

Interface Homme Machine Python-Tkinter Travaux Pratiques N° 1

Objetif :

- Réaliser une application avec une interface graphique pour calculer les éléments d'une facture.
- Les widgets à utiliser sont:
 - ✓ Label (Etiquette)
 - ✓ Entry (Zone de texte)
 - ✓ Button (Bouton de commande)
 - ✓ LabelFrame (Cadre avec étiquette)
- Utiliser les variables de contrôle à relier aux widgets pour effectuer les calculs (Voir annexe1 du TP)

L'interface graphique de l'application doit avoir le format ci-dessous.

Interface graphique de l'application

Données	
Montant Brut:	10000
Taux Remise:	0.02
Taux Escompte:	0.01
Taux de TVA:	0.2

Calculer

Résultats	
remise:	200.0
Net Commercial:	9800.0
Escompte:	98.0
Net Financier:	9702.0
Taxe:	1940.4
Net A Payer:	11642.4

Cet interface comporte les widgets suivants :

Widget	Option	Valeur
Définition de la fenêtre principale de l'application:		
<pre>root = Tk() root.title("Facturation")</pre>		
Lfdonnees : LabelFrame	Fenetre parent	root
	text	"Données"
	font	("Helvetica", 32)
	fg	"blue"
	bg	"#00ff00"
<pre>lfdonnees = LabelFrame(root, text="Données", font=("Helvetica", 32), fg="blue", bg="#00ff00")</pre>		
Position sur la fenêtre racine: 1 ^{ère} ligne, 1 ^{ère} colonne (row=0, column=0) <pre>lfdonnees.grid(row=0, column=0)</pre>		
Définition d'un objet font qui va être utilisé par les différents widgets:		
<pre>my_font = font.Font(root, ("courier new", 16, "bold italic"))</pre>		
lbMB : Label	fenetre	Lfdonnees : cadre avec etiquette défini ci-dessus
	text	"montant Brut : "
	font	My_font
	width	20
	anchor	"e"
	borderwidth	2
	relief	"sunken"
<pre>lbMB = Label(lfdonnees, text="Montant Brut:", font=my_font, width=20, anchor="e", borderwidth=2, relief='sunken')</pre>		
Position sur la fenêtre lfdonnees: 1 ^{ère} ligne, 1 ^{ère} colonne (row=0, column=0) <pre>lbMB.grid(row=0, column=0)</pre>		
Définition d'une variable de contrôle à relier avec le widget Entry entMB défini ci-dessous. <pre>mbvar = DoubleVar(root)</pre>		
entMB: Entry	fenetre	Lfdonnees : cadre avec etiquette défini ci-dessus
	textvariable	Mbvar
	font	my_font
	justify	"center",
	borderwidth	2
relief	"sunken"	
<pre>entMB = Entry(lfdonnees, textvariable=mbvar, font=my_font, justify="center",borderwidth=2, relief='sunken')</pre>		

