

LES DÉPLACEMENTS

1. Déplacements relatifs

Dans la catégorie **Mouvement**, intéressons-nous d'abord aux trois premiers blocs qui servent à avancer et à tourner :

avancer de 10 pas

tourner  de 15 degrés

tourner  de 15 degrés

Avancer

Lorsqu'on glisse à la souris le bloc **avancer de 10 pas** dans l'aire des scripts et que l'on clique dessus, le chat avance.

On peut modifier la valeur indiquée dans le bloc et cliquer dessus pour tester à nouveau. Le lutin avance plus ou moins loin selon la valeur indiquée dans le bloc. Ce nombre, appelé argument, peut aussi être négatif :

avancer de -50 pas fait reculer le lutin de 50 pixels.



En informatique, une image est constituée d'un ensemble de points que l'on appelle pixels. Cette unité provient de la contraction des mots anglais *picture* (image) et de *element*.

Ici, la scène a pour dimensions 480 pixels de longueur et 360 pixels de largeur (rapport 4/3). Ainsi, au départ, le lutin étant centré dans la scène, le bloc `avancer de 240 pas` lui fait rejoindre le bord droit.

Un glisser déposer à la souris permet de déplacer le lutin ailleurs sur la scène.

EXERCICE B1

Dans ce programme, avec le lutin `cat1`, que se passe-t-il quand on laisse la touche `espace` enfoncée ?

quand la touche `espace` est pressée

avancer de 10 pas

costume suivant



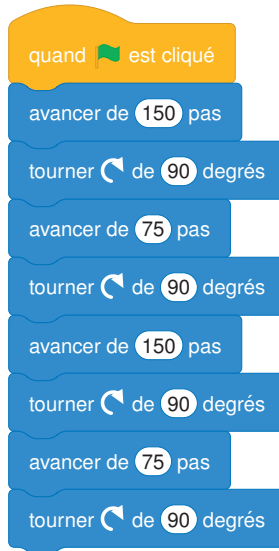
Tourner

Un lutin peut tourner dans le sens des aiguilles d'une montre ou dans le sens inverse.

L'unité utilisée est le degré. Ainsi, tourner de 180° fait faire un demi-tour au lutin. La valeur à entrer dans le bloc n'est pas limitée à 360° . En effet, tourner de $1\,000^\circ$ équivaut à tourner de 280° (deux tours de 360° non visibles et 280° car $1\,000 = 2 \times 360 + 280$).

EXERCICE B2

[Débogage] Ouvrir un nouveau fichier et assembler les blocs suivants. Pourquoi, lorsque l'on clique sur le drapeau vert, il ne se passe rien ?



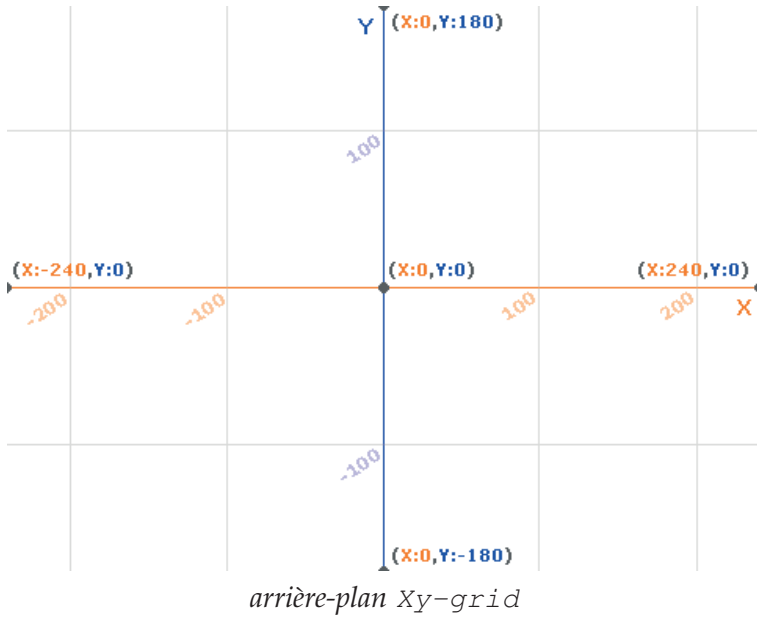
2. Déplacements absolus

Les déplacements vus précédemment (dits déplacement *relatifs*) dépendent de l'orientation du lutin au moment de l'exécution des blocs. Par exemple, si un lutin est orienté vers le haut, `avancer de 10 pas` le fait se déplacer vers le haut de dix pixels, alors que s'il est orienté vers la droite, ce même bloc le fait se déplacer vers la droite de dix pixels.

Les déplacements suivants (appelés déplacements *absolus*) n'en dépendent pas. Ils agissent sur les coordonnées des objets.

En effet, les lutins sont repérés sur la scène par leurs coordonnées, l'abscisse étant nommée x et l'ordonnée y . Le repère utilisé pour la scène est centré; la scène étant un rectangle de 480 pixels de longueur pour 360 pixels de largeur.

L'arrière-plan `xy-grid` que l'on peut trouver dans la bibliothèque permet d'afficher un repère.



