

## Modul 11 – Memainkan dan Memanipulasi Sound

---

### A. TUJUAN

- Mahasiswa dapat mengolah sound yang dapat menyesuaikan pergerakan object.
- Mahasiswa dapat mengolah sound yang terdengar sesuai jarak karakter terhadap sumber suara.
- Mahasiswa dapat mengatur sound dengan source code.

### B. PETUNJUK

1. Awali setiap kegiatan praktikum dengan berdoa
2. Baca dan pahami tujuan, dasar teori, dan latihan-latihan praktikum dengan baik
3. Kerjakan tugas-tugas praktikum dengan baik, sabar dan jujur
4. Tanyakan kepada dosen apabila ada hal-hal yang kurang jelas

### C. ALOKASI WAKTU: 6 jam pelajaran

### D. DASAR TEORI

Modul 11 adalah modul yang membahas cara untuk memainkan dan memanipulasi sound, sehingga game yang dibuat lebih menarik dan lebih nyata.

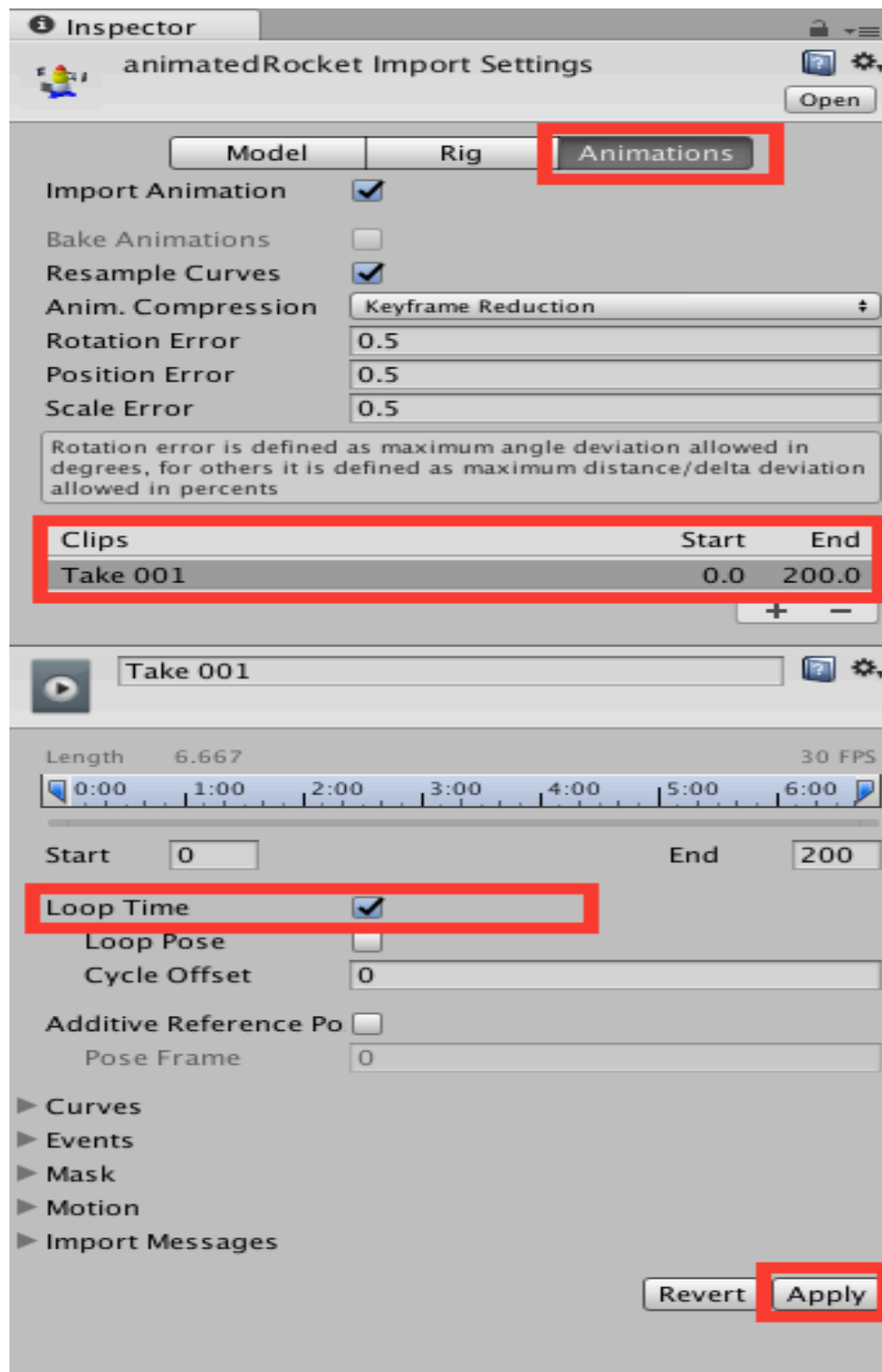
### E. LATIHAN PRAKTIKUM

#### 1. Matching audio pitch pada kecepatan animasi

Untuk melakukan praktikum ke-1, anda akan membutuhkan file `animatedRocket.fbx` dan `engineSound.wav` yang sudah disediakan pada folder `1362_09_01`. Berikut adalah beberapa langkah untuk mempersiapkan environment game:

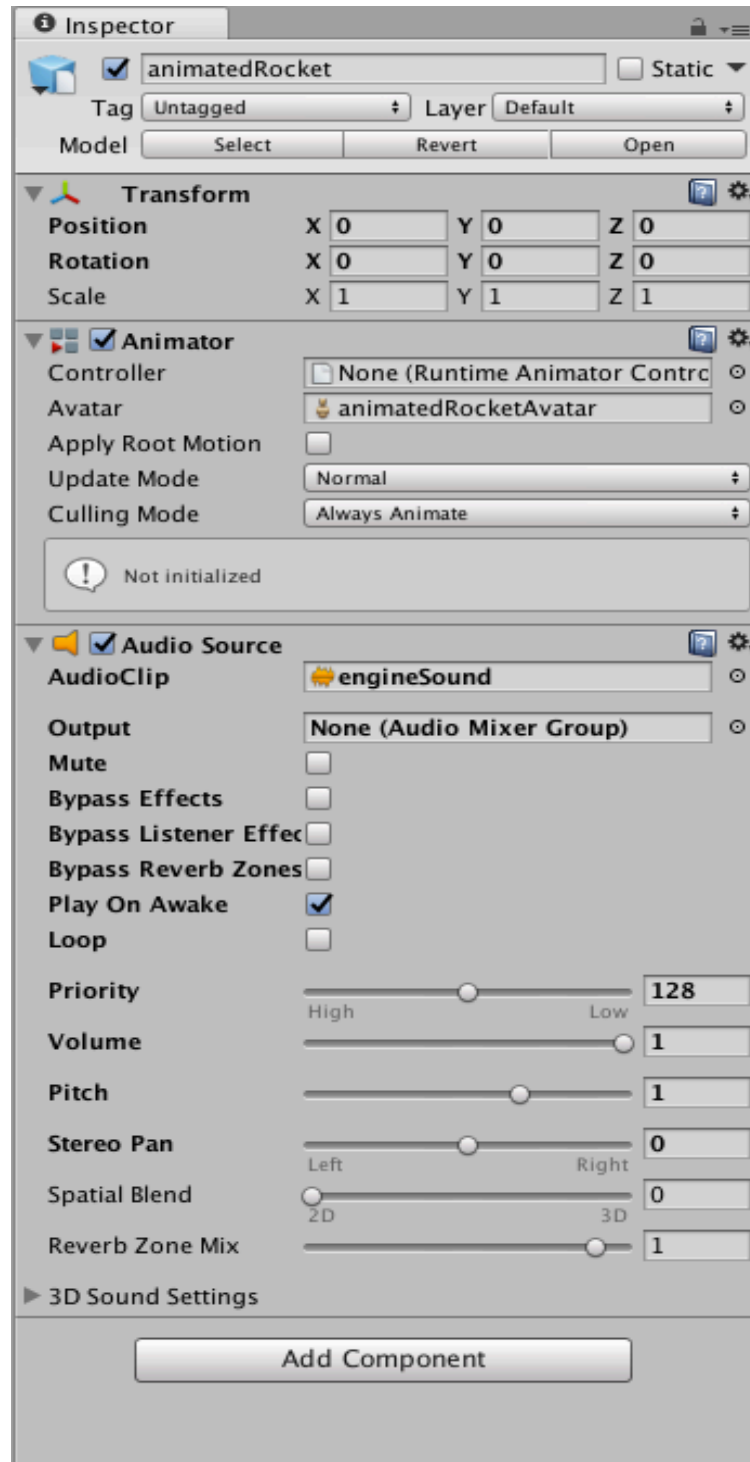
1. Buat project baru 3D.
2. Import file `animatedRocket.fbx` dari folder `1362_09_01`.
3. Pada **Project** view pilih `animatedRocket.fbx`, kemudian pada **Inspector** view amati **import Settings**. Pilih **Animation** tab kemudian pilih pada clip **Take 001**, dan pastikan

