De Scratch à Python

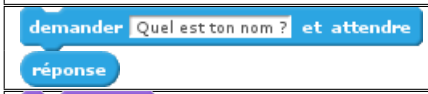
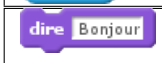
Opérations mathématiques et logiques

Scratch	Python	Remarques
	`x+y`	
	`x-y`	
	`x*y`	
	`x/y` `x//y`	Division dans $\mathbb{R}$ Quotient de la division euclidienne
	`x%y`	Reste de la division euclidienne
	`round(x)` `int(x)`	`round(2.3)`→`2.0` `int(2.3)`→`2`
	`abs(x)`	Distance à 0 du nombre
	`x<y` `x<=y`	 $x \leq y$
	`x==y` `x!=y`	Il faut deux "=" $x \neq y$
	`x>y` `x>=y`	 $x \leq y$
	`x and y`	
	`x or y`	
	`not x`	
	`import math` `from math import *`	Rajouter `math.` devant chaque commande Pas besoin de `math.`
	`sqrt(x)`	$\sqrt{x}$
	`floor(x)`	Arrondi par défaut
	`ceil(x)`	Arrondi par excès
	`sin(degrees(x))`	Python utilise des angles en radians
	`cos(degrees(x))`	`degrees(x)` convertit l'angle en degrés
	`tan(degrees(x))`	
	`degrees(asin(x))`	
	`degrees(acos(x))`	
	`degrees(atan(x))`	
	`log(x)`	Logarithme népérien
	`log10(x)`	Logarithme en base 10
	`e**x`	e^x
	`10**x` `x**y`	10^x x^y

Variables et listes

Scratch	Python	Remarques
	`x=expression`	Affecte la valeur de `expression` à `x`
	`x+=1`	Correspond à `x=x+1`
	`x-=1`	Correspond à `x=x-1`
	`liste=[]`	Création d'une liste vide
	`liste[0]`	Le premier élément de la liste
	`liste[i]`	L'élément `i` de la liste
	`liste[-1]`	Le dernier élément de la liste
	`liste+=[this]`	Rajoute un élément à la fin de la liste
	`liste=[this]+liste`	Rajoute un élément au début de la liste
	`liste=l1+l2`	`liste` contient les éléments de `l1` puis ceux de `l2`
	`liste[i]=this`	Remplace l'élément `i` de la liste
	`del liste[i]`	
	`len(liste)`	Le longueur de la liste vide est 0
	`thing in liste`	

Interactions avec l'utilisateur

Scratch	Python	Remarques
	`nom=input("Ton nom ?")`	La réponse est stockée dans `nom`
	`print("Bonjour")`	Les textes doivent être entre guillemets
	`print(nom)`	Affiche la valeur de la variable `nom`
	`print("Bonjour",nom)`	On peut afficher plusieurs éléments

Instructions conditionnelles et boucles

Attention, les indentations devant les instructions permettent de savoir dans quel bloc elles se trouvent. Il faut respecter ces indentations dans Python.

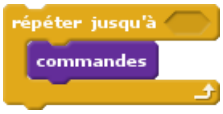
Scratch	Python	Remarques

	`if Condition:`
	`    commande1`
	`commande2`
		L'instruction `commande1` n'est exécuté que si la `Condition` est vérifiée.
		L'instruction `commande2` sera toujours exécutée.

	`if Condition:`
	`    commande1`
	`else:`
	`    commande2`
	`commande3`
		L'instruction `commande1` n'est exécuté que si la `Condition` est vérifiée.
		Sinon ce sera `commande2` qui sera exécutée.
		L'instruction `commande3` sera toujours exécutée.

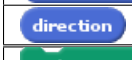
	`if Condition1:`
	`    commande1`
	`elif Condition2:`
	`    commande2`
	`else:`
	`    commande3`
		Cela permet de faire des études de cas.
		Le **`else`** n'est pas obligatoire.

	`for i in range(10):`
	`    commande1`
	`    commande2`
	`commande3`
		Les instructions `commande1` et `commande2` seront exécutées 10 fois avant que l'instruction `commande3` soit exécutée.
		La variable `i` peut être utilisée dans la boucle et prendra les valeurs 0, 1, ...jusqu'à 9.

	`while Condition:` `    commandes`	Dans Python, la boucle continue tant que `Condition` est vérifiée. Dans Scratch, la boucle s'arrête dès que `Condition` est vérifiée.
	`while True:` `    commandes`	La boucle ne s'arrête que si l'utilisateur interrompt l'exécution du programme.

Scratch	Python	Remarques

	`import turtle`
	
	`from turtle import *`
		Rajouter `turtle.` devant chaque commande
		Pas besoin de `turtle.`

	`forward(10)`	Avancer de 10 unités (pixels)
	`back(10)`	Reculer de 10 unités (pixels)
	`right(15)`	Tourner à droite de deg 15
	`left(15)`	Tourner à gauche de deg 15
	`goto(x,y)`	
	`setx(x)`	
	`sety(y)`	
	`setheading(0)`	Se tourner vers la droite
	`setheading(90)`	Se tourner vers la haut
	`setheading(180)`	Se tourner vers la gauche
	`setheading(270)`	Se tourner vers le bas
	`setheading(angle)`	S'orienter dans une direction donnée
	`position()`	Donne la position `(x,y)`
	`xcor()`	
	`ycor()`	
	`heading()`	
	`pendown()`	
	`pendup()`	
	`pendsize()`	
	`pencolor(couleur)`	`couleur` peut être `"red"`, `"blue"`...
	`pencolor((r,g,b))`	`r`, `g`, `b` sont entre 0 et 255
	`stamp()`	
	`clear()`	
	`hideturtle()`	
	`showturtle()`	