Les coordonnées de chaque lutin sont indiquées au dessus de la liste des lutins, en haut à droite :



Ces coordonnées sont par défaut celles du *centre* du lutin. Celui-ci peut être modifié en cliquant sur l'onglet *costume* puis en déplaçant le lutin dans la zone quadrillée. Le centre du lutin sera alors à l'emplacement de la discrète croix entourée d'un cercle :

Les déplacements des lutins peuvent donc être gérés par des blocs agissant sur leurs coordonnées, qui leur donnent des effets de translation :

aller à x: 0 y: 0

glisser en 1 secondes à x: 0 y: 0

ajouter 10 à x

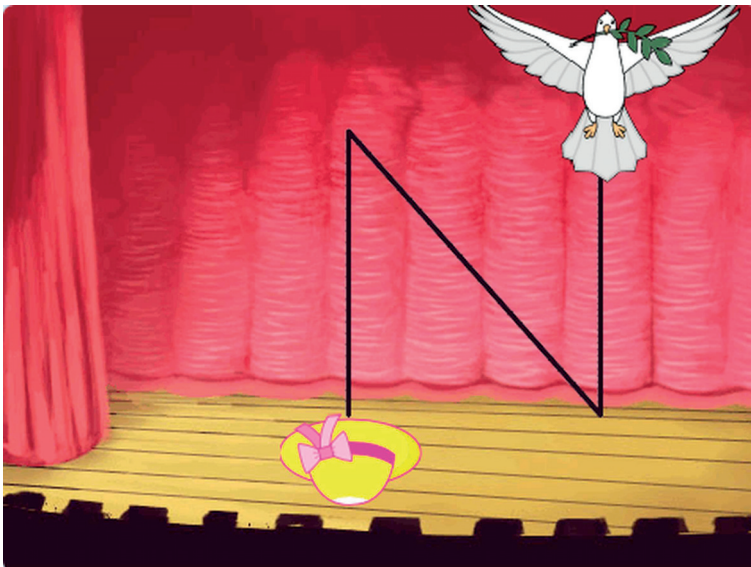
mettre x à 0

ajouter 10 à y

mettre y à 0

EXERCICE B3

Un chapeau est placé sur la scène, au point de coordonnées $(-20; -80)$. À l'aide du lutin Dove, faire surgir une colombe du chapeau puis la faire se déplacer comme sur l'illustration suivante.



Le bloc **s'orienter en direction de** permet de choisir l'orientation d'un lutin, indépendamment de celle de départ, selon la règle suivante : l'orientation à 0° correspond à l'orientation vers le haut. Puis, la mesure suit le sens des aiguilles d'une montre :






s'orienter en direction de -90 et **s'orienter en direction de 270** donnent le même résultat, à savoir l'orientation vers la gauche. Il est bien entendu possible d'écrire au clavier n'importe quelle autre valeur que celles proposées.

 Dans la liste des lutins, lorsque l'on clique sur la direction du lutin sélectionné :



- on peut visualiser sa direction actuelle ;
- on peut changer le style de rotation si nécessaire en cliquant

sur l'un des trois petits symboles    correspondants aux options de ce bloc :

fixer le sens de rotation gauche-droite ▾

- ✓ gauche-droite
- ne tourne pas
- tout autour

Enfin, un bloc permet de déplacer le lutin directement soit à la position du pointeur de la souris, soit à une position aléatoire sur la scène :

aller à position aléatoire ▾

- ✓ position aléatoire
- pointeur de souris

Si d'autres lutins sont déjà créés, alors il peut être déplacé à leur position :

aller à position aléatoire ▾

- ✓ position aléatoire
- pointeur de souris
- Sprite1
- Andie
- Casey

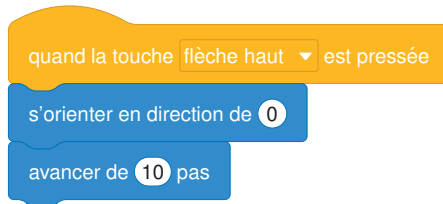
EXERCICE B4 *Quelle est la différence entre*

1. avancer de 10 pas et ajouter 10 à x ?
2. tourner  de 90 degrés et s'orienter en direction de 90 ?

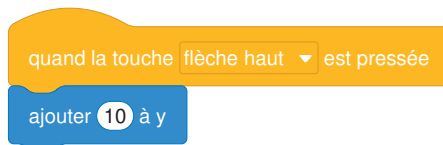
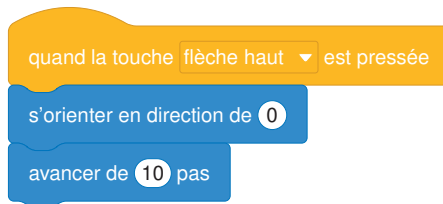
EXERCICE B5 *Écrire un programme qui, quand un lutin est cliqué, le téléporte ailleurs au hasard et lui fait changer de costume.*

EXERCICE B6*[Déplacements au clavier]*

1. *Que fait cet empilement ?*



2. *Quelle est la différence entre ces deux empilements ?*



3. *Pour chacune des deux méthodes ci-dessus, écrire les empilements de blocs correspondants aux trois autres touches fléchées du clavier.*

4. *Améliorer le programme pour créer un jeu à deux joueurs dans lequel un lutin est déplacé à l'aide des quatre flèches du clavier par un joueur. Un autre joueur doit cliquer sur le lutin avec la souris. A chaque clic sur le lutin, il rétrécit et se positionne ailleurs sur l'écran.*

EXERCICE B7

Écrire un programme dans lequel une balle (le lutin Soccer Ball par exemple) se déplace horizontalement vers la droite ou vers la gauche à l'aide des flèches du clavier. La balle devra donner l'effet qu'elle roule quand elle avance.