Position sur la fenêtre lfdonnees: 1 ^{ère} ligne, 2 ^{ème} colonne (row=0, column=1) entMB.grid(row=0, column=1)	
lbTR: Label	Défini de la même manière que le Label lbMB
lbTR = Label(lfdonnees, text="Taux Remise:", font=my_font, width=20, anchor="e", borderwidth=2, relief='sunken')	
Position sur la fenêtre lfdonnees: 3 ^{ème} ligne, 1 ^{ère} colonne (row=2, column=0) lbTR.grid(row=2, column=0)	
Définition d'une variable de contrôle à relier avec le widget Entry entTR défini ci-dessous. trvar = DoubleVar(root)	
entTR: Entry	Idem que entMB
entTR = Entry(lfdonnees, textvariable=trvar, font=my_font, justify="center", borderwidth=2, relief='raised')	
Position sur la fenêtre lfdonnees: 3 ^{ème} ligne, 1 ^{ère} colonne (row=2, column=1) entTR.grid(row=2, column=1)	
lbTE: Label	Idem que lbMB
lbTE = Label(lfdonnees, text="Taux Escompte:", font=my_font, width=20, anchor='e', borderwidth=2, relief='sunken')	
Position sur la fenêtre lfdonnees: 5 ^{ème} ligne, 1 ^{ère} colonne (row=4, column=0) lbTE.grid(row=4, column=0)	
Définition d'une variable de contrôle à relier avec le widget Entry entTE défini ci-dessous. tevar = DoubleVar(root)	
entTE: Entry	Idem que entMB
entTE = Entry(lfdonnees, textvariable=tevar, font=my_font, justify='center', borderwidth=2, relief='raised')	
Position sur la fenêtre lfdonnees: 5 ^{ème} ligne, 2 ^{ème} colonne (row=4, column=1) entTE.grid(row=4, column=1)	
lbTTVA: Label	Idem que lbMB
lbTTVA = Label(lfdonnees, text="Taux de TVA:", font=my_font, width=20, anchor='e', borderwidth=2, relief="sunken")	
Position sur la fenêtre lfdonnees: 7 ^{ème} ligne, 1 ^{ère} colonne (row=6, column=0)	

```
lbTTVA.grid(row=6, column=0)
```

Définition d'une variable de contrôle à relier avec le widget Entry entTTVA défini ci-dessous.

```
ttvavar = DoubleVar(root)
```

entTTVA:

Entry

Idem que entMB

```
entTTVA = Entry(lfdonnees, textvariable=ttvavar, font=my_font,
justify='center',borderwidth=2, relief='raised')
```

Position sur la fenêtre lfdonnees:

7ème ligne, 1ère colonne (row=6, column=1)

```
entTTVA.grid(row=6, column=1)
```

Définition de la fonction "calculer" qui s'exécute en réponse à l'événement click sur le bouton de commande "Calculer"

```
def calculer(mbvar, trvar, tevar, ttvavar, remisevar, ncvar, escomptevar,
nfvar, taxevar, napvar):
    remisevar.set(mbvar.get() * trvar.get())
    ncvar.set(mbvar.get() - remisevar.get())

    escomptevar.set(ncvar.get() * tevar.get())
    nfvar.set(ncvar.get() - escomptevar.get())

    taxevar.set(nfvar.get() * ttvavar.get())
    napvar.set(nfvar.get() + taxevar.get())
```

btnCalculer : Button	Fenetre parent	root : fenêtre racine
	text	"Calculer "
	font	("Helvetica", 32, 'bold italic')
	justify	'center'
	bg	"#00aa55"
	fg	"red",
	relief	"sunken"
	command	partial(calculer, mbvar, trvar, tevar, ttvavar, remisevar, ncvar, escomptevar, nfvar, taxevar, napvar)

```
btnCalculer = Button(root, text="Calculer",
font=("Helvetica", 32, 'bold italic'), justify='center',
bg="#00aa55", fg="red",
command=partial(calculer, mbvar, trvar, tevar, ttvavar,
remisevar, ncvar, escomptevar, nfvar,taxevar, napvar))
```

Position sur la fenêtre racine:

2ème ligne, 1ère colonne (row=1, column=0)

```
btnCalculer.grid(row=1, column=0)
```

lfresultats: LabelFrame	Fenetre parent	root
	text	"Données"
	font	=('Helvetica', 32, 'bold italic')
	fg	"#0000ff"
	bg	"#008888"
<pre>lfresultats = LabelFrame(root, text="Résultats", font=('Helvetica', 32, 'bold italic'), bg="#008888", fg="#0000ff")</pre>		
<p>Position sur la fenêtre racine: 3ème ligne, 1ère colonne (row=2, column=0) lfresultats.grid(row=2, column=0)</p>		

