

Blender 2.8 - 19 – Programmation en Python

*Ces notes de cours sont des notes personnelles et le fruit d'un long travail !
Je partage ces notes avec plaisir et j'espère que cet outil pourra vous apporter une aide précieuse.
Si vous y remarquez une quelconque erreur, ce serait gentil de me partager vos remarques.*

C.Brison

Voici le contenu de ces notes :

1. Python dans Blender	1
1. Python, les bases	2

1. Python dans Blender

Blender est un programme Open source, écrit en Python .
Il est donc possible de créer des routines en Python dans Blender.

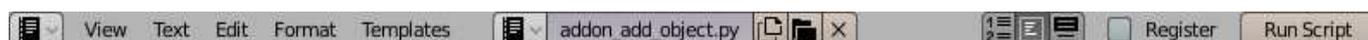
Il faut aller afficher une **fenêtre** de type « **Text Editor** »



Cocher les 3 icones en haut à droite

- Numéroté les lignes
- Retour à la ligne
- Mettre la syntaxe en surbrillance

+ clic sur **New** (des éléments en plus s'ajoutent dans la fenêtre)



Taper son programme en commençant par : **import bpy**

Une fois le programme écrit, cliquer ensuite sur « **Run Script** » pour exécuter celui-ci.

Astuce :

Comme aide, il est possible d'ouvrir la fenêtre de type « **Python Console** » (ce sont les coulisses de Blender)
On peut alors copier (Ctrl C) – coller (Ctrl V) des infos de cette fenêtre vers la fenêtre de Text Editor.

Pour info, super vidéo → <https://www.youtube.com/watch?v=FPfzGFYuOLU>

2. Python, les bases

	Code	Résultat
Affichage de texte Ecrit « Hello World »	<pre>Print(« Hello World »)</pre>	<i>Hello World</i>
Définir une variable texte Variable <i>oName</i> qui est égale à « Jacques » Variable <i>oName2</i> qui est égale à « Marie » Ecrit <i>oName</i> <i>Note : une chaîne de caractère est toujours entre guillemets.</i>	<pre>oName= « Jacques» oName2= « Marie » print(oName)</pre>	<i>Jacques</i>
Ajouter des caractères Ajoute « , tu es là ? » à la variable <i>oName</i> Ecrit <i>oName</i> , tu es là ?	<pre>oName= « Jacques» print(oName+ « , tu es là ? »)</pre>	<i>Jacques, tu es là ?</i>
Définir une variable nombre Variable <i>oInt</i> qui est égale à 5 Variable <i>oFloat</i> qui est égale à 5.5 Ecrit <i>oInt</i> (5) <i>Note : un nombre est défini sans guillemet, un nombre décimal est défini avec un point.</i>	<pre>oInt= 5 oFloat=5.5 print(oInt)</pre>	5
Faire des opérations entre nombres Variable <i>oInt1</i> qui est égale à 5 Variable <i>oInt2</i> qui est égale à 10 Ecrit le résultat de <i>oInt1</i> (5)+ <i>oInt2</i> (10) <i>Note : un nombre est défini sans guillemets, un nombre décimal est défini avec un point.</i>	<pre>oInt1= 5 oInt2=10 print(oInt1+oInt2)</pre>	15
Convertir un nombre en chaîne de caractères Variable <i>oName</i> qui est égale à « Paul » Variable <i>oInt</i> qui est égale à 10 Ecrit <i>oName</i> (Paul)+les caractères de <i>oInt</i> (10) <i>Note : une chaîne de caractères ne peut pas être additionnée à un nombre (→ ERREUR !) Il faut additionner des éléments du même type.</i>	<pre>oName= "Paul" oInt=10 print(oName+oInt) print(oName+str(oInt))</pre>	<i>ERREUR ! Paul10</i>
Convertir une chaîne de caractères en nombre Variable <i>oName</i> qui est égale à «10 » Variable <i>oInt</i> qui est égale à 10 Ecrit le résultat du nombre <i>oName</i> (10) + <i>oInt</i> (7) <i>Note : un nombre entre guillemets devient une chaîne de caractères. On ne peut additionner que des éléments du même type.</i>	<pre>oName= "10" oInt=17 print(oName+oInt) print(oName+str(oInt)) print(int(oName)+oInt)</pre>	<i>ERREUR ! 1010 17</i>

