

*Adventure Game Studio est un logiciel de création de jeux vidéo, et plus précisément de point and click, des jeu d'aventures à la souris.*

## Table des matières

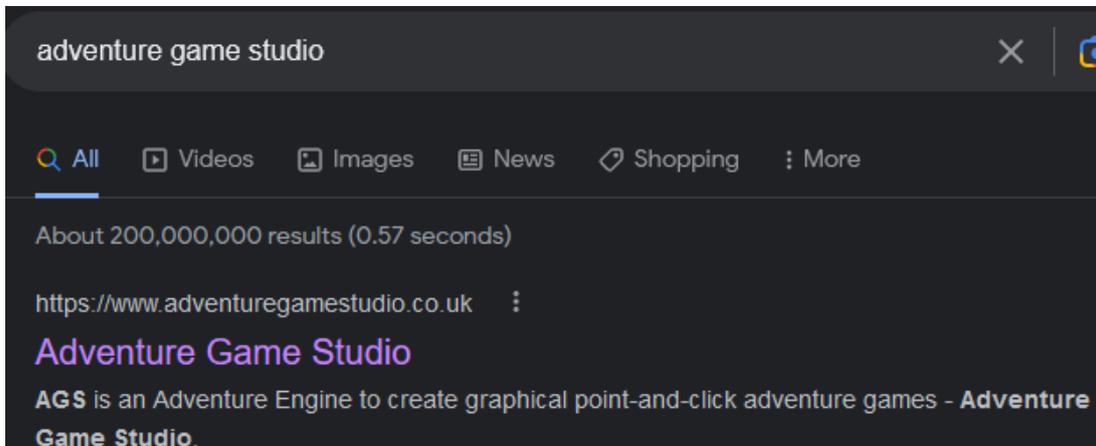
Présentation.....	2
Installation.....	3
Vocabulaire.....	4
Créer le jeu.....	5
Paramétrage.....	7
Résolution du jeu.....	7
Création de la première scène.....	9
Créer un décors.....	10
Créer des zones franchissables.....	10
Le personnage principal.....	12
Créer un personnage principal.....	12
Importer les sprites un par un.....	13
Importer tous les sprites en même temps.....	14
Créer une animation de marche.....	15
Paramétrer votre héro.....	20
Placer le héro dans votre décors.....	25
Interactions.....	30
Le personnage parle avec lui-même.....	30
Le personnage interagit avec un objet.....	32
Le personnage ramasse un objet.....	36
Ressources.....	37

# Présentation

Adventure Game Studio est un logiciel de création de jeux vidéo, et plus précisément de point and click, des jeu d'aventures à la souris. Il permet de travailler tous les aspects de votre jeu : le script pour ajouter des événements, la musique, les sprites...etc.  
Notez que si le présent tutoriel est en français, le logiciel lui est en anglais.

# Installation

Se rendre sur <https://www.adventuregamestudio.co.uk/> ou chercher sur le web "adventure Game Studio" et cliquer sur le lien suivant :



Cliquer sur « Download AGS ! » :



Enregistrer le fichier, le lancer en cliquant sur la flèche bleue en haut à droite de Firefox puis en cliquant sur le nom, et suivre le processus d'installation classique soit cliquez sur Oui puis cochez la case : I accept the agreement puis cliquez sur next — next ---finish. Si votre antivirus bloque, exécutez quand même le logiciel pour l'installer.

Pensez également à installer : [Microsoft .NET Framework 4.5](#)

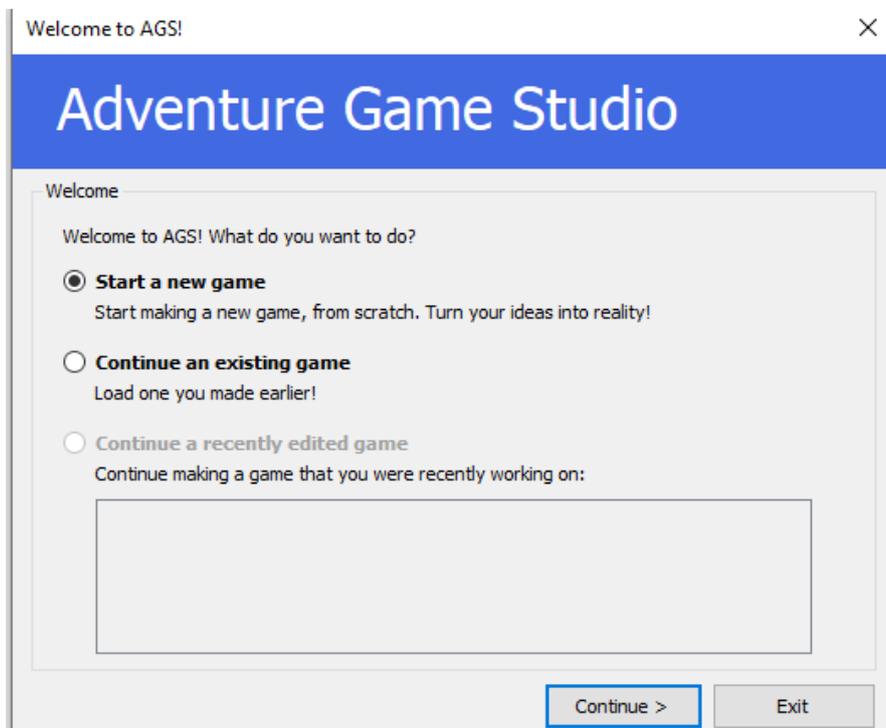
# Vocabulaire

AGS est en anglais, voici une traduction des termes techniques usuels :

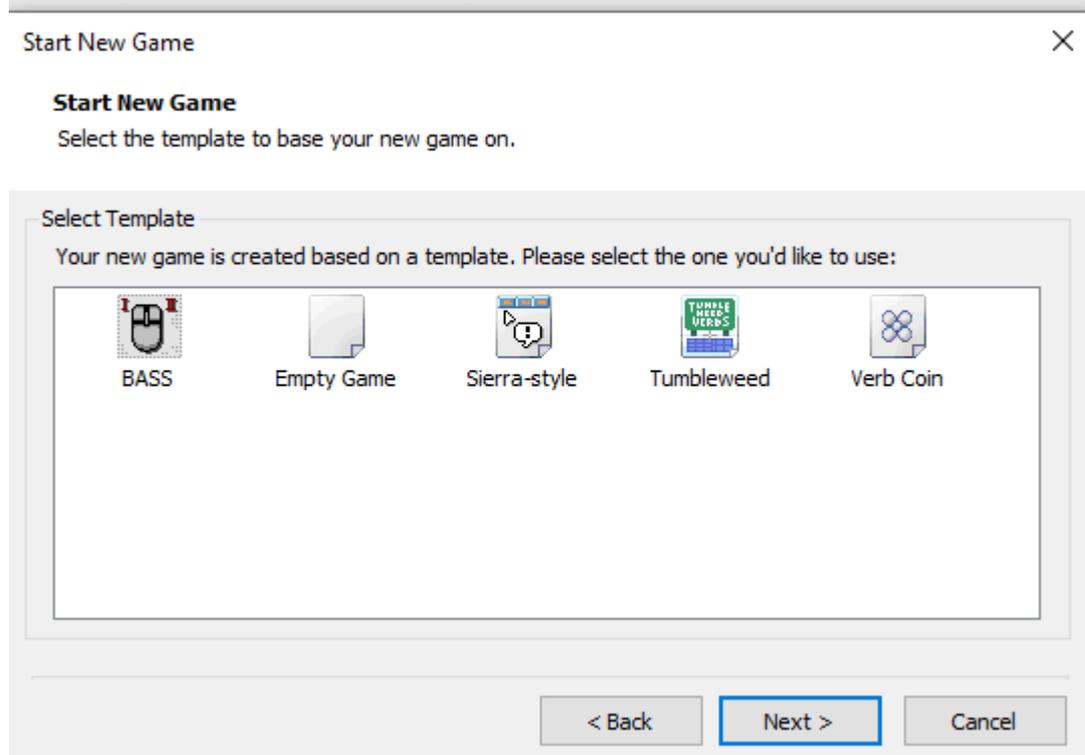
- **Array** : Tableau
- **Blocking script** : Bloquant l'exécution
- **Character** : Personnage
- **Clip (Audio)** : Séquence
- **Drawing Surface** : Surface de dessin
- **Frame** : Vignette (anciennement Image)
- **GUI** : Interface
- **Hotspot** : Zone interactive
- **Inventory Item** : Objet d'inventaire
- **Loop (dans une vue)** : Séquence (d'animation)
- **Object** : Objet
- **Overlay** : Calque
- **Plugin** : Greffon
- **Room** : Pièce (décor)
- **Speech** : Discours, Dialogue
- **Sprite** : Image
- **String** : Chaîne de caractères
- **View** : Vue
- **Voice (speech)** : Doublage
- **Walkable Area** : Zone de déplacement
- **Walkbehind Area** : Zone de plan intermédiaire/Zone couvrante

# Créer le jeu

Une fois l'installation terminée une fenêtre va s'ouvrir. Pour créer un jeu, choisissez :  
« **Start a new game** »



Puis cliquez sur : « **Continue** » puis « Next », une seconde fenêtre va s'ouvrir



Ici, vous devez choisir quel modèle vous aller utiliser pour votre nouveau jeu. Par simplicité nous allons sélectionner le modèle « **BASS** » puis cliquez sur « **Next** », une fenêtre va s'ouvrir :

Start New Game X

**Start New Game**  
Decide what your game's name will be.

New game details

What do you want the game to be called? This name will be displayed in the title bar of the window whilst playing. You can change this later in the General Settings editor.

New game|

Please enter the filename of the game below. This will be used to name the final build of your game. You can change this later in the General Settings editor.

Create in folder:

C:\Users\utilisateur\Documents ...

< Back Finish Cancel

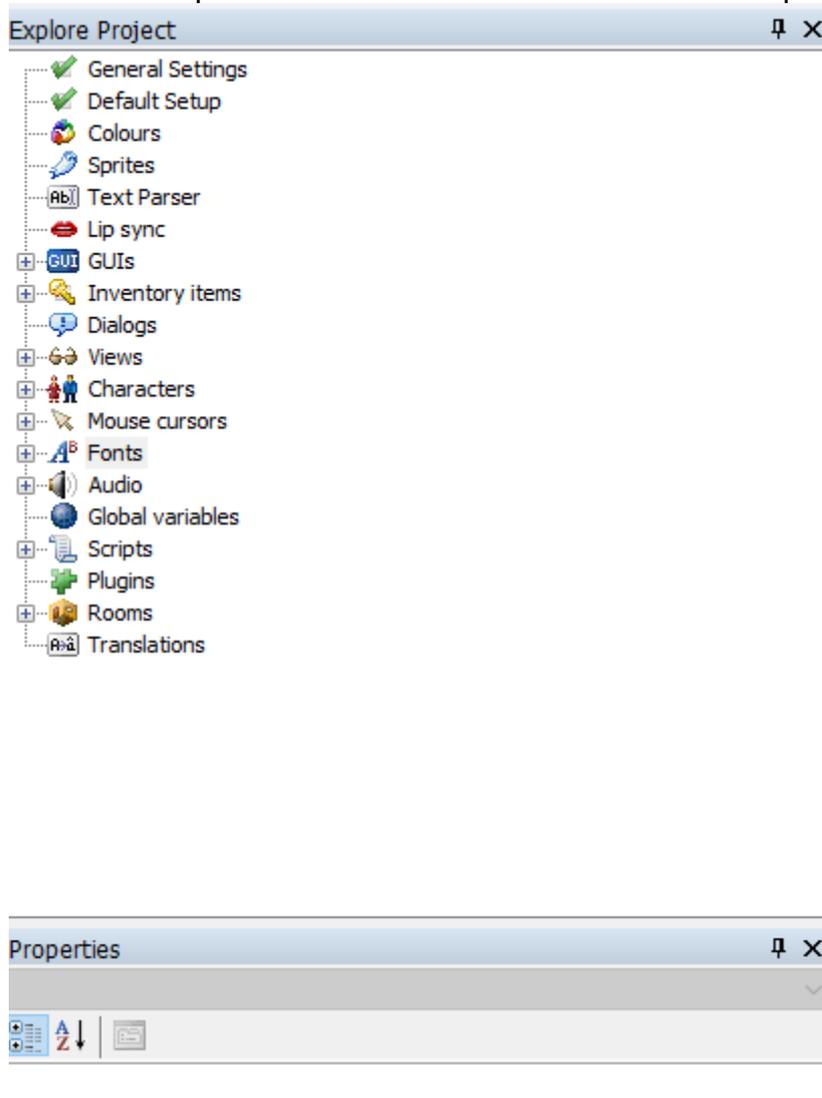
A la place de « **new game** » : donnez un nom à votre jeu (ici ce sera **Les aventures de l'EPN**).

La seconde zone vous demande le « **file name** » (**nom de fichier**) de votre jeu. Cela ne servira qu'au nom du répertoire dans lequel votre jeu sera créé et sauvegardé, par exemple le nom d'un des héros de votre jeu (**ici EPN**), puis cliquez sur « **Finish** ».

Votre jeu est prêt à être créé !

## Paramétrage

Pour naviguer dans l'éditeur, il faut utiliser l'**arbre du projet** dans le coin supérieur droit, celui-ci vous permet d'accéder à toutes les différentes parties de votre jeu.

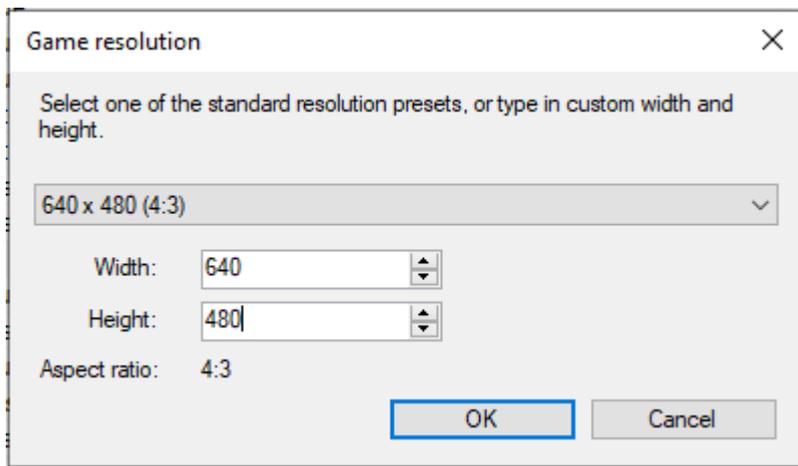
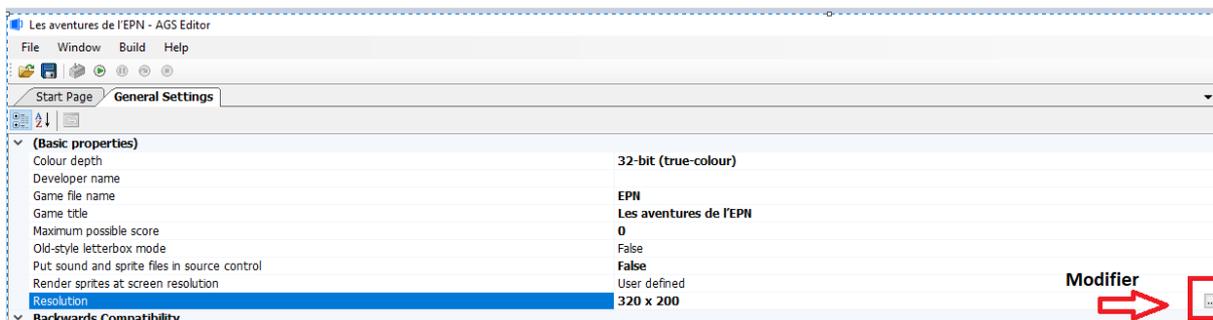


## Résolution du jeu

Pour éviter d'avoir un jeu trop petit on va changer la résolution du jeu qui est de : 320 par 200 pour le mettre en 640 par 480.

