

B4X Booklets

B4A

B4i

B4J

B4X JavaObject NativeObject

1	B4X platforms.....	5
2	General.....	6
3	B4A JavaObject.....	7
3.1	B4A JavaObject methods.....	8
3.1.1	CreateEvent.....	8
3.1.2	CreateEventFromUI.....	9
3.1.3	GetField.....	9
3.1.4	GetFieldJO.....	9
3.1.5	InitializeArray.....	9
3.1.6	InitializeContext.....	9
3.1.7	InitializeNewInstance.....	10
3.1.8	InitializeStatic.....	10
3.1.9	IsInitialized.....	10
3.1.10	RunMethod.....	11
3.1.11	RunMethodJO.....	12
3.1.12	SetField.....	12
3.2	How to use it.....	13
3.3	B4A examples.....	16
3.3.1	Get B4A Context.....	16
3.3.1.1	GetB4A.....	16
3.3.1.2	getContext.....	16
3.3.2	ViewUtils.....	17
3.3.2.1	getParent.....	17
3.3.2.2	getRootView.....	17
3.3.2.3	setRotation.....	17
3.3.2.4	setRotationX.....	17
3.3.2.5	setRotationY.....	18
3.3.2.6	setPivotX.....	18
3.3.2.7	getPivotX.....	18
3.3.2.8	setPivotY.....	18
3.3.2.9	getPivotY.....	18
3.3.3	EditText Utils.....	19
3.3.3.1	setCursorVisible.....	19
3.3.3.2	setTextIsSelectable.....	19
3.3.3.3	setSelection.....	19
3.3.3.4	getSelectionStart.....	19
3.3.3.5	getSelectionEnd.....	19
3.3.3.6	getLineHeight.....	20
3.3.3.7	getLineCount.....	20
3.3.3.8	getHighLightColor / setHighLightColor.....	20
3.3.3.9	GetXycursor.....	21
3.3.4	Label Utils.....	22
3.3.4.1	setEllipsize.....	22
3.3.4.2	resetEllipsize.....	22
3.3.4.3	setSingleLine.....	22
3.3.4.4	setLines.....	23
3.3.4.5	setMarqueeRepeat.....	23
3.3.4.6	Stop or run the horizontal scrolling.....	23
3.3.4.7	setShadowLayer.....	23
4	B4J JavaObject.....	24
4.1	B4J JavaObjec Methods.....	24
4.1.1	CreateEvent.....	24
4.1.2	CreateEventFromUI.....	26

4.1.3	GetField.....	26
4.1.4	GetFieldJO	26
4.1.5	InitializeArray	26
4.1.6	InitializeContext.....	26
4.1.7	InitializeNewInstance.....	27
4.1.8	InitializeStatic	27
4.1.9	IsInitialized.....	27
4.1.10	RunMethod.....	28
4.1.11	RunMethodJO	29
4.1.12	SetField	29
4.2	How to use it	30
4.3	B4J examples	31
4.3.1	setMouseTransparent	31
4.3.2	setTextAlignment.....	31
4.3.3	Graphics jCanvasExt.....	32
4.3.4	GoogleMaps	33
5	B4i NativeObject.....	34
5.1	B4i NativeObject methods	34
5.1.1	ArrayFromEdgeInsets	34
5.1.2	ArrayFromPoint	34
5.1.3	ArrayFromRange.....	34
5.1.4	ArrayFromRect	34
5.1.5	ArrayFromSize.....	35
5.1.6	ArrayToNSData	35
5.1.7	AsBoolean.....	35
5.1.8	AsNumber	35
5.1.9	AsString	35
5.1.10	ColorToUIColor.....	35
5.1.11	CreateBlock.....	36
5.1.12	GetField.....	36
5.1.13	Initialize(.....	36
5.1.14	IsInitialized.....	36
5.1.15	MakeEdgeInsets	36
5.1.16	MakePoint	37
5.1.17	MakeRange	37
5.1.18	MakeRect	37
5.1.19	MakeSize.....	37
5.1.20	NSDataToArray	37
5.1.21	RunMethod.....	38
5.1.22	RunMethodWithBlocks	38
5.1.23	SetField	38
5.1.24	Tag.....	38
5.1.25	UIColorToColor.....	38
5.2	How to use it	39
5.3	B4i examples.....	40
5.3.1	SetButtonTextColor	40
5.3.2	Hide the StatusBar.....	40
5.3.3	Setting the status bar to use a light theme	40
5.3.4	Check StatusBar state.....	40
5.3.5	Get Border properties.....	41
5.3.6	Getting current device language.....	41
5.3.7	Set the NavigationBar text color.....	41

Main contributors: Klaus Christl (klaus), Erel Uziel (Erel)

To search for a given word or sentence use the Search function in the Edit menu.

All the source code and files needed (layouts, images etc.) of the example projects in this booklet are included in the SourceCode folder.

Updated for:

B4A version 12.80

B4i version 8.50

B4J version 10.00

[B4X Booklets:](#)

B4X Getting Started

B4X Language

B4X IDE Integrated Development Environment

B4X Visual Designer

B4X Help tools

B4XPages Cross-platform projects

B4X CustomViews

B4X Graphics

B4X XUI B4X User Interface

B4X SQLite Database

B4X JavaObject NativeObject

B4R Example Projects

You can consult these booklets online in this link [\[B4X\] Documentation Booklets](#).

Be aware that external links don't work in the online display.

1 B4X platforms

B4X is a suite of programming languages for different platforms.

B4X suite supports more platforms than any other tool

ANDROID | IOS | WINDOWS | MAC | LINUX | ARDUINO | RASPBERRY PI | ESP8266 | AND MORE...

- **B4A**  **Android**

B4A is a **100% free** development tool for Android applications, it includes all the features needed to quickly develop any type of Android app.

- **B4i**  **iOS**

B4i is a development tool for native iOS applications.

B4i follows the same concepts as B4A, allowing you to reuse most of the code and build apps for both Android and iOS.

- **B4J**  **Java / Windows / Mac / Linux / Raspberry PI**

B4J is a **100% free** development tool for desktop, server and IoT solutions.

With B4J you can easily create desktop applications (UI), console programs (non-UI) and server solutions.

The compiled apps can run on Windows, Mac, Linux and ARM boards (such as Raspberry Pi).

- **B4R**  **ARDUINO** **Arduino / ESP8266**

B4R is a **100% free** development tool for native Arduino and ESP8266 programs.

B4R follows the same concepts of the other B4X tools, providing a simple and powerful development tool.