	Code	Résultat
<p>Définir une structure conditionnelle</p> <p>Variable <i>oName</i> qui est égale à « Pierre » Si <i>oName</i> est égal à « Pierre » Alors il faut écrire « Victoire » Sinon, dans tous les autres cas, Il faut écrire « Echec »</p> <p>Note : Mettre 2x= pour vérifier l'égalité et mettre une tabulation devant ce qui doit être fait si la condition est vérifiée.</p>	<pre>oName= "Pierre" if oName=="Pierre": print ("Victoire") else : print « Echec »</pre>	<p><i>Victoire</i></p>
<p>Ajouter une condition</p> <p>Variable <i>oName</i> qui est égale à « Pierre » Si <i>oName</i> est égal à « Pierre » Alors il faut écrire « Victoire » ou si <i>oName</i> == « Marie » Alors il faut écrire « Victoire encore » Sinon, dans tous les autres cas, Il faut écrire « Echec »</p>	<pre>oName= "Pierre" if oName=="Pierre": print ("Victoire") elif oName== « Marie » : print ("Victoire encore") else : print « Echec »</pre>	<p><i>Victoire encore</i></p>
<p>Invite de commande</p> <p>Invite l'utilisateur à entrer son age. (ici, l'utilisateur entre «42») Ecrit <i>oAge</i> (ce qui a été encodé par l'utilisateur)</p>	<pre>oAge=input(«Quel age as-tu ? ») print (oAge)</pre>	<p>Quel age as-tu ? <i>42 par exemple</i> 42</p>
<p>Structure conditionnelle de valeurs</p> <p>Invite l'utilisateur à entrer son age. (ici, l'utilisateur entre «18») Ecrit <i>oAge</i> (ce qui a été encodé par l'utilisateur)</p>	<pre>oAge=input(«Quel age as-tu ? ») if oAge>=18 : print ("Bravo tu es majeur !") elif oAge<18 : print ("Dommage tu es mineur")</pre>	<p>Quel age as-tu ? <i>Si 23 par exemple</i> <i>Bravo tu es majeur !</i> <i>Si 9 par exemple</i> <i>Dommage tu es mineur</i></p>
<p>La boucle « for »</p> <p>Répète l'action (imprimer <i>i</i>) suivant un nombre d'itérations demandé (de la place zéro à la place 10) Ecrit la valeur de <i>i</i> (<i>qui change à chaque tour</i>)</p> <p>En bref : Au premier tour, <i>i</i>=0 et il écrit la valeur de 0 ; au deuxième tour, <i>i</i>=1 et il écrit la valeur de 1 ; au troisième tour... au 10^{ème} tour, il la boucle s'arrête.</p> <p>Note : l'élément zéro est le premier élément d'une liste, 1 est le deuxième élément d'une liste.</p>	<pre>For i in range (0,10): print (i) On peut aussi taper *: For i in range (10): print (i) *avec une seule valeur entre parenthèses, on sous entant que le liste commence par zero.</pre>	<p>0 1 2 3 4 5 6 7 8 9</p>
<p>Les itérations</p> <p>Répète l'action (imprimer Lettre) tant que la liste le demande (passer en revue toutes les lettres de « Pierre ») Ecrit la valeur de <i>i</i></p>	<pre>oName= "Pierre" For Lettre in oName : print (Lettre)</pre>	<p>P I E R R E</p>
<p>Sélectionner un élément d'une liste</p> <p>Variable <i>oName</i> qui est égale à « Pierre » Ecrit l'élément de la liste placé à l'endroit zéro.</p> <p>Note une chaine de caractères n'est jamais qu'une liste de lettre)</p>	<pre>oName= "Pierre" print (oName[0])</pre>	<p>P</p>