Dans « l'arbre du projet » **double cliquez** sur « **General setting** » qui correspond **aux réglages généraux de votre jeu.**

Puis cliquez sur le bouton « modifier » et mettez la résolution désirée.



Puis cliquez sur « **OK** » et « **OUI** »

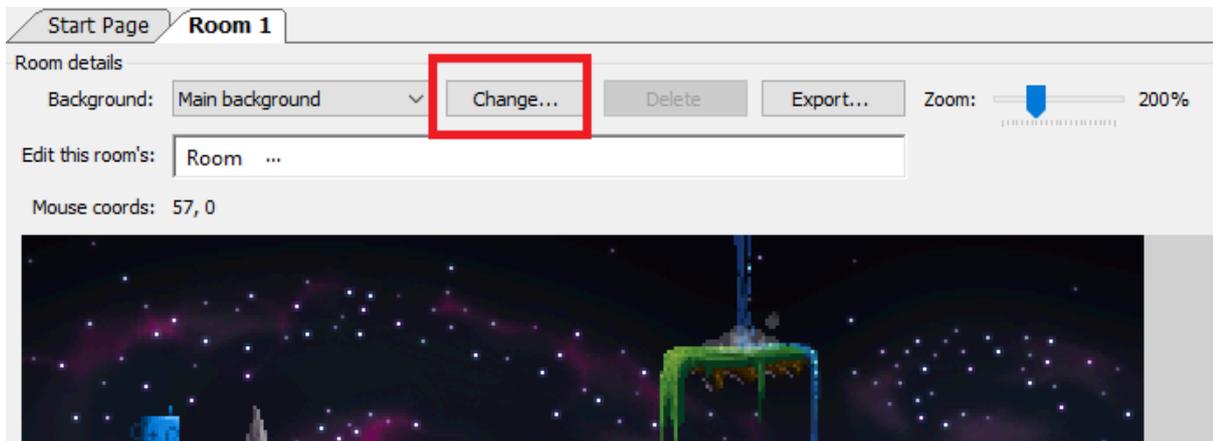
De ce fait, notre décor devra être de 640 \* 480 pixels

## Création de la première scène

En amont, créer un dossier « photo » sur votre pc et mettez l'image désirée à l'intérieur de ce dossier, notez que l'image de votre décors devra faire 640\*480 pixels ou elle sera rognée.

Puis sur l'**arbre du projet**, » **double cliquez** sur «**room**» qui correspond **aux pièces (scène) de votre jeu** puis **double cliquez** sur «**editroom**», une image de base de décor va apparaître.

Vous pouvez remplacer cette image par la votre en cliquant sur « **change** » qui signifie modifier.



Cherchez votre dossier « photo » sélectionner votre image de décor, sélectionnez celle-ci puis cliquez sur « ouvrir » puis répondez « oui » à la question. Pour mémoire, la taille de votre image doit être au moins de 480 de haut et ce n'est pas grave si elle est légèrement plus large car la caméra va se déplacer avec lui. Attention, en aucun cas votre image ne doit être plus petite.

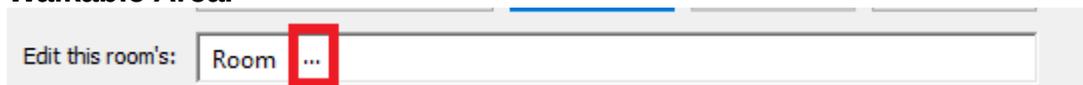
Retourner dans « l'arbre du projet » **double cliquez** sur « **General setting** » qui correspond **aux réglages généraux de votre jeu**, puis modifiez le « **colourdepth** » de 32 à 16 et dites « **oui** ». En effet, vous avez dû voir un petit message d'erreur lorsque vous avez importé votre décor, c'est dû aux couleurs 256 par défaut, **c'est pour cette raison qu'il faut choisir** le mode 16-bit color.

Pour information : vous verrez apparaître deux lignes jaunes sur votre décor. Celles-ci servent à déterminer les zones d'interaction où le joueur, une fois franchis une de ces lignes pourra passer sur un autre décors...

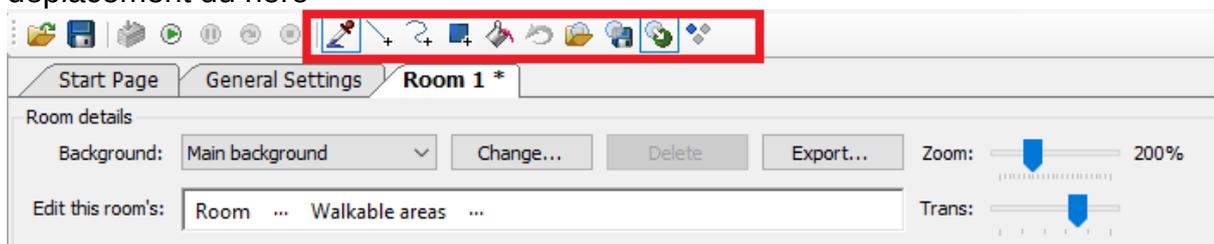
# Créer un décors

## Créer des zones franchissables

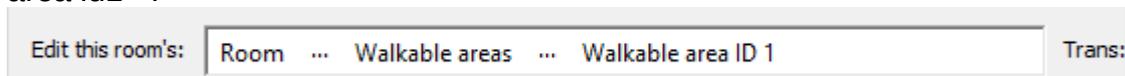
On va tracer un chemin sur lequel notre personnage va se déplacer dans ce décor. Pour ce faire nous allons utiliser l'outil **Walkable Area**, qui signifie Zone de déplacement. Dans « Edit this room's » (tout en haut), cliquez sur les 3 petits points et choisissez : **Walkable Area**.



Une série d'outil va apparaître au dessus qui vous permettront de déterminer les zones de déplacement du héros

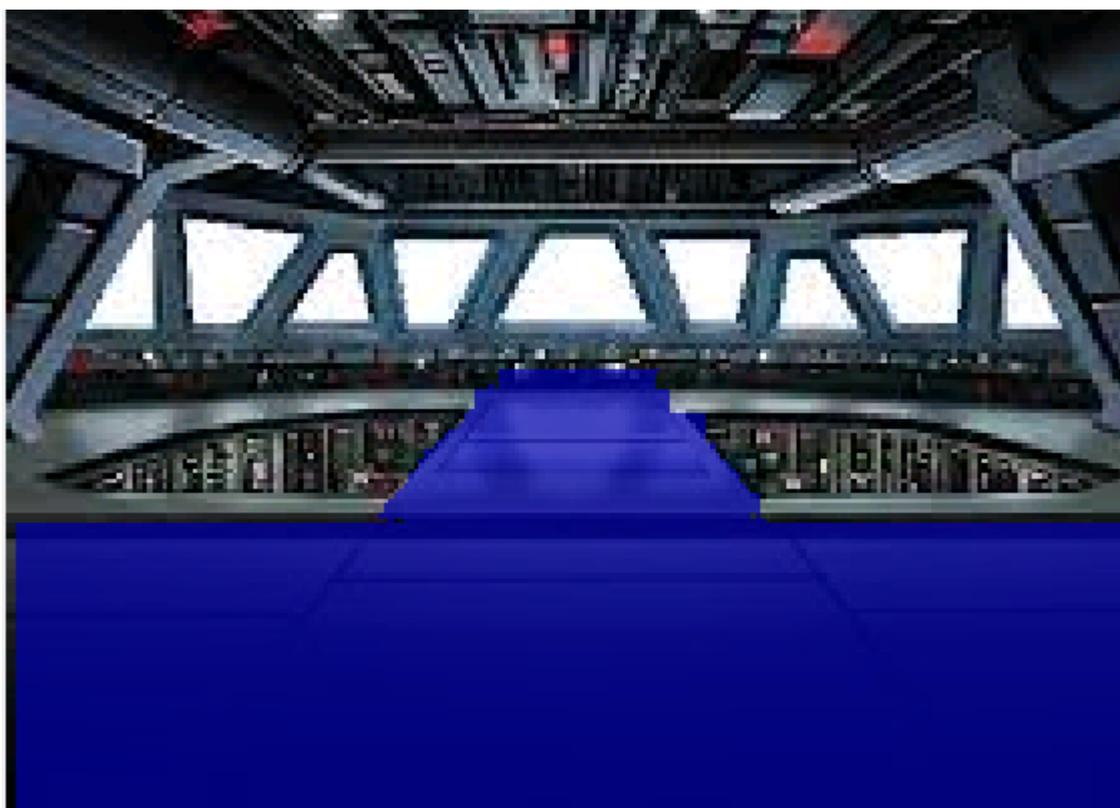


Il faut ensuite déterminer la zone de déplacement du héros, cliquez de nouveau sur les trois petits points qui se trouve à coté de « Walkable areas » et choisissez « Walkable area id1 ».



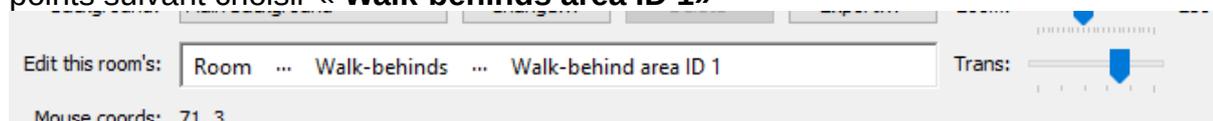
Puis avec l'outil rectangle  on va commencer à définir la zone de déplacement du héros, puis en fonction de son décor et de la zone de déplacement choisi, on peut utiliser les autres outils.

Attention : il faut bien entourer toute la zone franchissable par le héros, il ne peut pas y avoir de « trou de pixel », de ce fait n'hésitez pas à utiliser le **zoom**  
Vous allez obtenir à peu près ceci :



Toute la zone bleue correspond à la zone de déplacement de notre héros.

Si on veut faire marcher notre héros en arrière-plan, il faut dans Edit this room's (tout en haut) cliquer sur les 3 petits points et choisir : « **Walk-behinds** » puis dans les 3 petits points suivant choisir « **Walk-behinds area ID 1** »



Puis avec l'outil rectangle  et les autres outils, définissez la zone de déplacement du héros en arrière-plan.

# Le personnage principal

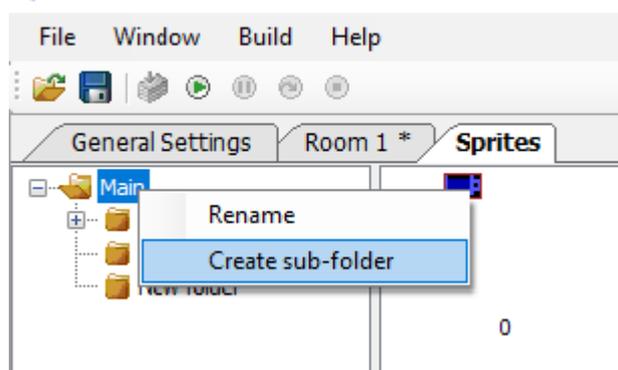
## Créer un personnage principal

Dans un 1<sup>er</sup> temps, on va importer des « sprites », ce sont des images correspondant aux diverses images d'animation de notre héros.

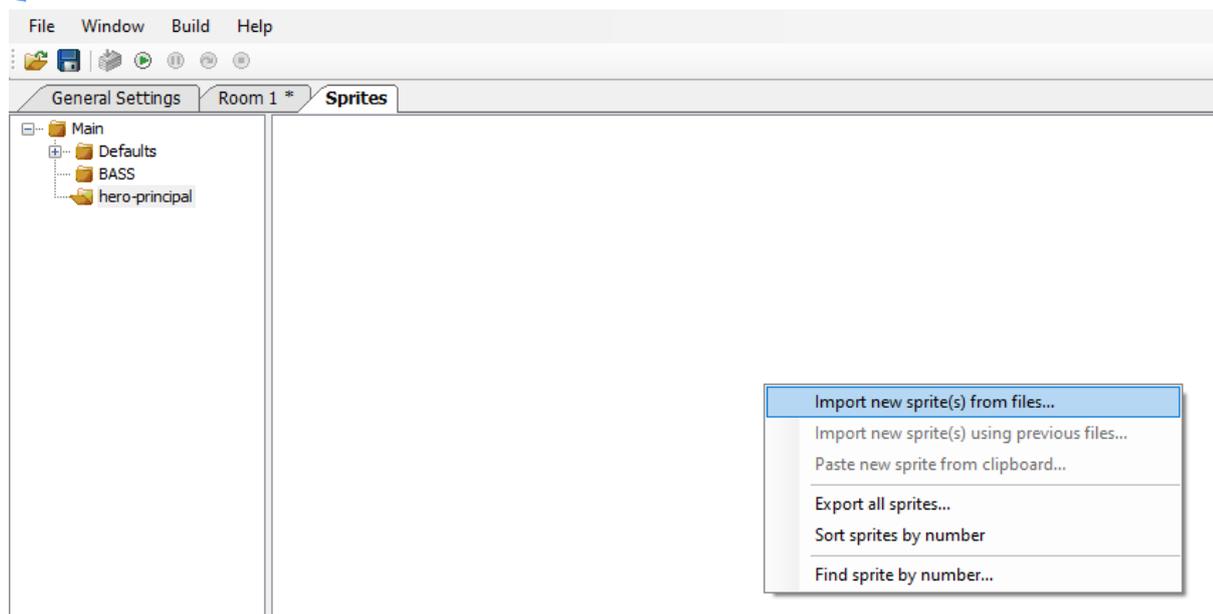
Retourner dans « l'arbre du projet » **double cliquez** sur « **Sprites** » puis sur « **Main** », faite un clic droit et sélectionnez « **createsub-folder** » qui signifie créer un sous-dossier, c'est à cet endroit que nous allons mettre les images d'animation de notre héros.

Ce dossier se nomme : **New folder**, il faut lui donner un nom, ici se sera : **hero-principal** et après dans la partie centrale (blanche), on va importer l'image.

Les aventures de l'EPN - AGS Editor



Les aventures de l'EPN - AGS Editor



**Vous trouverez ici des sprites à télécharger :**

**Attention : TRES IMPORTANT il faut impérativement lorsque vous choisissez vos sprites, qu'il y ait des images où le personnage regarde en BAS, en HAUT, à GAUCHE et à DROITE**

<https://www.pngegg.com/fr/png-yndfy>

<https://www.shutterstock.com/fr/image-vector/animated-robot-character-creating-scifi-video-780817981>

Télécharger votre sprite dans le dossier « photo » que vous avez précédemment créé sur votre PC.

