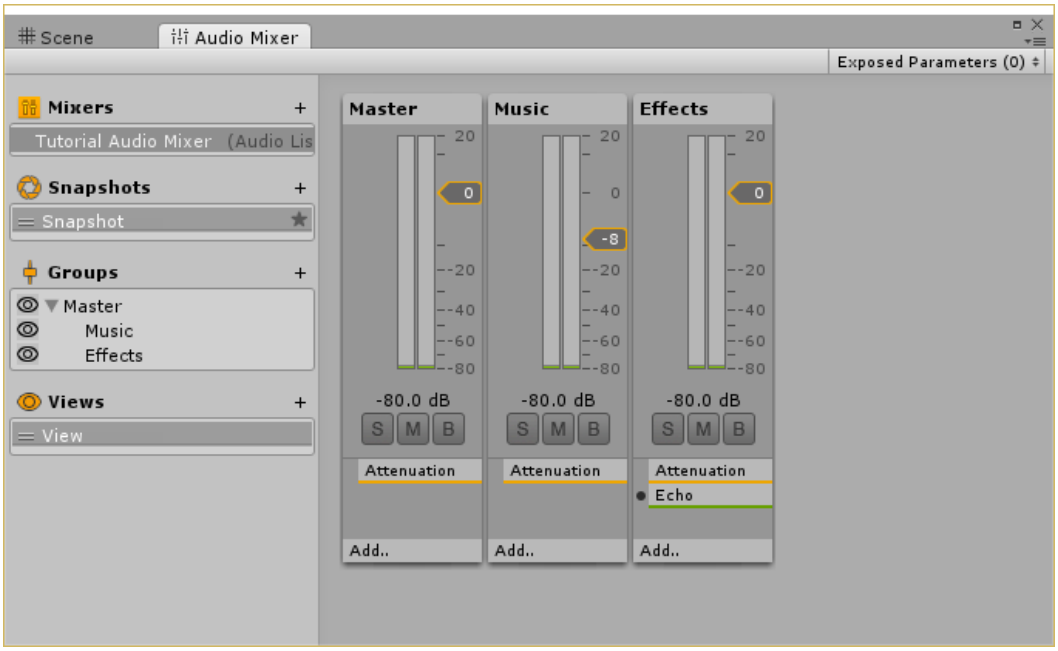
Tutorial – Audio in Unity

In this Tutorial we'll take a look at the audio functionality built into Unity. We'll start with a template project and add background music and a triggered sound effect.



Activity 0 - Prepare our project:

Download and unzip the starter project. It is available on the portal as "Audio - Tutorial Project - Starter.zip".

Also download and unzip the file "Audio - Tutorial Assets.zip". This contains the sounds we will be using in the tutorial.

Once the files are downloaded and unzipped, open the "Audio Tutorial" scene in the Unity project.

Activity 1 - Add a Character Controller:

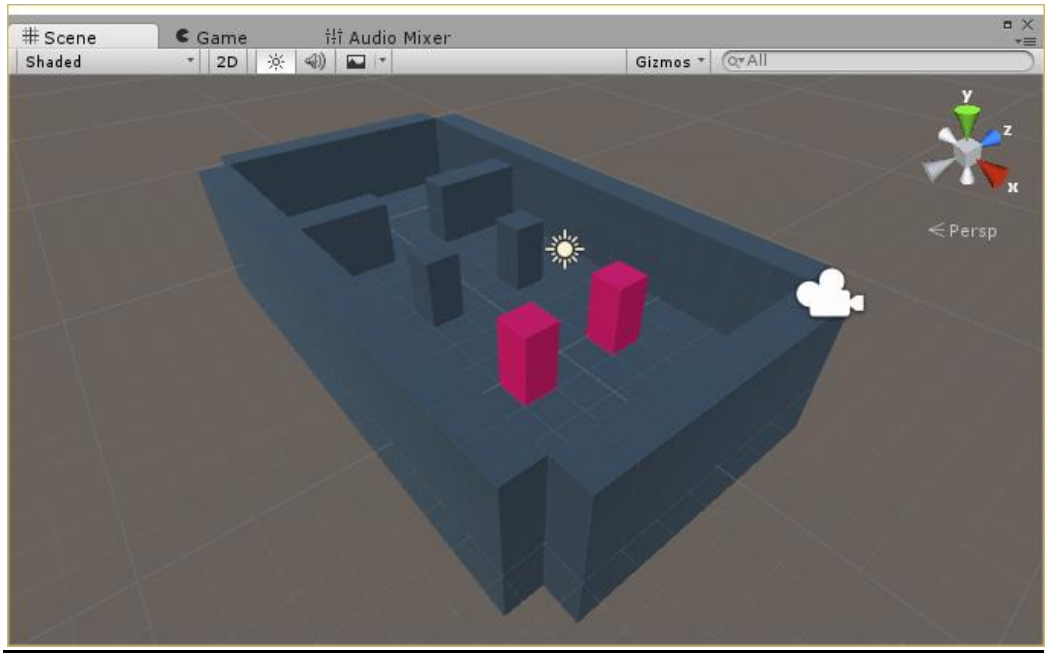
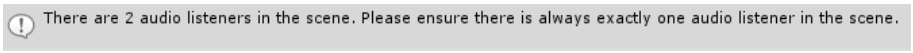