	Code	Résultat
<p>Ecrire la longueur d'une chaîne de caractère Variable <code>oName</code> qui est égale à « Pierre » Ecrit le nombre d'éléments qu'il y a dans la liste « Pierre » (il y a 6 lettres au total dans « Pierre »)</p>	<pre>oName= "Pierre" print (len(oName))</pre>	6
<p>Boucle qui reprend chaque caractère (l'un après l'autre) d'une chaîne de caractère Variable <code>oName</code> qui est égale à « Pierre » Répète l'action (imprimer « i ») tant que la liste le demande (passer en revue toutes les lettres de la chaîne de caractère « Pierre »)</p>	<pre>oName= "Pierre" for i range (0, len(oName)): print (oName[i])</pre>	P I E R R E
<p>Idem, en ajoutant la place des lettres dans la chaîne de caractère Variable <code>oName</code> qui est égale à « Pierre » Répète l'action (imprimer « n° : i ») tant que la liste le demande (passer en revue toutes les lettres de la chaîne de caractère « Pierre »)</p>	<pre>oName= "Pierre" for i range (0, len(oName)): print (str(i) « : » + oName[i])</pre>	0= P 1= I 2= E 3= R 4= R 5= E
<p>La boucle « While » Répète l'action (imprimer i) tant qu'une condition est vérifiée. Ecrit « Bravo, vous avez quitté le programme » si l'utilisateur a entré « o » à la question posée (voulez-vous quitter le programme ?) Si la réponse est différente de o, écrit « Dommage ! »</p>	<pre>oInput = « » while oInput != « o » : oInput=input ("Voulez-vous quitter le programme o/n ?") : print (« Bravo, vous avez quitté le programme ! ») whilenot oInput== « o »: print (« Dommage »)</pre>	Si o Bravo, vous avez quitté le programme ! Si n Dommage
<p>Les listes (caractères) Définir la liste de textes « Pierre, Paul, Jacques »</p>	<pre>oName list = [« Pierre », « Paul, « Jacques »] Print (oName list)</pre>	« Pierre », « Paul, « Jacques »
<p>Reprendre les éléments tout au long de la liste Définir la liste de textes « Pierre, Paul, Jacques » Ecrire chaque élément de la liste l'un après l'autre jusqu'à la fin de celle-ci.</p>	<pre>oName list = [« Pierre », « Paul, « Jacques »] For item in oName list : Print (item)</pre>	Pierre Paul Jacques
<p>Les listes (nombres) Définir la liste de nombres 0, 2, 5, 8 Ecrire la liste</p>	<pre>oInt list = [0, 2, 5, 8] Print (oInt list)</pre>	0, 2, 5, 8
<p>Reprendre un élément d'une liste Définir la liste la liste « A, B, C » Ecrit l'élément zéro de la liste Note zéro est toujours le premier élément d'une liste</p>	<pre>oName list = [« A », « B », « C »] Print (oName list[0])</pre>	A
<p>Reprendre une lettre d'un élément d'une liste Définir la liste la liste « Pierre, Paul, Jacques » Ecrit, dans l'élément zéro de la liste(Pierre), la lettre 1 (i)</p>	<pre>oName list = [« Pierre », « Paul », « Jacques »] Print (oName list[0][1])</pre>	i
<p>Ajouter un élément à une liste (Append) Définir la liste la liste « Pierre, Paul, Jacques » Ajouter « Marie » à la liste <code>oName list</code> Ecrit la liste. Note : Append ajoute un élément à la fin de la liste de départ.</p>	<pre>oName list = [« Pierre », « Paul », « Jacques »] oName list.append (« Marie ») Print (oName list)</pre>	« Pierre », « Paul », « Jacques », « Marie »