**Loop Time** di centang/check. Langkah terakhir klik button Apply untuk menyimpan perubahan.

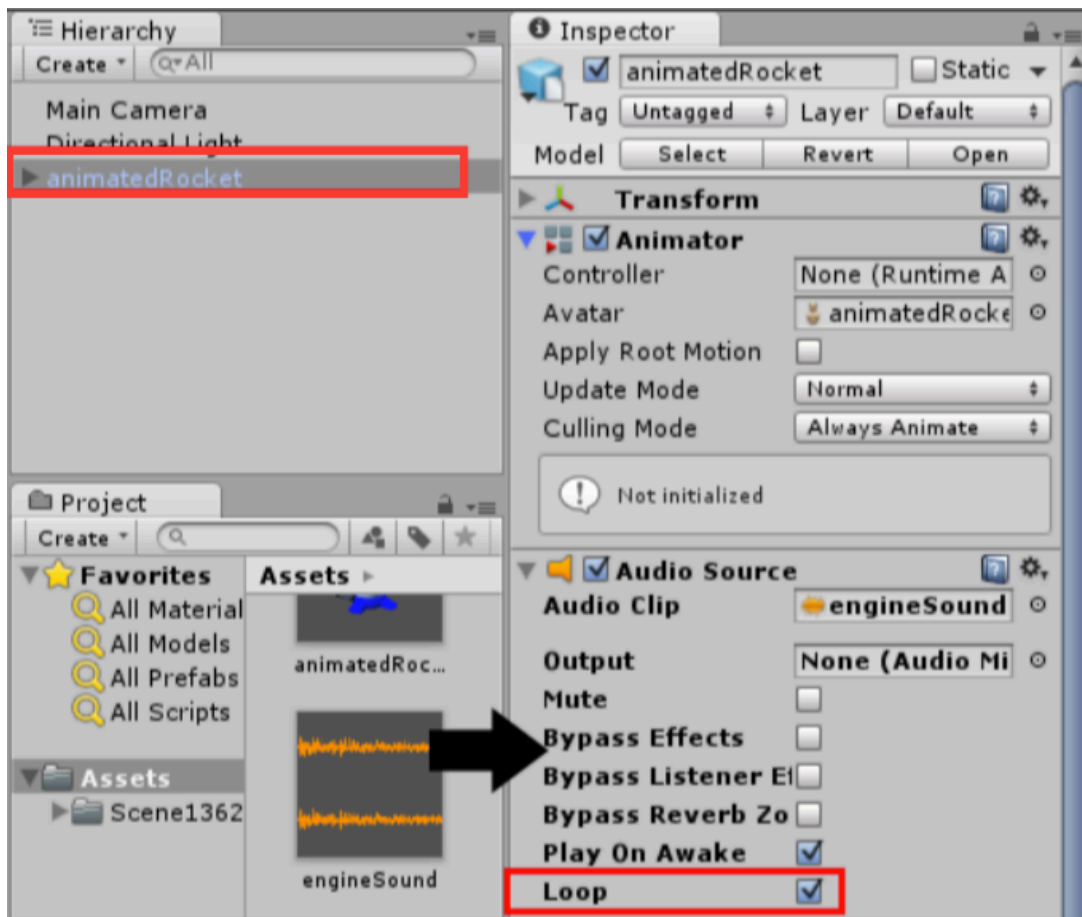


4. Drag animatedRocket.fbx pada **Hierarchy** view.
5. Import engineSound.wav dari folder 1362\_09\_01.

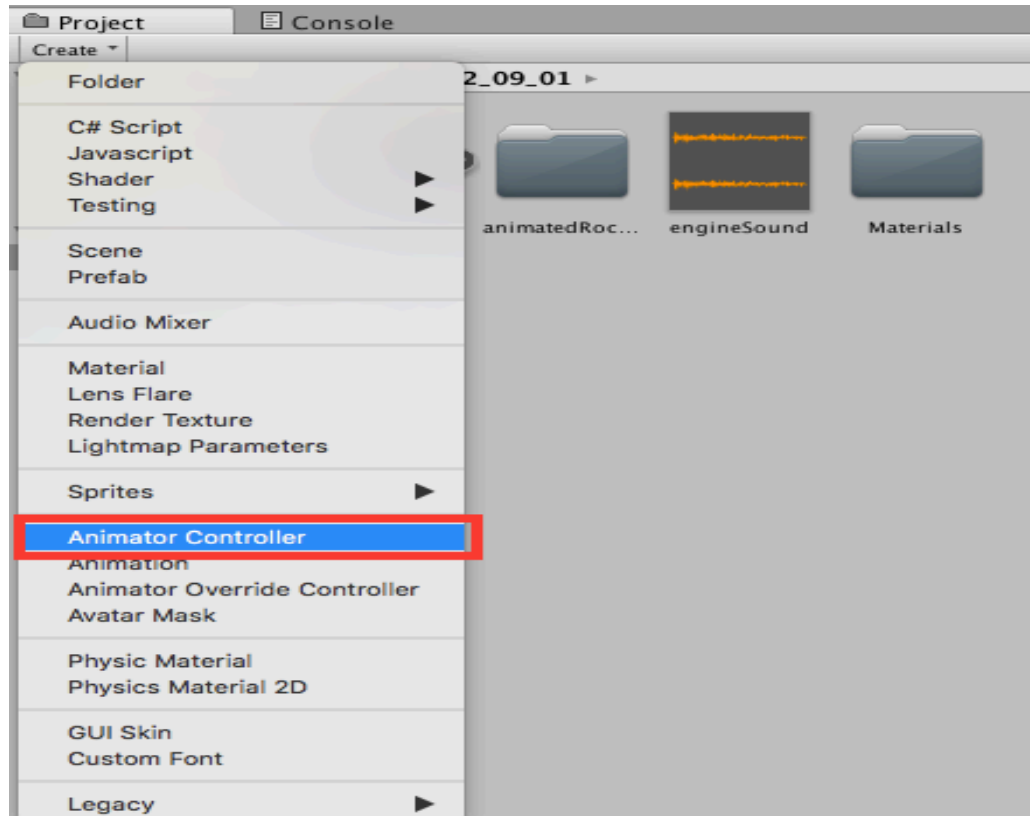
6. Tambahkan **Audio Source** pada Game Object **animatedRocket**. Dengan cara drag **engineSound** pada Object **animatedRocket**. Setelah di drag hasilnya sebagai berikut:



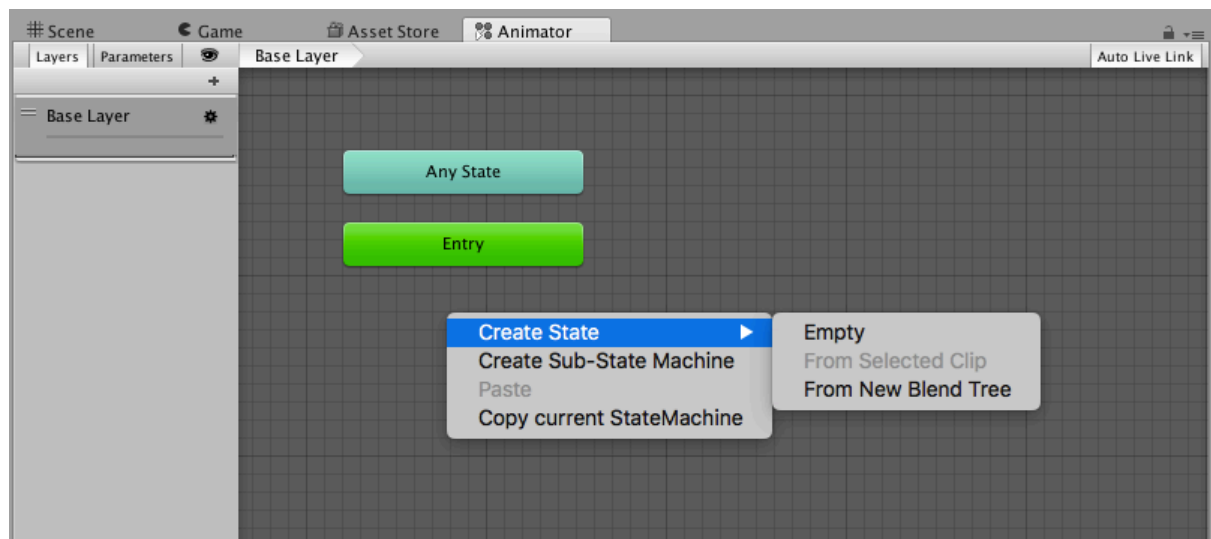
7. Pada komponen **Audio Source** dari **animatedRocket**, centang **loop**. Seperti pada gambar berikut:



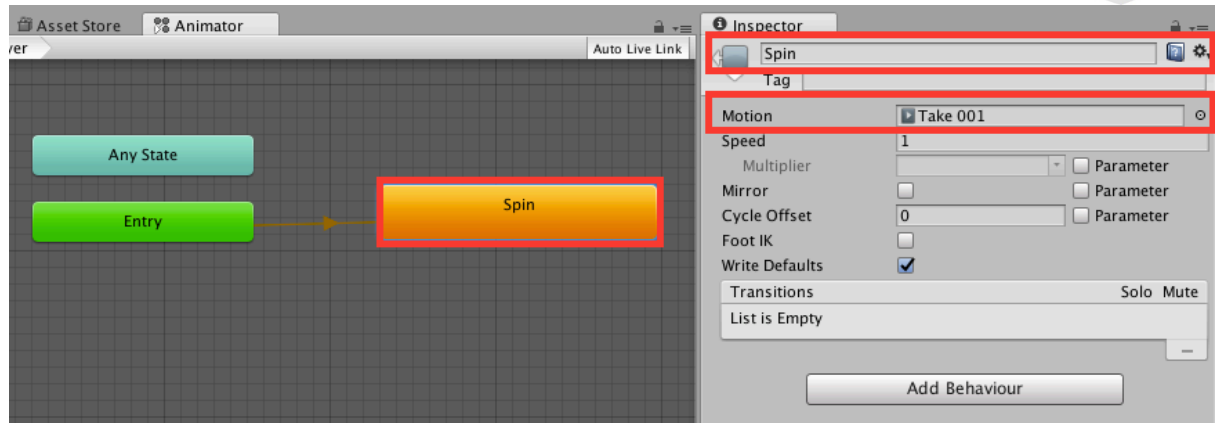
8. Selanjutnya buat Controller untuk **animatedRocket**. Pada **Project** view, klik **Create** dan pilih Animator Controller, kemudian beri nama sebagai **rocketController** .



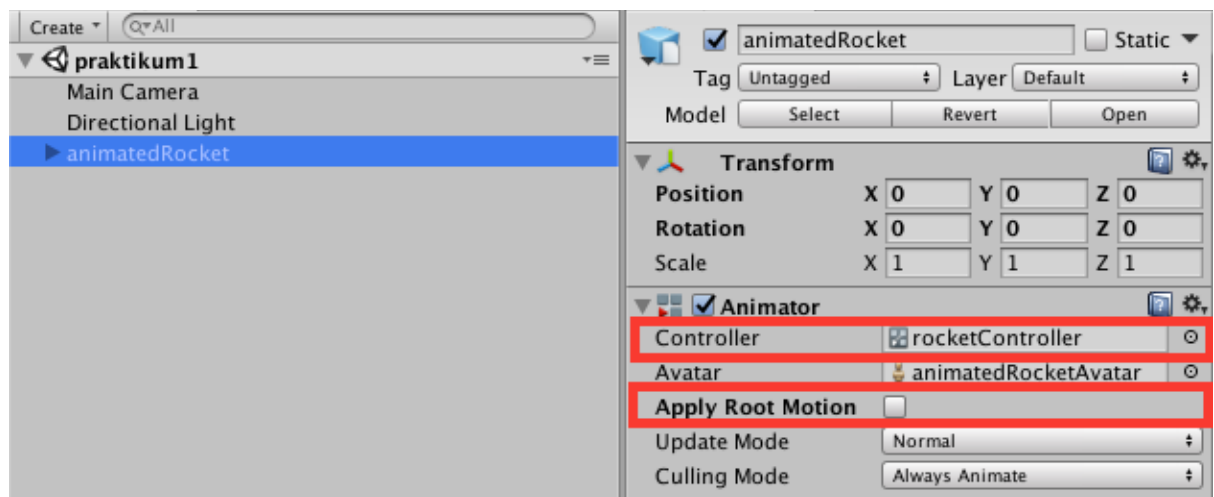
9. Doppelklik objek **rocketController** untuk membuka **Animator** window. Kemudian klik kanan pada grid area dan pilih **Create State | empty**.



