arguments.

Return Type

void

Titanium.UI.iPhone.NavigationGroup.toImage

function of Titanium.UI.iPhone.NavigationGroup

return a Blob image of the rendered view

Arguments

Name	Type	Description
f	function	function to be invoked upon completion. if non-null, this method will be performed asynchronously. if null, it will be performed immediately

Return Type

object

Titanium.UI.iPhone.ProgressBarStyle

object of Titanium.UI.iPhone

A set of constants for the bar styles used on the `style` property of Titanium.UI.ProgressBar.

Methods

This object has no methods

Properties

Name	Type	Description
BAR	int	The style of progress view that is used in a toolbar.
DEFAULT	int	The standard progress-view style. This is the default.

PLAIN	int	The standard progress-view style. Same as DEFAULT.
--------------	-----	--

Events

This object has no events

Titanium.UI.iPhone.RowAnimationStyle

object of Titanium.UI.iPhone

A set of constants for the Animation Styles used for transition on table view rows.

Methods

This object has no methods

Properties

Name	Type	Description
BOTTOM	int	The inserted row or rows slides in from the bottom; the deleted row or rows slides out toward the bottom.
FADE	int	The inserted or deleted row or rows fades into or out of the table view.
LEFT	int	The inserted row or rows slides in from the left; the deleted row or rows slides out to the left.
NONE	int	No animation is performed. The new cell value appears as if the cell had just been reloaded.
RIGHT	int	The inserted row or rows slides in from the right; the deleted row or rows slides out to the right.
TOP	int	The inserted row or rows slides in from the top; the deleted row or rows slides out toward the top.

Events

This object has no events

Titanium.UI.iPhone.ScrollIndicatorStyle

object of Titanium.UI.iPhone

A set of constants for the styles available for Titanium.UI.ActivityIndicator objects.

Methods

This object has no methods

Properties

Name	Type	Description
BLACK	int	A style of indicator which is black smaller than the default style. This style is good against a white content background.
DEFAULT	int	The default style of scroll indicator, which is black with a white border. This style is good against any content background.
WHITE	int	A style of indicator is white and smaller than the default style. This style is good against a black content background.

Events

This object has no events

Titanium.UI.iPhone.StatusBar

object of Titanium.UI.iPhone

A set of constants for the status bar style.

Methods

This object has no methods

Properties

Name	Type	Description
DEFAULT	int	The default status bar style
GRAY	int	The gray colored status bar style
OPAQUE_BLACK	int	The opaque black status bar style.
TRANSLUCENT_BLACK	int	The translucent black status bar style. This style provides some level of transparency to the device background.

Events

This object has no events

Titanium.UI.iPhone.SystemButtonStyle

Titanium.UI.iOS.SystemButtonStyle

object of Titanium.UI.iPhone

A set of constants for the system button styles that can be used for the button `style` property.

Methods

This object has no methods

Properties

Name	Type	Description
BORDERED	int	A simple button style with a border.
DONE	int	The style for a done button - for example, a button that completes some task and returns to the previous view.
PLAIN	int	Glowes when tapped. The default item style.

Events

This object has no events

Titanium.UI.iPhone.SystemIcon

object of Titanium.UI.iPhone

A set of constants for the system icon styles that can be used on a tab group tab.

Methods

This object has no methods

Properties

Name	Type	Description
BOOKMARKS	int	Bookmark style icon
CONTACTS	int	Contacts style icon
DOWNLOADS	int	Downloads style icon
FAVORITES	int	Favorites style icon
FEATURED	int	Featured style icon
HISTORY	int	

MORE	int	More style icon
MOST_RECENT	int	Most recent style icon
MOST_VIEWED	int	Most viewed style icon
RECENTS	int	Recents style icon
SEARCH	int	Search style icon
TOP_RATED	int	Top rated style icon

Events

This object has no events

Titanium.UI.iPhone.TableViewCellStyle

object of Titanium.UI.iPhone

A set of constants for the style that can be used for the `selectionStyle` property of Titanium.UI.TableViewRow.

Methods

This object has no methods

Properties

Name	Type	Description
BLUE	int	The cell when selected has a blue background. This is the default value.
GRAY	int	The cell when selected has a gray background.
NONE	int	The cell has no distinct style for when it is selected.

Events

This object has no events

Titanium.UI.iPhone.TableViewScrollPosition

A set of constants for the position value that can be used for the `position` property of Titanium.UI.TableView when invoking `scrollToIndex`.

Methods

This object has no methods

Properties

Name	Type	Description
BOTTOM	int	The table view scrolls the row of interest to the bottom of the visible table view.
MIDDLE	int	The table view scrolls the row of interest to the middle of the visible table view.
NONE	int	The table view scrolls the row of interest to be fully visible with a minimum of movement. If the row is already fully visible, no scrolling occurs. For example, if the row is above the visible area, the behavior is identical to that specified by TOP. This is the default.
TOP	int	The table view scrolls the row of interest to the top of the visible table view.