Figure 1: Our starting scene.

The first thing we'll need is some way to move around in our scene. Lets add a pre-built character controller, which will also act as our "listener" within the scene.

1. Find the "FPSController" asset in the Project view and drag it into the scene. Place it somewhere near the pink columns, and make sure that it is above the floor.
2. The "FPSController" GameObject has a child called "FirstPersonCharacter". Select this in the Hierarchy window and look at the Inspector. You will see that it has an AudioListener component attached. To play audio you need to have exactly one of these in the scene.
3. Play the scene. You'll notice that you get a message in the Console:

   This is because the "Main Camera" included in scenes by default also has an AudioListener attached. Almost anything that includes a Camera also includes an AudioListener by default, and we always need to remember to remove the ones we don't need.
4. In this case we don't need "Main Camera" at all, so delete it's GameObject from your scene. If you did still want to use the Camera you would remove the AudioListener component instead of deleting the GameObject.
5. Play the scene again. You'll notice that there are no messages in the console now. We're ready to start adding sounds to our scene.

Activity 2 - Background music:

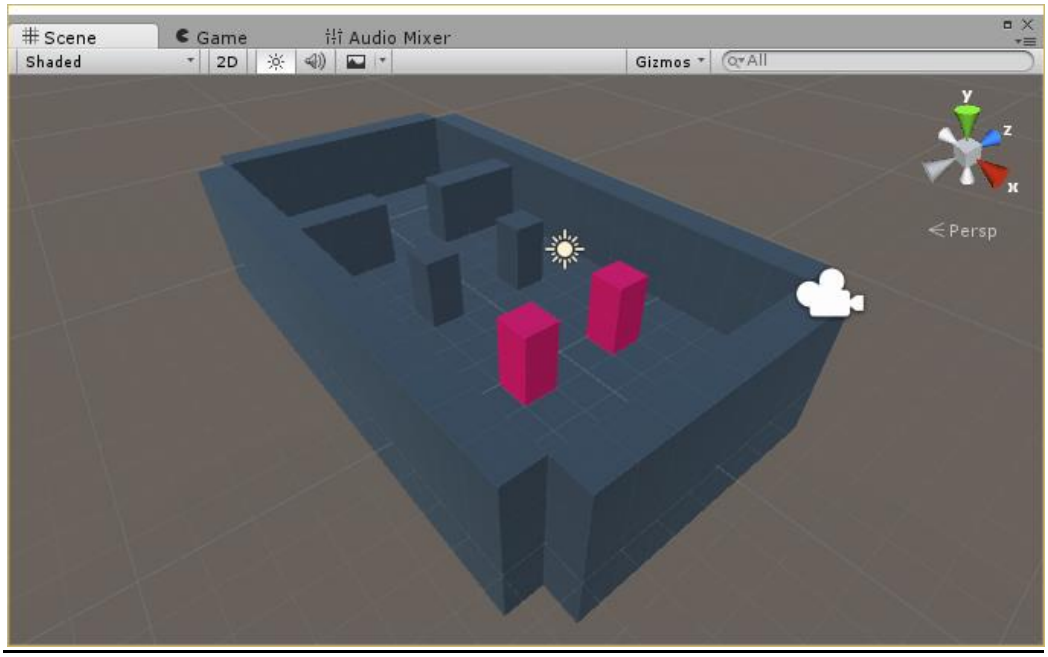
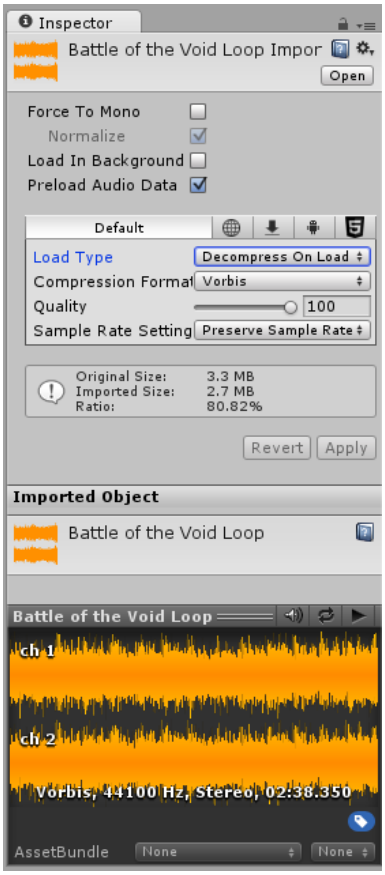