lbR: Label	Idem que lbMB
<pre>lbR = Label(lfresultats, text="remise:", font=my_font, width=20, anchor='e', borderwidth=2, relief='groove')</pre>	
<p>Position sur la fenêtre lfresultats: 1ère ligne, 1ère colonne (row=0, column=0) lbR.grid(row=0, column=0)</p>	
<p>Définition d'une variable de contrôle à relier avec le widget Entry entR défini ci-dessous. remisevar = DoubleVar(root)</p>	
entR: Entry	Idem que entMB
<pre>entR = Entry(lfresultats, textvariable=remisevar, bg="#0000ff", fg="#ff0000", font=my_font, justify='center', borderwidth=2, relief='raised')</pre>	
<p>Position sur la fenêtre lfresultats: 1ère ligne, 2ème colonne (row=0, column=1) entR.grid(row=0, column=1)</p>	
lbNC: Label	Défini de la même manière que le Label lbMB
<pre>lbNC = Label(lfresultats, text="Net Commercial:", font=my_font, width=20, anchor='e', borderwidth=2, relief='groove')</pre>	
<p>Position sur la fenêtre lfresultats: 2ème ligne, 1ère colonne (row=1, column=0) lbNC.grid(row=1, column=0)</p>	
<p>Définition d'une variable de contrôle à relier avec le widget Entry entNC défini ci-dessous. ncvar = DoubleVar(root)</p>	
entNC: Entry	Idem que entMB

```
entNC = Entry(lfresultats, textvariable=ncvar, bg="#0000ff", fg="#ff0000",
font=my_font, justify='center', borderwidth=2, relief='raised')
```

Position sur la fenêtre lfresultats:
2ème ligne, 2ème colonne (row=1, column=1)
entNC.grid(row=1, column=1)

lbEsc: Label	Idem que lbMB
------------------------	---------------

```
lbEsc = Label(lfresultats, text="Escompte:", font=my_font, width=20,
anchor='e', borderwidth=2, relief='groove')
```

Position sur la fenêtre lfresultats:
3ème ligne, 1ère colonne (row=2, column=0)
lbEsc.grid(row=2, column=0)

Définition d'une variable de contrôle à relier avec le widget Entry entEsc défini ci-dessous.
escomptevar = DoubleVar(root)

entEsc: Entry	Idem que entMB
-------------------------	----------------

```
entEsc = Entry(lfresultats, textvariable=escomptevar, bg="#0000ff",
fg="#ff0000", font=my_font, justify='center', borderwidth=2, relief='raised')
```

Position sur la fenêtre lfresultats:
3ème ligne, 2ème colonne (row=2, column=1)
entEsc.grid(row=2, column=1)

lbNF: Label	Idem que lbMB
-----------------------	---------------

```
lbNF = Label(lfresultats, text="Net Financier:", font=my_font, width=20,
anchor='e', borderwidth=2, relief='groove')
```

Position sur la fenêtre lfresultats:
4ème ligne, 1ère colonne (row=3, column=0)
lbNF.grid(row=3, column=0)

Définition d'une variable de contrôle à relier avec le widget Entry entNF défini ci-dessous.
nfvar = DoubleVar(root)

entNF: Entry	Idem que entMB
------------------------	----------------

```
entNF = Entry(lfresultats, textvariable=nfvar, bg="#0000ff", fg="#ff0000",
font=my_font, justify='center', borderwidth=2, relief='raised')
```