B4R, B4A, B4J and B4i together make the best development solution for the Internet of Things (IoT).

- **B4XPages**

B4XPages is an internal library for B4A, B4i and B4J allowing to develop easily cross-platform programs.

B4XPages is explained in detail in the B4XPages Cross-platform projects booklet.

Even, if you want to develop only in one platform it is interesting to use the B4XPages library it makes the program flow simpler especially for B4A.

2 General

JSONObject can be used to access Android or Java objects, or properties not directly exposed to B4A or B4J.

The library is the same for B4A and B4J.

NativeObject is like JSONObject but for B4i.

To use JSONObject or NativeObject you should know or learn how the operating systems are constructed and how they work.

The purpose of this booklet is to give an ‘entry point’ with examples, to better understand how to use it.

It is not an encyclopedia of routines.

3 B4A JavaObject

The purpose of JavaObject library is similar to the purpose of Reflection library. Both libraries allow you to directly call Java APIs based on Java reflection features.

JavaObject design is different than Reflection library and, in most cases, simpler to use. However, JavaObject doesn't replace Reflection library as it doesn't support all its features. In many cases you can use both libraries together (both are lightweight libraries).

JavaObject approach is more "object oriented". You declare a JavaObject object which then "wraps" any other object and provide three methods: SetField, GetField and RunMethod.

JavaObject variable is similar to an Object variable with the addition of the reflection methods.

Notes

- JavaObject can only access public methods and fields (unlike Reflection library).
- JavaObject doesn't include the helper methods to access the context and other fields as in Reflection library. You can use both libraries together when these fields are needed.
- There is almost no overhead for a JavaObject instance. It is better to create multiple JavaObjects instead of reusing a single instance.

3.1 B4A JavaObject methods

3.1.1 CreateEvent

CreateEvent (Interface As String, EventName As String, DefaultReturnValue As Object)

Creates an instance of the interface and binds it to the object.

Interface - The full interface name.

EventName - The prefix of the event sub.

DefaultReturnValue - This value will be returned if no value was returned from the event sub. This can happen if the Activity is paused for example.

B4A example:

```
Sub Activity_Create(FirstTime As Boolean)

    Dim btn As Button
    btn.Initialize("")
    Activity.AddView(btn, 0, 0, 200dip, 200dip)
    Dim jo As JavaObject = btn
    Dim e As Object = jo.CreateEvent("android.view.View.OnTouchListener", "btnTouch",
False)
    jo.RunMethod("setOnTouchListener", Array As Object(e))
End Sub
```

And the event routine:

```
Sub btnTouch_Event (MethodName As String, Args() As Object) As Object
    Dim MotionEvent As JavaObject = Args(1) 'args(0) is View
    Dim x As Float = MotionEvent.RunMethod("getX", Null)
    Dim y As Float = MotionEvent.RunMethod("getY", Null)
    Log(x & ", " & y)
    Dim Action As Int = MotionEvent.RunMethod("getAction", Null)
    Log("Action = " & Action)
    Return True
End Sub
```

And the Android documentation: [View.OnTouchListener](#), [MotionEvent](#).

float	<code>getX()</code> <code>getX(int)</code> for the first pointer index (may be an arbitrary pointer identifier).
float	<code>getXPrecision()</code> Return the precision of the X coordinates being reported.
float	<code>getY()</code> <code>getY(int)</code> for the first pointer index (may be an arbitrary pointer identifier).
int	<code>getAction()</code> Return the kind of action being performed.

If you look at the documentation, you may notice that there is a lot to learn.

3.1.2 CreateEventFromUI

CreateEventFromUI (Interface As String, EventName As String, ReturnValue As Object)

Similar to CreateEvent. The event will be sent to the message queue and then be processed (similar to CallSubDelayed).

3.1.3 GetField

GetField (Field As String)

Gets the value of the given field.

3.1.4 GetFieldJO

GetFieldJO (Field As String)

Similar to GetField. Returns a JavaObject instead of Object.

Returns a JavaObject.

3.1.5 InitializeArray

InitializeArray (ClassName As String, Values As Object())

Creates an array with the given class and values.

Returns a JavaObject.

3.1.6 InitializeContext

B4A only method.

Initializes the object with the current context (current Activity or Service).

Returns a JavaObject.

3.1.7 InitializeNewInstance

InitializeNewInstance (ClassName As String, Params As Object())

Creates a new instance of the given class.

ClassName - The full class name.

Params - An array of objects to pass to the constructor (or Null).

Returns a JavaObject.

3.1.8 InitializeStatic

InitializeStatic (ClassName As String)

Initializes the object. The object will wrap the given class (for static access).

ClassName - The full class name.

Returns a JavaObject.

3.1.9 IsInitialized

Returns a Boolean.

3.1.10 RunMethod

RunMethod (MethodName As String, Params As Object())

Runs the given method and returns the method return value.

MethodName - The case-sensitive method name.

Params - Method parameters (or Null).

Returns an Object.

B4A example:

Get and sets the Labels padding.

```
Private joLabel1 As JavaObject = Label1      'wrap the Label object
Log("Label1 Left padding = " & joLabel1.RunMethod("getPaddingLeft", Null))
Log("Label1 Top padding = " & joLabel1.RunMethod("getPaddingTop", Null))
Log("Label1 Right padding = " & joLabel1.RunMethod("getPaddingRight", Null))
Log("Label1 Bottom padding = " & joLabel1.RunMethod("getPaddingBottom", Null))

joLabel1.RunMethod("setPadding", Array As Object(10dip, 10dip, 10dip, 10dip))
```

Android documentation: [View](#).

Don't be afraid! It's a huge chapter! To see the padding methods, you need to scroll down very deep.

int	<code>getPaddingBottom()</code> Returns the bottom padding of this view.
int	<code>getPaddingEnd()</code> Returns the end padding of this view depending on its resolved layout direction.
int	<code>getPaddingLeft()</code> Returns the left padding of this view.
int	<code>getPaddingRight()</code> Returns the right padding of this view.
int	<code>getPaddingStart()</code> Returns the start padding of this view depending on its resolved layout direction.
int	<code>getPaddingTop()</code>
void	<code>setPadding(int left, int top, int right, int bottom)</code> Sets the padding.

3.1.11 RunMethodJO

RunMethodJO (MethodName As String, Params As Object())

Similar to RunMethod. Returns a JavaObject instead of Object.

Returns a JavaObject.

