

TUTO PLUG-INS

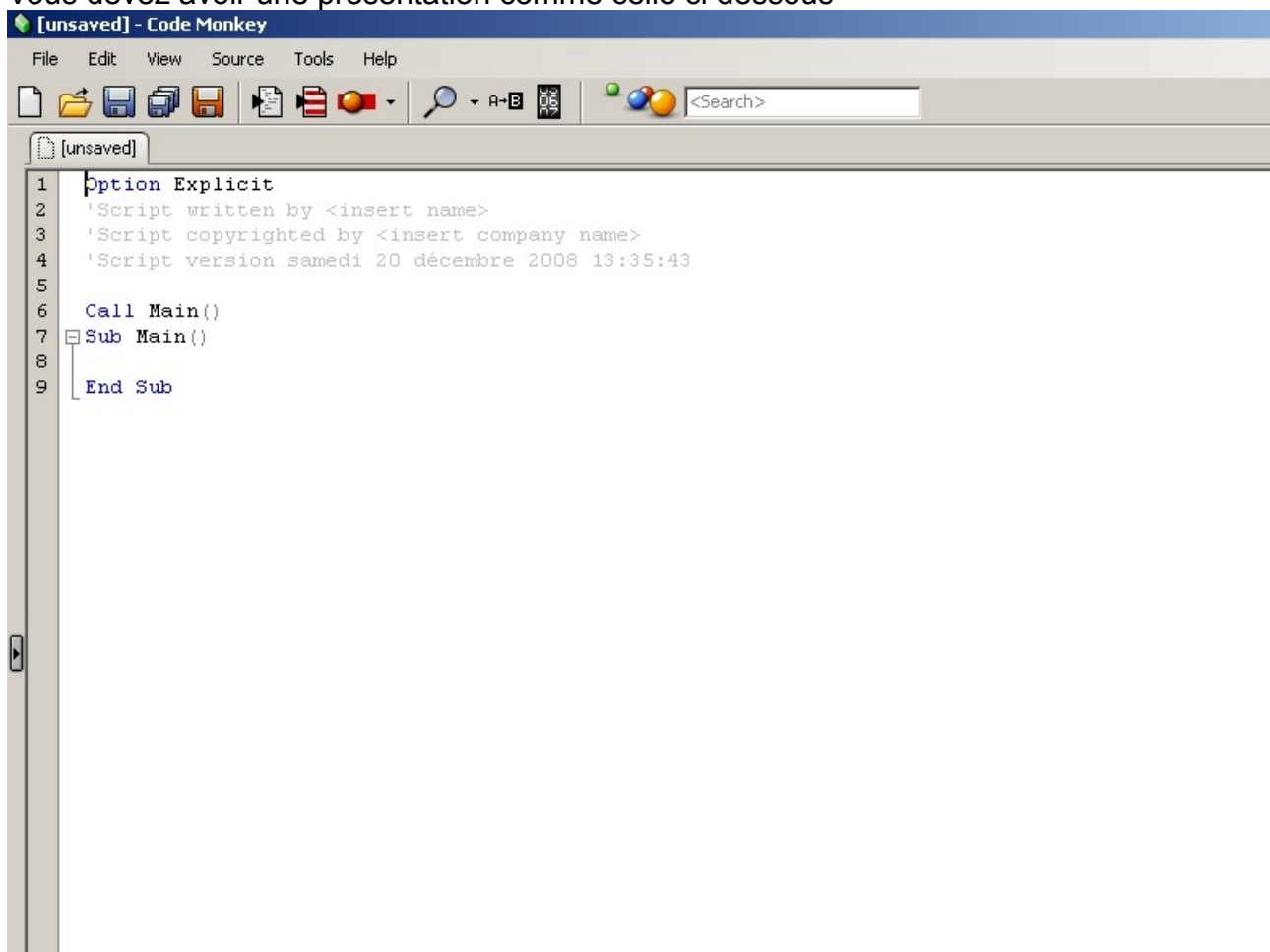
par Raphaël

Ce tuto à pour but d'expliquer comment faire un module pour RHINO 4 en VBA avec l'outil Monkey.