Position sur la fenêtre lfresultats:
4ème ligne, 1ère colonne (row=3, column=1)
entNF.grid(row=3, column=1)

lbTaxe:	Idem que lbMB
---------	---------------

Label	
<pre>lbTaxe = Label(lfresultats, text="Taxe:", font=my_font, width=20, anchor='e', borderwidth=2, relief='groove')</pre>	
<p>Position sur la fenêtre lfresultats: 5ème ligne, 1ère colonne (row=4, column=0) lbTaxe.grid(row=4, column=0)</p>	
<p>Définition d'une variable de contrôle à relier avec le widget Entry entTaxe défini ci-dessous. taxevar = DoubleVar(root)</p>	
entTaxe: Entry	Idem que entMB
<pre>entTaxe = Entry(lfresultats, textvariable=taxevar, bg="#0000ff", fg="#ff0000", font=my_font, justify='center', borderwidth=2, relief='raised')</pre>	
<p>Position sur la fenêtre lfresultats: 5ème ligne, 2ème colonne (row=4, column=1) entTaxe.grid(row=4, column=1)</p>	
lbNAP: Label	Idem que lbMB
<pre>lbNAP = Label(lfresultats, text="Net A Payer:", bg="#ff0000", fg="#00ff00", font=my_font, width=20, anchor='e', borderwidth=2, relief='solid')</pre>	
<p>Position sur la fenêtre lfresultats: 6ème ligne, 1ère colonne (row=5, column=0) lbNAP.grid(row=5, column=0)</p>	
<p>Définition d'une variable de contrôle à relier avec le widget Entry entNF défini ci-dessous. nfvar = DoubleVar(root)</p>	
entNAP: Entry	Idem que entMB
<pre>entNAP = Entry(lfresultats, textvariable=napvar, bg="#ff0000", fg="#00ff00", font=my_font, justify="center", borderwidth=2, relief='solid')</pre>	
<p>Position sur la fenêtre lfresultats: 6ème ligne, 1ère colonne (row=5, column=1) entNAP.grid(row=5, column=1)</p>	

Code complet source de l'application

```
from tkinter import *
from tkinter import font
from functools import partial

def calculer(mbvar, trvar, tevar, ttvavar, remisevar, ncvar, escomptevar,
nfvar, taxevar, napvar):
    remisevar.set(mbvar.get() * trvar.get())
    ncvar.set(mbvar.get() - remisevar.get())

    escomptevar.set(ncvar.get() * tevar.get())
    nfvar.set(ncvar.get() - escomptevar.get())

    taxevar.set(nfvar.get() * ttvavar.get())
    napvar.set(nfvar.get() + taxevar.get())

root = Tk()
root.title("Facturation")

my_font = font.Font(root, ("courier new", 16, "bold italic"))

# Variables de contrôle des Données
mbvar = DoubleVar(root)
trvar = DoubleVar(root)
tevar = DoubleVar(root)
ttvavar = DoubleVar(root)

# Variables de contrôle des Résultats
remisevar = DoubleVar(root)
ncvar = DoubleVar(root)
escomptevar = DoubleVar(root)
nfvar = DoubleVar(root)
taxevar = DoubleVar(root)
napvar = DoubleVar(root)

lfdonnees = LabelFrame(root, text="Données", font=("Helvetica", 32),
fg="blue", bg="#00ff00")

lbMB = Label(lfdonnees, text="Montant Brut:", font=my_font, width=20,
anchor="e", borderwidth=2, relief='sunken')

entMB = Entry(lfdonnees, textvariable=mbvar, font=my_font,
justify="center",borderwidth=2, relief='sunken')

lbMB.grid(row=0, column=0)
entMB.grid(row=0, column=1)

lbTR = Label(lfdonnees, text="Taux Remise:", font=my_font, width=20,
anchor="e", borderwidth=2, relief='sunken')

entTR = Entry(lfdonnees, textvariable=trvar, font=my_font,
justify="center", borderwidth=2, relief='raised')
lbTR.grid(row=2, column=0)
entTR.grid(row=2, column=1)
```



```

lbTE = Label(lfdonnees, text="Taux Escompte:", font=my_font, width=20,
anchor='e', borderwidth=2, relief='sunken')

entTE = Entry(lfdonnees, textvariable=tevar, font=my_font,
justify='center', borderwidth=2, relief='raised')

lbTE.grid(row=4, column=0)
entTE.grid(row=4, column=1)

lbTTVA = Label(lfdonnees, text="Taux de TVA:", font=my_font, width=20,
anchor='e', borderwidth=2, relief="sunken")
entTTVA = Entry(lfdonnees, textvariable=ttvavar, font=my_font,
justify='center', borderwidth=2, relief='raised')
lbTTVA.grid(row=6, column=0)
entTTVA.grid(row=6, column=1)

btnCalculer = Button(root, text="Calculer", font=("Helvetica", 32, 'bold
italic'), justify='center', bg="#00aa55", fg="red",
command=partial(calculer, mbvar, trvar, tevar,
ttvavar, remisevar, ncvar, escomptevar, nfvar,
taxevar, napvar))

lfresultats = LabelFrame(root, text="Résultats", font=('Helvetica', 32,
'bold italic'), bg="#008888", fg="#0000ff")

lbR = Label(lfresultats, text="remise:", font=my_font, width=20,
anchor='e', borderwidth=2, relief='groove')

entR = Entry(lfresultats, textvariable=remisevar, bg="#0000ff",
fg="#ff0000", font=my_font, justify='center', borderwidth=2,
relief='raised')

lbR.grid(row=0, column=0)
entR.grid(row=0, column=1)

lbNC = Label(lfresultats, text="Net Commercial:", font=my_font, width=20,
anchor='e', borderwidth=2, relief='groove')

entNC = Entry(lfresultats, textvariable=ncvar, bg="#0000ff",
fg="#ff0000", font=my_font, justify='center', borderwidth=2,
relief='raised')

lbNC.grid(row=1, column=0)
entNC.grid(row=1, column=1)

lbEsc = Label(lfresultats, text="Escompte:", font=my_font, width=20,
anchor='e', borderwidth=2, relief='groove')
entEsc = Entry(lfresultats, textvariable=escomptevar, bg="#0000ff",
fg="#ff0000", font=my_font, justify='center', borderwidth=2,
relief='raised')
lbEsc.grid(row=2, column=0)
entEsc.grid(row=2, column=1)

lbNF = Label(lfresultats, text="Net Financier:", font=my_font, width=20,
anchor='e', borderwidth=2, relief='groove')
entNF = Entry(lfresultats, textvariable=nfvar, bg="#0000ff",
fg="#ff0000", font=my_font, justify='center', borderwidth=2,
relief='raised')
lbNF.grid(row=3, column=0)

