|  |
| --- |
| 4205G4MO – Technologie émergentes des applications |
| Projet synthèse: Laboratoire |
| Collège Montmorency |

|  |
| --- |
| Soliman, Alexandre |

Ce laboratoire nécessite Godot pour être réalisé. Il est également recommandé de lire les notes documentées qui expliquent de nombreuses normes de conception des niveaux ainsi que les subtilités de l'intelligence artificielle des ennemis.

L'objectif n'est pas de simplement copier les instructions ci-dessous, mais de comprendre Godot suffisamment pour travailler sur vos propres projets, si vous le souhaitez.  
  
Commencer par installer Godot : <https://godotengine.org/download/windows/>

Une image contenant texte, capture d’écran, Police, Graphique

Description générée automatiquement

Choisissez cette option.

Décompressez le contenu du dossier téléchargé et enregistrez-le où vous le souhaitez.

Cliquer sur ce fichier.

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

Ensuite, cliquer sur create. Ce menu devrait s’afficher :

Assurez de placer votre projet dans le dossier laboratoire où se trouve ce document.

**Commençons à créer!**

Bien que vous soyez familier avec C++, Python, etc. Godot utilise son propre langage, GDScript.

Pour commencer, nous allons créer 3 dossiers appelés assets, scenes et scripts. Vous pouvez le faire en cliquant avec le bouton droit sur le dossier res situé en bas à gauche de votre application.

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

**Assets**

L'une des premières choses à faire est d'obtenir nos composants. Heureusement pour vous, ce laboratoire est accompagné d'une sélection de ressources conçues pour l’usage publique que vous pouvez utiliser dans le cadre de ce projet ou d'autres.

Pour l'instant, il suffit de faire glisser le contenu du dossier assets où se trouve ce laboratoire dans le dossier assets de votre projet.

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

**Fonctionnement**

Nous commencerons par créer nos nœuds. Comme indiqué dans les notes de cours, les nœuds englobent les menus, les personnages et sont en quelque sorte la base de tout jeu vidéo.

Comme il s'agit d'un jeu en 2D, commençons par créer un nœud racine en 2D

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquementEnsuite, renommez-le à Game

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquementEnsuite sauvegarder le avec Ctrl +S >aller à « scenes »

**Joueur 1**

Commençons par créer notre premier personnage.   
Créer une nouvelle scène.



Une image contenant texte, capture d’écran, Police, symbole

Description générée automatiquement

Ensuite, créer un nouveau nœud racine pour notre personnage.

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

« CharacterBody2D » est un corps spécialisé pour les personnages d’un script 2D.

Cependant, nous devons aussi donner une forme à ce personnage.

Créer un nouveau nœud (CTRL+A).

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Comme ce personnage va être contrôlé par un joueur, nous utiliseront « AnimatedSprite2D ».

Sélectionnez AnimatedSprite2D dans le menu à gauche. Ces informations seront affichées dans l’inspecteur à droite. Aller dans « Animation » et sélectionnez « New Sprite Frames »

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Ceci affichera au bas un menu d’animation.

Une image contenant Logiciel multimédia, logiciel, Logiciel de graphisme, texte

Description générée automatiquement

Une image contenant texte, Police, capture d’écran, nombre

Description générée automatiquement

Choisissez cette option ou faites la combinaison Ctrl+Shift+O.

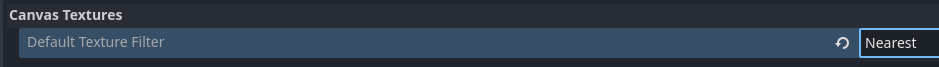
Allez ensuite dans Assets/Sprites. Choisissez knight.png, qui contient les animations de notre personnage principal.

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Modifier la grille pour qu'elle soit de 8x8

Ensuite, choisissez les 4 premières cases dans la rangée « idle ». Cliquez sur « Add 4 Frames ».

Une image contenant texte, Police, capture d’écran, Rectangle

Description générée automatiquementUne image contenant texte, Police, capture d’écran, conception

Description générée automatiquementVous remarquerez que notre bonhomme est un peu flou. Une image contenant capture d’écran, texte, Logiciel multimédia, Logiciel de graphisme

Description générée automatiquementVoici comment régler ceci :

Une image contenant pixel, capture d’écran, Caractère coloré, Graphique

Description générée automatiquementEt voilà! Regarder comment il est beau.