Exercice: plug-in qui compte le nombre d'objet afficher dans le document

- étape1: il faut télécharger Monkey et l'installer [ici](#)
- étape2: lancer Rhino puis lancer Monkey (menu Monkey->Monkey Editor)

Vous devez avoir une présentation comme celle ci dessous



The screenshot shows the Code Monkey editor window. The title bar reads "[unsaved] - Code Monkey". The menu bar includes File, Edit, View, Source, Tools, and Help. The toolbar contains icons for file operations (New, Open, Save, Print, Copy, Paste, Undo, Redo), a search icon, and a search input field with the placeholder text "<Search>". The main editing area shows a VBA script template with the following code:

```
1 Option Explicit
2 'Script written by <insert name>
3 'Script copyrighted by <insert company name>
4 'Script version samedi 20 décembre 2008 13:35:43
5
6 Call Main()
7 Sub Main()
8
9 End Sub
```

le programme de base ce fera entre sub Main () et le End sub

On va commencés par déclarer les variables.

```
Dim mesobjet,x,nombre
```

le préfixe **Dim** déclare toutes les variable situé après lui séparé par des virgule " , "

mesobjet,x,nombre Comme vous l'avez deviné ces mots sont des Variables.

On écrit une petite ligne de code

```
mesobjet = Rhino.AllObjects
```

Rhino. => Fonction qui commande RHINO.

AllObjects => Renvois les identifiants de tous les objets affichés dans le document.

Ici c'est une équation, la commande AllObjects va renvoyer les identifiants de tous les objets dans la variable (tableau) mesobjet.

Une variable de type tableau ou (Array) est une variable qui contient plusieurs données.

```
If isnull(mesobjet) Then
    rhino.print ("il n'y a pas d'objet visible!")
    Exit Sub
End If
```

La ça veut dire si la variable (mesobjet) est vide, afficher à l'écran (il n'y a pas d'objet visible!) puis quitter le programme.

Maintenant on va afficher le nombre d'objets qui sont présent dans notre chère variable (mesobjet) à l'aide d'une boucle.

```
For x=lbound (mesobjet) To ubound (mesobjet)
    rhino.print (x)
Next
```

Le mot clef lbound signifie le point le plus bas du tableau (0).

Le mot clef ubound signifie le point le plus haut du tableau (1,2,3,4....100000 etc...).

La variable x est la position courante du tableau (elle commence à 0 et se fini à ubound)

La fonction print affiche un texte ou le contenu d'une variable.

Dernière ligne de code

```
nombre= ubound (mesobjet)+ 1 'ici on ajout + 1 au niveau le plus haut de (mesobjet)
```

```
rhino.Print ("il y a "&nombre&" objet(s)") 'on affiche le résultat
```

Notre code doit maintenant être comme ci dessous.

```
Option Explicit
```

```
'Script written by <insert name>
```

```
'Script copyrighted by <insert company name>
```

```
'Script version samedi 20 décembre 2008 18:32:10
```

```
Call Main()
```

```
Sub Main()
```

```
#####
```

```
'Variable
```

```
Dim mesobjet,x,nombre
```

```
mesobjet = Rhino.AllObjects
```

```
If isnull(mesobjet) Then
```

```
    rhino.print ("il n'y à pas d'objet visible!")
```

```
    Exit Sub
```

```
End If
```

```
For x=lbound (mesobjet) To ubound (mesobjet)
```

```
    rhino.Print (x)
```