Voici un exemple de sprite :

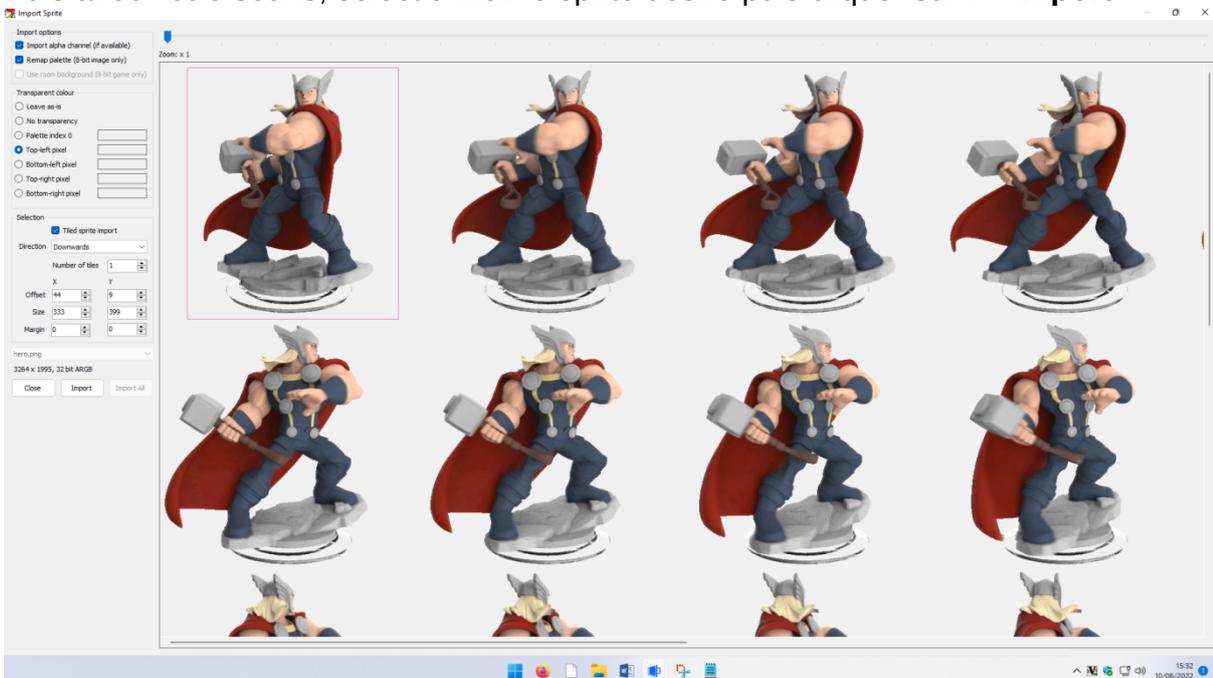


Si l'image que vous trouvez n'est pas en png, il faudra sans doute la retravailler sous Gimp pour qu'elle ait un arrière-plan transparent.

### Importer les sprites un par un

Après avoir importé, votre image, une fenêtre va s'ouvrir, il faudra alors cocher la case : **tile sprite import**.

Puis avec votre souris, sélectionnez le sprite désiré puis cliquer sur : « **Import** ».



Renouveler cette opération autant de fois que nécessaire pour importer chaque sprite.

## Importer tous les sprites en même temps

**Attention : dans un premier temps : il faut faire un clic droit sur votre image de sprite afin de connaître sa hauteur et sa largeur – faire clic droit sur l'image et cliquez sur l'onglet détails et noter la hauteur et la largeur**

Après avoir importé, votre image une fenêtre va s'ouvrir, il faudra alors cocher la case : **top left pixel**.

Dans selection : il faudra alors cocher la case : **tilesprite import** ;

Dans direction : choisir **to the right** ;

En amont, il faut compter le nombre de sprites (images) que vous avez, ici on en a 40 ;

Dans «**number of titles** »: il faut donc mettre le nombre 40 ;

Et **dans size**, il faudra pour :

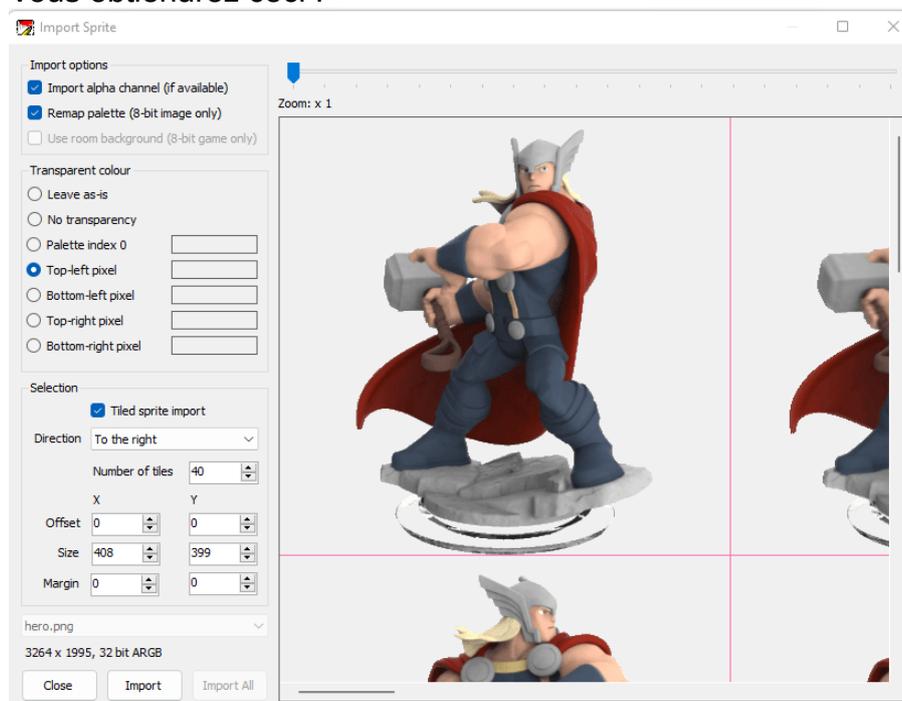
**Size x** : il faudra diviser la largeur de l'image par le nombre de personnage en largeur.

Dans cet exemple, il faut faire  $3264 / 8 = 408$  ;

**Size y** : il faudra diviser la hauteur de l'image par le nombre de personnage en hauteur.

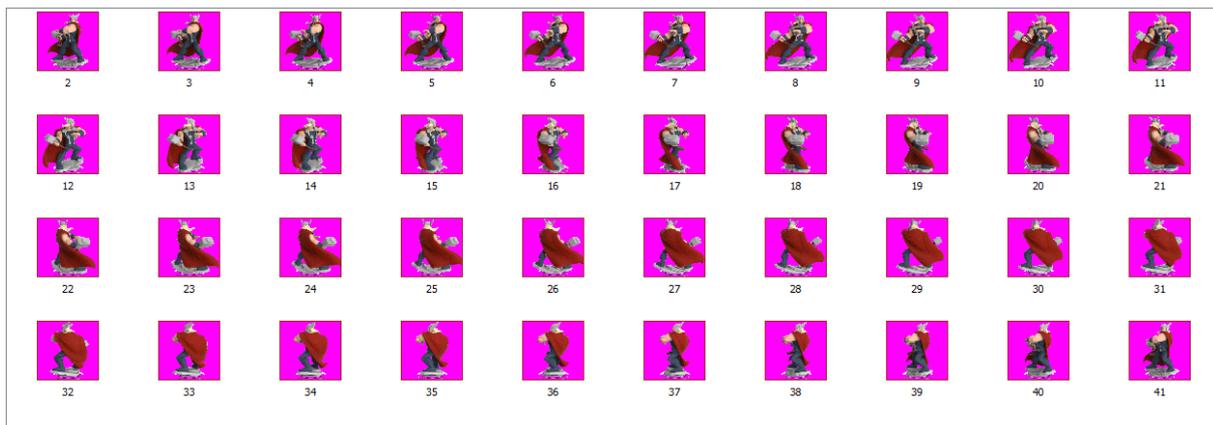
Dans cet exemple, il faut faire  $1995 / 5 = 399$  ;

Vous obtiendrez ceci :



Cliquez ensuite sur le bouton « import ».

Après l'import, vous verrez que tous vos sprites sont en rose c'est-à-dire qu'ils sont sur un fond transparents qui permettra de voir le décors.



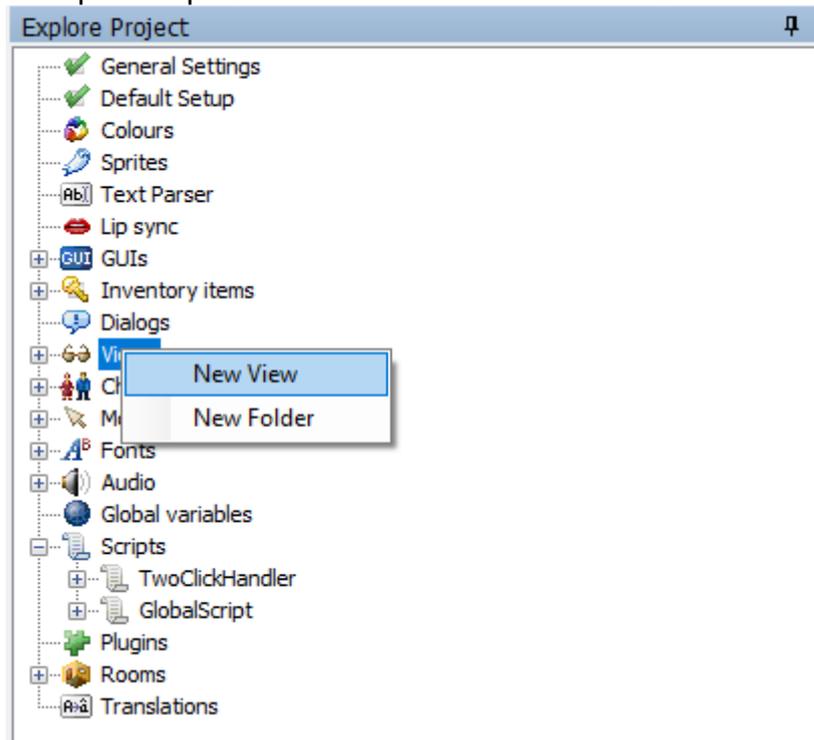
Si vous n'êtes pas satisfait du résultat vous pouvez toujours importer vos sprites un par un

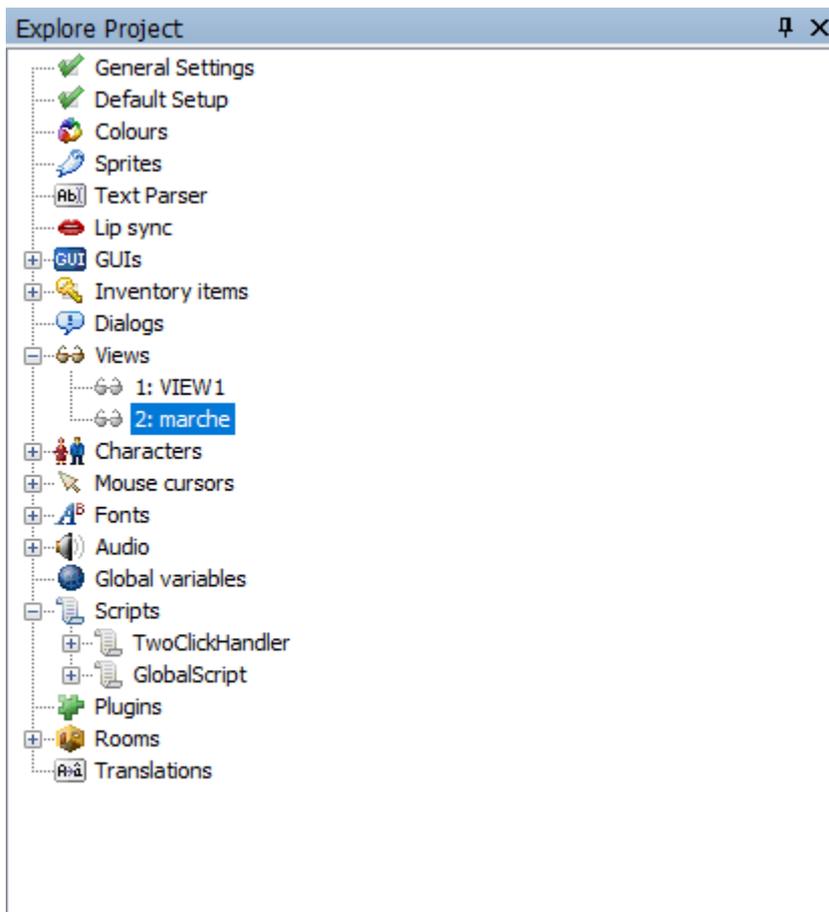
## Créer une animation de marche

On va créer une « view » cela correspond à une série de sprites qui vont s'enchaîner les uns après les autres et vont ainsi donner l'illusion du mouvement.

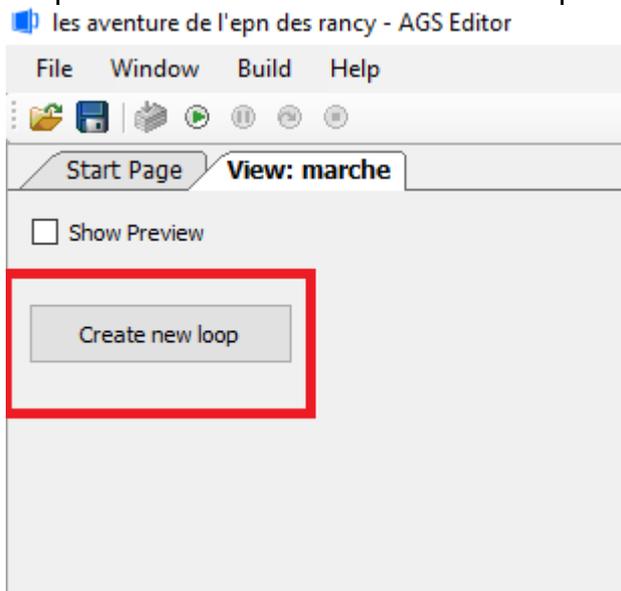
On va donc créer une view qui se nommera « marche ».

Pour ce faire, Retourner dans « l'arbre du projet » cliquez sur « **Views** » et faire un clic droit puis cliquez sur « New View » et nommez-la « marche ».