3.1.12 SetField

SetField (FieldName As String, Value As Object)

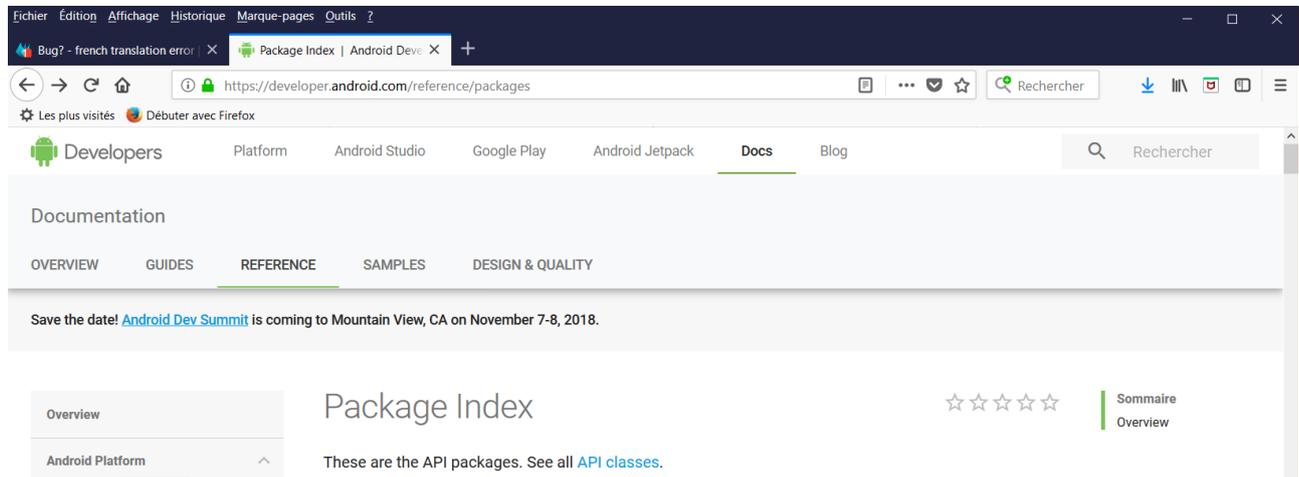
Sets the value of the given field.

Returns Void.

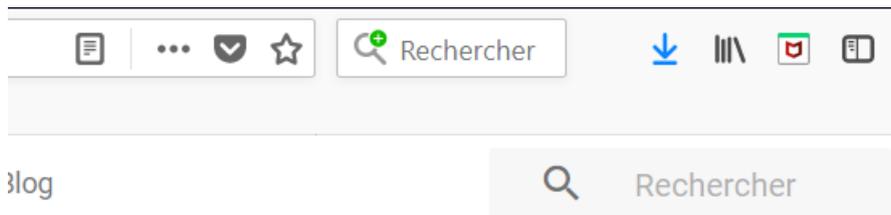
3.2 How to use it

Let's add the Ellipse property to a view. This property is implemented in B4A, but not the MARQUEE mode which allows scrolling the text in the TextView.

First, we need to look at the [Android Documentation](https://developer.android.com/reference/packages).



Enter 'ellipse' in the Search field in the top right corner.



The first search result:

Résultats de recherche pour **ellipse**

[TextView | Android Developers](https://developer.android.com/reference/android/widget/TextView)

<https://developer.android.com/reference/android/widget/TextView>

android:ellipse, If set, causes words that are longer than the view is wide to be ellipsized instead of broken in the middle. android:ems, Makes the TextView be ...

Click on: [TextView | Android Developers](https://developer.android.com/reference/android/widget/TextView)

Below this title **Public methods** you find all public methods of the TextView.

It is a huge 'chapter'!

Scroll down until you find `setEllipsize`.

<code>void</code>	<code>setEllipsize(TextUtils.TruncateAt where)</code>
	Causes words in the text that are longer than the view's width to be ellipsized instead of broken in the middle.

setEllipsize

added in API level 1

```
public void setEllipsize (TextUtils.TruncateAt where)
```

Causes words in the text that are longer than the view's width to be ellipsized instead of broken in the middle. You may also want to `setSingleLine()` or `setHorizontallyScrolling(boolean)` to constrain the text to a single line. Use `null` to turn off ellipsizing. If `setMaxLines(int)` has been used to set two or more lines, only `TextUtils.TruncateAt.END` and `TextUtils.TruncateAt.MARQUEE` are supported (other ellipsizing types will not do anything).

Related XML Attributes:

[android:ellipsize](#)

Parameters	
<code>where</code>	<code>TextUtils.TruncateAt</code>

In text we see that we need to `setSingleLine`

We see that we must add a parameter. Click on [TextUtils.TruncateAt](#).

Here find the different possible parameters.

Enum values	
<code>TextUtils.TruncateAt</code>	<code>END</code>
<code>TextUtils.TruncateAt</code>	<code>MARQUEE</code>
<code>TextUtils.TruncateAt</code>	<code>MIDDLE</code>
<code>TextUtils.TruncateAt</code>	<code>START</code>

Public methods	
<code>static TextUtils.TruncateAt</code>	<code>valueOf(String name)</code>
<code>static final TruncateAt[]</code>	<code>values()</code>

Example code for ellipsize MARQUEE:

```
Private lblTest As Label

Private jo = TextView As JavaObject
jo.RunMethod("setSingleLine", Array As Object(True))
jo.RunMethod("setEllipsize", Array As Object("MARQUEE"))
jo.RunMethod("setSelected", Array As Object(True)) ' needed for MARQUEE
```

And a generalized routine:

```
'Sets the Ellipsizing of a TextView (Label, EditText)
'original text 'This is a test text'
'Mode can be
'START"    ... This is a te
'MIDDLE"   This is ... text
'END"      This is a tes...
'MARQUEE"  This is a test t  scrolling text
Sub setEllipsize(TextView As Label, Mode As String)
  Dim jo = TextView As JavaObject
  jo.RunMethod("setSingleLine", Array As Object(True))
  jo.RunMethod("setEllipsize", Array As Object(Mode))
  jo.RunMethod("setSelected", Array As Object(True)) ' needed for MARQUEE
End Sub
```

To disable the Ellipsize property use:

```
jo.RunMethod("setEllipsize", Null)
```

You can also limit the number of scrolling:

setMarqueeRepeatLimit

added in API level 2

```
public void setMarqueeRepeatLimit (int marqueeLimit)
```

Sets how many times to repeat the marquee animation. Only applied if the TextView has marquee enabled. Set to -1 to repeat indefinitely.