```

```

entNF.grid(row=3, column=1)

lbTaxe = Label(lfresultats, text="Taxe:", font=my_font, width=20,
anchor='e', borderwidth=2, relief='groove')
entTaxe = Entry(lfresultats, textvariable=taxevar, bg="#0000ff",
fg="#ff0000", font=my_font, justify='center', borderwidth=2,
relief='raised')
lbTaxe.grid(row=4, column=0)
entTaxe.grid(row=4, column=1)

lbNAP = Label(lfresultats, text="Net A Payer:", bg="#ff0000",
fg="#00ff00", font=my_font, width=20, anchor='e', borderwidth=2,
relief='solid')
entNAP = Entry(lfresultats, textvariable=napvar, bg="#ff0000",
fg="#00ff00", font=my_font, justify="center", borderwidth=2,
relief='solid')
lbNAP.grid(row=5, column=0)
entNAP.grid(row=5, column=1)

lfdonnees.grid(row=0, column=0)
btnCalculer.grid(row=1, column=0)
lfresultats.grid(row=2, column=0)

# root.geometry("800x800+100+100")

root.mainloop()

```

Annexe1: Variables de contrôle reliés aux widgets

Une variable de contrôle de tkinter est un **objet spécial** qui se comporte comme une variable ordinaire de Python en ce sens que c'est un conteneur pour une valeur comme un nombre ou une chaîne de caractères.

Une même variable de contrôle peut être reliée à plusieurs widgets.

