

TP3 Gestion du Son

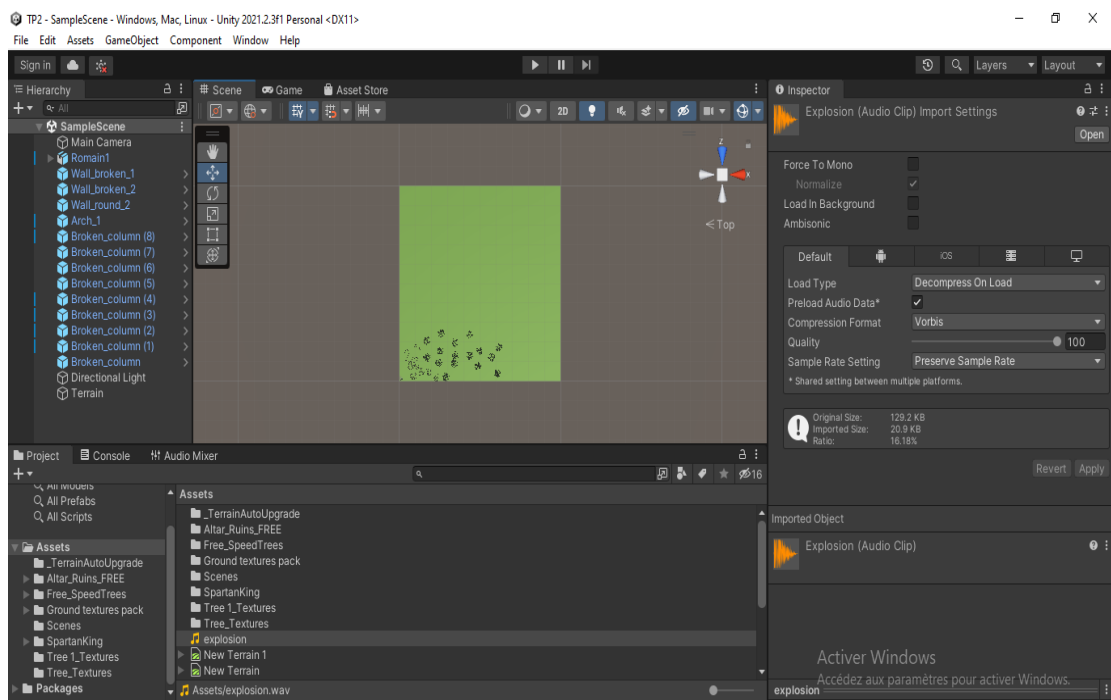
L'objectif du TP est d'ajouter du son à la scène virtuelle.

Vous pouvez télécharger Unity et Unity hub via les liens suivant : <https://unity.com/fr/download>