Related XML Attributes:

[android:marqueeRepeatLimit](#)

Parameters

marqueeLimit	int
--------------	-----

```
jo.RunMethod("setMarqueeRepeatLimit", Array As Object(Limit))
```

You need to explore and test a lot to become more experienced.

3.3 B4A examples

Some of these examples exist in the forum in [View Utils](#).

3.3.1 Get B4A Context

The two routines are used in conjunction.

3.3.1.1 GetB4A

```
Sub GetBA As JavaObject
  Dim jo As JavaObject
  Dim cls As String = Me
  cls = cls.SubString("class ".Length)
  jo.InitializeStatic(cls)
  Return jo.GetFieldJO("processBA")
End Sub
```

3.3.1.2 getContext

```
Sub GetContext As JavaObject
  Return GetBA.GetField("context")
End Sub
```

3.3.2 Screen size in inches

Gets the number of pixels per inch in both directions.

To get the number of pixels per millimeter divide the xdpi and ydpi values by 25.4.

```
Private jo As JavaObject
Private xdpi, ydpi As Double
jo = Root
jo = jo.RunMethodJO("getResources", Null)
jo = jo.RunMethodJO("getDisplayMetrics", Null)
xdpi = jo.GetField("xdpi")
ydpi = jo.GetField("ydpi")
```

Or the short way:

```
Private jo As JavaObject
Private xdpi, ydpi As Double
jo = Root
jo = jo.RunMethodJO("getResources", Null).RunMethodJO("getDisplayMetrics", Null)
xdpi = jo.GetField("xdpi")
ydpi = jo.GetField("ydpi")
```

3.3.3 ViewUtils

These methods concern Views and are published in the [forum](#).

3.3.3.1 getParent

This property has been added to B4A.

```
'Returns the parent view of the given view
Sub getParent(v As View) As View
    Dim jo = v As JavaObject
    Return jo.RunMethod("getParent", Null)
End Sub
```

3.3.3.2 getRootView

```
'Returns the root view of the given view, it's mainly the Activity
Sub getRootView(v As View) As View
    Dim jo = v As JavaObject
    Return jo.RunMethod("getRootView", Null)
End Sub
```

3.3.3.3 setRotation

```
'Sets the rotation angle of the given view
'v = view
'Angle = rotation angle in degrees
Sub setRotation(v As View, Angle As Float)
    Dim jo = v As JavaObject
    jo.RunMethod("setRotation", Array As Object(Angle))
End Sub
```

3.3.3.4 setRotationX

```
'Sets the rotation angle around the X axis of the given view
'v = view
'Angle = rotation angle in degrees
Sub setRotationX(v As View, Angle As Float)
    Dim jo = v As JavaObject
    jo.RunMethod("setRotationX", Array As Object(Angle))
End Sub
```

3.3.3.5 setRotationY

```
'Sets the rotation angle around the Y axis of the given view
'v = view
'Angle = rotation angle in degrees
Sub setRotationY(v As View, Angle As Float)
    Dim jo = v As JavaObject
    jo.RunMethod("setRotationY", Array As Object(Angle))
End Sub
```

3.3.3.6 setPivotX

```
'Sets the X pivot point of the given view
'v = view
'X = X coordinate of the pivot in pixels
'reference upper left corner, default pivot middle of the view
Sub setPivotX(v As View, X As Float)
    Dim jo = v As JavaObject
    jo.RunMethod("setPivotX", Array As Object(X))
End Sub
```

3.3.3.7 getPivotX

```
'Gets the X pivot point of the given view
Sub getPivotX(v As View) As Float
    Dim jo = v As JavaObject
    Return jo.RunMethod("getPivotX", Null)
End Sub
```

3.3.3.8 setPivotY

```
'Sets the Y pivot point of the given view
'v = view
'Y = Y coordinate of the pivot in pixels
'reference upper left corner, default pivot middle of the view
Sub setPivotY(v As View, Y As Float)
    Dim jo = v As JavaObject
    jo.RunMethod("setPivotY", Array As Object(Y))
End Sub
```

3.3.3.9 getPivotY

```
'Gets the Y pivot point of the given view
Sub getPivotY(v As View) As Float
    Dim jo = v As JavaObject
    Return jo.RunMethod("getPivotY", Null)
End Sub
```

3.3.4 EditText Utils

These methods concern EditText views and are in the forum [EditText Utils](#).

All the methods below can be used without a sub like:

```
Dim jo = edt As JavaObject

jo.RunMethod("setTextIsSelectable", Array As Object(True))
jo.RunMethod("setCursorVisible", Array As Object(True))
jo.RunMethod("setSelection", Array As Object(5, 10))
```

3.3.4.1 setCursorVisible

```
'Sets the cursor visible or hides it
Sub setCursorVisible(edt As EditText, Visible As Boolean)
    Dim jo = edt As JavaObject
    jo.RunMethod("setCursorVisible", Array As Object(Visible))
End Sub
```

3.3.4.2 setTextIsSelectable

```
'Sets the text selectable or not selectable
Sub setTextIsSelectable(edt As EditText, Selectable As Boolean)
    Dim jo = edt As JavaObject
    jo.RunMethod("setTextIsSelectable", Array As Object(Selectable))
End Sub
```

3.3.4.3 setSelection

```
'Selects the text between the two indexes.
Sub setSelection(edt As EditText, StartIndex As Int, EndIndex As Int)
    Dim jo = edt As JavaObject
    jo.RunMethod("setSelection", Array As Object(StartIndex, EndIndex))
End Sub
```

3.3.4.4 getSelectionStart

```
'Gets the selection start index
Sub getSelectionStart(edt As EditText) As Int
    Dim jo = edt As JavaObject
    Return jo.RunMethod("getSelectionStart", Null)
End Sub
```

3.3.4.5 getSelectionEnd

```
'Gets the selection end index
Sub getSelectionEnd(edt As EditText) As Int
    Dim jo = edt As JavaObject
    Return jo.RunMethod("getSelectionEnd", Null)
End Sub
```

3.3.4.6 getLineHeight

```
'Gets the line height
Sub getLineHeight(edt As EditText) As Int
    Dim jo = edt As JavaObject
    Return jo.RunMethod("getLineHeight", Null)
End Sub
```

3.3.4.7 getLineCount

```
'Gets the line count
Sub getLineCount(edt As EditText) As Int
    Dim jo = edt As JavaObject
    Return jo.RunMethod("getLineCount", Null)
End Sub
```

