

```

1 ;*****
2 ;CalculatriceSTI2D (version finale) Auteur: Philippe HOARAU
3 ;
4 ;Le code suivant permet de gérer les différentes fonctions de saisies et de calculs utilisées sur l'onglet "Calculatrice" de la
5 ;"CalculletteSTI2D".
6 ;=====
7 ;Déclaration des variables
8 ;=====
9 Dim $Operande1 ;Premier opérande
10 Dim $Operande2 ;Deuxième opérande
11 Dim $Nv1Saisie ;Drapeau signalant une nouvelle saisie (Vrai / Faux)
12 Dim $$saisieOperande1 ;Drapeau signalant la saisie du premier ou du deuxième opérande (Vrai=>Opérande1, Faux=>Opérande2)
13 Dim $OperEnAttente ;Drapeau signalant une opération en attente d'exécution (Vrai / Faux)
14 Dim $VirguleSaisie ;Drapeau signalant qu'une virgule a déjà été saisie (Vrai / Faux)
15 ;=====
16 ;Initialisation des variables
17
18 $Operande1="" ;Initialisation de l'opérande 1
19 Initialiser() ;Appel de la fonction d'initialisation des autres variables (voir fin du script).
20 ; L'initialisation des autres variables (sauf $Operande1) intervenant également à chaque appui sur les touches
21 ; [=] et [CE], une fonction Initialiser() a été créée en fin de programme.
22 ;=====
23 ;=====
24 ;Gestion des touches numériques [0]...[9]
25
26 Func BtnCal0Click()
27 AffichTchNum("0") ;La fonction "AffichTchNum" est définie en fin de programme.
28 EndFunc
29 Func BtnCal1Click()
30 AffichTchNum("1")
31 EndFunc
32 Func BtnCal2Click()
33 AffichTchNum("2")
34 EndFunc
35 Func BtnCal3Click()
36 AffichTchNum("3")
37 EndFunc
38 Func BtnCal4Click()
39 AffichTchNum("4")
40 EndFunc
41 Func BtnCal5Click()
42 AffichTchNum("5")
43 EndFunc
44 Func BtnCal6Click()
45 AffichTchNum("6")
46 EndFunc
47 Func BtnCal7Click()
48 AffichTchNum("7")
49 EndFunc
50 Func BtnCal8Click()
51 AffichTchNum("8")
52 EndFunc
53 Func BtnCal9Click()
54 AffichTchNum("9")
55 EndFunc
56 ;=====
57 ;Gestion du point décimal [.]
58
59 Func BtnCalVirguleClick()
60 if Not $VirguleSaisie then ;On vérifie si la virgule a déjà été saisie.
61 AffichTchNum(".") ;Indicateur de saisie de la virgule pour
62 $VirguleSaisie=True ;interdire la saisie d'une deuxième
63 EndIf
64 EndFunc
65 ;=====
66 ;Gestion de la touche [-] (Effacement du dernier caractère saisi)
67
68 Func BtnCalBackClick()
69 If GUICtrlread($Afficheur)<> "0" then ;On vérifie si la valeur affichée est
différente de 0.
70 if StringRight(GUICtrlread($Afficheur),1)=". " Then ;Si le dernier caractère affiché est la virgule
71 $VirguleSaisie=False ;On signale qu'aucune virgule n'a été saisie
(avant de l'effacer à l'instruction suivante).
72 EndIf
73 GUICtrlSetData($Afficheur,StringTrimRight(GUICtrlread($Afficheur),1)) ;Suppression du caractère de droite de la
valeur affichée.
74 if $$saisieOperande1=True Then ;On vérifie si la saisie en cours concerne le
premier ou le deuxième opérande.
75 $Operande1=GUICtrlread($Afficheur) ;Mise à jour du premier ou deuxième opérande
avec la nouvelle valeur affichée.
76 Else
77 $Operande2=GUICtrlread($Afficheur)
78 EndIf
79 if GUICtrlread($Afficheur)= " " Then ;Si tous les caractères ont été supprimés on
affiche "0" et on autorise une nouvelle saisie.
80 GUICtrlSetData($Afficheur,"0")
81 $Nv1Saisie=True
82 EndIf
83 EndIf
84 EndFunc
85 ;=====
86 ;Gestion de la touche [CE] (Reinitialisation de la calculatrice)
87
88 Func BtnCalCEClick()
89 $Operande1="" ;Initialisation de l'opérande 1
90 Initialiser() ;Appel de la fonction d'initialisation des
variables (voir fin du script).
91 GUICtrlSetData($Afficheur,"0") ;Affichage de la valeur "0".
92 EndFunc
93 ;=====
94 ;Gestion des touches opérations [+],[-],[x],[/]
95 ;
96 ;La fonction "Calculer" est définie en fin de programme.
97 ;Pour chaque opération on appelle la fonction "Calculer" et faisant passer comme paramètres: l'opérande 1, L'opération et l'opérande 2
98