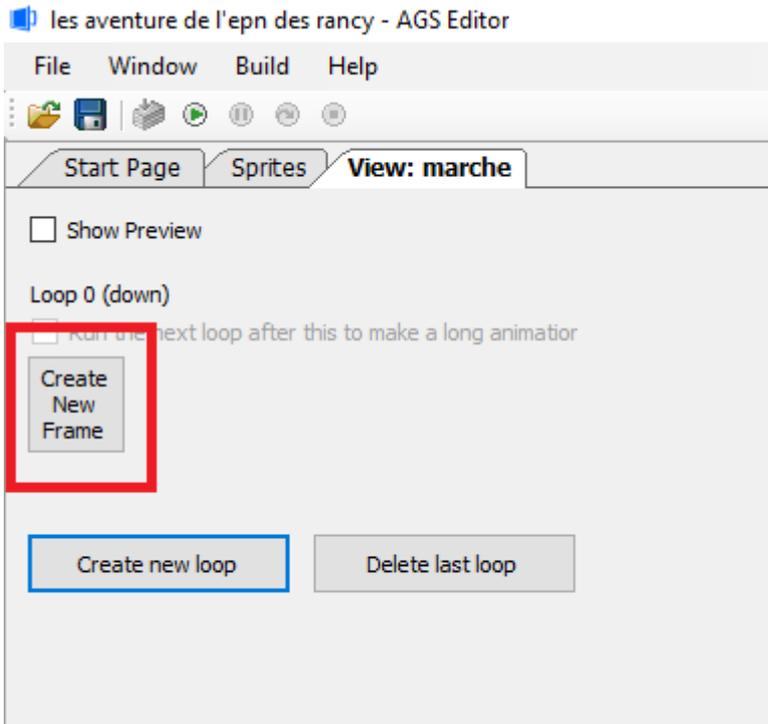
Cliquez ensuite sur « Create a new loop » :



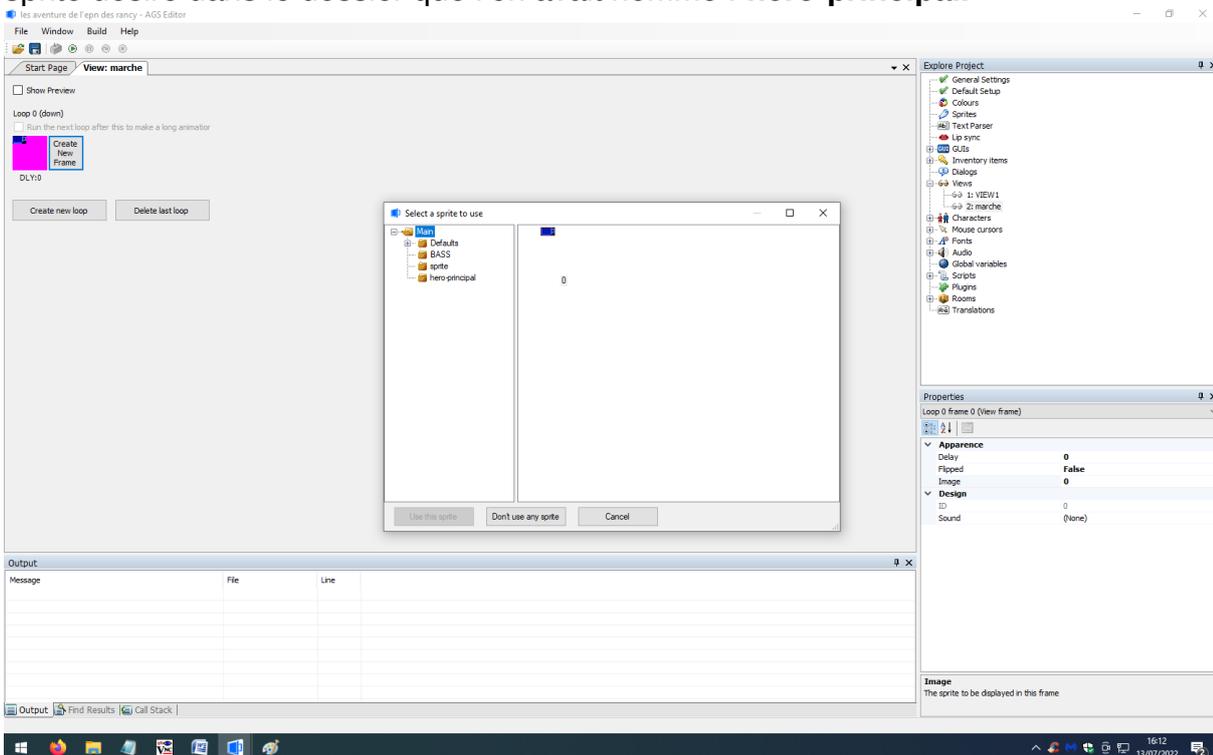
Pour information : Une loop est une animation qui est en boucle, chaque « view » est décomposée en plusieurs loop (animation) : une pour le haut, une pour le bas, une pour la gauche...etc.

On va choisir une loop où le personnage **marche vers le bas** (loop 0 (down)).  
Après avoir cliqué sur : « **create new loop** », une fenêtre va s'ouvrir.

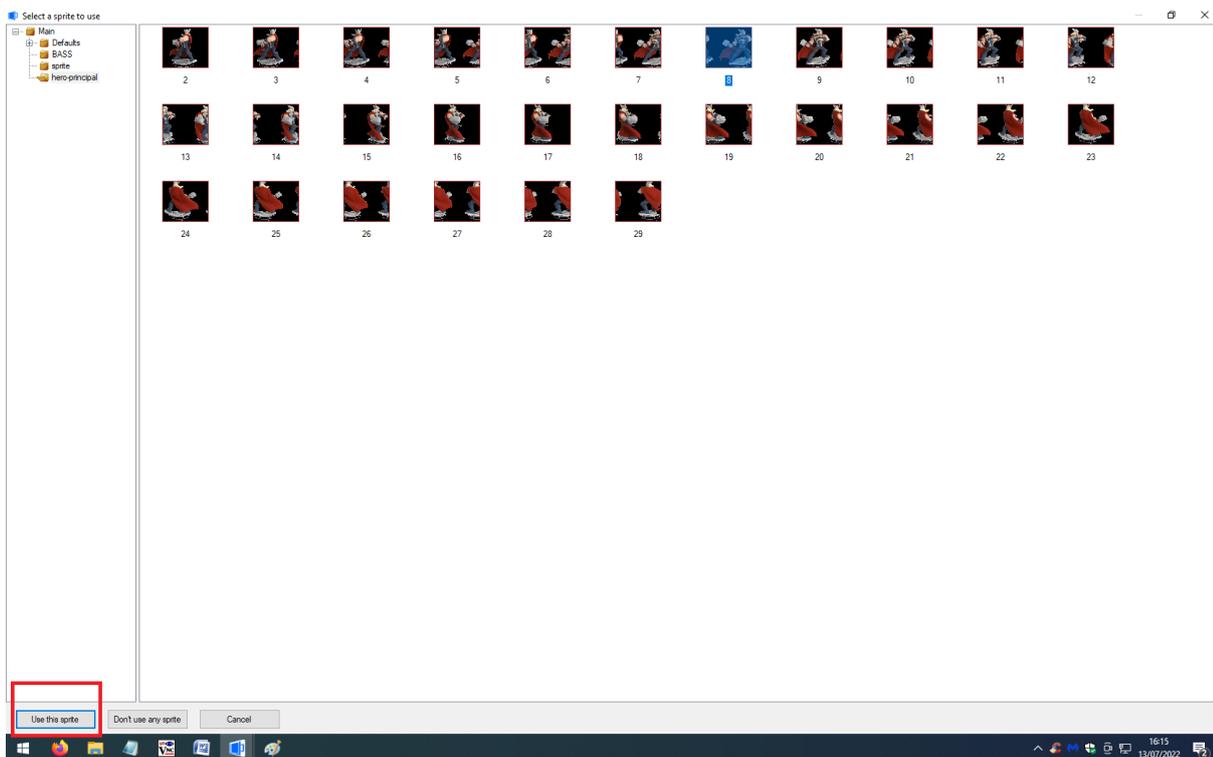
Cliquez sur « Create New Frame » :



En cliquant sur le carré rose une fenêtre va s'ouvrir. Elle vous permettra de sélectionner le sprite désiré dans le dossier que l'on avait nommé : **hero-principal**.

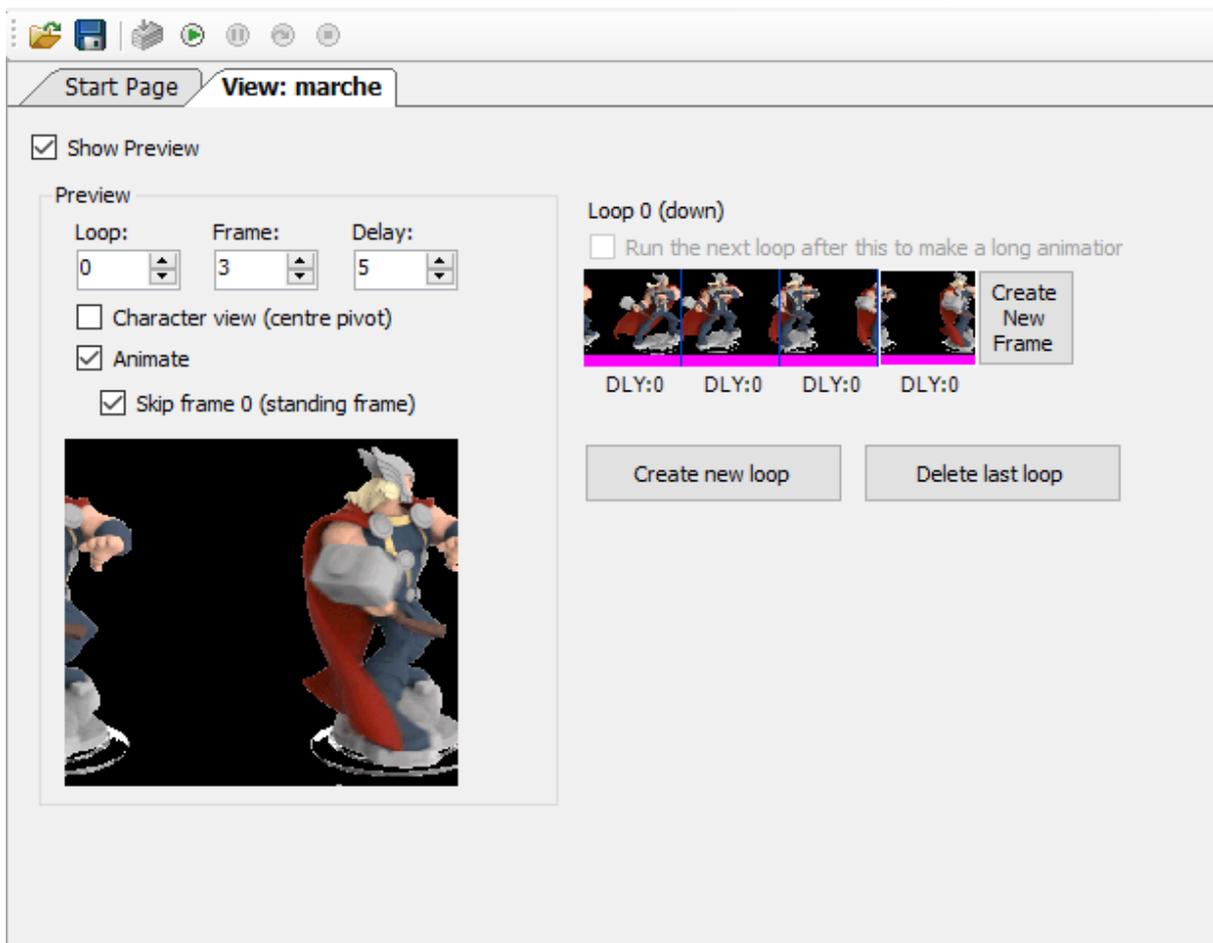


Sélectionnez donc le sprite désiré et tout en bas cliquez sur : « use this sprite ».



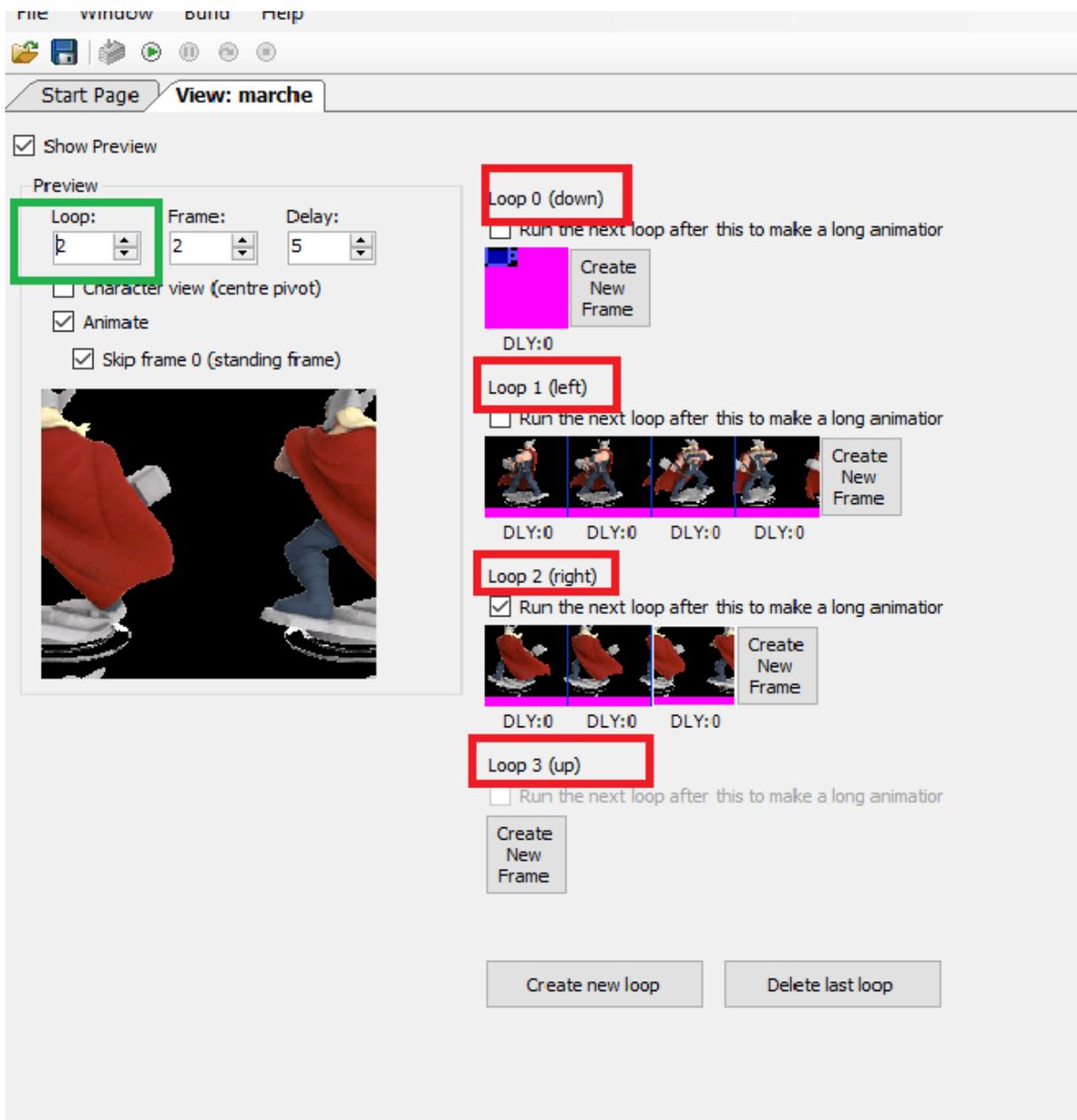
Cliquez à nouveau sur « **create new frame** » puis cliquez sur le carré rose, une fenêtre va alors s'ouvrir. Retournez dans le dossier principal et choisissez un **2<sup>ème</sup> sprite** qui marche vers la gauche puis tout en bas **cliquez sur : « use this sprite »**. Refaites cette opération autant de fois que nécessaire.

**Une fois que vous avez fini il faut cocher la case « show preview » puis cocher la case « animate » et cocher la case « skip frame » ainsi on pourra voir ce que donne notre loop (animation).**



Il faut ensuite recommencer cette opération pour la droite et en haut, n'hésitez pas à tester vos animations pour en vérifier le rendu.