3.3.4.8 getHighLightColor / setHighLightColor

```
'Sets the highlight color
Sub setHighLightColor(edt As EditText, Color As Int)
    Private joEdt = edt As JavaObject
    joEdt.RunMethod("setHighLightColor", Array(Color))
End Sub
```

```
'Gets the highlight color
Sub getHighLightColor(edt As EditText) As Int
    Private joEdt = edt As JavaObject
    Return joEdt.RunMethod("getHighLightColor", Null)
End Sub
```

3.3.4.9 GetXYCursor

```
'gets the x and y coordinates, in pixels, of the cursor in an EditText view
'the Y coordinate is at the base line, on top or on bottom of the line
'retuens an array of Ints
'X = xy(0) and Y = xy(1)
'the reference is the top left corner of the EditText
'YPosition can be "Baseline", "TopOfLine", "BottomOfLine"
Private Sub GetXYCursor(edt As EditText, YPosition As String) As Int()
    Private joEditText, joLayout As JavaObject
    Private PaddingLeft, PaddingTop, ScrollY, Pos, Line, LineBaseline, LineTop,
LineBottom As Int
    Private xy(2) As Int

    joEditText = edt
    PaddingLeft = joEditText.RunMethod("getPaddingLeft", Null)
    PaddingTop = joEditText.RunMethod("getPaddingTop", Null)
    ScrollY = joEditText.RunMethod("getScrollY", Null)
' pos = edt.SelectionStart
    Pos = joEditText.RunMethod("getSelectionStart", Null)
    joLayout = joEditText.RunMethod("getLayout", Null)
    Line = joLayout.RunMethod("getLineForOffset", Array As Object(Pos)) 'line numbsr
    LineBaseline = joLayout.RunMethod("getLineBaseline", Array As Object(Line))
    LineTop = joLayout.RunMethod("getLineTop", Array As Object(Line))
    LineBottom = joLayout.RunMethod("getLineBottom", Array As Object(Line))

    xy(0) = joLayout.RunMethod("getPrimaryHorizontal", Array As Object(Pos)) +
PaddingLeft ' X coordinate
    Select Case YPosition
        Case "BaseLine"
            xy(1) = LineBaseline + PaddingTop - ScrollY 'base line Y coordinate
        Case "TopOfLine"
            xy(1) = LineTop + PaddingTop - ScrollY 'top of the line Y coordinate
        Case "BottomOfLine"
            xy(1) = LineBottom + PaddingTop - ScrollY 'text bottom line
    End Select
    Return xy
End Sub
```

3.3.5 Label Utils

Examples from the forum [LabelUtils](#).

The methods below can also be used without a sub, like:

```
Private jobj As JavaObject
Private lblTest As Label
'
'
jobj = lblTest
jobj.RunMethod("setLines", Array As Object(2))
jobj.RunMethod("setEllipsize", Array As Object("END"))
```

3.3.5.1 setEllipsize

This property has been added in B4A.

```
'Sets the Ellipsizing of a TextView (Label, EditText)
' Mode can be
'   original text 'This is a test text'
'"START"         ... This is a te
'"MIDDLE"        This is ... text
'"END"           This is a tes...
'"MARQUEE       This is a test t
Sub setEllipsize(TextView As Label, Mode As String)
    Dim jo = TextView As JavaObject
    jo.RunMethod("setSingleLine", Array As Object(True))
    jo.RunMethod("setEllipsize", Array As Object(Mode))
    jo.RunMethod("setSelected", Array As Object(True)) ' needed for MARQUEE
End Sub
```

3.3.5.2 resetEllipsize

```
'Removes the Ellipsizing of a TextView (Label, EditText)
Sub resetEllipsize(TextView As Label)
    Dim jo = TextView As JavaObject
    jo.RunMethod("setSingleLine", Array As Object(False))
End Sub
```

3.3.5.3 setSingleLine

```
'Sets the TextView to single line
Sub setSingleLine(TextView As Label, SingleLine As Boolean)
    Dim jo = TextView As JavaObject
    jo.RunMethod("setSingleLine", Array As Object(SingleLine))
End Sub
```

3.3.5.4 setLines

```
'Limits the line number to the given value
Sub setLines(TextView As Label, LineNumber As Int)
    Dim jo = TextView As JavaObject
    jo.RunMethod("setLines", Array As Object(LineNumber))
End Sub
```

3.3.5.5 setMarqueeRepeat

```
'Sets the Marquee repeat limit
' default is three times
' set -1 for repeat indefinitely
Sub setMarqueeRepeatLimit(TextView As Label, Limit As Int)
    Dim jo = TextView As JavaObject
    jo.RunMethod("setMarqueeRepeatLimit", Array As Object(Limit))
End Sub
```

3.3.5.6 Stop or run the horizontal scrolling

```
jobj.RunMethod("setHorizontallyScrolling", Array As Object(False))
'or
jobj.RunMethod("setHorizontallyScrolling", Array As Object(True))
```

3.3.5.7 setShadowLayer

Test

```
'Adds a shadow to a text
'lbl = the view to add a shadow, can be and EditText or a Label
'Radius = parameter for blur, radius = 1 no blur, the bigger radius the more blur
'dx = horizontal offset of the shadow in pixels
'dy = vertical offset of the shadow in pixels
'Color = shadow color
Sub setShadowLayer(lbl As View, Radius As Float, dx As Float, dy As Float, Color As Int)
    Dim jo = lbl As JavaObject
    jo.RunMethod("setShadowLayer", Array(Radius, dx, dy , Color))
End Sub
```

4 B4J JavaObject

4.1 B4J JavaObjec Methods

4.1.1 CreateEvent

CreateEvent (Interface As String, EventName As String, DefaultReturnValue As Object)

Creates an instance of the interface and binds it to the object.

Interface - The full interface name.

EventName - The prefix of the event sub.

DefaultReturnValue - This value will be returned if no value was returned from the event sub. This can happen if the Activity is paused for example.

B4A example:

```
Sub Activity_Create(FirstTime As Boolean)

    Dim btn As Button
    btn.Initialize("")
    Activity.AddView(btn, 0, 0, 200dip, 200dip)
    Dim jo As JavaObject = btn
    Dim e As Object = jo.CreateEvent("android.view.View.OnTouchListener", "btnTouch",
False)
    jo.RunMethod("setOnTouchListener", Array As Object(e))
End Sub
```

And the event routine:

```
Sub btnTouch_Event (MethodName As String, Args() As Object) As Object
    Dim MotionEvent As JavaObject = Args(1) 'args(0) is View
    Dim x As Float = MotionEvent.RunMethod("getX", Null)
    Dim y As Float = MotionEvent.RunMethod("getY", Null)
    Log(x & ", " & y)
    Dim Action As Int = MotionEvent.RunMethod("getAction", Null)
    Log("Action = " & Action)
    Return True
End Sub
```

And the Java documentation: [View.OnTouchListener](#), [MotionEvent](#).

float	<code>getX()</code> <code>getX(int)</code> for the first pointer index (may be an arbitrary pointer identifier).
float	<code>getXPrecision()</code> Return the precision of the X coordinates being reported.
float	<code>getY()</code> <code>getY(int)</code> for the first pointer index (may be an arbitrary pointer identifier).