```

```

99 Func BtnCalPlusClick()
100     Calculer($Operande1,"+", $Operande2)
101 EndFunc
102
103 Func BtnCalMoinsClick()
104     Calculer($Operande1,"-", $Operande2)
105 EndFunc
106
107 Func BtnCalMultiClick()
108     Calculer($Operande1,"*", $Operande2)
109 EndFunc
110
111 Func BtnCalDivClick()
112     Calculer($Operande1,"/", $Operande2)
113 EndFunc
114 ;=====
115 ;Gestion de la touche [=]
116 ;
117 ;L'appui sur la touche [=] n'est pris en compte que si une opération est en attente.
118 ;On applique alors cette opération entre l'opérande 1 mémorisé et la valeur affichée.
119 ;On initialise ensuite la calculatrice pour une nouvelle saisie.
120
121 Func BtnCalEgalClick()
122     If $OperEnAttente <> "" Then
123         $Operande2=GUICtrlread($Afficheur)
124         Calculer($Operande1,$OperEnAttente,$Operande2)
125     EndIf
126     Initialiser()
127 EndFunc
128 ;=====
129 ;Gestion de la touche [%] (Pourcentage)
130 ;
131 ;L'appui sur la touche [%] n'est pris en compte que si une opération est en attente.
132 ;On applique alors cette opération entre l'opérande 1 et la valeur partielle de l'opérande1 correspondant au pourcentage affiché.
133 ;On initialise ensuite la calculatrice pour une nouvelle saisie.
134
135 Func BtnCall100Click()
136     If $OperEnAttente <> "" Then
137         $Operande2=($Operande1*GUICtrlread($Afficheur))/100 ;Calcul du % de $Operande1.
138         Calculer($Operande1,$OperEnAttente,$Operande2)
139     EndIf
140     Initialiser()
141 EndFunc
142 ;=====
143 ;Gestion de la touche [+/-] (Pourcentage)
144 ;
145 ;Si la valeur affichée est différente de "0" on l'inverse.
146 ;On met ensuite à jour l'opérande 1 ou l'opérande 2 suivant celui qui est en cours de saisie.
147
148 Func BtnInverseClick()
149     If GUICtrlread($Afficheur)<> 0 then
150         GUICtrlSetData($Afficheur,-GUICtrlread($Afficheur)) ;Inversion de la valeur affichée si elle est différente de 0.
151         if $SaisieOperande1=True Then
152             $Operande1=GUICtrlread($Afficheur)
153         Else
154             $Operande2=GUICtrlread($Afficheur)
155         EndIf
156     EndIf
157 EndFunc
158 ;=====
159 ;Action exécutée lorsque l'on ferme la calculatrice.
160
161 Func MaCalculletteClose()
162     exit
163 EndFunc
164
165 ;*****
166 ;
167 ;***** ZONE DE DEFINITION DES FONCTIONS *****
168 ;*****
169 ;Fonction "AffichTchNum"
170 ;
171 ;Le paramètre attendu par cette fonction est un chiffre (0,...9) ou le point décimal (.).
172 ;S'il s'agit d'une nouvelle saisie on affiche "0." ou le chiffre passé en paramètre.
173 ;S'il d'une saisie en cours on complète la valeur affichée en écrivant le nouveau chiffre (ou le point décimal) à droite de la valeur
174 ;affichée.
175 ;On met ensuite à jour l'opérande 1 ou l'opérande 2 suivant celui qui est en cours de saisie.
176
177 Func AffichTchNum($ValTch)
178     if $NvlSaisie=True Then
179         If $ValTch = "." Then
180             GUICtrlSetData($Afficheur,"0.")
181         Else
182             GUICtrlSetData($Afficheur,$ValTch)
183         EndIf
184     Else
185         $NvlSaisie=False ;Drapeau de nouvelle saisie à FAUX pour que la prochaine saisie
186         se fasse à droite de la valeur affichée.
187     EndIf
188     GUICtrlSetData($Afficheur,GUICtrlread($Afficheur) & $ValTch)
189 EndFunc
190
191 if $SaisieOperande1=True Then
192     $Operande1=GUICtrlread($Afficheur)
193 Else
194     $Operande2=GUICtrlread($Afficheur)
195 EndIf
196 EndFunc
197 ;*****
198 ;*****
199 ;Fonction "Calculer"
200 ;Affiche le chiffre correspondant à "$ValTch" si la valeur courante affichée = 0
201 ;sinon la valeur "$ValTch" est affichée à droite de la valeur courante affichée
202
203 Func Calculer($Op1,$Oper,$Op2)
204     dim $Resultat
205     dim $Formule

```

```

203     If $Op2<>" " then ;L'opération n'est effectuée que si l'opérande 1 et l'opérande 2 sont
        saisis
204     $Formule=$Op1 & " " & $OperEnAttente & " " & $Op2 ;Elaboration de la formule à calculer
205     $Resultat=Execute($Formule) ;Exécution de la formule
206     GUICtrlSetData($Afficheur,$Resultat) ;Affichage du resultat
207     $Operande1=$Resultat ;Transfert de la valeur affichée dans $Operande1 (pour pouvoir
        enchaîner les opérations suivantes)
208     $Operande2="" ;Initialisation de l'opérande 2
209     $OperEnAttente=$Oper
210     Else
211     $SaisieOperande1=false ;Validation de la saisie de l'opérande 2.
212     $OperEnAttente=$Oper ;Si l'opérande 2 n'est pas encore saisi on met l'opération en attente.
213     EndIf
214     $NvlSaisie=True ;Validation de la saisie d'un nouveau nombre
215     $VirguleSaisie=False ;Validation de la saisie d'une virgule
216 EndFunc
217 ;*****
218
219 ;*****
220 ;Fonction "Initialiser"
221 ;Initialisation des variables au début du programme et ensuite à chaque fois que
222 ;la touche [=] ou [CE] est appuyée
223
224 func Initialiser()
225     $Operande2="" ;Initialisation de l'opérande 2
226     $NvlSaisie=True ;Initialisation de l'indicateur de nouvelle saisie
227     $SaisieOperande1=true ;Initialisation de l'indicateur de saisie du premier opérande
228     $OperEnAttente="" ;Initialisation de l'opération en attente d'exécution
229     $VirguleSaisie=False ;Initialisation de la variable indiquant si une virgule a été saisie
        ou non.
230 EndFunc
231 ;=====

```