

Loader Object: A Unity Asset

Description: A small package that facilitates development by implementing Scriptable Objects as a quick method to save and load various types of Data from a JSON file.

Version: 1.0.0

Asset Store: [Link](#)

Location of the JSON files: C:\Users\Username\AppData\LocalLow\Company\Game\Loader Objects

How to use this Package

1. Create a class that inherits from LoaderObjectData<TData, TValue>

```
[CreateAssetMenu(fileName = "Audio LoaderData", menuName = "Data/Loader
Object/Audio")]
public class AudioLoaderData : LoaderObjectData<MixerChannel, float>
{
    protected override string FileName { get; }

    public override void SaveData(MixerChannel channel)
    { }

    public override void LoadData(MixerChannel channel)
    { }
}
```

TData = Type of the Data that is passed from LoaderMono of this Scriptable Object.

TValue = Type of the value that will be saved and loaded into the Json file. (e.g. class, float, struct, etc.)

Remark: **TValue** needs to be Serializable in order to save into Json.

`[System.Serializable]` - for classes, structs

[SerializeField] - for fields

2. Create a class that inherits from LoaderMono<TObject, TData>

```
public class AudioManager : LoaderMono<AudioLoaderData, MixerChannel>
{
    protected override void Awake()
    { }
}
```

```
[System.Serializable]
public class MixerChannel
{
    [SerializeField] private string channelName;
    [SerializeField] private string displayName;

    private AudioManager _audioMixer;

    public string DisplayName => displayName;
    public float SliderVolume { get; private set; }

    internal void SetMixer(AudioMixer audioMixer) => _audioMixer =
audioMixer;

    public void SetVolume(float sliderVolume)
    {
        SliderVolume = sliderVolume;
        _audioMixer.SetFloat(channelName, GetChannelVolume());

        float GetChannelVolume() => Mathf.Log(sliderVolume) * 20;
    }
}
```

TObject = Type of the Scriptable Object that inherits from LoaderObjectData.

TData = Type of the Data that will be passed to the Scriptable Object, indicated by TObject. (e.g. class, float, struct, etc.)

3. Call **PassData()** function in the LoaderMono

```
public class AudioManager : LoaderMono<AudioLoaderData, MixerChannel>
{
    [SerializeField] private AudioMixer audioMixer;
    [SerializeField] private MixerChannel[] mixerChannels;

    protected override void Awake()
    {
        foreach (var channel in mixerChannels)
            channel.SetMixer(audioMixer);

        PassData(mixerChannels);
    }
}
```

PassData(TData dataElement) = Pass a single Data that will Load and Save to Json.

PassData(TData[] dataArray) = Pass an array of Data that will Load and Save to Json.

PassData(List<TData> dataList) = Pass a list of Data that will Load and Save to Json.

4. Implement saving and loading in the LoaderObjectData

```
[CreateAssetMenu(fileName = "Audio LoaderData", menuName = "Data/Loader
Object/Audio")]
public class AudioLoaderData : LoaderObjectData<MixerChannel, float>
{
    protected override string FileName { get; } = "Audio";

    public override void SaveData(MixerChannel channel)
    {
        SetValue(channel.DisplayName, channel.SliderVolume);
    }

    public override void LoadData(MixerChannel channel)
    {
        float sliderVolume = GetValue(channel.DisplayName, 1f);
        channel.SetVolume(sliderVolume);
    }
}
```

FileName = The unique name of the Json File (e.g: PostProcessing, PlayerPositions)

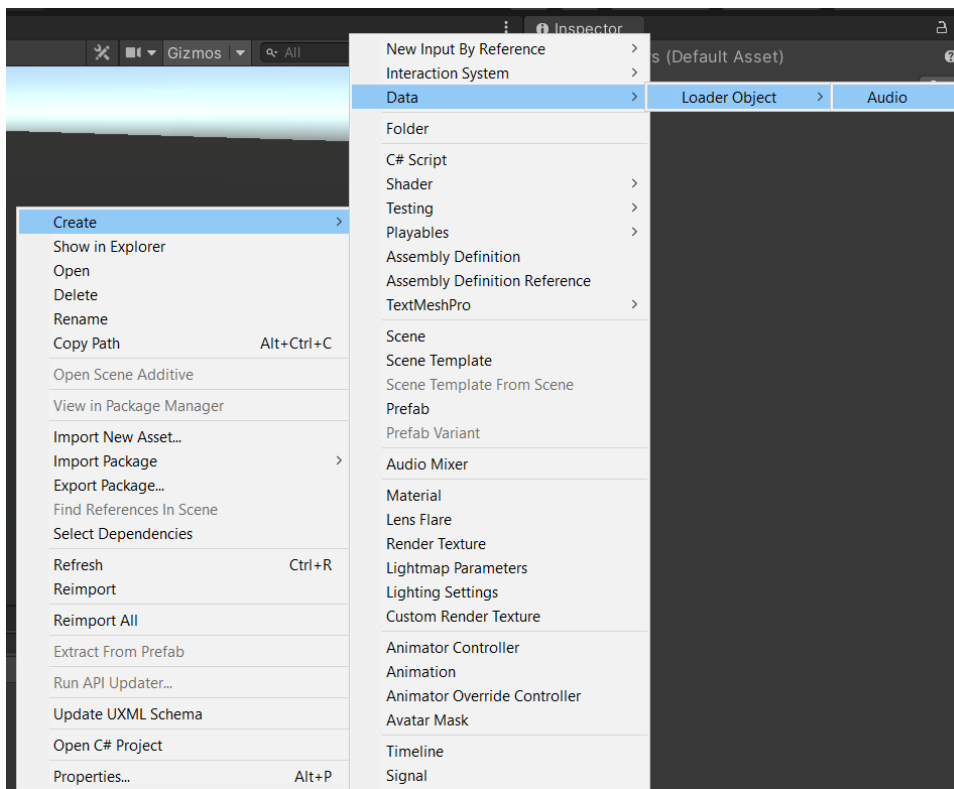
`SaveData(TData data)` = Called when the LoaderMono of this Scriptable Object is Disabled. Use `SetValue` function to save the Data.

`LoadData(TData data)` = Called when the LoaderMono of this Scriptable Object is Awakened. Use `GetValue` function to get the Data.

`SetValue(string dataName, TValue dataValue)` = Set the value of the Data, indicated by `dataName`. If the Data is not found, a new one is created. (This can happen whether the object is not found in the Scriptable Object or the Json file was not generated)

`GetValue(string dataName, TValue defaultValue = default)` = Get the value of the Data, indicated by `dataName`. If the Data is not found, a new one is created with the `defaultValue` value. (This can happen whether the object is not found in the Scriptable Object or the Json file was not generated)

5. Create the LoaderObjectData in Unity



6. Attach the LoaderMono script on a GameObject and assign the LoaderObjectData