<code>int</code>	<code>getAction()</code> Return the kind of action being performed.
------------------	--

If you look at the documentation, you may notice that there is a lot to learn.

4.1.2 CreateEventFromUI

CreateEventFromUI (Interface As String, EventName As String, ReturnValue As Object)

Similar to CreateEvent. The event will be sent to the message queue and then be processed (similar to CallSubDelayed).

4.1.3 GetField

GetField (Field As String)

Gets the value of the given field.

4.1.4 GetFieldJO

GetFieldJO (Field As String)

Similar to GetField. Returns a JavaObject instead of Object.

Returns a JavaObject.

4.1.5 InitializeArray

InitializeArray (ClassName As String, Values As Object())

Creates an array with the given class and values.

Returns a JavaObject.

4.1.6 InitializeContext

B4A only method.

Initializes the object with the current context (current Activity or Service).

Returns a JavaObject.

4.1.7 InitializeNewInstance

InitializeNewInstance (ClassName As String, Params As Object())

Creates a new instance of the given class.

ClassName - The full class name.

Params - An array of objects to pass to the constructor (or Null).

Returns a JavaObject.

4.1.8 InitializeStatic

InitializeStatic (ClassName As String)

Initializes the object. The object will wrap the given class (for static access).

ClassName - The full class name.

Returns a JavaObject.

4.1.9 IsInitialized

Returns a Boolean.

4.1.10 RunMethod

RunMethod (MethodName As String, Params As Object())

Runs the given method and returns the method return value.

MethodName - The case-sensitive method name.

Params - Method parameters (or Null).

Returns an Object.

B4A example:

Get and sets the Labels padding.

```
Private joLabel1 As JavaObject = Label1      'wrap the Label object
Log("Label1 Left padding = " & joLabel1.RunMethod("getPaddingLeft", Null))
Log("Label1 Top padding = " & joLabel1.RunMethod("getPaddingTop", Null))
Log("Label1 Right padding = " & joLabel1.RunMethod("getPaddingRight", Null))
Log("Label1 Bottom padding = " & joLabel1.RunMethod("getPaddingBottom", Null))

joLabel1.RunMethod("setPadding", Array As Object(10dip, 10dip, 10dip, 10dip))
```

Android documentation: [View](#).

Don't be afraid! It's a huge chapter! To see the padding methods, you need to scroll down very deep.

int	<code>getPaddingBottom()</code> Returns the bottom padding of this view.
int	<code>getPaddingEnd()</code> Returns the end padding of this view depending on its resolved layout direction.
int	<code>getPaddingLeft()</code> Returns the left padding of this view.
int	<code>getPaddingRight()</code> Returns the right padding of this view.
int	<code>getPaddingStart()</code> Returns the start padding of this view depending on its resolved layout direction.
int	<code>getPaddingTop()</code>
void	<code>setPadding(int left, int top, int right, int bottom)</code> Sets the padding.

4.1.11 RunMethodJO

RunMethodJO (MethodName As String, Params As Object())

Similar to RunMethod. Returns a JavaObject instead of Object.

Returns a JavaObject.

4.1.12 SetField

SetField (FieldName As String, Value As Object)

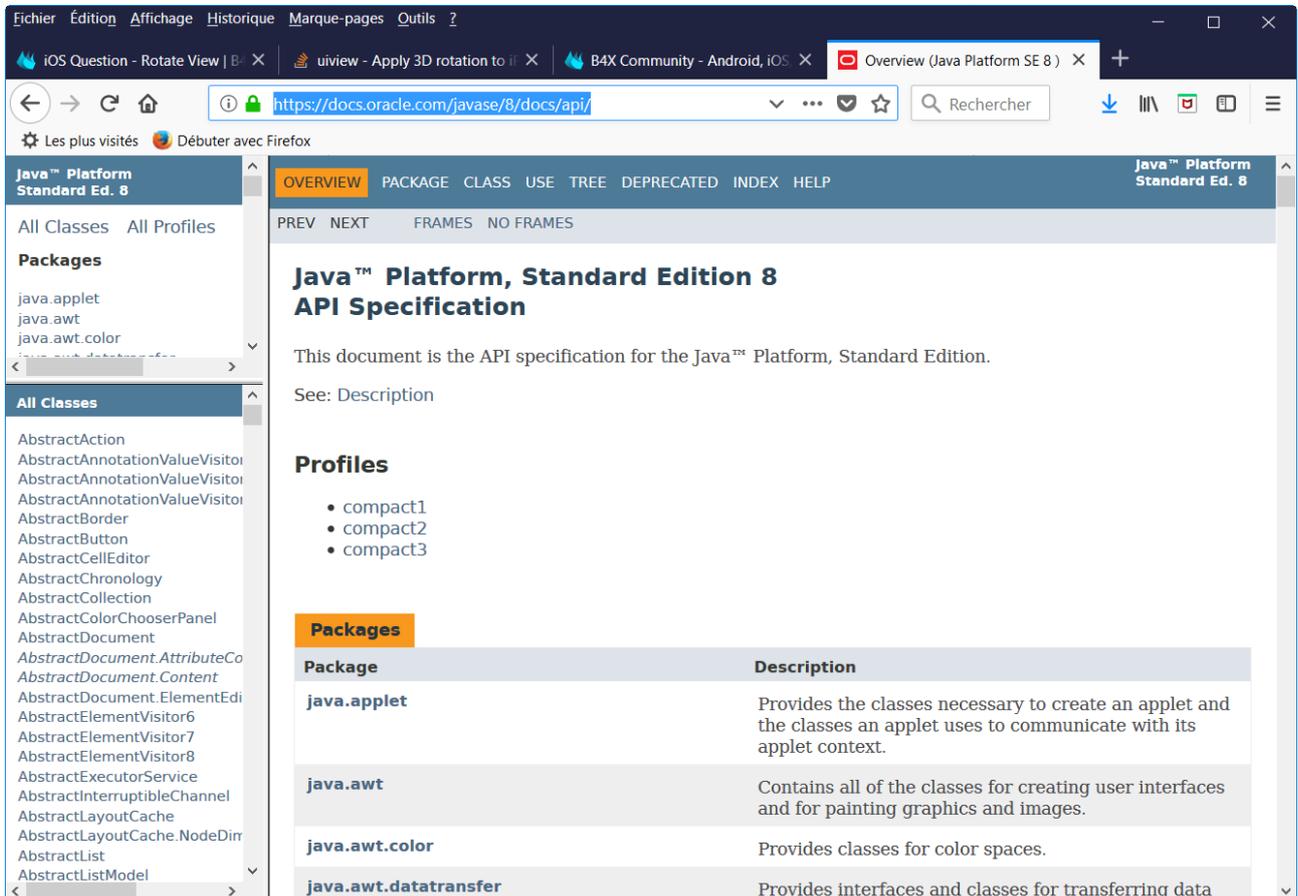
Sets the value of the given field.