10. Beri nama state baru dengan nama **Spin** dan ubah nilai Motion Field dengan **Take 001**.



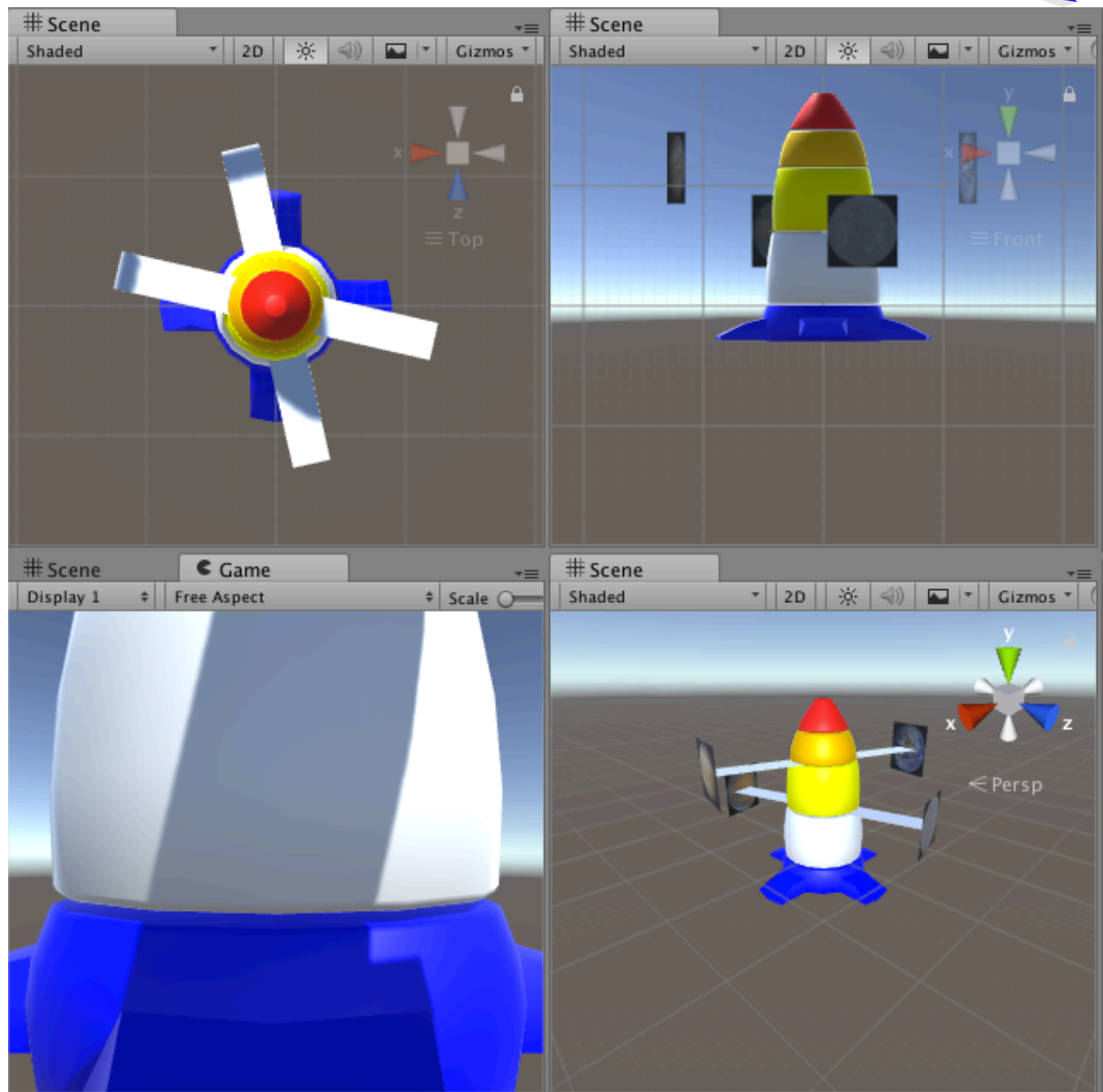
11. Dari **Hierarchy** view, pilih **animatedRocket**. Kemudian pada komponen **Animator** ubah nilai **Controller** menjadi rocketController dan pastikan **Apply Root Motion** tidak dicentang.



12. Pada project view buat **C# Script** dengan nama ChangePitch.
13. Klik double ChangePitch, ubah source code yang ada dengan source code berikut:

```
using UnityEngine;
public class ChangePitch : MonoBehaviour{
    public float accel = 0.05f;
    public float minSpeed = 0.0f;
    public float maxSpeed = 2.0f;
    public float animationSoundRatio = 1.0f;
    private float speed = 0.0f;
    private Animator animator;
    private AudioSource audioSource;
    void Start(){
        animator = GetComponent<Animator>();
        audioSource = GetComponent<AudioSource>();
        speed = animator.speed;
        AccelRocket (0f);
    }
    void Update(){
        if (Input.GetKey (KeyCode.Alpha1))
            AccelRocket(accel);
        if (Input.GetKey (KeyCode.Alpha2))
            AccelRocket(-accel);
    }
    public void AccelRocket(float accel){
        speed += accel;
        speed = Mathf.Clamp(speed,minSpeed,maxSpeed);
        animator.speed = speed = Mathf.Clamp (speed, 0, maxSpeed);
        float soundPitch = animator.speed * animationSoundRatio;
        audioSource.pitch = soundPitch;
    }
}
```

14. Simpan perubahan script dan tambahkan ChangePitch sebagai komponen GameObject animatedRocket.
15. Play project yang telah dibuat hasilnya adalah seperti pada gambar dibawah ini.  
Kecepatan animasi dapat dengan menekan button 1 (percepatan) dan button 2 (memperlambat) pada keyboard. Audio akan mengikuti kecepatan object.



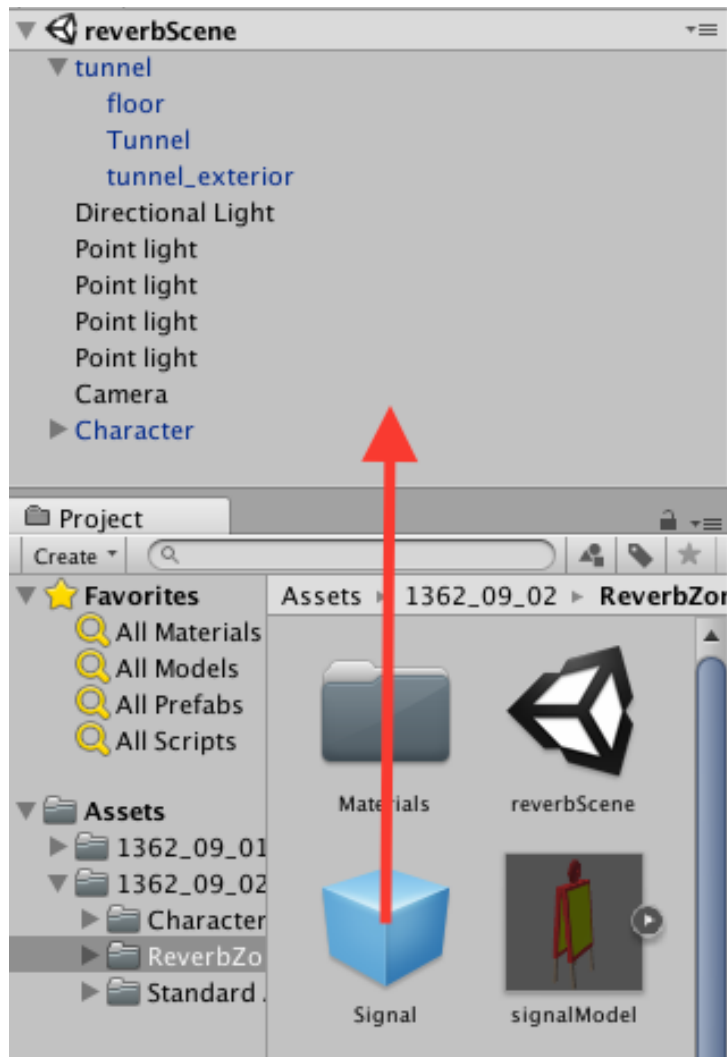
## 2. Simulating Acoustic environment with reverb zone