	Code	Résultat
<p>Ajouter un élément à une liste (Insert) Définir la liste la liste « Pierre, Paul, Jacques » Ajouter « Marie » à la place 1 de la liste Ecrit la liste.</p> <p>Note : Insert ajoute un élément un certain endroit précis de la liste de départ.</p>	<pre>oName list = [« Pierre », « Paul », « Jacques »] oName list.insert (1,« Marie ») Print (oName list)</pre>	« Pierre », « Marie », « Paul », « Jacques »
<p>Retirer un élément à une liste Définir la liste la liste « Pierre, Paul, Jacques » Retirer « Pierre » de la liste Ecrit la liste.</p>	<pre>oName list = [« Pierre », « Paul », « Jacques »] oName list.remove (« Pierre ») Print (oName list)</pre>	« Paul », « Jacques »
<p>Retirer le dernier élément à une liste Définir la liste la liste « Pierre, Paul, Jacques » Retirer le dernier élément (Jacques) de la liste Ecrit la liste.</p>	<pre>oName list = [« Pierre », « Paul », « Jacques »] oName list.pop () Print (oName list)</pre>	« Pierre », « Paul »
<p>Retirer un élément précis à une liste Définir la liste la liste « Pierre, Paul, Jacques » Retirer l'élément 1 (Paul) de la liste Ecrit la liste.</p>	<pre>oName list = [« Pierre », « Paul », « Jacques »] oName list.pop (1) Print (oName list)</pre>	« Pierre », « Jacques »
<p>Inverser une liste Définir la liste la liste « Pierre, Paul, Jacques » Inverser tous les éléments de la liste Ecrit la liste.</p>	<pre>oName list = [« Pierre », « Paul », « Jacques »] oName list.reverse Print (oName list)</pre>	« Jacques », « Paul », « Pierre »
<p>Trier une liste (ordre alphabétique) Définir la liste la liste « Pierre, Paul, Jacques » Inverser tous les éléments de la liste Ecrit la liste.</p>	<pre>oName list = [« Pierre », « Paul », « Jacques »] oName list.sort() Print (oName list)</pre>	« Jacques », « Paul », « Pierre »
<p>Ecrire des commentaires dans le code Permet d'écrire un commentaire personnel que le code ne considèrera lors de son exécution</p>	# commentaire personnel	
<p>Ecrire des commentaires dans le code Permet d'écrire plusieurs lignes de commentaires personnels que le code ne considèrera lors de son exécution</p>	<pre>""" Plusieurs lignes de commentaires """</pre>	
<p>Vérifier si un élément est un nombre Inviter l'utilisateur à entrer un nombre dans la variable oInput Ecrire la variable oInput si c'est un nombre (TRUE) Note : il n'y a que deux possibilités (TRUE ou FALSE)</p>	<pre>oInput=Input (SVP, donnez un nombre) print (oInput.isdigit())</pre>	<p><i>Si 5 par exemple</i> 5</p> <p><i>Si Bernard par exemple</i> (il n'écrit rien)</p>

	Code	Résultat
<p>Les booléens</p> <p>Inviter l'utilisateur à entrer un nombre dans la variable oInput Ecrire la variable oBool TRUE si c'est un nombre et FALSE si ce n'est pas un nombre</p> <p>Note : Booléen est une variable qui a deux valeurs possibles : TRUE ou FALSE</p>	<pre>oInput=Input (SVP, donnez un nombre) oBool=oInput.isdigit() print (oBool)</pre>	<p><i>Si 5 par exemple</i> TRUE <i>Si Bernard par exemple</i> FALSE</p>
<p>Les booléens et les conditions</p> <p>Inviter l'utilisateur à entrer un nombre dans la variable oInput Vérifier si oInput est bien un nombre et mettre le résultat dans la variable oBool Si la variable oBool est TRUE, écrire « Continuez » Si la variable oBool est FALSE, écrire « Erreur »</p>	<pre>oInput=Input (SVP, donnez un nombre) oBool=oInput.isdigit() if oBool==TRUE: print ("Continuez") elif oBool===FALSE: print ("Erreur")</pre>	<p><i>Si 5 par exemple</i> Continuez <i>Si Bernard par exemple</i> Erreur</p>
<p>Exemple de petit programme</p> <p>Commentaires sur l'objectif et le déroulement du programme</p> <p>Inviter l'utilisateur à entrer un nombre dans la variable oInput</p> <p>Vérifier si l'entrée est bien un nombre</p> <p>Si la variable oBool est TRUE, Convertir oInput en nombre</p> <p>Ecrire la valeur de 0 (en texte) « x » la valeur de oInput (en texte) « = » la valeur de oInput (en texte) pour tous les éléments de la liste (1,11)</p> <p>Si la variable oBool est FALSE, écrire « Erreur »</p>	<pre>""" programme qui écrit les tables de multiplication suivant un nombre à entrer. Le programme doit vérifier si l'entrée est bien un nombre """ oInput=Input (SVP, donnez un nombre) oBool=oInput.isdigit() if oBool==TRUE: oInput=Int(oInput) for l in range (0,11): print (str) i +"x"(str) oInput + "=" + (str) oInput elif oBool===FALSE: print ("Erreur")</pre>	<p><i>Si 5 par exemple</i> 0 x 5 = 0 1 x 5 = 5 2 x 5 = 10 3 x 5 = 15 4 x 5 = 20 5 x 5 = 25 6 x 5 = 30 7 x 5 = 35 8 x 5 = 40 9 x 5 = 45 10 x 5 = 50</p> <p><i>Si Bernard par exemple</i> Erreur</p>