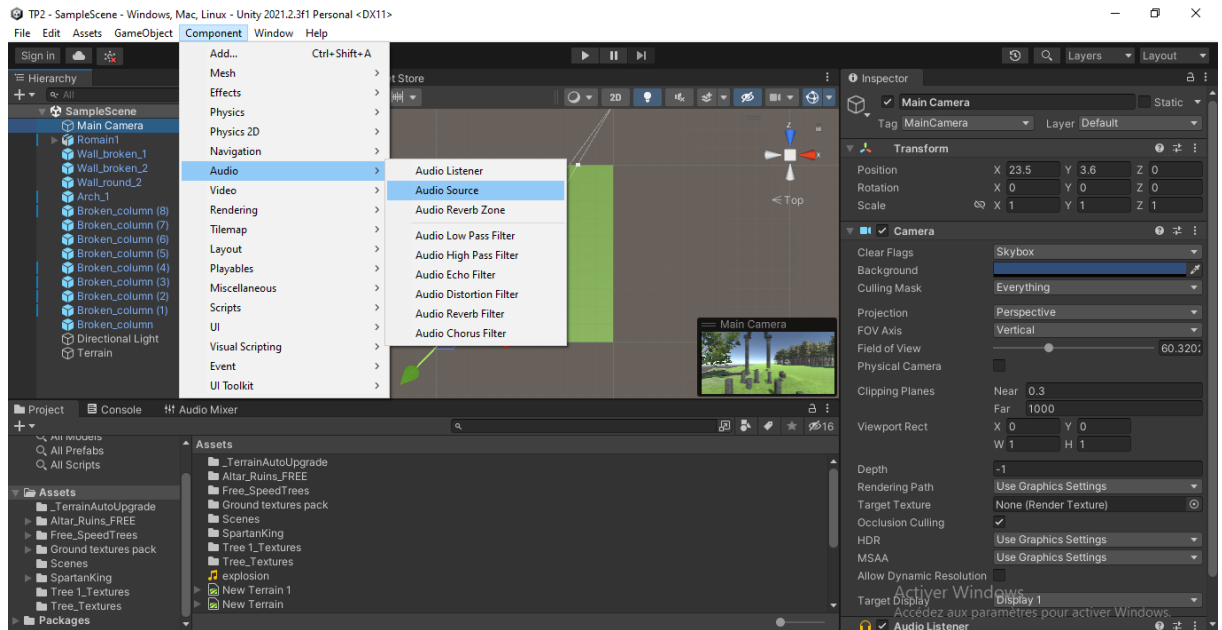
Télécharger les Assets via l'Asset Store <https://assetstore.unity.com/>

Exercice 1 : Ajoutez un son de fond

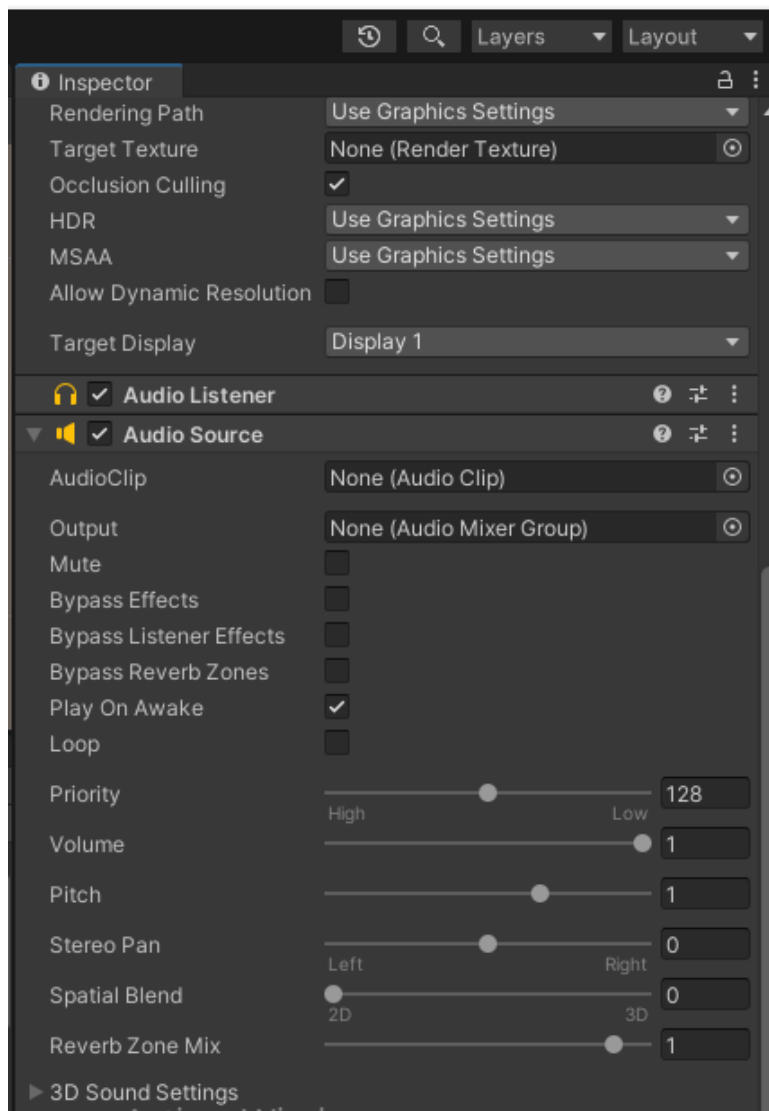
1. Démarrez un nouveau projet en sélectionnant 3D comme modèle de projet.
2. Placez vos fichiers sonores dans votre répertoire Assets, vous pouvez y créer un sous répertoire Son. Les fichiers sonores acceptés sur Unity sont : **.aif .wav .mp3 .ogg**