Pour cela, il faut cocher la case « **show preview** » puis cocher la case « **animate** » et cocher la case « **skip frame** » ainsi on pourra voir ce que donne notre loop (animation). Attention chaque loop a un chiffre, donc pensez à mettre le chiffre qui correspond à l'animation que vous désirez voir

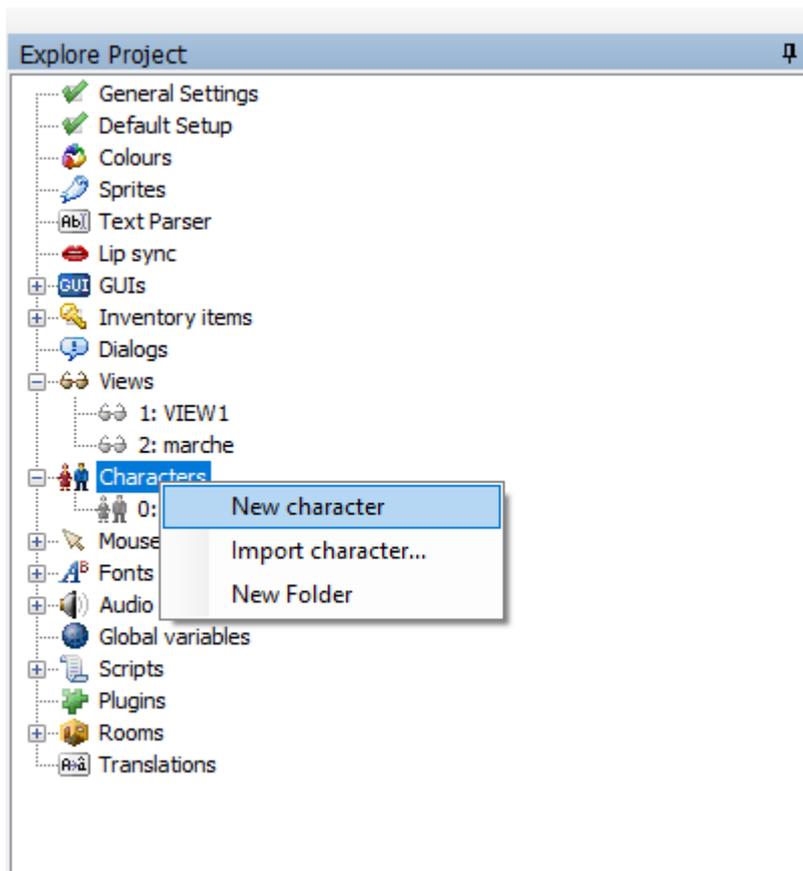


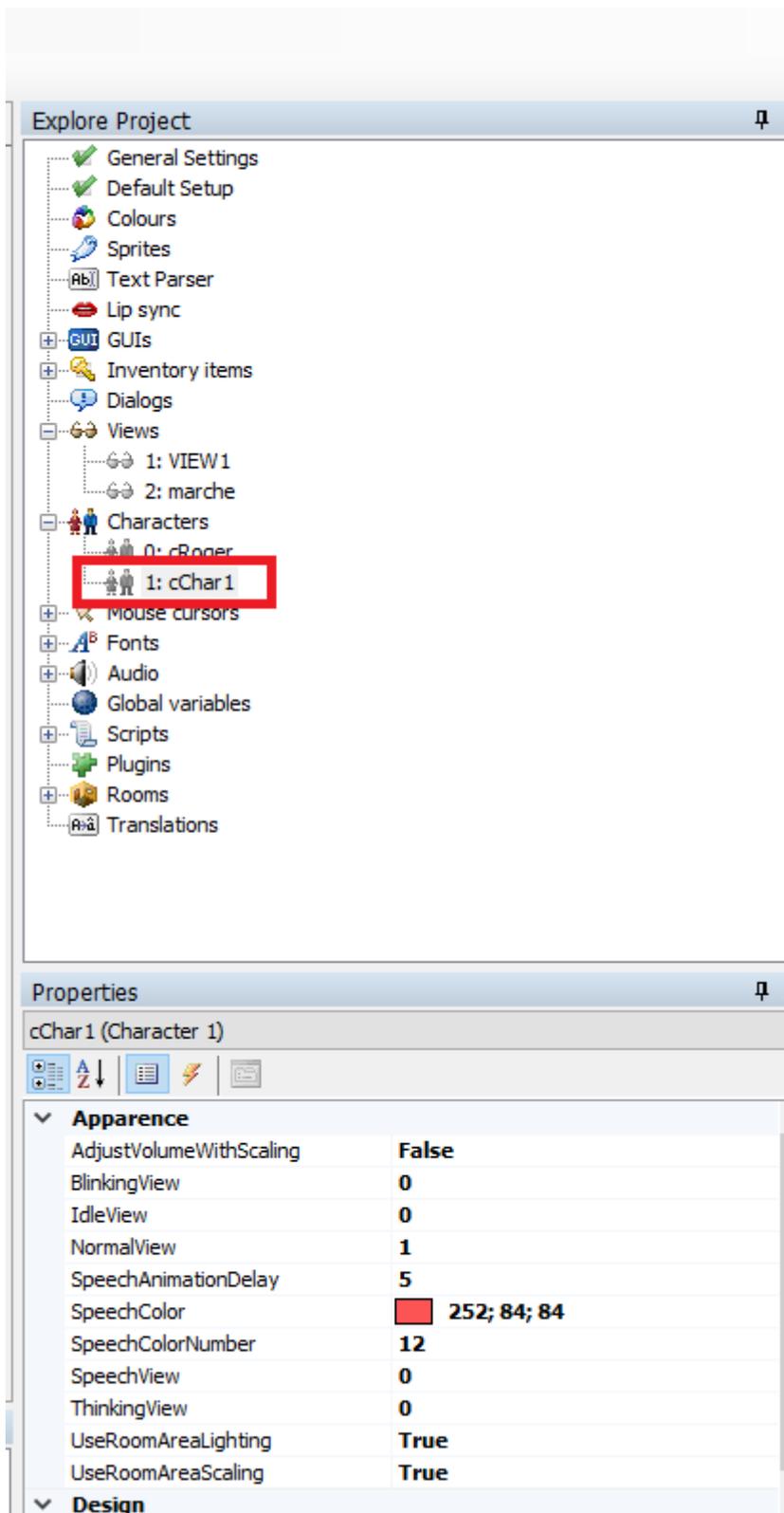
En dessous de preview dans loop (carré vert) si vous mettez :

- 0 vous verrez les loop de l'animation correspondant au bas
- 1 vous verrez les loop de l'animation correspondant a la gauche
- 2 vous verrez les loop de l'animation correspondant a la droite
- 3 vous verrez les loop de l'animation correspondant au haut

## Paramétrer votre héros

Retourner dans « l'arbre du projet », **cliquez** sur « **Characters** » puis faire un clic droit et cliquer sur « **new character** ».

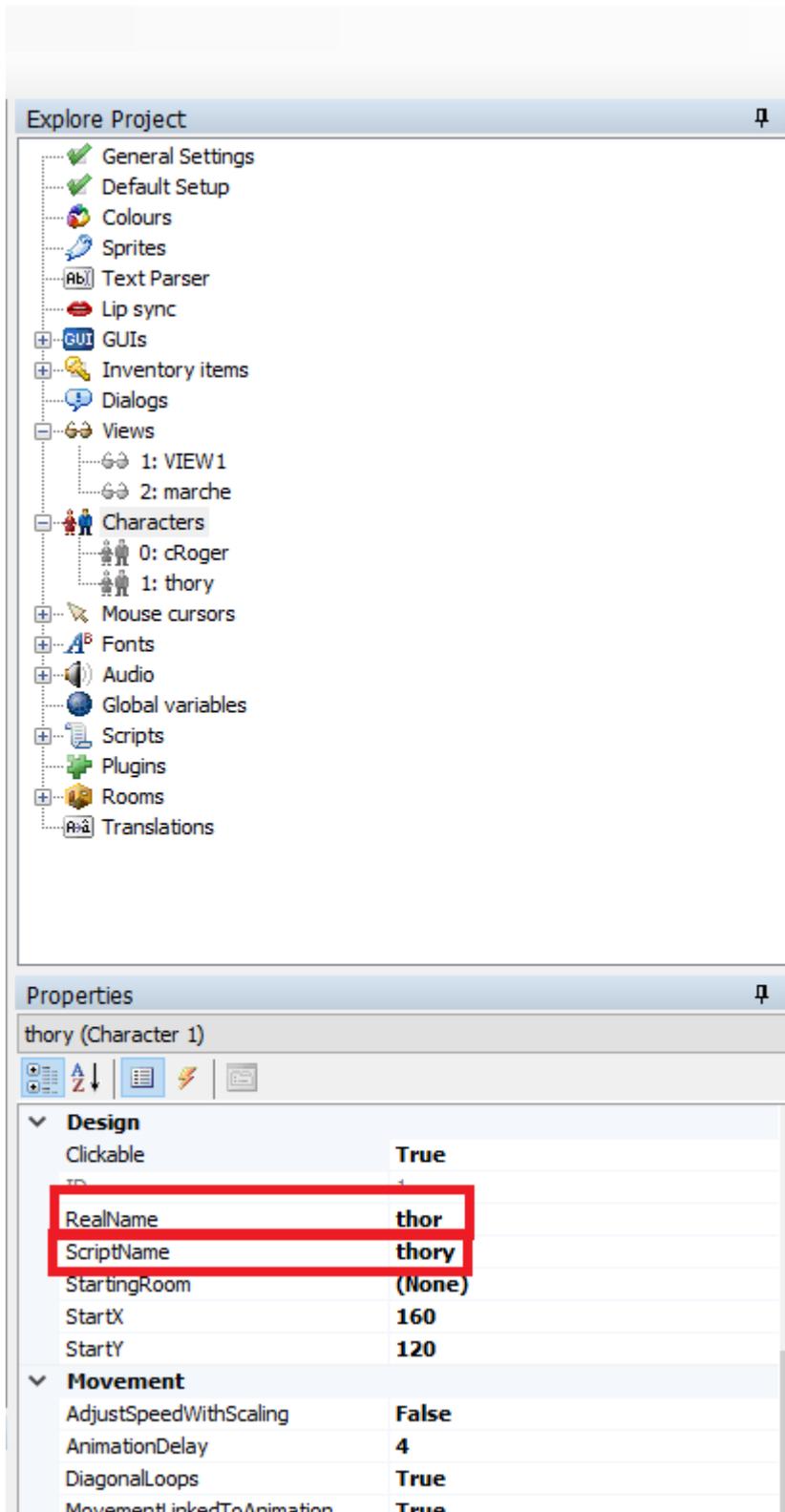




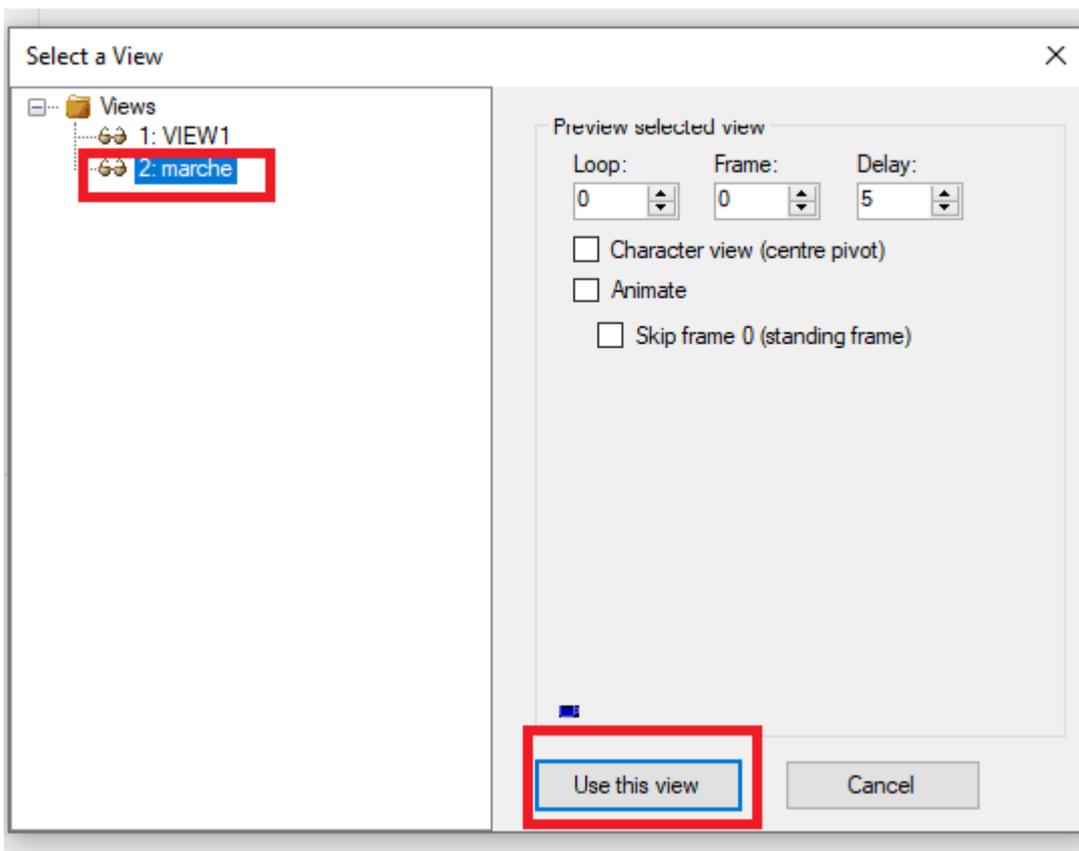
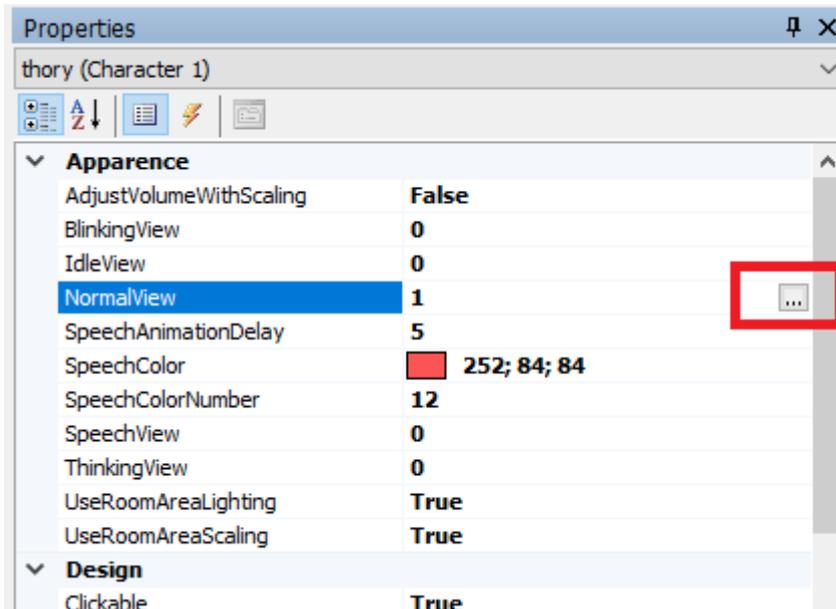
Cliquez et positionnez-vous sur « cChar1 ». Dans **Properties** (en bas) nous allons faire des modifications :

A côté de « **ScriptName** » on va donner un nom à notre personnage (par exemple Thory )  
– Ce nom correspond à celui par lequel il sera appelé dans le script au moment de programmer.

On va ensuite à côté de « **RealName** » : donner un nom (par exemple Thor), celui-ci va correspondre au nom qui apparaîtra quand le curseur de la souris passera sur lui.