Returns Void.

4.2 How to use it

You need to look at the [Java documentation](https://docs.oracle.com/javase/8/docs/api/).

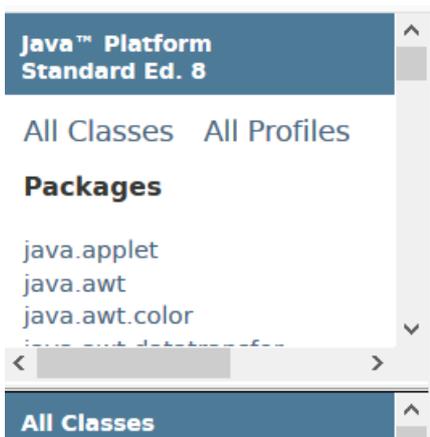
It looks like this:



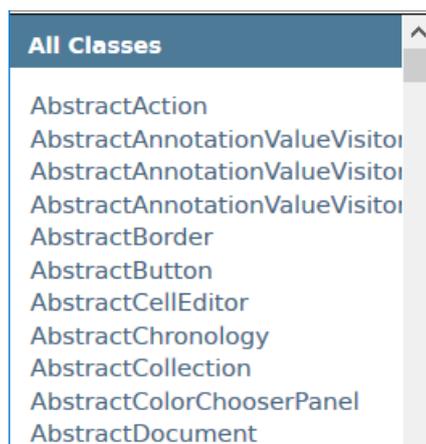
The screenshot shows a web browser displaying the Java Platform Standard Edition 8 API Specification page. The browser's address bar shows the URL <https://docs.oracle.com/javase/8/docs/api/>. The page title is "Java™ Platform, Standard Edition 8 API Specification". The main content area includes a navigation menu with "OVERVIEW", "PACKAGE", "CLASS", "USE", "TREE", "DEPRECATED", "INDEX", and "HELP". Below the navigation menu, there is a section for "Profiles" with a list of "compact1", "compact2", and "compact3". A "Packages" section is highlighted, showing a table with columns "Package" and "Description".

Package	Description
java.applet	Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.
java.awt	Contains all of the classes for creating user interfaces and for painting graphics and images.
java.awt.color	Provides classes for color spaces.
java.awt.datatransfer	Provides interfaces and classes for transferring data

On the top left you find all the Packages, and just below, all classes:



This image shows a close-up of the left sidebar of the Java API documentation. The top section is titled "Packages" and lists several packages: "java.applet", "java.awt", "java.awt.color", and "java.awt.datatransfer". Below this is a section titled "All Classes" which lists various classes and interfaces.



This image shows a close-up of the "All Classes" section of the Java API documentation. It lists a long list of classes and interfaces, including "AbstractAction", "AbstractAnnotationValueVisitor", "AbstractBorder", "AbstractButton", "AbstractCellEditor", "AbstractChronology", "AbstractCollection", "AbstractColorChooserPanel", and "AbstractDocument".

4.3 B4J examples

<https://www.b4x.com/android/forum/threads/google-map-with-custom-style.84894/#content>

<https://www.b4x.com/android/forum/threads/jgauges.70538/#content>

<https://www.b4x.com/android/forum/threads/excel-charts-and-templates-with-jpoi-library.57689/#content>

<https://www.b4x.com/android/forum/threads/class-textflow-similar-to-b4a-b4i-richstring.61237/#content>

<https://www.b4x.com/android/forum/threads/jcanvasext-extends-canvas-methods.86407/#content>

<https://www.b4x.com/android/forum/threads/using-javaobject-and-javafx-linechart.34777/#content>

4.3.1 setMouseTransparent

Enables mouse events being transmitted to underlying nodes.

In B4J, when a node covers other nodes, like a Pane, the mouse events are not submitted to underlying nodes.

This is the same as in B4i, it is inverse in B4A.

In B4i, you can use the `UserInteractionEnabled` property.

In B4J you can use the code below to enable mouse events being submitted to underlying nodes.

```
Private jo = MyNode As JavaObject
jo.RunMethod("setMouseTransparent", Array As Object(True))
```

The value is `False` by default.

4.3.2 setTextAlignment

The standard `Alignment` property does not work properly with multiline labels. It aligns the content based on the first line.

You can use this code instead:

```
Dim jo As JavaObject = Label1
jo.RunMethod("setTextAlignment", Array("CENTER"))
```

4.3.3 Graphics jCanvasExt

The jCanvasExt class can be found in the forum in this thread:

[jCanvasExt Extends Canvas methods.](#)

The Java documentation:

<https://docs.oracle.com/javase/8/javafx/api/javafx/scene/canvas/GraphicsContext.html>

4.3.4 GoogleMaps

The GoogleMaps library is based on this open-source project GMapsFX.
Under the hood it uses JavaFX WebView with GoogleMaps JavaScript API V3.

The [Library Documentation](#).

<https://www.b4x.com/android/forum/threads/googlemapsdemo.91906/#content>

<https://www.b4x.com/android/forum/threads/class-googlemapsextra.56871/#content>

5 B4i NativeObject

5.1 B4i NativeObject methods

5.1.1 ArrayFromEdgeInsets

ArrayFromEdgeInsets (EdgeInsets As NSData*)

Converts a UIEdgeInsets to an array with the Left, Top, Right and Bottom values.

Returns: Float()

5.1.2 ArrayFromPoint

ArrayFromPoint(Point As NSData*)

Converts a CGPoint to an array with the X and Y values.

Returns: Float()

5.1.3 ArrayFromRange

ArrayFromRange (Range As NSData*)

Converts a NSRange to an array with the location and length values.

