Contents



QWidget Class Reference

Public Types

Properties





Public Functions

NameSame and any state of the control of

Reimplemented Public Functions
Public Slots Variable Variab
Signals
Static Public Members
Protected Functions Section Continue
Reimplemented Protected Functions
Protected Slots
Related Non-Members
Macros
Detailed Description

Top-Level and Child Widgets

Widget Style Sheets



Transparency and Double Buffering

Contents Propagated (Default) With autoFillB	ackground set With WA_OpaquePaintEvent set
<u>O</u> ptions	<u>O</u> ptions
☐ C <u>a</u> se sensitive ☐ W <u>h</u> ole words	☐ Case sensitive ☐ Whole words
■ Search forwards ☐ From start of text	■ Search forwards ☐ From start of text
Contents Propagated (Default)	With autoFillBackground set
Creating Translucent Windows Native Widgets vs Alien Widgets	
Softkeys	
Mombor Type Decumentation	
Member Type Documentation enum QWidget::RenderFlag flags QWidget::RenderFlags	
The second secon	

Property Documentation

acceptDrops: bool

NAME OF THE PROPERTY OF THE PR
The state of the s
Marketon
accessibleDescription: QString
NATIONAL ATTEMPT
Management of the Control of the Con
-
accessibleName: QString
The second secon
Management 1
autoFillPaakground + haal
autoFillBackground: bool
Anna Anna Anna Anna Anna Anna Anna Anna
Name
Security Control of the Control of t
See May 1 (Fee), played with the 1 (Fee) And played before process of the long proce
baseSize : QSize
-
baseSize : QSize
-
Section of the sectio
childrenRect : const QRect
childrenRect : const QRect childrenRegion : const QRegion
childrenRect : const QRect childrenRegion : const QRegion
childrenRect : const QRect childrenRegion : const QRegion
childrenRect : const QRect childrenRegion : const QRegion
childrenRect : const QRect childrenRegion : const QRegion childrenRegion : const QRegion contextMenuPolicy : Qt::ContextMenuPolicy
childrenRect : const QRect childrenRegion : const QRegion
childrenRect : const QRect childrenRegion : const QRegion childrenRegion : const QRegion contextMenuPolicy : Qt::ContextMenuPolicy
childrenRect : const QRect childrenRegion : const QRegion childrenRegion : const QRegion contextMenuPolicy : Qt::ContextMenuPolicy

The state of the s
enabled : bool
Name of the second of the seco
focus : const bool
focusPolicy : Qt::FocusPolicy
Manager State of the Control of the
font : QFont
National content of the company of the content of t
frameGeometry : const QRect
Marine Commission of the Commi
frameSize : const QSize
Full Courses and a second hand
fullScreen : const bool
geometry : QRect

height : const int
inputMethodHints : Qt::InputMethodHints
Table desired and the second s
isActiveWindow : const bool
CANADATA
layoutDirection : Qt::LayoutDirection
Tentania di America di
Legale - Ol cools
locale : QLocale
the Marian
maximized : const bool
maximumHeight: int
antena a mangina a mangina

maximumSize : QSize

-
maximumWidth: int
maximumvidin : ini
Auto-Autority Auto-Autority Auto-Autority Autority Autori
No. No. allocated and anti-state and
mentalises a manufactur and distribution of the second o
Name of the state
minimized : const bool
November and Novem
Name of the Control o
AMAZONIA.
minimumHeight: int
Mark Hame
NAME AND ADDRESS A
minimum Sizo + OSizo
minimumSize : QSize
Name of the content o
The second secon
The second secon
The second secon
The second secon
minimumSizeHint : const QSize
minimumSizeHint : const QSize minimumWidth : int
minimumSizeHint : const QSize minimumWidth : int
minimumSizeHint : const QSize minimumWidth : int
minimumSizeHint: const QSize minimumWidth: int
minimumSizeHint: const QSize minimumWidth: int
minimumSizeHint : const QSize minimumWidth : int modal : const bool
minimumSizeHint: const QSize minimumWidth: int modal: const bool
minimumSizeHint: const QSize minimumWidth: int minimumWidth: int modal: const bool
minimumSizeHint: const QSize minimumWidth: int minimumWidth: int modal: const bool
minimumSizeHint: const QSize minimumWidth: int minimumWidth: int modal: const bool
minimumSizeHint: const QSize minimumWidth: int minimumWidth: int modal: const bool

Malaconducty
normalGeometry : const QRect
entender De minimipal
palette : QPalette
Name of the Control o
relative inflate Applies in Appli
MARIEM CONTROL OF THE
pos: QPoint
Nagarantasanapananapan
Same Control of the C
Manager (A)
rect : const QRect
Name and the property of the second s
And A supple and the Control of the
size: QSize
Nagaration of Administration
Name Associated Associ
size Hint a parent OCina
sizeHint : const QSize
eater date eater
sizeIncrement : QSize