Figure 2: Our scene with added FPSController.

Next we'll add some music to our scene.

1. Before we can add the music to our Scene we must first add it to our Project. In your Project window, create a folder called "Audio" and inside that a folder called "Music". Inside the "Audio - Tutorial Assets" folder you unzipped you will find a file called "Battle of the Void Loop.ogg". Import this into your "Music" folder. This may take a few moments to process.
2. Note that in the "Audio - Tutorial Assets" folder there is also a text file called "Attributions.txt". Take a look at its contents, then drag this into your "Audio" folder. It's always a good idea to keep records of where you got any 3rd party assets you're using. Also note the licensing terms of any assets you use, and make sure you abide by them.

3. Select the "Battle of the Void Loop" file in your Project view and take a look at the Inspector.



"Battle of the Void" is quite long, as it is a music track. We don't want to keep the whole track in memory at once, so change the "Load Type" to "Streaming" and click "Apply". The other default options are all ok, so we don't need to change anything else.

4. Create an Audio Source in your scene by selecting GameObject -> Audio -> Audio Source. Rename the GameObject this creates to something like "Background Music".
5. With your "Background Music" GameObject selected look at the Audio Source properties in the Inspector.
    - For the "AudioClip" property select the "Battle of the Void Loop" clip we imported.
    - We want the music to start when the Scene is loaded, so ensure that "Play On Awake" is ticked.
    - Background music shouldn't stop when the track is finished, so we want it to loop. Make sure that "Loop" is ticked.
    - Ensure that "Spatial Blend" is set to 2D. This property is actually a number between 0 and 1, as sound can be blended between 2D and 3D. To make a sound fully 2D set the number to 0.
6. Make sure that your speakers or headphones are plugged in and turned on as necessary. Also make sure that their volume is relatively low! It's hard to tell how loud the first sound you add to a scene will be, so it's best to start with low volume on your speakers or headphones and ease it up to a comfortable listening level after you've got some sounds playing.

7. Play your scene. You should hear the music playing. If you wait until the end of the music track it should seamlessly loop back to the start and continue playing.
8. With the scene still playing, walk around. Notice that the sound of the music does not change based on your movement.
9. With your "Background Music" GameObject selected change the "Spatial Blend" property to fully 3D. Then place the GameObject near one of the pink columns.
10. Play the scene and walk around again. This time you will notice that the sound does change based on your movement. The volume will decrease as you move away, and it will pan between speakers are you look in different directions. This usually isn't very useful for background music, but it's very useful for sound effects.
11. With your "Background Music" GameObject selected open the "3D Sounds Settings" section at the bottom of the "Audio Source" Inspector. The properties you see here determine how position and movement effect the sound from this Audio Source.
12. Set "Spatial Blend" back to fully 2D.

Activity 3 - Add a sound effect for an event:

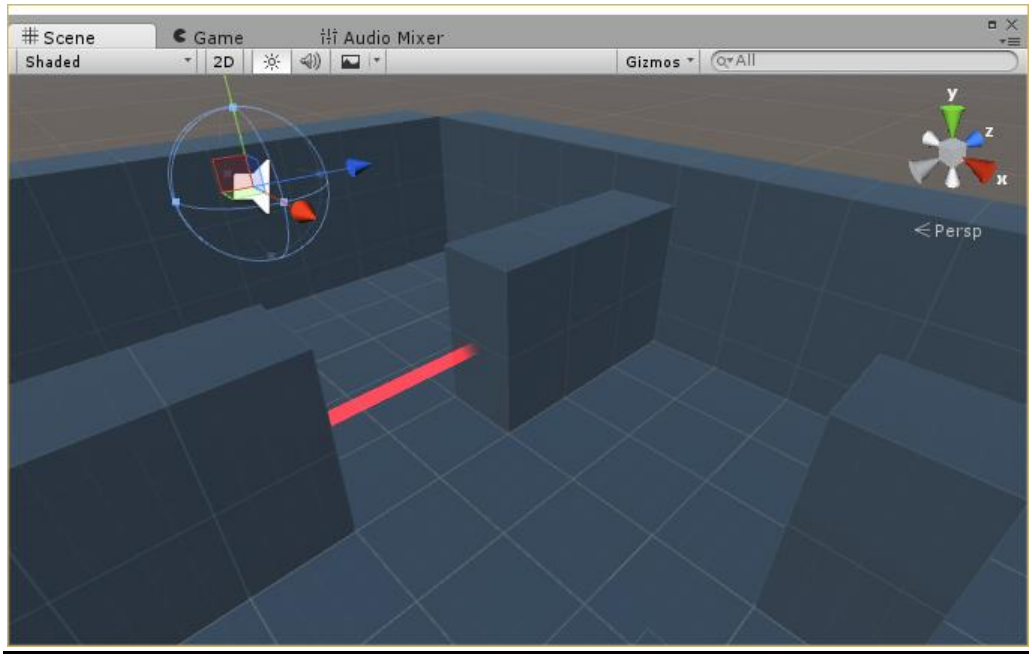


Figure 3: An alarm with a trip beam.

Music is relatively straightforward, because in simple cases we can just add the Audio Source to the scene and let it go. We often want sound effects to play in response to something happening, though.

1. Inside your "Audio" folder make another folder called "Effects". Inside the "Audio - Tutorial Assets" folder you unzipped you will find a file called "alarm.ogg". Import this into your "Effects" folder.
2. This effect is very short, so in the Import Settings we'll leave its "Load Type" set to "Decompress on Load". This uses additional memory, but has the least performance overhead to play.

3. Create another AudioSource in your scene and place it somewhere near the back of our level, in the room at the back. Name its GameObject something like "Alarm".
4. With your "Alarm" GameObject selected look at the Audio Source properties in the Inspector.
    - For the "AudioClip" property select the "alarm" clip we imported.
    - Make sure that "Play On Awake" is unticked. We only want this sound to play when we tell it to.
    - Make sure that "Loop" is unticked. We only want this sound to play once when we tell it to, then stop.
    - Ensure that "Spatial Blend" is set to 3D.
5. If you play the scene now your "Alarm" won't make any sound. This is because the Audio Source is not playing.
6. Make a laser beam which we can use to set off the alarm.
    - GameObject -> 3D Object -> Cube.
    - Name it something like "Alarm Beam".
    - Set its Box Collider's "Is Trigger" property is ticked.
    - Place it in the gap in the wall and scale it to a beam, as shown in the image above.
    - Optionally, give it a custom material.
7. Create a new script called "AlarmAudioTrigger".
    - Add a public AudioSource field called "m_audioSource".
    - Remove the "Start()" and "Update()" methods, as we don't need them.
    - Add an "OnTriggerEnter(...)" method to the AlarmAudioTrigger script. See the Scripting API reference for details.
    - In your "OnTriggerEnter(...)" method, call the "Play()" method on "m_audioSource" if it is not null. If it is null, log an error message instead.

    Make sure that this script compiles correctly before continuing.
8. Add the "AlarmAudioTrigger" component to your "Alarm Beam" GameObject. In its Inspector, for the "Audio Source" property select your "Alarm" GameObject's Audio Source.
9. Play the scene. When you walk through the beam the alarm sound effect should play.

Activity 4 - A simple mixer configuration:

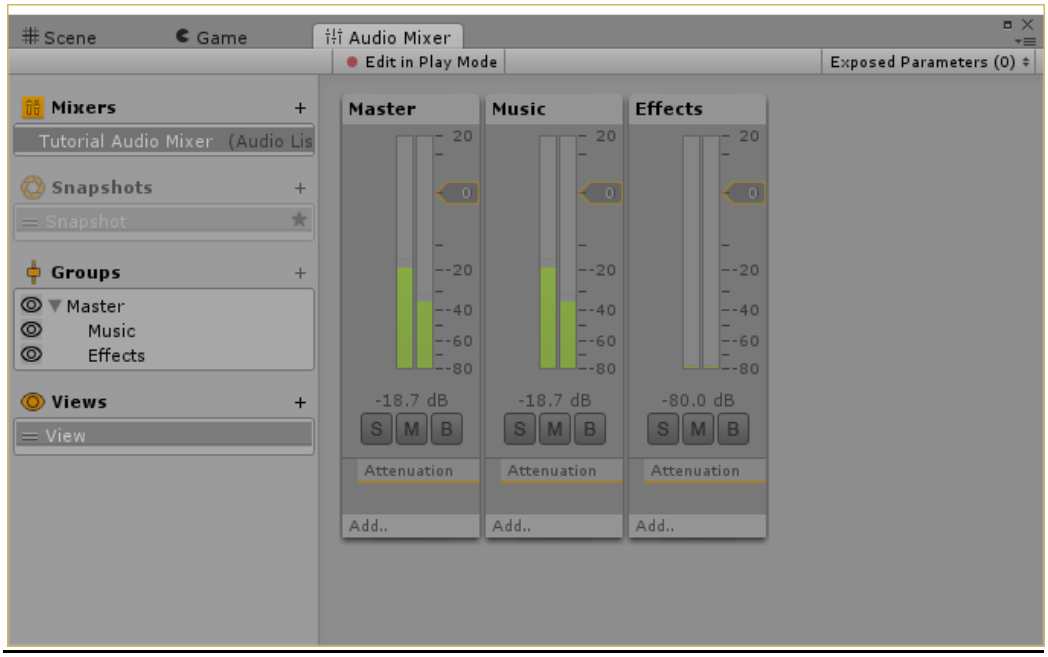


Figure 4: Our simple mixer configuration.

When putting audio in your game it's important to make sure that all of the different sounds are of appropriate volumes relative to one another. This is called "balancing". For example, the music needs to be loud enough to be clear without being so loud that it drowns out your sound effects. Also, important sound effects, such as instructional dialog, should be clearly audible above other sounds that may be playing at the same time.

While it's possible to do this by modifying the volume property of every Audio Source in your game it is much easier if we take advantage of the built in "Audio Mixer" system. We'll take a very quick look at this now.

1. Create an Audio Mixer resource by selecting Assets -> Create -> Audio Mixer. This will create a new item in your Project window. Name this something like "Tutorial Audio Mixer".
2. Double-click on your Audio Mixer resource to open the "Audio Mixer" Editor tab. Depending on where this loads you may want to re-dock it to be more easily used.
3. By default your Audio Mixer will have one Group called "Master". Groups are used to organise Audio Sources together. Audio Sources are assigned to a single Group, and any settings applied to that Group are applied to all of its Audio Sources. So, if you lower the volume of a Group you also lower the volume of all of its Audio Sources.
4. Using the "+" button to the right of the Groups heading, add two new Groups to your mixer. Call one "Music" and the other "Effects". Note that Groups can be nested within one another. Make sure that your new Groups are children of the "Master" Group and are not nested within one another - you can drag them around just like GameObjects in the Hierarchy window.
5. In your Scene, select your "Background Music" GameObject. In its Audio Source Inspector set the "Output" property to your "Music" Group.

6. Select your "Alarm" GameObject. In its Audio Source Inspector set the "Output" property to your "Effects" group.
7. Before continuing, make sure that you can see both your Game window and your Audio Mixer window at the same time.
8. Play your scene. In your Audio Mixer window you should see activity in the "Master" and "Music" groups. When you walk through the "Alarm Beam" you should also see activity in the "Effects" Group. What you see in the "Master" group is the sum of all of its child groups.
9. To balance your game's audio you can now change the volume of Groups rather than individual Audio Sources. Right now our music is quite loud, so the alarm sound isn't always clear. In the mixer, drag the volume tag for the "Music" Group down until its value is -8 and walk through the alarm beam again. Note that these values are decibels, so a small change can make a large difference.
    - If the Scene is playing you will first have to click "Edit in Play Mode" at the top of the window. This is because since the Audio Mixer is not a part of a Scene its settings are not reset when you exit Play mode.
10. As well as volume you can also add various effects to your Groups. Try this by pressing the "Add..." button at the bottom of your "Effects" Group and then selecting "Echo", then walk through the "Alarm Beam" again.
11. Many effects have properties which can be modified in the Inspector when their Group is selected. Experiment with the properties on the Echo effect on your "Effects" Group.
12. Try adding the "FPSController" GameObject's AudioSource to your "Effects" Group as well. What do you think was happening with its audio when it didn't have an Output Group set?

The Audio Mixer functionality is very useful when you have large scenes with many Audio Sources used for different purposes. When you make scenes with complex audio also be sure to check out the "Snapshot" functionality, which lets you set up multiple configurations for your Audio Mixer and switch between them while playing your game. For example, this allows you to easily have different balancing and effects during dialog than during exploration or combat.