Untuk praktikum ke-2, anda akan membutuhkan file `ReverbZone.unitypackage` yang sudah disediakan pada folder `1362_09_02`. Pada file tersebut terdapat basic level dengan nama `reverbScene`. Berikut adalah beberapa langkah untuk mempersiapkan environment game:

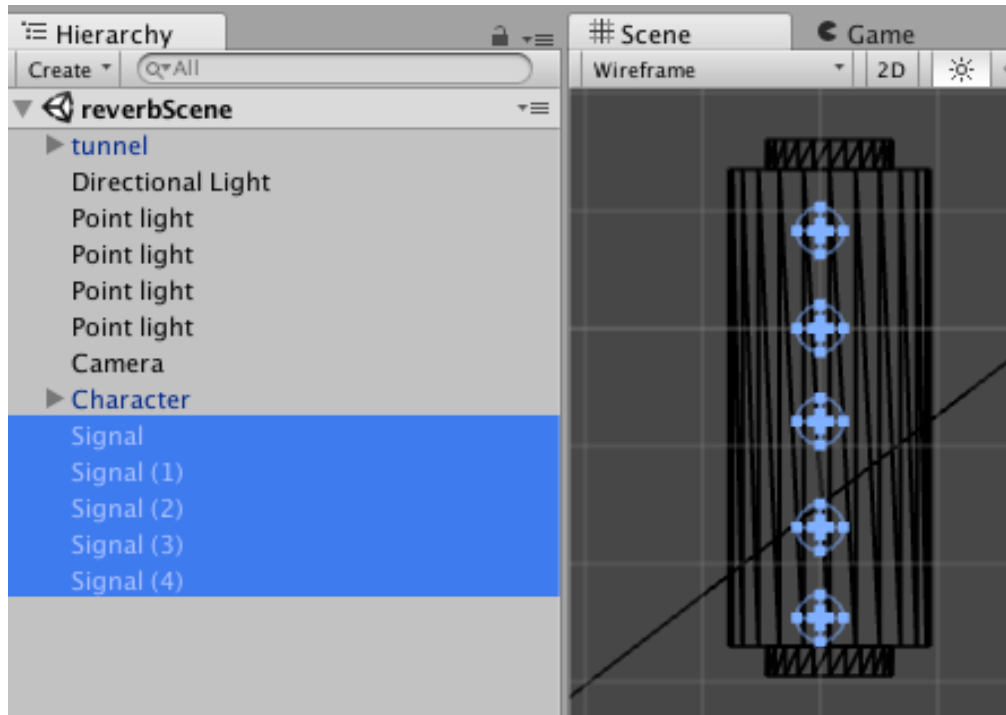
1. Import file `ReverbZone` dari folder `1362_09_02` ke Unity Project.
2. Pada Project view, klik doble pada **reverbScene** yang terdapat pada folder `ReverbZones`. Ini adalah basic scene pada praktikum ke-2 ini, yang terdiri dari terowongan / tunnel, karakter dan kontroler untuk mengatur gerakan karakter.



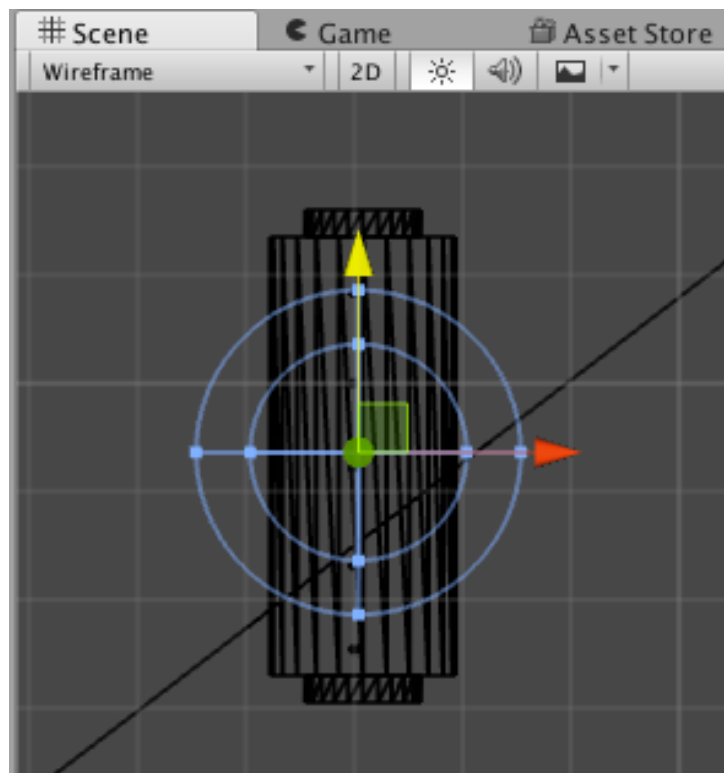
3. Setelah **reverbScene** dibuka, drag **Signal** prefab pada **Hierarchy** view. Letakkan Signal di tengah terowongan. Hal ini digunakan untuk menambahkan pemancar suara.