6 100 (6 10 10 10 10 10 10 10 10 10 10 10 10 10
NAME OF THE PROPERTY OF THE PR

sizePolicy: QSizePolicy
A STATE OF THE STA
statusTip: QString
The state of the s
and the second s
styleSheet: QString
Table States Andrews A
to all Tipe a OCtain as
toolTip : QString
Table 1
updatesEnabled : bool
visible: bool
The state of the s
whatsThis: QString
Name of the second of the seco
width: const int

Market Colonial Colon
windowFilePath: QString
Contraction of the Contraction o
The Administration of the Conference of the Conf
Management
windowFlags: Qt::WindowFlags
Tenning State (State of State
windowlcon: Qlcon
No. No. 1 Acres 1 Acre
windowlconText: QString
The state of the s
window Modality: Qt:: Window Modality
Service desired to the
windowModified : bool
The Administration of
windowOpacity: double

windowTitle: QString

The state of the s
x : const int
STATE OF THE STATE
y : const int

Member Function Documentation
QWidget:: QWidget (QWidget * $parent = 0$, Qt::WindowFlags $f = 0$)
QWidget::~QWidget ()
void OWidest uggtion Event (OAction Event * event) [virtual protected]
<pre>void QWidget::actionEvent (QActionEvent * event) [virtual protected]</pre>
QList <qaction *=""> QWidget::actions () const</qaction>
void QWidget::activateWindow ()
void QWidget:: addAction (QAction * action)
void QWidget ::addActions (QList <qaction *=""> actions)</qaction>
void OWidgetvadiustSize O
void QWidget:: adjustSize ()

QPalette::ColorRole QWidget:: backgroundRole () const
void QWidget:: changeEvent (QEvent * event) [virtual protected]
QWidget * QWidget::childAt (int x, int y) const
QWidget * QWidget::childAt (const QPoint & p) const
void QWidget:: clearFocus ()
——————————————————————————————————————
void QWidget:: clearMask ()
had Outidean along O [elek]
bool QWidget::close () [slot]
void QWidget:: closeEvent (QCloseEvent * event) [virtual protected]
QMargins QWidget::contentsMargins () const
QRect QWidget::contentsRect () const
void QWidget:: contextMenuEvent (QContextMenuEvent * event) [virtual protected]
void QWidget ::create (WId window = 0 , bool initializeWindow = true , bool destroyOldWindow =
true) [protected]
void QWidget::customContextMenuRequested (const QPoint & pos) [signal]

<pre>void QWidget::destroy (bool destroyWindow = true, bool destroySubWindows = true) [protected]</pre>
void QWidget:: dragEnterEvent (QDragEnterEvent * event) [virtual protected]
<pre>void QWidget::dragLeaveEvent (QDragLeaveEvent * event) [virtual protected]</pre>
void QWidget:: dragMoveEvent (QDragMoveEvent * event) [virtual protected]
void QWidget:: dropEvent (QDropEvent * event) [virtual protected]
Wld QWidget::effectiveWinId () const
void QWidget:: ensurePolished () const
void QWidget:: enterEvent (QEvent * event) [virtual protected]
bool QWidget:: event (QEvent * event) [virtual protected]
QWidget * QWidget::find (Wld id) [static]
void QWidget:: focusInEvent (QFocusEvent * event) [virtual protected]
bool QWidget:: focusNextChild () [protected]

bool QWidget:: focusNextPrevChild (bool <i>next</i>) [virtual protected]
void QWidget:: focusOutEvent (QFocusEvent * event) [virtual protected]
bool QWidget:: focusPreviousChild () [protected]
QWidget * QWidget::focusProxy () const
QWidget * QWidget::focusWidget () const
QFontInfo QWidget::fontInfo () const
QFontMetrics QWidget::fontMetrics () const ——— QPalette::ColorRole QWidget::foregroundRole () const
void QWidget ::getContentsMargins (int * <i>left</i> , int * <i>top</i> , int * <i>right</i> , int * <i>bottom</i>) const
HDC QWidget::getDC () const [virtual]
<pre>void QWidget::grabGesture (Qt::GestureType gesture, Qt::GestureFlags flags = Qt::GestureFlags())</pre>
void QWidget:: grabKeyboard ()
void QWidget:: grabMouse ()