Pour attribuer une valeur à une variable de contrôle, on utilise la méthode `set` (`<widget>.set(<valeur>)`), et pour récupérer sa valeur on utilise sa méthode `get` (`<valeur>=<widget>.get()`).

Lorsqu'on attribue une valeur à une variable de contrôle avec la méthode `set` tous les widgets qui lui sont reliés sont automatiquement mis à jour sur l'écran.

Tkinter utilise des variables de contrôles pour réaliser de nombreuses fonctionnalités, par exemple:

- Les boîtes à cocher, *Checkbuttons*, utilisent une variable de contrôle pour mémoriser leur état courant (on ou off, 0 ou 1, True ou False).
- Un groupe de boutons radio, *radiobuttons*, partage une variable de contrôle qui peut être utilisée pour connaître le bouton qui est actuellement coché. Quand l'utilisateur clique sur l'un des boutons d'un groupe, le mécanisme qui assure qu'un seul des boutons est coché à un moment donné s'appuie sur cette variable de contrôle partagée.
- Les variables de contrôles contiennent des textes pour de nombreuses applications. Normalement, le texte affiché dans un widget de saisie, *Entry*, est relié à une variable de contrôle. Il est aussi possible d'utiliser une telle variable de contrôle pour gérer les textes des étiquettes des cases à cocher, des boutons radio et le contenu des widgets étiquettes, *Label*.

Par exemple, vous pourriez relier un widget `Entry` (boîte de saisie) à un widget `Label` (étiquette) de telle sorte que quand l'utilisateur modifie le texte du premier, le second soit automatiquement mis à jour pour montrer le même texte.

Pour obtenir une variable de contrôle, utilisez l'un de ces constructeurs, en fonction du type de valeur que vous souhaitez mémoriser :

```
v = DoubleVar() # Mémorise un flottant; sa valeur par défaut est 0.0
v = IntVar()    # Mémorise un entier; sa valeur par défaut est 0
v = StringVar() # Mémorise une string; sa valeur par défaut est ""
v = BooleanVar() # Mémorise une booléenne; sa valeur par défaut est False
```

Toutes les variables de contrôle disposent de ces méthodes :

<code>get()</code>	Retourne la valeur courante de la variable de contrôle.
<code>set(valeur)</code>	Modifie la valeur courante de la variable de contrôle. Si les options d'un ou plusieurs widgets sont reliées à cette variable, ces widgets seront automatiquement mis à jour quand la boucle principale sera à nouveau en attente;

Comment relier une variable de contrôle à un widget.