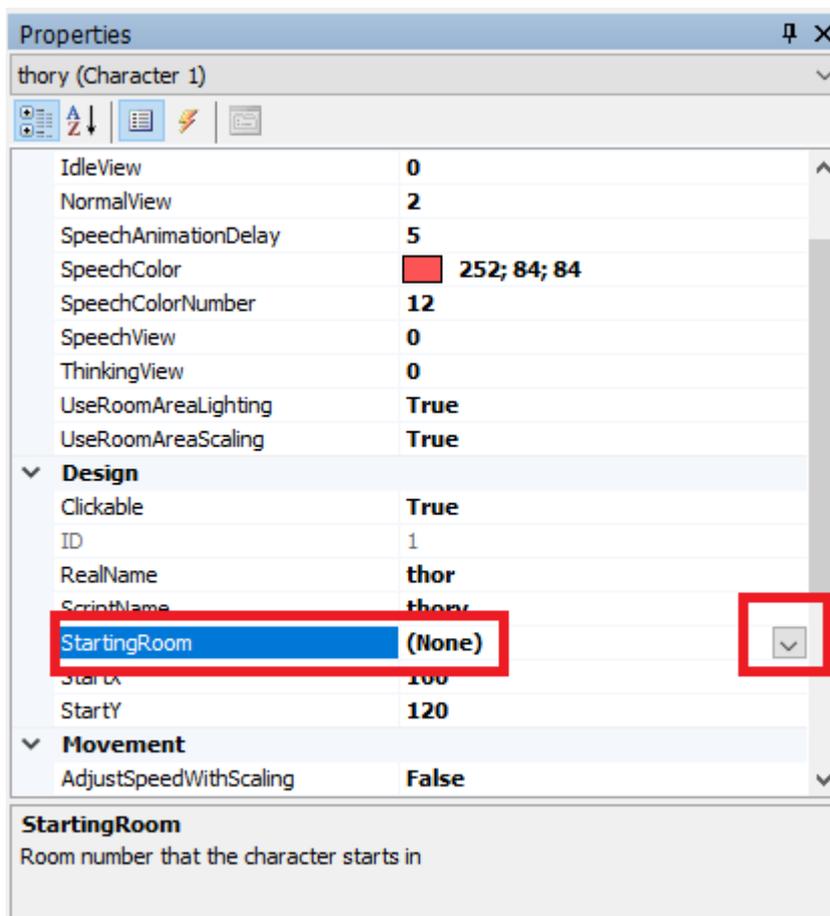


Ensuite, à côté de « **NormalView** », on va aller sélectionner l'animation de marche, cliquez sur les ... (qui correspond à parcourir) et sélectionner votre view qui se nomme **marche**.



Puis cliquez sur « **Use this view** »

On va placer notre personnage dans la room que nous avons créé au tout début, donc toujours dans propriétés à côté de « **StartingRoom** », cliquez sur la flèche :



Puis cliquez sur 1 qui correspond à votre 1<sup>ère</sup> room (la 1<sup>ère</sup> scène (pièce) ou vous avez mis votre décor.

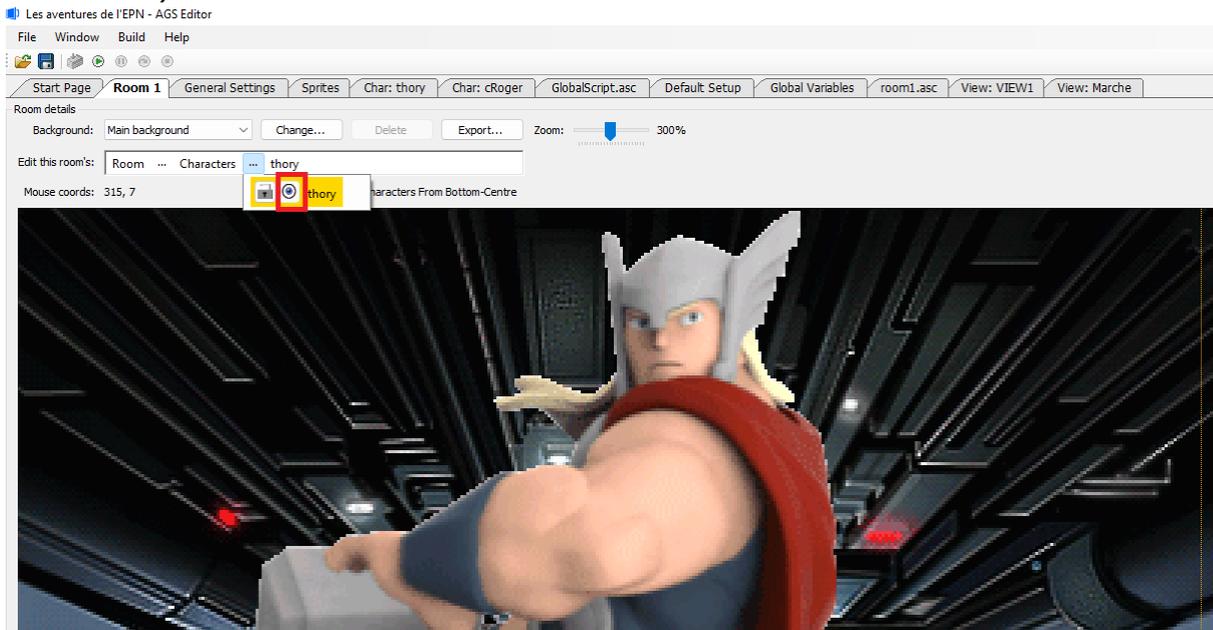
Ensuite, Il va falloir supprimer le personnage que nous a imposé AGS, pour ce faire il faut retourner dans l'arbre du projet et cliquez sur « **character** » puis cliquez sur « 0cRoger » puis dans propriétés, et dans « **StartingRoom** » : il faut mettre **(None)**, ainsi ce personnage n'apparaîtra pas dans votre jeu.

## Placer le héros dans vos décors

Retournez dans « l'arbre du projet », cliquez sur « **room** » puis faites un clic droit et cliquez sur « **editroom** ».

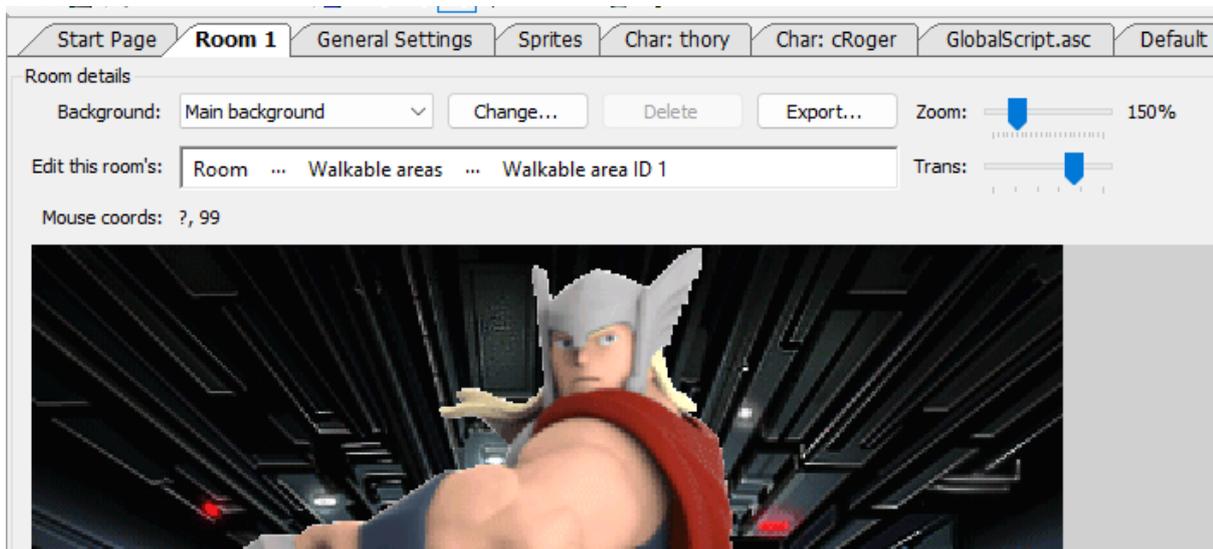
Tout en haut dans « **editroom** », après « **room** » il faut cliquer sur les 3 petits points et choisissez : « **characters** » puis dans les 3 petits points suivants choisissez « **thory** » (qui correspond au nom que l'on a donné à notre personnage).

Attention si votre personnage n'apparaît pas, cliquez sur l'œil à côté de thory (voir l'image ci-dessous) :

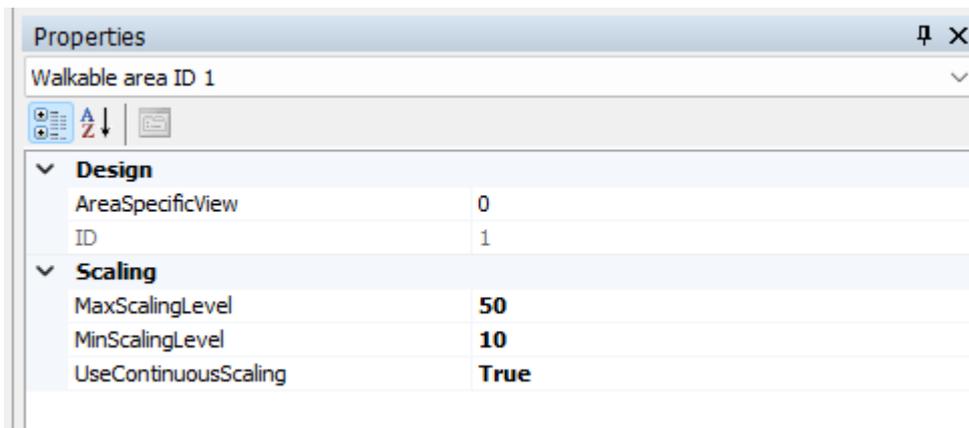


Comme vous pouvez le constater, le personnage « thor » est énorme, pour régler la taille du personnage, il faut :

Tout en haut dans « **editroom** », après « **room** » il faut cliquer sur les 3 petits points et choisissez : « **walkable areas** » puis dans les 3 petits points suivant choisir « **walkable area ID 1** »

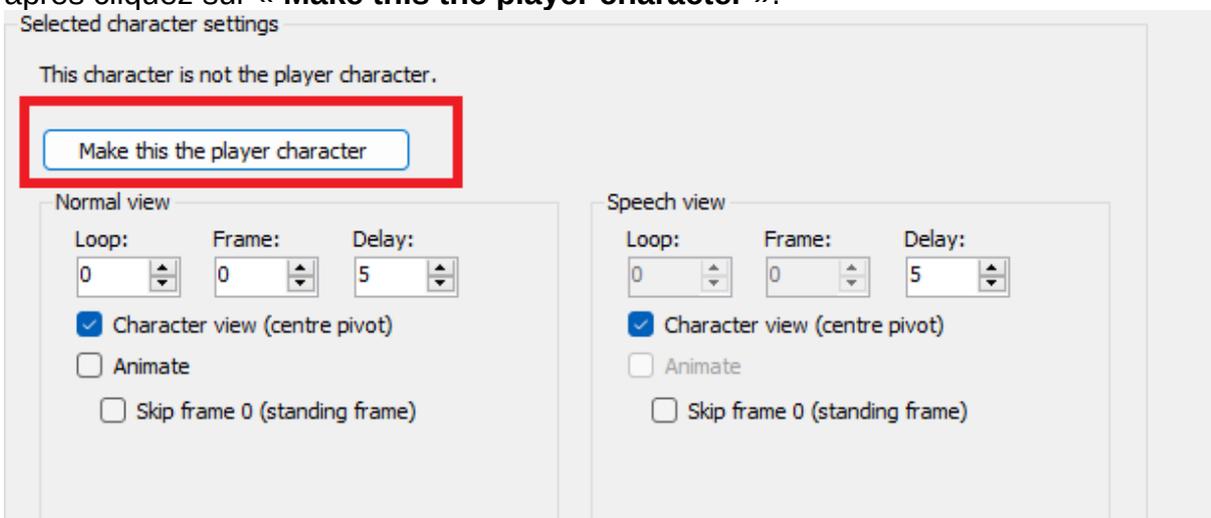


Puis à droite dans « propriétés » modifiez la valeur « **MaxScalingLevel** » ici à 50 - , cela correspond à l'échelle du personnage de près.  
Passez « **MaxScalingLevel** » ici à 10 - , cela correspond à l'échelle du personnage de loin.

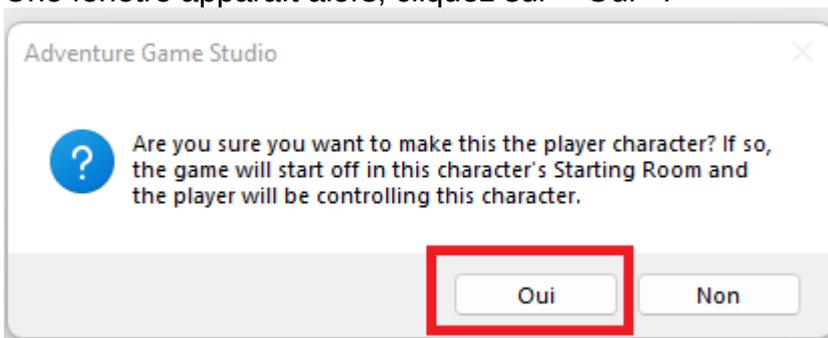


Il faut ensuite faire en sorte que votre personnage «thor » démarre dans la room que vous avez choisi :

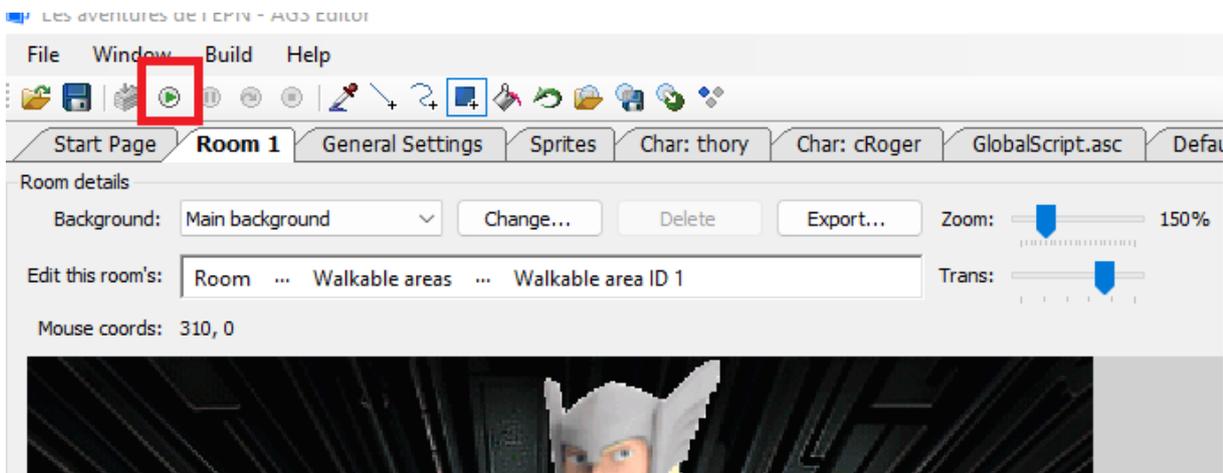
Retournez dans « l'arbre du projet », **cliquez** sur « **characters** » puis sur « **thory** » et après cliquez sur « **Make this the player character** ».



Une fenêtre apparaît alors, cliquez sur « Oui ».



Pour vous faire une idée de ce que donne votre jeu avec votre personnage, tout en haut cliquez sur la touche play :



Avec la souris, votre personnage va se déplacer sur les zones de marche que vous avez créées.