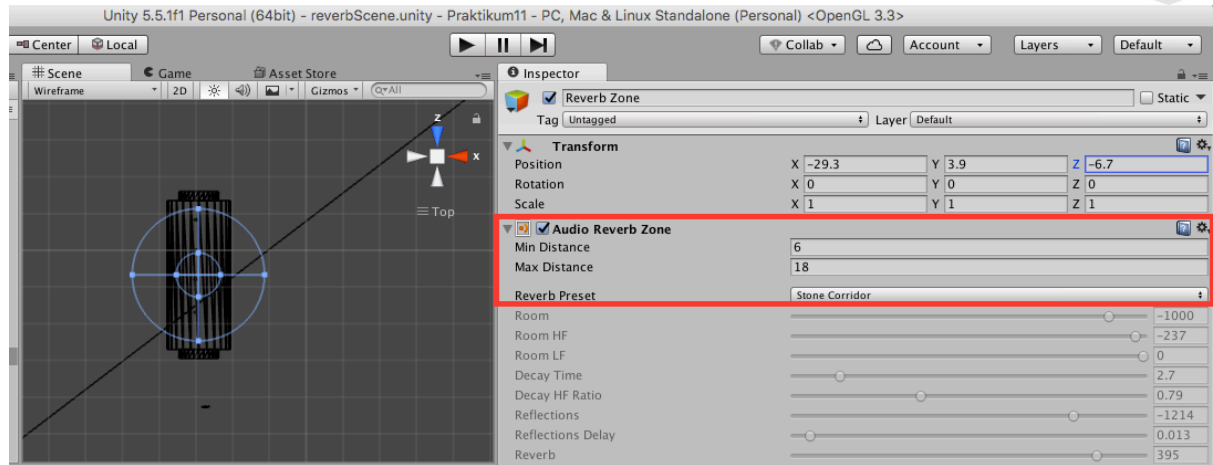
4. Copy **Signal** sebanyak 5 kali dan atur posisinya seperti pada gambar dibawah ini.



5. Pada **Hierarchy** view buat Audio Reverb Zone dengan cara **Create | Audio | Audio Reverb Zone**, kemudian letakkan di tengah terowongan.



6. Pilih game Object **Reverb Zone**. Pada **Inspector** view, ubah parameter komponen **Reverb Zone**, seperti berikut **Min Distance:6; Max Distance:18** dan **Preset:StoneCorridor**. seperti pada gambar berikut:

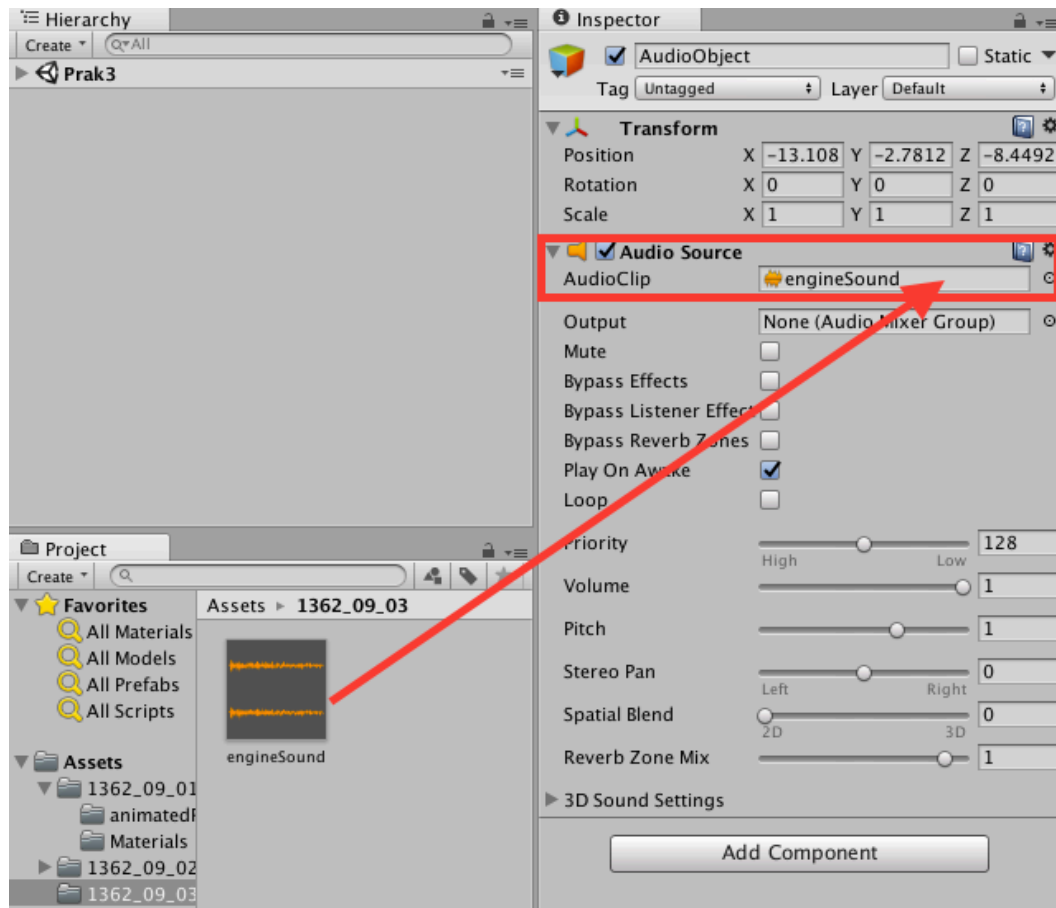


7. Play project yang telah dibuat. Anda dapat menggerakkan karakter dengan menekan button WASD dan Shift untuk berlari. Hasil akhirnya adalah anda dapat mendengar suara bergema pada Reverb Zone, sebaliknya suara akan semakin mengecil jika menjauhi Reverb Zone.

### 3. Preventing an Audio Clip from restarting if it already playing

Untuk melakukan praktikum ke-3, anda akan membutuhkan file animatedRocket.fbx dan engineSound.wav yang sudah disediakan pada folder 1362\_09\_03. Berikut adalah beberapa langkah untuk mempersiapkan environment game:

1. Buat project baru 2D.
2. Buat **Empty** GameObject dan beri nama sebagai **AudioObject**. Selanjutnya tambahkan komponen **Audio Source** dengan cara Select **AudioObject** | **Add Component** | **Audio** | **Audio Source**.
3. Import **engineSound.wav** Audio clip dari folder 1362\_09\_03. Kemudian drag **engineSound** dari Project view ke parameter Audio Clip pada **AudioObject**.