Button	<p>Vous pouvez renseigner son option textvariable avec une <i>StringVar</i>. À chaque fois que cette variable est modifiée, le texte du bouton sera mis à jour pour afficher la nouvelle valeur. Il n'est pas nécessaire d'utiliser cela sauf si le texte du bouton doit changer au cours du temps: utilisez l'option text si l'étiquette du bouton ne change pas.</p>
Checkbutton	<p>Normalement, vous relierez son option variable à une <i>IntVar</i>, et cette variable aura la valeur 1 lorsque la case est cochée, la valeur 0 lorsqu'elle ne l'est pas. Cependant, vous pouvez choisir d'autres valeurs pour ces deux états en utilisant les options onvalue et offvalue.</p> <p>Vous pouvez même utiliser une <i>StringVar</i> comme variable de la case à cocher en fournissant des chaînes de caractères aux options onvalue et offvalue. Voici un exemple:</p> <pre>spamVar = StringVar() spamCB = Checkbutton(root, text='Spam?', variable=spamVar, onvalue='oui', offvalue='non')</pre> <p>Si cette case est cochée, <i>spamVar.get()</i> retournera 'oui'; Si elle ne l'est pas, la même instruction renverra 'non'. De plus, votre programme peut cocher la case en appelant <i>spamVar.set('oui')</i>.</p> <p>Vous pouvez encore indiquer une <i>StringVar</i> pour son option textvariable. Il vous suffit alors d'utiliser sa méthode <i>set()</i> pour actualiser l'étiquette de cette case à cocher.</p>
Entry	<p>Positionnez son option textvariable avec une <i>StringVar</i>. Utilisez alors sa méthode <i>get()</i> pour récupérer le texte actuellement affiché dans la boîte de saisie. Vous pouvez aussi modifier ce texte en utilisant sa méthode <i>.set()</i>.</p>
Label	<p>Positionnez son option textvariable avec une <i>StringVar</i>. En appelant sa méthode <i>set()</i>, vous pourrez modifier le texte affiché sur cette étiquette. Il n'est pas nécessaire d'utiliser cela; si vous ne prévoyez pas de changer son texte, utilisez plutôt son option text pour renseigner son contenu.</p>
Menubutton	<p>Si vous souhaitez être capable de modifier le texte affiché sur un bouton de menu, positionnez son option textvariable avec une <i>StringVar</i> et utilisez sa méthode <i>set()</i> pour modifier le texte affiché.</p>
Radiobutton	<p>Son option variable doit être réglée avec une variable de contrôle, soit une <i>IntVar</i> soit une <i>StringVar</i>. Tous les boutons radios qui forment un groupe doivent alors partager cette variable de contrôle.</p> <p>Régler l'option value de chaque bouton radio du groupe à une valeur différente. Lorsque l'utilisateur coche un bouton, la variable mémorisera la valeur de l'option value de ce bouton et tous les autres boutons seront décochés.</p> <p>Vous pourriez vous demander dans quel état se trouve un groupe de boutons radio lorsque leur variable de contrôle n'a pas encore reçu de valeur et que l'utilisateur n'a pas encore coché l'un des boutons ? Chaque variable de contrôle possède une valeur par défaut: 0 pour une <i>IntVar</i>, 0.0 pour une <i>DoubleVar</i>, et '' pour une <i>StringVar</i>.</p>

	<p>Si l'un des boutons radio a cette valeur, il est coché initialement. Si aucun d'eux n'a cette valeur, aucun n'est coché.</p> <p>Si vous souhaitez modifier l'étiquette d'un bouton radio pendant l'exécution du programme, régler son option <i>textvariable</i> avec une <i>StringVar</i>. Vous serez alors en mesure de la modifier en utilisant la méthode <i>set()</i> de cette variable de contrôle.</p>
<p><i>Scale</i></p>	<p>Pour un widget «curseur», <i>Scale</i>, positionnez son option variable avec une variable de contrôle du type voulu et réglez ses options <i>from_</i> et <i>to</i> aux valeurs limites qui apparaissent à chaque extrémité du widget.</p> <p>Par exemple, vous pourriez utiliser une <i>IntVar</i> en combinaison avec <i>from_=0</i> et <i>to=100</i>. Alors, à chaque fois que l'utilisateur modifie la position du curseur, la variable de contrôle est mise à jour avec la valeur sélectionnée de l'intervalle <i>[0; 100]</i>.</p> <p>Votre programme peut aussi déplacer le curseur du widget en utilisant la méthode <i>set()</i> de la variable de contrôle. Dans l'exemple précédent, l'instruction <i>v.set(75)</i> déplacer le curseur au 3/4 de sa barre.</p> <p>Pour utiliser un widget <i>Scale</i> avec des valeur flottantes, utilisez une <i>DoubleVar</i>.</p> <p>Vous pouvez utiliser une <i>StringVar</i> comme variable de contrôle d'un widget <i>Scale</i>. Il sera tout de même nécessaire de préciser des valeurs numériques pour les options <i>from_</i> et <i>to</i>, Mais les valeurs numériques du widget seront converties en une chaîne de caractères pour être mémorisées dans la <i>StringVar</i>. Utilisez l'option digit du widget pour contrôler la précision avec laquelle cette conversion est réalisée.</p>

Annexe2 : Valeurs possible de relief et leur apparence