3. Ajoutez un Audio a votre camera. **Sélectionnez la caméra ->cliquez sur Component -> Audio->Audio Source**



4. Ajustez les paramètres de votre Audio



AudioClip: sélectionnez un audio parmi ceux déposées dans votre répertoire Assets/.

Output: Par défaut le son sera dirigé vers l'Audio Listener de la scène. Vous pouvez également diriger le son vers un audio mixer.

Mute : Permet de mettre l'Audio Source en sourdine. S'il y a d'autres Audio sources dans la scène, elles continueront de jouer. Ex. GetComponent().mute = {true ou false}

Play On Awake: Si coché, le son sera lu dès le lancement de la scène ou dès que le gameObject est activé. S'il est décoché, le son sera en attente et pourra être déclenché avec la commande

Play(). Ex. GetComponent().Play()

Loop: Détermine si le son joue en boucle ou non.

Priority : Détermine la priorité de cette Audio Source. 0 = Plus important, 256 = le moins important, 128 = valeur par défaut

Volume: Contrôle le volume du son (0 à 1).

Pitch: Contrôle la vitesse de lecture du son. (Entre -3 et 3)

Ex. 1 = vitesse normale, 2 = double de la vitesse normale, 0.5 = la moitié de la vitesse normale, etc.

Stereo Pan : Permet de définir la spatialisation stéréo du son (effet gauche/droite). (Entre -1 et 1.) Ex. 0 = son centré, -1 = seulement à gauche, 1 = seulement à droite, 0.75 = 75% à droite et 25% à gauche, etc.

Spatial blend : Entre 0 et 1. Permet de définir s'il s'agit d'un son 2D (valeur de 0) ou 3D (valeur de 1). Une valeur intermédiaire signifiera que le son sera en partie géré comme son 2D et comme son 3D.

5. Exercice 2 : Jouer le son suite à une action de l'utilisateur

Modifier le script c# de sorte à :

- Déclarer les variables de type AudioSource et AudioClip.
`AudioSource source;`
`public AudioClip son;`
- Récupérer le component AudioSource,
`source = gameObject.GetComponent<AudioSource>();`
- Lancer l'audio quand le joueur appuie sur la touche du clavier.
- `if (Input.GetKey(inputShoot))`
- `{source.PlayOneShot(son); }`
- Renseignez les variables ajoutée (son, InputShoot) dans l'inspecteur d'objet en sélectionnant un audio parmi ceux dans votre Assets, et choisir la touche du clavier.

```
public float rotateV; //vitesse de déplacement rotation
//Direction du Déplacement
public string InputFront;
public string InputBack;
public string InputLeft;
public string InputRight;
public string InputShoot;
//
AudioSource asd;
public AudioClip son;

// Start is called before the first frame update
// Message Unity 5.0 references
void Start() {
    animations = gameObject.GetComponent<Animation>();
    asd = gameObject.GetComponent<AudioSource>();
}

// Update is called once per frame
// Message Unity 5.0 references
void Update()
{
    //marche avant
    if (Input.GetKey(InputFront))...
    //marche arriere
    if (Input.GetKey(InputBack))...
    //tourner a gauche
    if (Input.GetKey(InputLeft))...
    //tourner a droite
    if (Input.GetKey(InputRight))...
    if (Input.GetKey(InputShoot))
    {
        asd.PlayOneShot(son);
    }
}
```