Events

This object has no events

Titanium.UI.iPhone.TableViewSeparatorStyle

object of Titanium.UI.iPhone

A set of constants for the style that can be used for the `separatorStyle` property of Titanium.UI.TableView.

Methods

This object has no methods

Properties

Name	Type	Description
NONE	int	The separator cell has no distinct style.
SINGLE_LINE	int	The separator cell has a single line running across its width. This is the default value.

Events

This object has no events

Titanium.UI.iPhone.TableViewStyle

object of Titanium.UI.iPhone

A set of constants for the style that can be used for the button `style` property of Titanium.UI.TableView.

Methods

This object has no methods

Properties

Name	Type	Description
GROUPED	int	A table view whose sections present distinct groups of rows. The section headers and footers do not float.
PLAIN	int	A plain table view. Any section headers or footers are displayed as inline separators and float when the table view is scrolled.

Events

This object has no events

Titanium.UI.iPhone.addEventListener

function of Titanium.UI.iPhone

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.UI.iPhone.createNavigationGroup

function of Titanium.UI.iPhone

create and return an instance of Titanium.UI.iPhone.NavigationGroup

Arguments

Name	Type	Description
parameters	object	(optional) a dictionary object properties defined in Titanium.UI.iPhone.NavigationGroup

Return Type

object

Titanium.UI.iPhone.fireEvent

function of Titanium.UI.iPhone

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.UI.iPhone.hideStatusBar

function of Titanium.UI.iPhone

convenience method to hide the status bar

Arguments

Name	Type	Description
animated	boolean	boolean to indicate if the action should be animated
animationStyle	int	the animation style

Return Type

void

Titanium.UI.iPhone.removeEventListener

function of Titanium.UI.iPhone

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

Return Type

void

Titanium.UI.iPhone.showStatusBar

function of Titanium.UI.iPhone

convenience method to show the status bar

Arguments

Name	Type	Description
animated	boolean	boolean to indicate if the action should be animated
animationStyle	int	the animation style

Return Type

void

Titanium.UI.removeEventListener

function of Titanium.UI

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Utils

submodule of Titanium

The top level Utils module. The Utils module is a set of common JavaScript functions that are useful for applications.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
base64decode	decode a Base64 string

base64encode	encode a string into Base64
fireEvent	fire a synthesized event to the views listener
md5HexDigest	compute a MD5 hash algorithm against the input and return a hex-based string
removeEventListener	remove a previously added event listener

Properties

This module has no properties

Events

This module has no events

Titanium.Utils.addEventListener

function of Titanium.Utils

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Utils.base64decode

function of Titanium.Utils

decode a Base64 string

Arguments

Name	Type	Description
str	string	the string to use for the input

Return Type

string

Titanium.Utils.base64encode

Titanium.Utils.encode

function of Titanium.Utils

encode a string into Base64

Arguments

Name	Type	Description
str	string	the string to use for the input

Return Type

string

Titanium.Utils.fireEvent

function of Titanium.Utils

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Utils.md5HexDigest

function of Titanium.Utils

compute a MD5 hash algorithm against the input and return a hex-based string

Arguments

Name	Type	Description
str	string	the string to use for the input

Return Type

string

Titanium.Utils.removeEventListener

function of Titanium.Utils

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.XML

submodule of Titanium

The top level XML module. The XML module is used parsing and processing XML-based content.

Objects

Name	Description
Titanium.XML.DOMDocument	The DOMDocument returned from Titanium.XML.parseString. The result is an object that implementes the DOM Level 2 API.

Methods

Name	Description
addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
parseString	parse an XML string into a DOMDocument
removeEventListener	remove a previously added event listener
serializeToString	serialize a DOMDocument or DOMNode and its descendants into an XML string

Properties

This module has no properties

Events

This module has no events

Titanium.XML.DOMDocument

object of Titanium.XML

The DOMDocument returned from Titanium.XML.parseString. The result is an object that implementes the DOM Level 2 API.

Methods

This object has no methods

Properties

This object has no properties

Events

This object has no events

Titanium.XML.addEventListener

function of Titanium.XML

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.XML.fireEvent

function of Titanium.XML

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.XML.parseString

function of Titanium.XML

parse an XML string into a DOMDocument

Arguments

Name	Type	Description
xml	string	the XML content as a string

Return Type

object

Titanium.XML.removeEventListener

function of Titanium.XML

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.XML.serializeToString

function of Titanium.XML

serialize a DOMDocument or DOMNode and its descendants into an XML string

Arguments

Name	Type	Description
node	DOMNode	the XML DOMNode or DOMDocument to serialize

Return Type

string

Titanium.Yahoo

submodule of Titanium

The top level Yahoo module. The Yahoo module is used for accessing Yahoo related API services.

