

Les bases graphiques : la librairie « Standard Widget Toolkit » (fin)

Table des matières

- LES BASES GRAPHIQUES : LA LIBRAIRIE « STANDARD WIDGET TOOLKIT » (FIN)..... 1**
- TABLE DES MATIERES 2
- L'EDITION DE TEXTE (1) 3
 - la classe *StyledText*..... 3
 - les différents *listeners* 3
- L'EDITION DE TEXTE (2) 4
 - quelques méthodes de *StyledText* 4
- L'EDITION DE TEXTE (2) 5
 - quelques méthodes de *StyledText* (suite)..... 5
- L'EDITION DE TEXTE (2) 6
 - la classe *StyleRange* 6
 - la manipulation des *StyleRange* dans un *StyledText*..... 6
- LES BARRES D'OUTILS (2) 7
 - quelques généralités 7
 - quelques méthodes de la classe *ToolBar* 7
 - quelques méthodes de la classe *ToolItem* 7
- LES BARRES D'OUTILS (2) 8
 - un exemple simple..... 8
- UN DERNIER COMPOSITE 9
 - la classe *ViewForm*..... 9

la classe StyledText

- un éditeur de texte basique personnalisable supportant de façon native :
la gestion du curseur, le déplacement via les flèches, la sélection de portion de texte, la gestion du clipboard
- permettant de :
coloriser le texte, de contrôler les retours à la ligne, d'associer des actions à certaines touches

les différents listeners

- **SelectionListener** : appelé lorsque le StyledText est sélectionné
- **LineBackgroundListener** : appelé avant l'affichage d'une ligne pour permettre la définition du background
- **LineStyleListener** : appelé avant l'affichage d'une ligne pour permettre la définition du style de la ligne
- **VerifyKeyListener** : appelé lorsqu'une touche est frappé
- **VerifyListener** : appelé avant modification du texte dans le StyledText
- **ModifyListener** : appelé après modification du texte dans le StyledText
- **ExtendedModifyListener** : étend ModifyListener

quelques méthodes de StyledText

void setContent(Composite fils)
void setAlwaysShowScrollBars(boolean show), boolean getAlwaysShowScrollBars()
void setMinimumSize(int width, int height)
void setOrigin(int x, int y), void setOrigin(Point orig), point getOrigin()

- La gestion du texte
String getText(), String getText(int start, int length)
void insert(String aTxt), void append(String aTxt), void replaceTextRange(int start, int length, String aTxt)
- La gestion des sélections
Point getSelectionRange(), String getSelectionText(), paste()
- la manipulation du clipboard
cut(), copy(), paste()
- l'impression
Runnable print(Printer aP)
Runnable print(Printer aP, StyledTextPrintOptions aOpts)

quelques méthodes de StyledText (suite)

- l'association d'action (prédéfinie) à des touches
int getKeyBinding(int aKey), void setKeyBinding(int aKey, int aAction)

```
.....  
int altAction = stText.getKeyBinding('a' | SWT.ALT);  
int altAction = stText.getKeyBinding(SWT.SHIFT | SWT.ALT);  
stText.setKeyBinding('t' | SWT.ALT, ST.TOGGLE_OVERWRITE);  
.....
```

Action prédéfinie : constantes de la classe ST

```
TEXT_START, TEXT_END  
WORD_NEXT, WORD_PREVIOUS  
.....  
COPY, CUT, PASTE  
DELETE_NEXT, DELETE_PREVIOUS  
DELETE_WORD_NEXT,  
DELETE_WORD__PREVIOUS  
LINE_DOWN, LINE_UP,  
.....  
SELECT_TEXT_START, SELECT_TEXT_END  
SELECT_PAGE_START, SELECT_PAGE_END  
.....
```

la classe StyleRange

- définit une portion contigüe de texte ayant des caractéristiques propres :
start, length + background, foreground + fontStyle (SWT.NORMAL, SWT.BOLD)
- quelques ctors : **StyleRange(), StyleRange(int start, int length, Color foreground, Color background)**

```
StyleRange sr = new StyleRange();  
sr.start = 0;  
sr.length = 5;  
sr.setForeground(display.getSystemColor(SWT.COLOR_GREEN));  
sr.fontStyle = SWT.BOLD;
```

la manipulation des StyleRange dans un StyledText

- la classe StyledText offre des setters et des getters
StyleRange[] getStyleRanges(), StyleRange[] getStyleRanges(int start, int length)
void replaceStyleRanges(int start, int length, StyleRange[] aRanges)
void setStyleRange(StyleRange aRange), void setStyleRanges(StyleRange[] aRanges)

```
styledText.setStyleRange(new StyleRange(0, 5, orange, null) ;
```

Les barres d'outils (2)

quelques généralités

- barre d'outils : **ToolBar** est une barre de **ToolItem** (boutons standards ou toggle ou menus dropdown)
- sous 2 formes : toolbar et coolbar
- les types possibles pour la barre sont des constantes de **SWT** : **FLAT, WRAP, RIGHT, HORIZONTAL, VERTICAL, SHADOW_OUT**
- les types possibles pour les items : **CHECK, DROP_DOWN, PUSH, RADIO, SEPARATOR**

quelques méthodes de la classe ToolBar

ToolItem getItem(int index), ToolItem[] getItems(), ToolItem getItem(Point point), int getItemCount(), int indexOf(ToolItem itm)

création de groupes de controls
SWT.SEPARATOR est un délimiteur
de fin de groupe !!!

quelques méthodes de la classe ToolItem

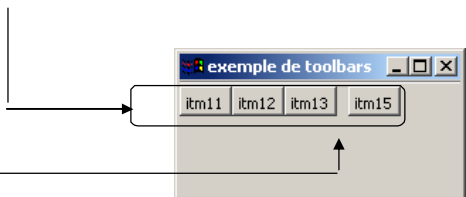
Control getControl(), void setControl(Control ctl)
String getText(), void setText(String txt),
Image getImage(), Image getDisabledImage(), Image getHotImage(), void setImage(image img),
String getToolTipText(), void setToolTipText(String txt)

void addSelectionListener(SelectionListener lst)

Les barres d'outils (2)

un exemple simple

```
shell.setLayout(new FillLayout());  
.....  
ToolBar tb1 = new ToolBar(shell, SWT.SHADOW_OUT + SWT.HORIZONTAL);  
ToolItem itm11 = new ToolItem(tb1, SWT.PUSH);  
itm11.setText("itm11");  
ToolItem itm12 = new ToolItem(tb1, SWT.PUSH);  
itm12.setText("itm12");  
ToolItem itm13 = new ToolItem(tb1, SWT.PUSH);  
itm13.setText("itm13");  
ToolItem itm14 = new ToolItem(tb1, SWT.SEPARATOR);  
itm14.setText("itm14");  
ToolItem itm15 = new ToolItem(tb1, SWT.CHECK);  
itm15.setText("itm15");  
.....  
.....
```



Un dernier composite

la classe ViewForm

- un composite muni de 3 controls sur une ligne (en haut)
 - une image (à gauche) + un texte
 - une toolbar
 - un bouton de fermeture