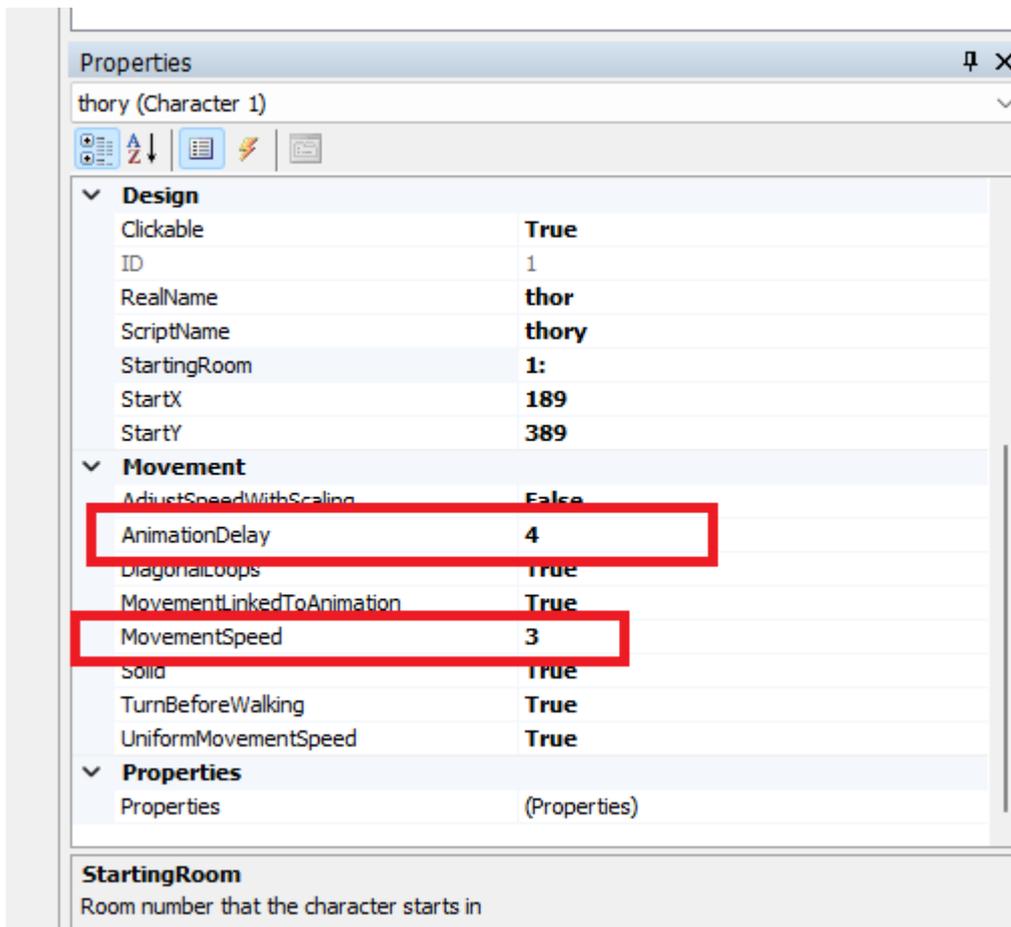
Si vous trouvez que votre personnage marche trop lentement, vous pouvez le faire accélérer :

Retournez dans « l'arbre du projet », cliquez sur « **Characters** », cliquez sur « **thory** » et regardez dans « **properties** ».

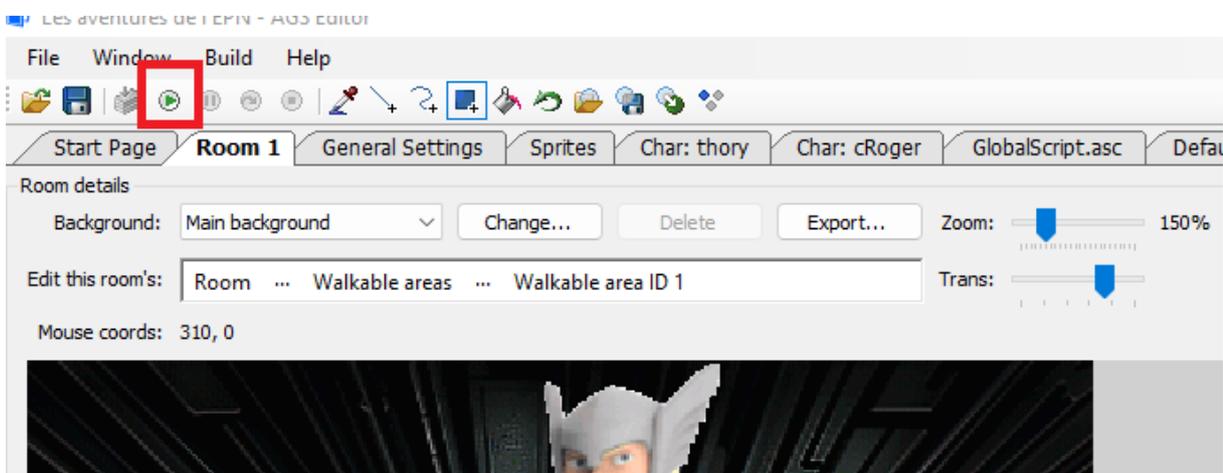
Dans « **animationDelay** » on va diminuer le délai entre 2 sprites (animations), on va mettre par exemple 3.

Et enfin on va augmenter « **MovementSpeed** » qui correspond à la vitesse de l'animation de marche, on va mettre par exemple 4.

Voir l'image ci-dessous



Vous pouvez lancer à nouveau le jeu, pour voir si cela vous convient. Pour mémoire, il suffit d'appuyer sur play :



# Interactions

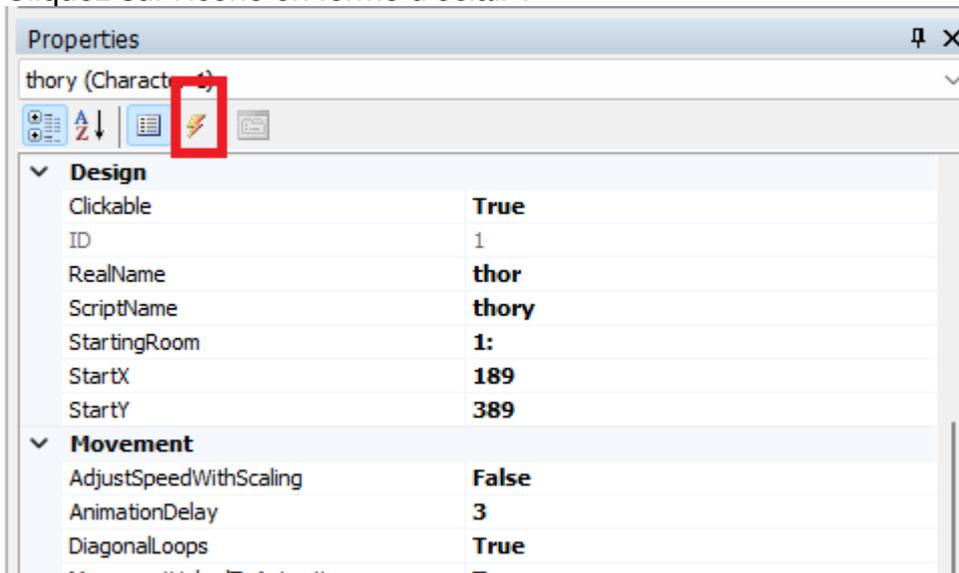
## Le personnage parle avec lui-même

Nous allons ici paramétrer ce que va dire le personnage principal si le joueur clique sur lui.

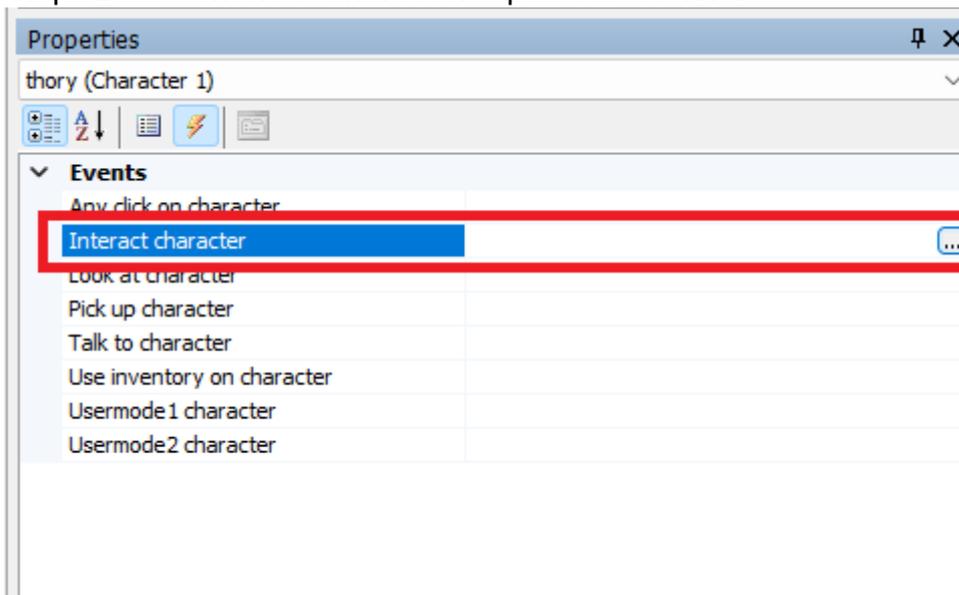
Retournez dans « l'arbre du projet », cliquez sur « **Characters** », sur « **thory** » et regardez dans « **properties** ».

Nous allons commencer à faire parler « thor » et plus exactement il va se parler à lui-même. Pour ce faire, on va utiliser la fonction « **interagir** ».

Cliquez sur l'icône en forme d'éclair :



Cliquez sur « **Interact character** » puis sur les ... :



Une fenêtre s'ouvre alors :

```

// no free slots found
return -1;
}

function btnSaveGame_OnClick(GUIControl *control, MouseButton button)
{
    int gameSlotToSaveInto = find_save_slot(txtNewSaveName.Text);

    if (gameSlotToSaveInto < 0)
    {
        Display("No more free save slots!");
    }
    else
    {
        SaveGameSlot(gameSlotToSaveInto, txtNewSaveName.Text);
        close_owning_gui(control);
    }
}

function btnRestoreGame_OnClick(GUIControl *control, MouseButton button)
{
    if (lstRestoreGamesList.SelectedIndex >= 0)
    {
        RestoreGameSlot(lstRestoreGamesList.SaveGameSlots[lstRestoreGamesList.SelectedIndex]);
    }

    close_owning_gui(control);
}

function lstSaveGamesList_OnSelectionCh(GUIControl *control)
{
    txtNewSaveName.Text = lstSaveGamesList.Items[lstSaveGamesList.SelectedIndex];
}

function txtNewSaveName_OnActivate(GUIControl *control)
{
    // pressing Return in the text box simulates clicking the save button
    btnSaveGame_OnClick(control, eMouseLeft);
}

function btnDeleteSave_OnClick(GUIControl *control, MouseButton button)
{
    if (lstSaveGamesList.SelectedIndex >= 0)
    {
        DeleteSaveSlot(lstSaveGamesList.SaveGameSlots[lstSaveGamesList.SelectedIndex]);
        lstSaveGamesList.FillSaveGameList();
    }
}

function thory_Interact()
{
}

```

En dessous de « **function thory\_Interact** on va écrire :  
« **Player.say** » et **mettre en mettre guillemet** le texte que l'on veut lui faire dire :

```

327 {
328 // pressing Return in the text box simulates clicking the save button
329 btnSaveGame_OnClick(control, eMouseLeft);
330 }
331
332 function btnDeleteSave_OnClick(GUIControl *control, MouseButton button)
333 {
334     SelectedIndex >= 0)
335     SaveGamesList.SaveGameSlots[1stSaveGamesList.SelectedIndex]);
336     SaveGameList ();
337     SaveGameList ();
338     player
339     PlayFlic
340     PlayVideo
341     Point
342     protected
343     player|
344 }
345

```

Tapez le début du mot « **player** », après **player** **mettre un point** puis tapez le début du mot « **say** » :

```

330 }
331
332 function btnDeleteSave_OnClick(GUIControl *control, MouseButton button)
333 {
334     SelectedIndex >= 0)
335     SaveGamesList.SaveGameSlots[1stSaveGamesList.SelectedIndex]);
336     SaveGameList ();
337     SaveGameList ();
338     RunInteraction
339     Room
340     Say
341     SayAt
342     player.s
343 }
344
345

```

Il suffit ensuite d'ajouter des parenthèse et de rédiger votre texte entre guillemets. Attention, n'oubliez pas le point-virgule après la dernière parenthèse.

## Le personnage interagit avec un objet

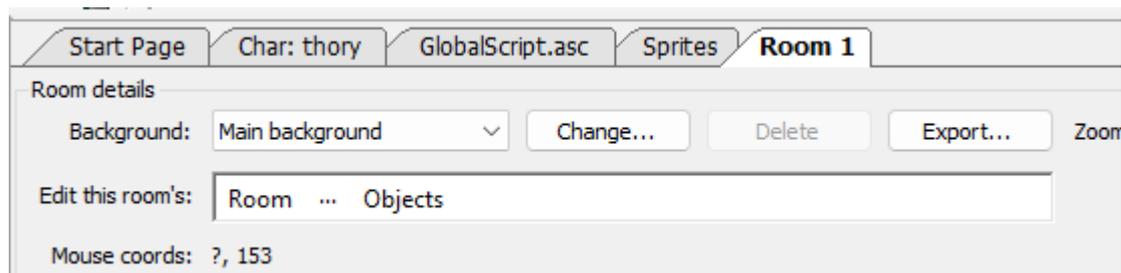
Retourner dans « l'arbre du projet » **double cliquez** sur « **Sprites** » puis sur « **Main** » ; faite un clic droit et sélectionnez « **createsub-folder** » qui signifie créer un sous-dossier. C'est à cet endroit que nous allons mettre les images d'animation d'objet.

Ce dossier se nomme : **New folder**, il faut lui donner un nom, ici se sera : **objet** et après dans la partie centrale (blanche), on va importer une image d'objet.

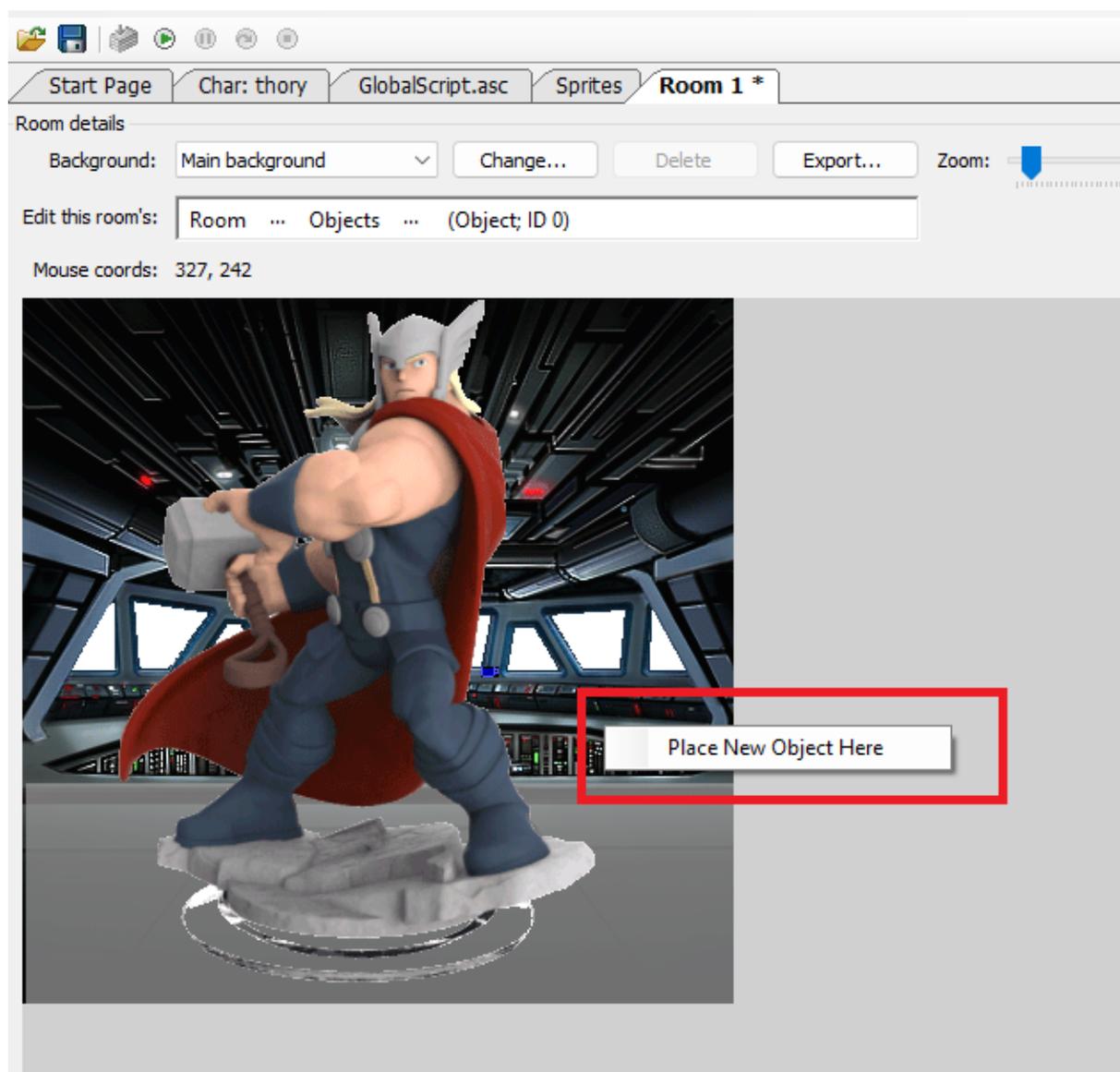
**Attention cet objet doit être plus petit que votre personnage, utilisez un logiciel comme Gimp pour réduire la taille de votre image si besoin.**

Ensuite Retourner dans « l'arbre du projet » **double cliquez** sur « **room** » puis sur « **Editroom** ».