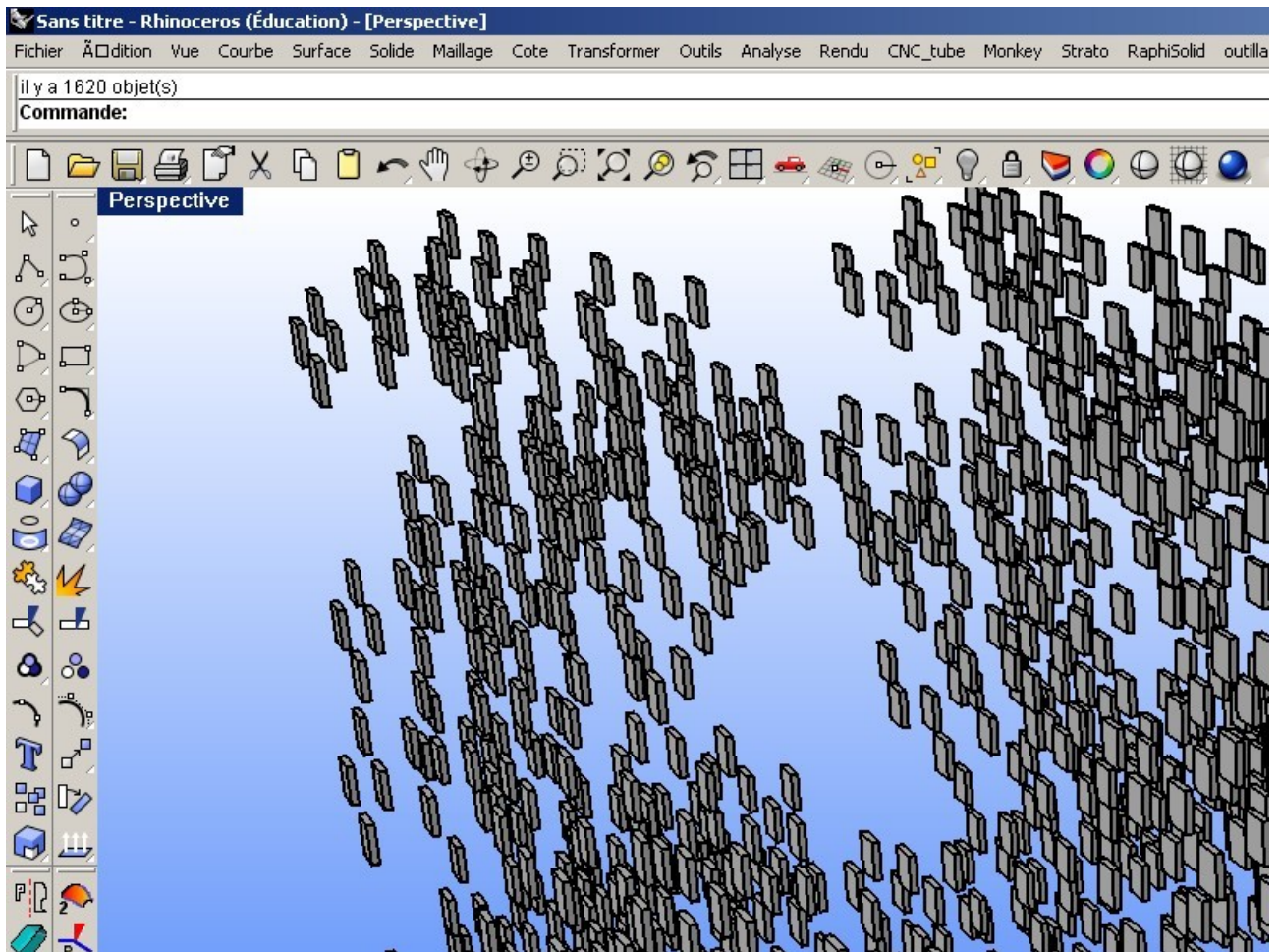
```
Next
```

```
nombre= ubound (mesobjet)+ 1
```

```
rhino.Print ("il y a "&nombre&" objet(s)")
```

```
End Sub
```

dernière parti teston le programme

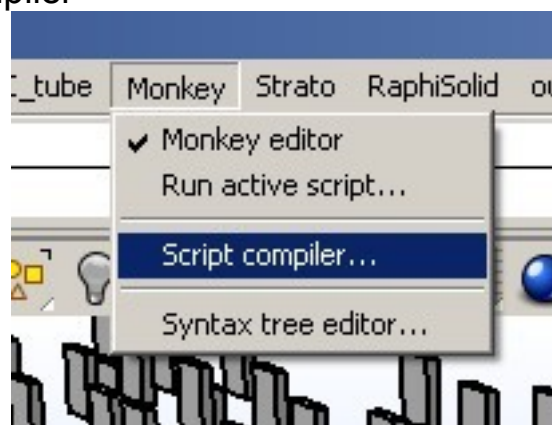


HOURRA ÇA MARCHE!!!!

Cette fois ci ces la dernière partie promis
pensez quand même à enregistrer.

Partie compilation

Lançons le script compiler



Bien maintenant il vous reste plus qu'à installer le plug-in sur Rhino et à l'utiliser si vous en sentez le besoin.

Je vous remercie d'avoir suivi ce cours tuto, si vous avez des soucis ou des questions, n'hésitez pas à me demander.