void QWidget:: grabMouse (const QCursor & <i>cursor</i>)
<pre>int QWidget::grabShortcut (const QKeySequence & key, Qt::ShortcutContext context = Qt::WindowShortcut)</pre>
QGraphicsEffect * QWidget::graphicsEffect () const
QGraphicsProxyWidget * QWidget::graphicsProxyWidget () const
bool QWidget:: hasEditFocus () const
int OWide at the circlet CovWidth (int w) count [virtual]
int QWidget:: heightForWidth (int w) const [virtual]
void QWidget:: hide () [slot]
void QWidget:: hideEvent (QHideEvent * event) [virtual protected]
QInputContext * QWidget::inputContext ()
void QWidget ::inputMethodEvent (QInputMethodEvent * event) [virtual protected]
QVariant QWidget:: inputMethodQuery (Qt::InputMethodQuery query) const [virtual]
void QWidget:: insertAction (QAction * before, QAction * action)

with a finish of the colonial agents the colonial and the
<pre>void QWidget::insertActions (QAction * before, QList<qaction *=""> actions)</qaction></pre>
bool QWidget:: isAncestorOf (const QWidget * <i>child</i>) const
bool QWidget::isEnabledTo (QWidget * ancestor) const
bool QWidget:: isHidden () const
The state of the s
bool QWidget::isVisibleTo (QWidget * ancestor) const
bool QWidget::isWindow () const
<pre>void QWidget::keyPressEvent (QKeyEvent * event) [virtual protected]</pre>
void QWidget ::keyReleaseEvent (QKeyEvent * event) [virtual protected]
QWidget * QWidget::keyboardGrabber () [static]
QLayout * QWidget::layout () const
<pre>void QWidget::leaveEvent (QEvent * event) [virtual protected]</pre>

void QWidget:: lower () [slot]
Qt::HANDLE QWidget:: macCGHandle () const
bool QWidget:: macEvent (EventHandlerCallRef <i>caller</i> , EventRef <i>event</i>) [virtual protected]
Qt::HANDLE QWidget:: macQDHandle () const
QPoint QWidget::mapFrom (QWidget * parent, const QPoint & pos) const
QPoint QWidget::mapFromGlobal (const QPoint & pos) const
QPoint QWidget::mapFromParent (const QPoint & pos) const
OD-int OMI doubt was To (OMI doubt to a root of OD-int O may) and to
QPoint QWidget:: mapTo (QWidget * parent, const QPoint & pos) const
OPoint OWidgetumen To Clobal (const OPoint & nos) const
QPoint QWidget::mapToGlobal (const QPoint & pos) const
QPoint QWidget::mapToParent (const QPoint & pos) const
QRegion QWidget::mask () const
<pre>int QWidget::metric (PaintDeviceMetric m) const [virtual protected]</pre>
void QWidget ::mouseDoubleClickEvent (QMouseEvent * event) [virtual protected]
QWidget * QWidget::mouseGrabber () [static]

void QWidget ::mouseMoveEvent (QMouseEvent * event) [virtual protected]
void QWidget:: mousePressEvent (QMouseEvent * event) [virtual protected]
void QWidget:: mouseReleaseEvent (QMouseEvent * event) [virtual protected]
void QWidget:: moveEvent (QMoveEvent * event) [virtual protected]
QWidget * QWidget::nativeParentWidget () const
QWidget * QWidget::nextInFocusChain () const
void QWidget:: overrideWindowFlags (Qt::WindowFlags flags)
QPaintEngine * QWidget::paintEngine () const [virtual]
void QWidget: :paintEvent (QPaintEvent * event) [virtual protected]
QWidget * QWidget::parentWidget () const
QPlatformWindow * QWidget::platformWindow () const

QPiatformWindowFormat Qwidget::piatformWindowFormat () const
ONA idea at the ONA idea at a second a
QWidget * QWidget::previousInFocusChain () const
heal OWidgeturgus Front (OWS Front towent) [virtual protected]
bool QWidget ::qwsEvent (QWSEvent * event) [virtual protected]
void QWidget:: raise () [slot]
void OWidesturalesse DC (LIDC bds) senst [virtual]
void QWidget ::releaseDC (HDC <i>hdc</i>) const [virtual]
void QWidget:: releaseKeyboard ()
void QWidget:: releaseMouse ()
—
void QWidget:: releaseShortcut (int id)
void QWidget:: removeAction (QAction * action)
void QWidget:: render (QPaintDevice * target, const QPoint & targetOffset = QPoint() , const QRegion & sourceRegion = QRegion() , RenderFlags renderFlags = RenderFlags(
DrawWindowBackground DrawChildren))
void QWidget:: render (QPainter * painter, const QPoint & targetOffset = QPoint() , const QRegion & sourceRegion = QRegion() , RenderFlags renderFlags = RenderFlags(DrawWindowBackground DrawChildren))
void QWidget:: repaint () [slot]