Returns: Int()

5.1.4 ArrayFromRect

ArrayFromRect (Rect As NSData*)

Converts a CGRect to an array with the X, Y, Width and Height values.

Returns: Float()

5.1.5 ArrayFromSize

ArrayFromSize (Size As NSData*)

Converts a CGSize to an array with the Width and Height values.

Returns: Float()

5.1.6 ArrayToNSData

ArrayToNSData (Arr As Byte())

Converts an array of bytes to NSData object.

Returns : NSObject

5.1.7 AsBoolean

Returns the wrapped object as a Boolean value.

Returns : BOOL

5.1.8 AsNumber

Returns the wrapped object as a numeric value.

Returns: NSObject

5.1.9 AsString

Returns the wrapped object as a String value.

Returns: NSString

5.1.10 ColorToUIColor

ColorToUIColor (Color As Int)

Converts B4i color value to UIColor.

Returns: UIColor

5.1.11 CreateBlock

CreateBlock (bi As B4I*, SubName As NSString*, NumberOfParameters As Int, Delayed As BOOL)

Creates a block that raises the given sub.

SubName - The event prefix. The sub signature should be: SubName_Event (Args() As Object) As Object.

NumberOfParameters - The number of parameters passed to the block. Should be between 0 to 2.

Delayed - Whether the event is raised immediately or it is sent to the message queue.

Returns: Id

5.1.12 GetField

GetField (Field As NSString*)

Gets the value of the given property or field.

Returns: B4INativeObject

5.1.13 Initialize(

Initialize (ClassName As NSString*)

Initializes a new object and sets it to reference the given class.

Returns: B4INativeObject

5.1.14 IsInitialized

Tests whether this object was initialized.

Returns: BOOL

5.1.15 MakeEdgeInsets

MakeEdgeInsets (Left As Float, Top As Float, Right As Float, Bottom As Float)

Creates a UIEdgeInsets struct.

Returns: NSData

5.1.16 MakePoint

Creates a CGPoint struct.

Returns: NSData

5.1.17 MakeRange

MakeRange (StartLocation As Int, Length As Int)

Creates a NSRange struct.

Returns: NSData

5.1.18 MakeRect

MakeRect (X As Float, Y As Float, Width As Float, Height As Float)

Creates a CGRect struct.

Returns: NSData

5.1.19 MakeSize

MakeSize (Width As Float, Height As Float)

Creates a CGSize struct.

Returns: NSData

5.1.20 NSDataToArray

NSDataToArray (Data As NSObject*)

Converts an NSData object to an array of bytes.

Returns: Byte()

5.1.21 RunMethod

RunMethod (MethodName As NSString*, Params As NSObject())

Runs the given method. Params can be Null if there are no parameters.
Note that a method name is the full name with the colons.

Returns: B4INativeObject

5.1.22 RunMethodWithBlocks

RunMethodWithBlocks (MethodName As NSString*, Params As NSObject())

Similar to RunMethod. Runs a method that expects a block.

Returns: B4INativeObject

5.1.23 SetField

SetField (FieldName As NSString*, Value As NSObject*)

Sets the value of the given property or field.

Returns: Void

5.1.24 Tag

Gets or sets the Tag object. This is a placeholder for any object you like to tie to this object.

Readable As NSObject

Writable As NSObject

5.1.25 UIColorToColor

UIColorToColor (UIColor As UIColor*)

Converts a UIColor to B4i color value.

Returns: Int

5.2 How to use it

5.3 B4i examples

5.3.1 SetButtonTextColor

```
'state: 0 = normal, 1 = pressed, 2 = disabled
Sub SetButtonTextColor(btn As Button, clr As Int, state As Int)
    Dim no As NativeObject = btn
    no.RunMethod("setTitleColor:forState:", Array(no.ColorToUIColor(clr), state))
End Sub
```

```
SetButtonTextColor(Button1, Colors.Green, 1)
```

5.3.2 Hide the StatusBar

1. Add this attribute:

```
#PlistExtra: <key>UIViewControllerBasedStatusBarAppearance</key><false/>
```

2. Add this code:

```
Dim no As NativeObject = app
no.RunMethod("setStatusBarHidden:animated:", Array(True, False))
```

5.3.3 Setting the status bar to use a light theme

1. Add this attribute:

```
#PlistExtra: <key>UIViewControllerBasedStatusBarAppearance</key><false/>
```

2. Add this code:

```
Sub SetStatusBarStyleLight
    Dim no As NativeObject = app
    no.RunMethod("setStatusBarStyle:", Array(1))
End Sub
```

5.3.4 Check StatusBar state

```
Sub IsStatusBarHidden As Boolean
    Dim no As NativeObject = App
    Return no.GetField("statusBarHidden").AsBoolean
End Sub
```

5.3.5 Get Border properties

Gets the Border properties of a Panel :

```
Dim no As NativeObject = Panel1
Dim BorderWidth As Float = no.GetField("layer").GetField("borderWidth").AsNumber
Dim BorderRadius As Int = no.GetField("layer").GetField("cornerRadius").AsNumber
Dim BorderUIColor As Int= no.UIColorToColor(no.GetField("layer").GetField("borderColor"
))
```

5.3.6 Getting current device language

```
Sub GetPreferredLanguage As String
    Dim no As NativeObject
    Return no.Initialize("NSLocale").RunMethod("preferredLanguages", Null).RunMethod("obj
ectAtIndex:", Array(0)).AsString
End Sub
```

Or, without the county ID.

```
Sub GetPreferredLanguage As String
    Dim no As NativeObject
    Dim s As String = no.Initialize("NSLocale").RunMethod("preferredLanguages", Null).Run
Method("objectAtIndex:", Array(0)).AsString
    If s.Length > 2 Then s = s.Substring2(0, 2)
    Return s
End Sub
```

5.3.7 Set the NavigationBar text color

```
Private Sub Application_Start (Nav As NavigationController)
    NavControl = Nav
    Page1.Initialize("Page1")
    Page1.RootPanel.LoadLayout("1")
    NavControl.ShowPage(Page1)
    Page1.Title = "Title"
    SetTitleColor(Nav, Colors.Red)
End Sub
```

```
Sub SetTitleColor(nav As NavigationController, clr As Int)
    Dim attributes As NativeObject
    attributes = attributes.Initialize("B4IAttributedString").RunMethod("createAttribute
s::", Array(Font.CreateNew(18), attributes.ColorToUIColor(clr)))
    Dim no As NativeObject = nav
    no.GetField("navigationBar").RunMethod("setTitleTextAttributes:", Array(attributes))
End Sub
```