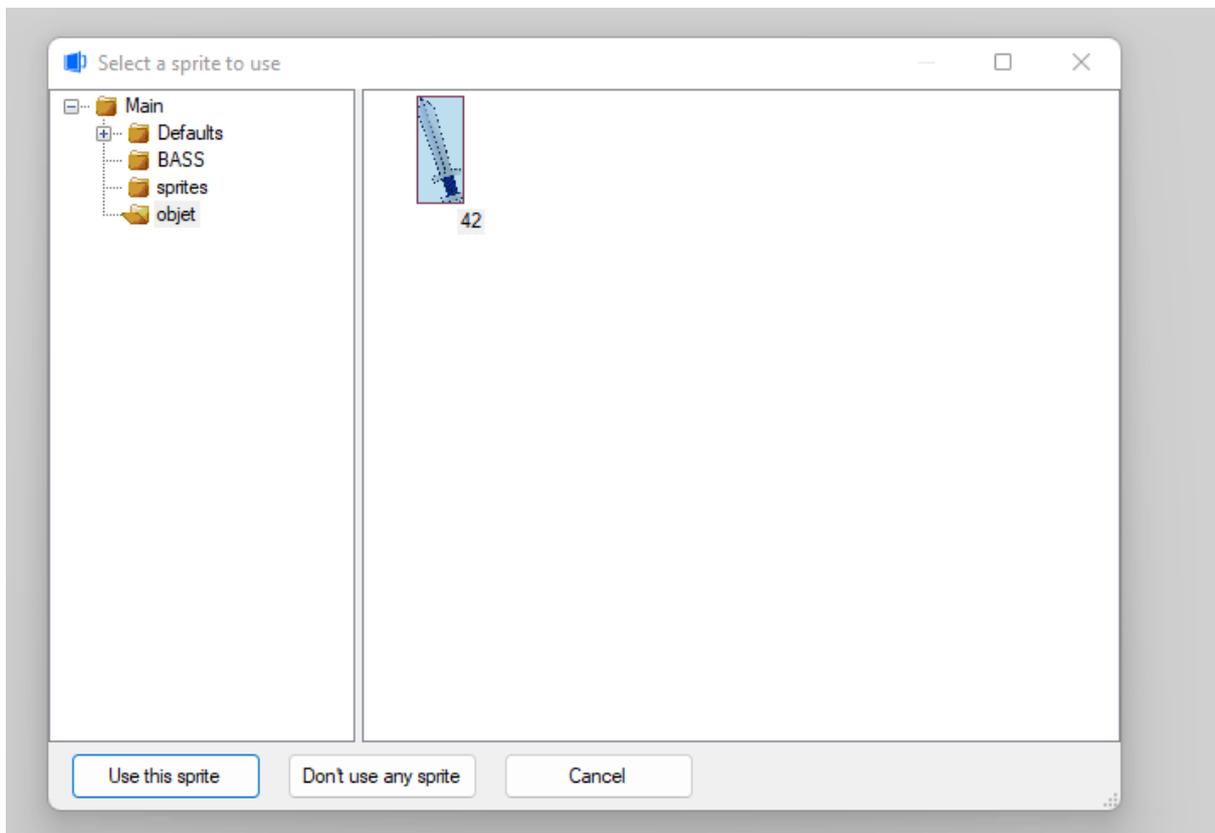
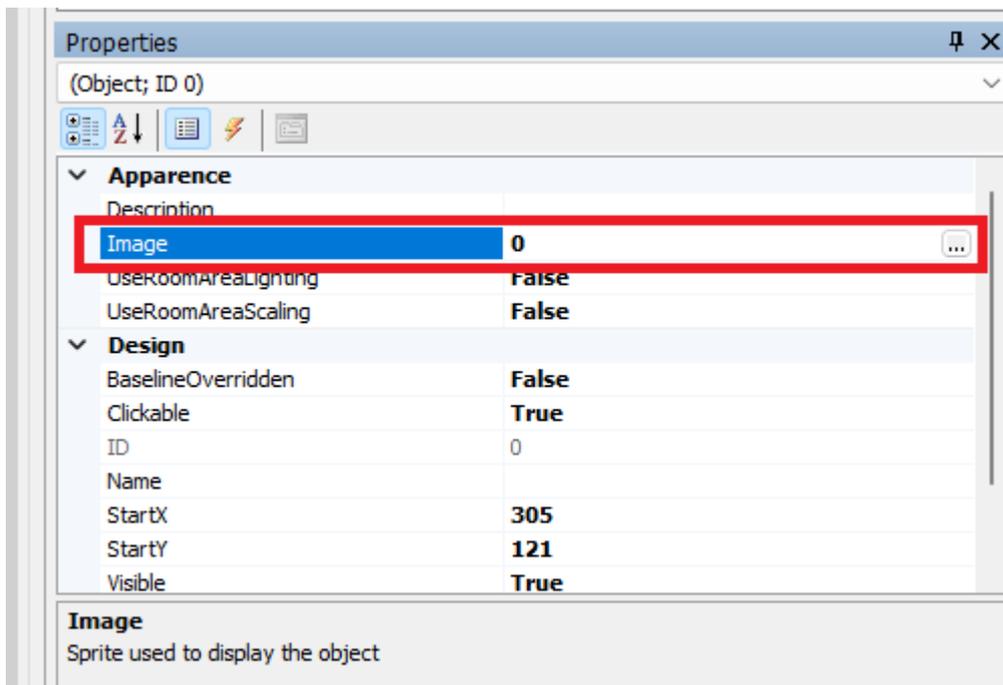
Dans « Edit this room's » (tout en haut) vous aurez « **room** » puis cliquez sur les 3 petits points et choisissez : « **objects** ».



Puis après faire un clic droit sur l'écran pour créer un nouvel objet.

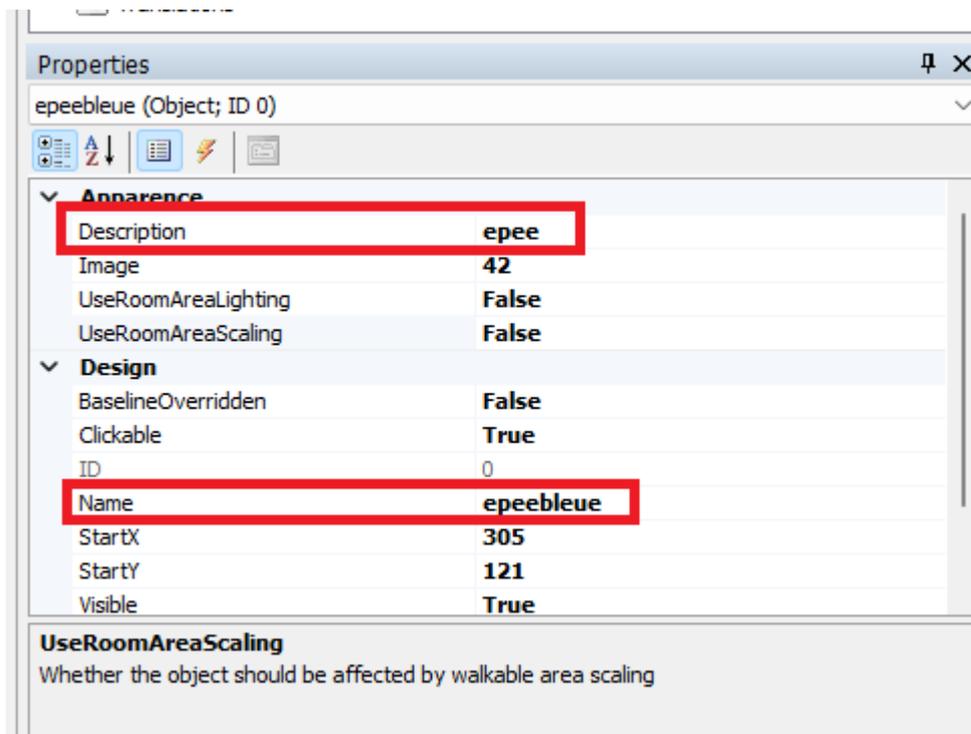


Puis sur le côté droit dans « **properties** » on va cliquer sur « **image** » puis les ... et on va sélectionner l'image de notre objet, dans notre cas une épée.



Puis cliquez sur « **use this sprite** ».

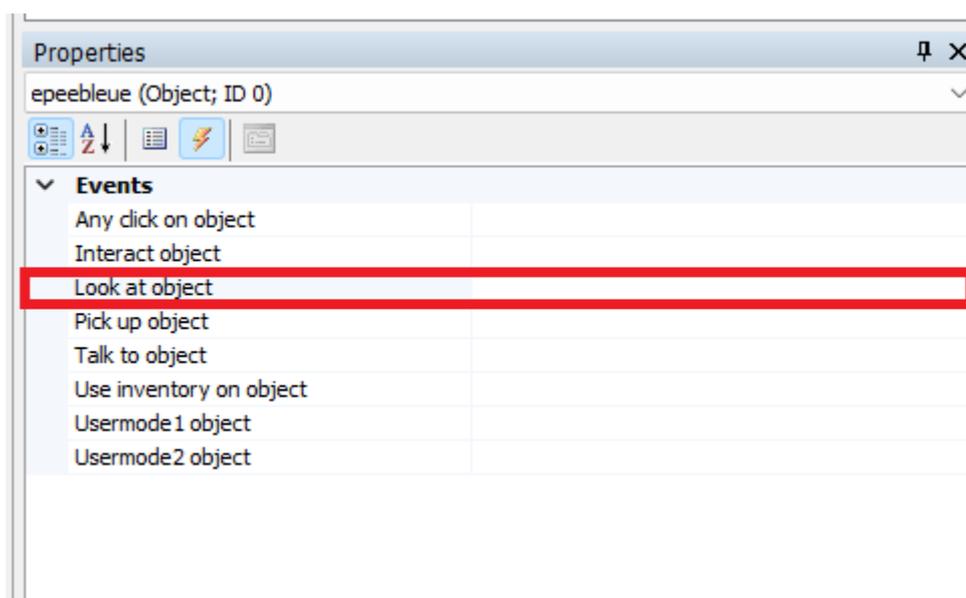
A côté de « **Name** » on va donner un nom à notre objet (par exemple epeebleue) – Ce nom correspond à celui par lequel il sera appelé dans le script au moment de programmer. On va ensuite à côté de « **description** » : donner un nom (par exemple épée), celui-ci va correspondre au nom qui apparaîtra pour les joueurs.



On va faire en sorte que thor interagisse lorsqu'il voit notre objet (épée), cliquez sur la fonction « **interagir** » (le petit éclair jaune).

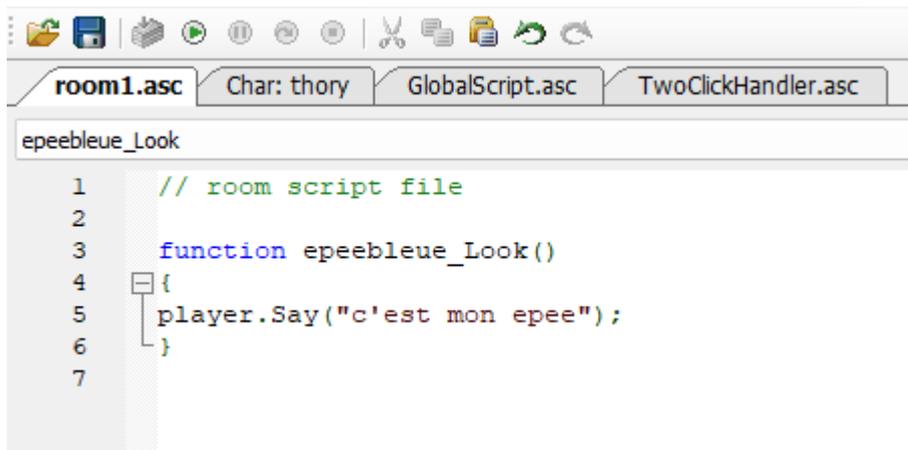


Puis cliquez sur « look at objet » puis sur les 3 petits points ...



Pour information le clic droit correspond à l'action de regarder.

Une fenêtre va alors s'ouvrir, Et on va faire dire à thor que c'est son épée.



```
room1.asc Char: thory GlobalScript.asc TwoClickHandler.asc
epeebleue_Look
1 // room script file
2
3 function epeebleue_Look()
4 {
5     player.Say("c'est mon epee");
6 }
7
```

## Le personnage ramasse un objet

Pour ramasser un objet et l'ajouter à l'inventaire du joueur, il faut retourner dans les fonctions d'interaction de la pièce (room), et cliquer sur le petit éclair jaune.



Choisissez ensuite « Interact object » et cliquez sur les trois petits points à droite, ce qui ouvrira le script.

Tapez ensuite la commande suivante :

```
function coffee_Interact ()
{
    sandrine.AddInventory(icoffee);
    coffee.Visible=false;
}
```

sandrine.AddInventory (icoffee) ;

Cette ligne ajoute l'objet « coffee » à l'inventaire du joueur (ici Sandrine). Le i placé avant le nom de l'objet signale qu'il s'agit du même objet, mais placé dans l'inventaire, pour ne pas le confondre avec l'objet placé dans la pièce, les deux peuvent avoir des propriétés différentes.

Remplacez « sandrine » par le nom de votre personnage, « icoffee » par le nom de votre objet, précédé d'un i.

coffee.Visible=false ;

Cette ligne fait disparaître l'objet de la pièce quand le joueur clique dessus et l'ajoute à son inventaire. C'est comme ça qu'il « ramasse » un objet.

Ici aussi remplacez « coffee » par le nom de votre objet.

Vous pouvez tester votre jeu, désormais lorsque le joueur cliquera sur l'objet celui-ci disparaître pour se placer dans l'inventaire en haut de l'écran !

## Ressources

Ce tutoriel a été fait à partir des ressources suivantes :

- <http://admin.no.uchi.free.fr/dokuwiki-2008-05-05/doku.php?id=debuter>
- [https://www.youtube.com/watch?v=9fuTnY3Z500&list=PLHEkYM\\_5NDNx1qH4EgE0Kh7K-KsjBAcbg](https://www.youtube.com/watch?v=9fuTnY3Z500&list=PLHEkYM_5NDNx1qH4EgE0Kh7K-KsjBAcbg)

N'hésitez pas à les consulter pour approfondir ce tutoriel.