Name of the contract production of the contract
void QWidget:: repaint (int x, int y, int w, int h)
void QWidget:: repaint (const QRect & rect)
void QWidget:: repaint (const QRegion & rgn)
void QWidget:: resizeEvent (QResizeEvent * event) [virtual protected]
bool QWidget:: restoreGeometry (const QByteArray & geometry)
QByteArray QWidget:: saveGeometry () const
void QWidget:: scroll (int dx, int dy)
void QWidget:: scroll (int <i>dx</i> , int <i>dy</i> , const QRect & <i>r</i>)
void QWidget:: setAttribute (Qt::WidgetAttribute attribute, bool on = true)
void QWidget:: setBackgroundRole (QPalette::ColorRole <i>role</i>)
void QWidget:: setContentsMargins (int <i>left</i> , int <i>top</i> , int <i>right</i> , int <i>bottom</i>)

void QWidget:: setContentsMargins (const QMargins & <i>margins</i>)
void QWidget:: setDisabled (bool <i>disable</i>) [slot]
void QWidget:: setEditFocus (bool <i>enable</i>)
void QWidget:: setFixedHeight (int <i>h</i>)
void QWidget:: setFixedSize (const QSize & s)
void QWidget:: setFixedSize (int w, int h)
void QWidget:: setFixedWidth (int w)
void QWidget:: setFocus (Qt::FocusReason <i>reason</i>)
void QWidget:: setFocus () [slot]
void QWidget:: setFocusProxy (QWidget * w)
void QWidget:: setForegroundRole (QPalette::ColorRole <i>role</i>)
void QWidget:: setGraphicsEffect (QGraphicsEffect * effect)

void QWidget:: setHidden (bool <i>hidden</i>) [slot]
void QWidget::setInputContext (QInputContext * context)
void QWidget:: setLayout (QLayout * layout)
void QWidget:: setMask (const QBitmap & bitmap)
void QWidget:: setMask (const QRegion & region)
void QWidget:: setParent (QWidget * parent)
<pre>void QWidget::setParent (QWidget * parent, Qt::WindowFlags f)</pre>
void QWidget:: setPlatformWindow (QPlatformWindow * window)
void QWidget:: setPlatformWindowFormat (const QPlatformWindowFormat & format)
void QWidget:: setShortcutAutoRepeat (int <i>id</i> , bool <i>enable</i> = true)

National personal of the Control of
void QWidget:: setShortcutEnabled (int <i>id</i> , bool <i>enable</i> = true)
void QWidget::setStyle (QStyle * style)
void QWidget:: setTabOrder (QWidget * <i>first</i> , QWidget * <i>second</i>) [static
void QWidget:: setWindowRole (const QString & role)
void QWidget:: setWindowState (Qt::WindowStates <i>windowState</i>)
void QWidget:: setWindowSurface (QWindowSurface * surface)
void QWidget:: setupUi (QWidget * <i>widget</i>)
void QWidget:: show () [slot]
void QWidget ::showEvent (QShowEvent * event) [virtual protected]
void QWidget:: showFullScreen () [slot]

void QWidget:: showMaximized () [slot]
void QWidget:: showMinimized () [slot]
and ONG Local about the supplication of Colored
void QWidget:: showNormal () [slot]
void QWidget:: stackUnder (QWidget * w)
QStyle * QWidget::style () const
void QWidget:: tabletEvent (QTabletEvent * event) [virtual protected]
bool QWidget:: testAttribute (Qt::WidgetAttribute attribute) const
bool QWidget::underMouse () const
void QWidget:: ungrabGesture (Qt::GestureType <i>gesture</i>)
void QWidget:: update () [slot]
void QWidget:: update (int x, int y, int w, int h)
void QWidget:: update (const QRect & rect)
void QWidget:: update (const QRegion & <i>rgn</i>)
void QWidget:: updateGeometry ()

to application agree, and considerate and cons
void QWidget::updateMicroFocus () [protected slot]
QRegion QWidget::visibleRegion () const
void QWidget:: wheelEvent (QWheelEvent * event) [virtual protected]
bool QWidget:: winEvent (MSG * message, long * result) [virtual protected]
Wld QWidget::winld () const
- OWidget * OWidget: window () const
QWidget * QWidget::window () const
QString QWidget::windowRole () const
Qt::WindowStates QWidget::windowState () const
QWindowSurface * QWidget::windowSurface () const
Qt::WindowType QWidget::windowType () const
bool QWidget:: x11Event (XEvent * event) [virtual protected]
const QX11Info & QWidget::x11Info () const
Qt::HANDLE QWidget:: x11PictureHandle () const

Related Non-Members
typedef QWidgetList
typedef Wld
_
Macro Documentation
Macro Documentation QWIDGETSIZE_MAX