Cliquer sur ce bouton situé dans le menu d’animation.



Il bouge! Maintenant, renommez l’animation à idle et cliquez sur la flèche autoplay.

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Placer le sur la ligne rouge en utilisant les flèches de votre clavier.

Une image contenant capture d’écran, diagramme, pixel, ligne

Description générée automatiquement

Pour permettre au personnage d’interagir avec son entourage, créer une node CollisionShape2D. Cela donnera un «Une image contenant texte, capture d’écran, logiciel, Police

Description générée automatiquement

Ensuite, dans l’inspecteur, choisissez CircleShape.

Une image contenant texte, Police, capture d’écran, nombre

Description générée automatiquement

Une image contenant capture d’écran, diagramme, Graphique, Caractère coloré

Description générée automatiquement

Ajuster la taille pour qu’elle ressemble à ceci. Comme vous pouvez l’observer, le nœud de collision est plus petit que le modèle de notre personnage. Ceci est voulu! Un personnage avec une grosse boite de collision est très énervant à contrôler.

Une image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquement

Renommez-le a « Joueur ». Ensuite, sauvegardez-le.

Une image contenant texte, ordinateur, capture d’écran, logiciel

Description générée automatiquement

Une image contenant Police, texte, capture d’écran, nombre

Description générée automatiquementRetournez à game.

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Prenez le joueur.tcsn et glissez le dans l’espace game.

Ensuite, créer un nœud Camera2D et mettez le en 4x4 dans l’inspecteurUne image contenant texte, logiciel, Logiciel multimédia, capture d’écran

Description générée automatiquement

Retournez à joueur.



Cliquez sur ce bouton pour créer un nouveau script.



Godot possède un script déjà préconçu pour les mouvements de base. Choisissez-le.

Une image contenant texte, Police, capture d’écran, conception

Description générée automatiquement  
Assurez-vous de sauvegarder le tout dans scripts.

Appuyez sur create.

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Finalement, du code!

Mais nous ne toucherons pas à ça pour l’instant.

D’abord, il faut créer un espace pour que notre personnage ne tombe bas dans le vide.

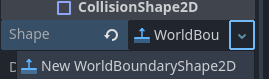
Retournez à game.   
Choisissez un StaticBody2D Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Et ensuite, un collisionShape2D pour celui-ci.

Une image contenant texte, Police, capture d’écran, nombre

Description générée automatiquement



Choisissez dans l’inspecteur un WorldBoundary.



Bougez-le avec cet outil et placez-le en dessous du perso.

Vous pouvez maintenant lancer l’application avec la flèche du haut à droite.

Vous apercevrez que le joueur bouge trop rapidement et qu’il saute très haut. Corrigez cela en modifiant ses attributs de vitesse et de vitesse de saut dans le script.

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquementLancer encore l’application. Vous verrez que notre personnage se déplace plus comme un personnage de jeu vidéo habituel.

Cependant, notre pauvre chevalier vit actuellement dans un espace de gris total. Donnons un peu de vie à son monde.

**Le Monde**

Supprimez le StaticBody, puisque nous allons maintenant donner au personnage de la « vrai » terre où se mettre debout.

Créer ensuite un TileMap Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Ensuite, créer un nouveau TileSet

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement



En bas, naviguer sur l’onglet TileSet.

Ensuite, glisser world\_tileset de assets/sprites dans l’espace tiles.

Une image contenant texte, capture d’écran, Logiciel multimédia, logiciel

Description générée automatiquement

Appuyez sur Yes.

Une image contenant capture d’écran, texte, pixel

Description générée automatiquementCeci devrait s’afficher.

Maintenant que nous avons no TileSets, on peut commencer la création du TileMap. Ceci dépend entièrement de vos goûts. Vous sélectionner une tuile dans le TileMap, et ensuite, dans l’espace game, vous pouvez dessiner avec celle-ci. Si vous commettez une erreur, vous pouvez revenir en arrière en faisant un clic droit sur la tuile que vous avez placé. Vous pouvez sélectionner plusieurs tuiles an gardant le clic durant la sélection.

Une image contenant capture d’écran

Description générée automatiquement

Vous devrez avoir quelque chose qui ressemble à ceci avec au moins une surface où le personnage peut se déplacer.

Cependant, si vous lancer l’application maintenant, vous verrez que le pauvre chevalier ne fait que tomber à travers le sol.

