

# Game Programming

## Lecture XII

### Game Audio

Izzet Fatih Senturk

# Background Music

- Create an empty game object
  - Rename as Audio\_Manager
- Create an empty game object within the Audio\_manager dedicated to background music
  - Rename as Background
  - Add a component: Audio Source
- Set audio source
  - Set AudioClip: music\_background from the Audio folder
  - Set Play on Awake and Loop true
- Save and play

# Laser Shot Sound

- Play a sound effect when the Player fires laser
- Update the FireLaser method in the Player script
  - When should we play the audio, before or after the laser visual?
- Light is faster than sound
  - Play the audio after the shooting visual effect

```
void FireLaser()
{
    _nextFire = Time.time + _fireRate;

    if (_isTripleShotActive == true)
    {
        Instantiate(_tripleShotPrefab, transform.position, Quaternion.identity);
    }
    else
    {
        Instantiate(_laserPrefab, transform.position, Quaternion.identity);
    }
    // Play the laser audio clip
}
```

# Laser Shot Sound

- Select the Player on the Hierarchy window and add the component: Audio Source
  - Do not set AudioClip yet, it will be set with the script
  - Set Play on Awake false
- Create a variable to store the audio clip and serialize it
  - Set Laser Sound Clip from the inspector: laser\_shot
- Create a handler variable for the AudioSource component

```
[SerializeField]  
0 references  
private AudioClip _laserSoundClip;  
0 references  
private AudioSource _audioSource;
```

# Laser Shot Sound

- Get the AudioSource component at start
  - Make a null check and set the audio source clip
- Update FireLaser method
  - Play the audio source after the laser visual

```
void Start()
{
    transform.position = new Vector3(0, 0, 0);
    _spawnManager_sc = GameObject.Find("Spawn_Manager").GetComponent<SpawnManager>();
    _uiManager_sc = GameObject.Find("Canvas").GetComponent<UIManager>();
    _audioSource = GetComponent<AudioSource>();
}
```

```
if (_audioSource == null)
{
    Debug.LogError("Audio Source is NULL");
}
else
{
    _audioSource.clip = _laserSoundClip;
}
```

```
void FireLaser()
{
    _nextFire = Time.time + _fireRate;

    if (_isTripleShotActive == true)
    {
        Instantiate(_tripleShotPrefab, transform.position, Quaternion.identity);
    }
    else
    {
        Instantiate(_laserPrefab, transform.position, Quaternion.identity);
    }
    _audioSource.Play();
}
```

# Explosion Sound

- Play the explosion sound when the Asteroid explodes
- There is also an Explosion prefab
  - Attach the Audio Source component to the Explosion prefab
  - Set AudioClip: explosion\_sound
  - Set Play on Awake true
  - Save and play
- Asteroid uses Explosion prefab but Enemy objects have a different explosion mechanism
  - Select the Enemy game object and add the component: AudioSource
  - Set Play on Awake false
  - Set AudioClip: explosion\_sound

# Explosion Sound

- Update Enemy script
  - Create a handler variable for the AudioSource component
  - Get component at start and make a null check
  - Play the AudioSource at two locations: when hit by the laser or collide with the Player

```
private AudioSource _audioSource;

0 references
void Start()
{
    _player_sc = GameObject.Find("Player").GetCo
    _anim = GetComponent<Animator>();
    _audioSource = GetComponent<AudioSource>();
}
```

```
//Damage the player
Player_sc player = other.transform.GetC
if (player != null)
{
    player.Damage();
}
_anim.SetTrigger("OnEnemyDeath");
_speed = 0;
_audioSource.Play();
Destroy(this.gameObject, 2.8f);
```

# Powerup Sound

- Play the powerup sound when the Player collects a powerup item
- Update Powerup script
  - We check collision with the Player
  - We can play the sound in the OnTriggerEnter2D method
  - But the object is deleted right after
- How to play audio clip on destroyed game objects?
  - PlayClipAtPoint method: Plays an AudioClip at a given position
  - The method creates an AudioSource but automatically disposes of it once the clip has finished playing
  - It is like an instantiation method for audio



# Powerup Sound

- Update Powerup script
  - Create an AudioClip variable and serialize it
- Set the AudioClip from inspector
  - Select all of the three powerup prefabs and set Audio Clip: power\_up\_sound
- Play clip when the Player collides with the Powerups
  - Update OnTriggerEnter2D method

```
[SerializeField]
```

```
0 references
```

```
private AudioClip _audioClip;
```

```
private void OnTriggerEnter2D(Collider2D other)
{
    if (other.tag == "Player")
    {
        Player_sc player = other.transform.GetComponent<Player_sc>();
        AudioSource.PlayClipAtPoint(_audioClip, transform.position);
        if (player != null)
        {

```