Methods

Name	Description
Yahoo	Creates a new instance of the Yahoo module.

addEventListener	add an event listener for the instance to receive view triggered events
fireEvent	fire a synthesized event to the views listener
removeEventListener	remove a previously added event listener

Properties

This module has no properties

Events

This module has no events

Titanium.Yahoo.addEventListener

function of Titanium.Yahoo

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.Yahoo.fireEvent

function of Titanium.Yahoo

fire a synthesized event to the views listener

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.Yahoo.removeEventListener

function of Titanium.Yahoo

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium.Yahoo.yql

function of Titanium.Yahoo

invoke a Yahoo YQL query

Arguments

Name	Type	Description
yql	string	the YQL query to execute
callback	function	the function to execute when the query completes. The event will contain the boolean property success if successful. If success is false, the message property will contain the error message. If success is true, the data property will contain the data payload received from the YQL.

Return Type

void

Titanium.addEventListener

function of Titanium

add an event listener for the instance to receive view triggered events

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function to invoke when the event is fired

Return Type

void

Titanium.fireEvent

function of Titanium

Arguments

Name	Type	Description
name	string	name of the event
event	object	event object

Return Type

void

Titanium.include

function of Titanium

one or more filenames to include as if the Javascript code was written in place. This is similar to a C #include function.

Arguments

Name	Type	Description
name	string	filename to include

Return Type

void

Titanium.removeEventListener

function of Titanium

remove a previously added event listener

Arguments

Name	Type	Description
name	string	name of the event
callback	function	callback function passed in addEventListener

Return Type

void

Titanium Mobile 1.4.0 - 7/26/2010

This release includes general bug fixes and enhancements for Android and iPhone.

Please see the updated Titanium Mobile Reference Documentation for 1.4.

We have fixed 200+ reported (and unreported) issues since 1.3. Thanks for everyone's reported issues and patience as we have worked a little longer on this release than we had originally anticipated.

We encountered many differences between the latest version of iOS 4 and the previous version we supported (3.1.3) and many differences between the various devices than we have never experienced before with iPhone OS.

iPhone Version Support

We have tested this release with iPhone OS 3.1.3 and iOS 4.0.1. We recommend that you build your applications using iOS 4.0.1. If you build with 4.0.1, your application will continue to work on older versions of iPhone OS back to 3.1.

We have tested this release on iPhone OS 3.2 for iPad.

We have tested this release on 1G, 2G, 3G, 3GS and 4G including (iPhone and iPod Touch) and iPad.

iOS4

In this release, we have added support for the following iOS4 capabilities:

- Fast application switching
- Retina display
- Hi-res icons
- Background audio

In the next release, we will add additional support for background location and background processing of your application (so you can run your application logic while the application is in the background), and local notifications.

Custom Info.plist changes

If you have used a custom Info.plist.template in the past, you will now need to make a slight change to the location and name of the template. To better support custom template and version migrations, we now look for the Info.plist file in the root folder of your project directory. If you place a final Info.plist in this directory, we will use this file as-is.

iPhone GPS/Location Changes

Starting in iPhone 4.0, you must set the Titanium.Geolocation.purpose property to the reason your application is requesting Location information before using Geolocation APIs. If you don't set this field, GPS will not work and you will get a warning in the application log. This is an Apple requirement.

iPhone Packaging Changes

As of July 22, 2010, Apple now requires applications be submitted either through XCode or through the iPhone Application Loader application. The Titanium build for distribution step has been modified to comply with these new build mechanisms. Titanium will no longer generate a zip file for distribution. Instead, Titanium will automatically build your application using the "Build and Archive" xcode build stage and your application will show up in the XCode organizer. You can validate your application passes Apple tests prior to submission. You can also submit your application directly to iTunes from Organizer. You can optionally use the Application Loader to submit your application to iTunes. If you're on iOS 4, you'll have the Application Loader pre-installed on your system under `/Developer/Applications/Utilities`. If you're on a previous version, you will need to download the application from iTunes Connect.

We have found that the current version XCode Organizer provides no visual feedback when you perform tasks such as Validate Application, etc. We have also found that no error indication is provided if you attempt to Validate or Submit your application to iTunes and you do not have a previously created application in iTunes Connect. These buttons will simply do nothing. You will first need to create an application in the portal before using these functions.

Android Changes

This release is required if you've recently installed the Android SDK Tools version 6 or have added new platforms using the version 6 tools. Google changed the naming scheme again.

We have made a number of improvements in Android and fixed a lot of bugs. Some major issues:

- General Tableview improvements in sections, getData and more.
- Improved Ti.UI.Notification with more control over placement and look-n-feel of Toast notifications
- Geolocation: preferredProvider support
- Android Module SDK

Android Module SDK

This version formally introduces the Android Module SDK for extending Titanium with native module code for Android. A separate module documentation is being developed that should be available soon.

Continuous Builds from Git available

We are now performing continuous builds of Titanium from the git master. You can find more details about this release from our documentation site

Please note that we do not offer any support for running from master (unless directed by Appcelerator Premium Support). Git is where we do *active* development and the master branch is not stable. We tag our repo with stable tags if you'd like to build an older version of Titanium from a stable release. We appreciate everyone's desire to help us test and run through issues as they're being developed! However, please remember that master means unstable until officially released. Please use with caution!

Revision History

9/17/2010	Initial release
-----------	-----------------