DIALOGUE SPEAKER DOCUMENTATION

GETTING STARTED

1. Create an empty game object and add the "*Dialogue Speaker*" component. You will also find that an AudioSource has been added. This is the central audio that the Dialogue Speaker will use.

2. You will also find that this added AudioSource has been set in the **Central Audio** property. If you wish to use another audio source. Simply drag/drop your audio source to the central audio property.

3. Create a Text Mesh Pro UI text and position and style it as you please. Then drag/drop this object to **Subtitles Text** property as this will be the subtitle for the dialogues.

4. Make sure **Use Subtitles** property is checked in order for the subtitles to be printed out.

5. Set the wanted number for your dialogues inside the **Dialogues** property list.

6. As you expand each element inside the dialogues list you will notice that each element contains 4 properties: ***Audio*** (AudioClip), ***Time*** (float), ***Subtitles*** (string) and finally ***ScriptToEnable*** (MonoBehaviour)
   - Audio: is an AudioClip and it's the actual audio clip you want to play.
   - Time: is a float and it's the amount of time passed before playing the audio. Useful for adding a little breathing room between dialogues.
   - Subtitles**:** takes a string and is the actual subtitles text you want printed out.
   - TriggerEvent: is any public method to trigger when the audio plays.

7. After setting all the above it's time for the ***Script Options.***

8. The **Fire Finish Event** property is to set whether you want to trigger a certain method when the entire dialogue finishes.

9. **Finished Event** takes a method that you want to trigger when the dialogue finishes.

10. Now that you've finished setting the Dialogue Speaker you can get the "DialogueSpeaker" component via script and run **Play()**. Or simply **Set Play On Awake** property to true in the inspector.

APIs AND PROPERTIES:

index *– (type int) returns the current index of the playing dialogue.*

Play() *– starts playing the dialogues.*

Pause() *– stops the current dialogue playing for later resume.*

Resume() *– resumes the dialogue from pause.*

Skip() *– skips the current playing dialogue and jumps to the next one. (if available)*

PlayAudioIndex(int audioIndex) *- plays dialogue of certain index.*

isFinished *- (type bool) returns whether the dialogue has finished or not.*

isStarted *- (type bool) returns whether the dialogue has started once before or not. (if true will never go back to false)*