Pour changer ceci, cliquer sur l’onglet TileMap, et dans l’inspecteur, cliquer sur Physics Layer et ensuite Add element.

Maintenant, allez sur l’onglet TileSet, dans la page paint.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Choisissez Physics Layer 0.

Ensuite, sélectionner toutes tuiles qui correspond à du sol. Éviter de sélectionner les arbres ou les champignons si vous ne voulez pas qu’ils servent d’obstacle. Si vous voulez annuler, cliquez sur C et après sélectionner les tuiles, pour retourner à la peinture de collisions, cliquez sur F.

Une image contenant capture d’écran, diagramme, pixel, conception

Description générée automatiquement

Vous pouvez aussi choisir la taille de la collision dans l’image associée à la tuile. N’oubliez pas de cliquer sur Ctrl+S après chaque tuile.

Si vous lancez l’application maintenant, vous verrez que le chevalier peut maintenant marcher sur le sol. Malheureusement, vous constaterez aussi que la caméra ne bouge pas avec lui, donc il est bloqué dans l’espace original qu’on lui a accordé. Pour changer ceci, cliquer sur Camera2D, glissez-le comme attribut dans la scène joueur.

Une image contenant texte, Police, capture d’écran, nombre

Description générée automatiquement

Et ensuite effectuez ceci dans l’inspecteur. Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

La caméra devrait maintenant suivre notre chevalier. Liberté!  
Cependant, notre chevalier vit dans un monde très…calme. C’est un chevalier, il doit avoir la vie plus difficile.

**Les ennemis…et la mort**

Avant d’ajouter des ennemis, il faut s’assurer que notre personnage peut mourir. Sinon, les ennemis ne servent à rien. Cela permet aussi de réinitialiser la position du chevalier s’il commet l’erreur de tomber dans le vide.

Une image contenant texte, capture d’écran

Description générée automatiquementPour commencer, nous allons limiter notre caméra. En cliquant sur R, vous pouvez mesurer le font de votre niveau.

Une image contenant texte, capture d’écran

Description générée automatiquementEnsuite, aller dans l’inspecteur de la caméra pour choisir votre limite de fond.

Créer une nouvelle scène avec Area2D comme nœud racine.Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Une image contenant texte, capture d’écran, Appareil électronique, nombre

Description générée automatiquement

Changer le Collision Mask de ce nœud à 2.

Une image contenant texte, capture d’écran, Police, conception

Description générée automatiquement

Renommer le nœud à killzone, et sauvegarder le dans le dossier scenes.

Téléversez le nœud killzone dans la scène game. Ensuite, ajoutez un CollisionShape2D à celui-ci.

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Placez dans le fond de votre niveau, assurez vous qu’on ne peut y accéder quand tombant. Une image contenant capture d’écran, Logiciel de jeu vidéo

Description générée automatiquement

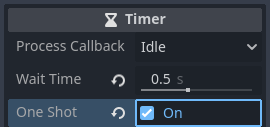
Retourner à la scène killzone, et ajouter un script comme-ci : Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Ajoutez ensuite un nœud timer à la scène killzone.

Une image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquement



Une image contenant texte, capture d’écran, logiciel, Police

Description générée automatiquement

Copier ce code dans votre script. La méthode -on\_body\_entered déclanche le timer, qui ensuite déclanche la méthode \_on\_timer\_timeout quand il fini son déconte. Celle-ci reinitialise le niveau.

Maintenant, ajoutons une nouvelle scène pour l’ennemi. C’est un processus très semblable a celui du joueur.

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Choisissez l’image slime\_green

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Une image contenant capture d’écran, texte, Graphique, graphisme

Description générée automatiquement

Puisque c’est un monstre nocif, ajoutez la scène killzone comme paramètre.

Ensuite, ajouter un CollisionShape2D en mode rectangle à celle-ci

Une image contenant pixel, diagramme, capture d’écran, Caractère coloré

Description générée automatiquement

Ajouter le à la scène game. Vous verrez maintenant que quand notre personnage le touche, il meurt.

Une image contenant capture d’écran

Description générée automatiquement

Pour avoir une animation de mort, copier coller ce code dans le script killzone.gd

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Ceci ralenti le jeu et enlève la collision, qui fait tomber dans le vide notre chevalier quand il entre en contact avec le monstre.

Et voilà! Vous avez créer un jeu de base, qui possède un joueur, un ennemi et des interacton entre les deux.