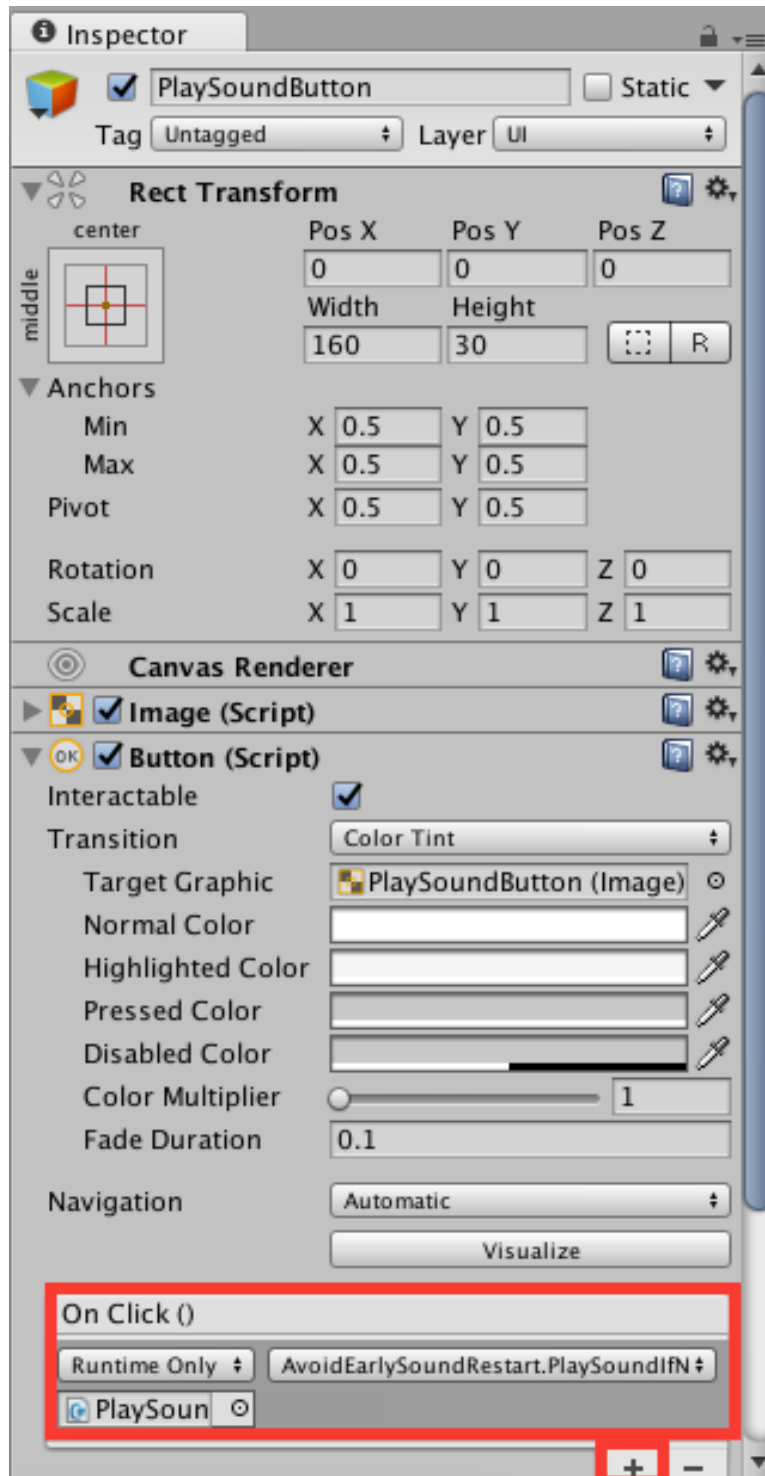


4. Pada langkah keempat, buat UI Button dengan nama **PlaySoundButton** pada hierarchy view dan tambahkan C# script pada **AvoidEarlySoundRestart**. Berikut adalah source codednya:

```
using UnityEngine;
using System.Collections;
using UnityEngine.UI;

public class AvoidEarlySoundRestart : MonoBehaviour {
    public AudioSource audioSource;
    public Text message;
    void Update(){
        string statusMessage = "Play sound";
        if(audioSource.isPlaying)
            statusMessage = "(sound playing)";
        message.text = statusMessage;
    }
    // button click handler
    public void PlaySoundIfNotPlaying(){
        if( !audioSource.isPlaying)
            audioSource.Play();
    }
}
```

5. Pada langkah kelima pilih **PlaySoundButton** pada hierarchy view selanjutnya tambahkan event baru. Untuk menambahkan event pada **PlaySoundButton** dengan cara tekan “+” pada event selanjutnya **PlaySoundButton** untuk mengisi object, selengkapnya seperti pada gambar berikut:



6. Pada langkah terakhir Play project yang telah dibuat. Hasil akhirnya adalah sound dimainkan hanya sekali, setelah itu akan akan berhenti dan text pada button akan berubah menjadi **Play Sound**. Jika pada saat text button adalah Play Sound, button dapat di klik untuk memainkan sound, selanjutnya teks pada button berubah menjadi **Sound Playing**.

#### A. TUGAS PRAKTIKUM

- a. Kerjakan seluruh project yang ada dimodul ini dan kumpulkan. Berikan penjelasan yang kalian dapatkan setelah melakukan praktikum.
- b. Buat project menggunakan package pada folder Tugas, tambahkan Praktikum ke 1 dan Praktikum 2 pada package yang telah disediakan. Ulangi pembuatan project supaya lebih paham.

--- SELAMAT BELAJAR ---