```
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class SoundManager : SingletonMonoBehaviour<SoundManager>
{
    [SerializeField, Range(0, 1), Tooltip("マスタ音量")]
    static float volume = 1;
    [SerializeField, Range(0, 1), Tooltip("BGMの音量")]
    static float bgmVolume = 1;
    [SerializeField, Range(0, 1), Tooltip("SEの音量")]
    static float seVolume = 1;

    AudioClip[] bgm;
    AudioClip[] se;

    Dictionary<string, int> bgmIndex = new Dictionary<string, int>();
    Dictionary<string, int> seIndex = new Dictionary<string, int>();

    AudioSource bgmAudioSource;
    AudioSource seAudioSource;


    public float Volume
    {
        set
        {
            volume = Mathf.Clamp01(value);
            bgmAudioSource.volume = bgmVolume * volume;
            seAudioSource.volume = seVolume * volume;
        }
        get
        {
            return volume;
        }
    }

    public float BgmVolume
    {
        set
        {
            bgmVolume = Mathf.Clamp01(value);
            bgmAudioSource.volume = bgmVolume * volume;
        }
        get
        {
            return bgmVolume;
        }
    }

    public float SeVolume
    {
        set
        {
            seVolume = Mathf.Clamp01(value);
            seAudioSource.volume = seVolume * volume;
        }
        get
        {
            return seVolume;
        }
    }

    public void Awake()
    {
        if (this != Instance)
        {
            Destroy(gameObject);
            return;
        }


        DontDestroyOnLoad(gameObject);

        bgmAudioSource = gameObject.AddComponent<AudioSource>();
        seAudioSource = gameObject.AddComponent<AudioSource>();

        bgm = Resources.LoadAll<AudioClip>("Audio/BGM");
        se = Resources.LoadAll<AudioClip>("Audio/SE");



        for (int i = 0; i < bgm.Length; i++)
        {
            bgmIndex.Add(bgm[i].name, i);
        }

        for (int i = 0; i < se.Length; i++)
        {
            seIndex.Add(se[i].name, i);
        }
        if (SceneManager.GetActiveScene().name == "menu")
        { // Menuシーンでのみやりたい処理
            PlayBgmByName("sou");

                    PlaySeByName("oto");
            }

        else if(SceneManager.GetActiveScene().name == "option")
        { // Optionのシーンでやりたい処理
            PlayBgmByName("cat");
        }
    }

    public int GetBgmIndex(string name)
    {
        if (bgmIndex.ContainsKey(name))
        {
            return bgmIndex[name];
        }
        else
        {
            Debug.LogError("指定された名前のBGMファイルが存在しません。");
            return 0;
        }
    }

    public int GetSeIndex(string name)
    {
        if (seIndex.ContainsKey(name))
        {
            return seIndex[name];
        }
        else
        {
            Debug.LogError("指定された名前のSEファイルが存在しません。");
            return 0;
        }
    }

    //BGM再生
    public void PlayBgm(int index)
    {
        index = Mathf.Clamp(index, 0, bgm.Length);

        bgmAudioSource.clip = bgm[index];
        bgmAudioSource.loop = true;
        bgmAudioSource.volume = BgmVolume * Volume;
        bgmAudioSource.Play();
    }

    public void PlayBgmByName(string name)
    {
        PlayBgm(GetBgmIndex(name));
    }

    public void StopBgm()
    {
        bgmAudioSource.Stop();
        bgmAudioSource.clip = null;
    }

    //SE再生
    public void PlaySe(int index)
    {
        index = Mathf.Clamp(index, 0, se.Length);

        seAudioSource.PlayOneShot(se[index], SeVolume * Volume);
    }

    public  void PlaySeByName(string name)
    {
        PlaySe(GetSeIndex(name));
    }

    public void StopSe()
    {
        seAudioSource.Stop();
        seAudioSource.clip = null;
    }

}
```