

# Dialogue Styling Conventions

This document is to demonstrate all the ways we will style our dialogue in the game. Some of which come from **BB Code** (included in Godot's RichTextLabel node), some are from **Nathan Hoad's [Dialogue Manager](#)** plugin, and some are **custom functions** we created.

## Table of Contents

<b>Table of Contents</b> .....	<b>1</b>
<b>BB Code</b> .....	<b>2</b>
Font-Size:.....	2
Shake:.....	2
Wave:.....	3
Italics:.....	3
Color:.....	3
<b>Dialogue Manager</b> .....	<b>4</b>
Speed:.....	4
Wait:.....	4
<b>Custom Functions</b> .....	<b>5</b>
Interrupt Dialogue:.....	5
Custom Wait:.....	5
Fade:.....	5
Music Fade:.....	5
Play Sound:.....	6

# BB Code

## Font-Size:

How to use:

- [font\_size=X] text with new font size [/font\_size]
- Standard text size (without label) is 50
- Never go above 100 or below 30
- Changing font size mid-dialogue line
  - Going from a larger font size to a smaller font size is completely fine (especially if a line is getting progressively smaller)
  - Generally avoid going from a smaller font size to a larger font size. If you start small and then go larger, the text is shifted down in the text box slightly as it's printing out, which can look strange
  - Going down then back up is fine, and vice versa.

Why to use:

- Larger font sizes (>50): use if a character is angry, shocked, upset, or excited. Anything where someone would raise their voice
- Smaller font sizes (<50): use if a character is embarrassed, sad, weak, shy, or awkward. Anything where they might lower their voice
- Text progressively getting smaller: great for awkward or embarrassed.
- Singular words with larger font size: great for emphasis
- Singular words with smaller font size: great for awkwardness, or any onomatopoeia (such as "Ahem")

## Shake:

How to use:

- [shake level=X rate=X] text to shake [/shake]
- Level should be between 7-10
- Rate should be between 20-50
- Standard shake is level 7 and rate 30
- There's also a "connected" parameter, which defaults to 1 (true). If the section of text which will shake is very small, it might look awkward. Try setting connected to 0
- For larger font sizes (80-100), use level 10 and rate 50

Why to use:

- Great to express anger, frustration, or villain monologues ("BEHOLD!" vibe)
- Generally good for emphasis
- Combine with italics for an even greater emphasis

## Wave:

How to use:

- `[wave freq=X amp=X] text to wave [/wave]`
- Freq should be between 7-10
- Amp should be between 20-50
- Standard wave is freq 7 and amp 30
- There's also a "connected" parameter, which defaults to 1 (true). If the section of text which will wave is very small, it might look awkward. Try setting connected to 0

Why to use:

- Good for enthusiasm or innocent excitement, sarcasm, or taunting

## Italics:

How to use

- `[i] italicized text [/i]`
- If there's punctuation right after the text, italicize it as well (especially ! or ?)

Why to use

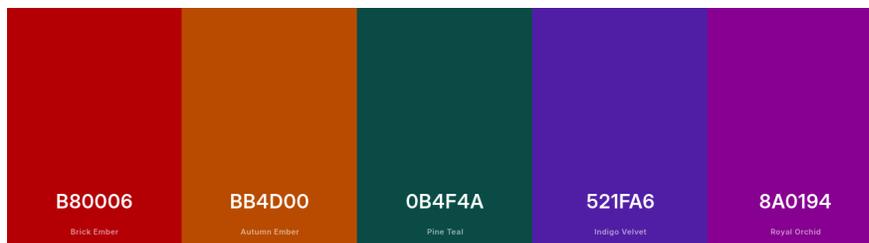
- Use for emphasis, generally

## Color:

How to use

- `[color="X"]Colored Text[/color]` followed by non-colored text
- For colored text, ensure that the individual word or phrase is the *only* thing colored.
- For each color, see the use case below.

Why to use



- Red (B80006) for People
- Orange (BB4D00) for N/A
- Green (0B4F4A) for Locations
- Indigo (521FA6) for Important Info
- Purple (8A0194) for Wizards

# Dialogue Manager

## Speed:

How to use

- This text will play at x1 speed. [speed=2] This text will play at x2 speed. [speed=0.5]  
This text will play at half speed.
- Generally don't go below 0.5 or above 1.5
- You can go above 1.5 if the player *is not supposed to read the text*
  - Ex: Fenn rambling followed by another character interrupting him

Why to use

- For legible fast text (>1.0, <1.5), it's good for an excited or anxious character
- For illegible Fast Text (>1.5), it's good for an overly excited character, where the information present is not important.
  - When Fenn goes on rants, followed by another character interrupting him.
- Slower text is great for emphasizing or exaggerating certain information

## Wait:

How to use

- This line has a dramatic [wait=X] pause.
- This function waits the allotted amount of seconds
- Almost *always* use 0.3
- Note: Using [wait] *ensures that a dialogue text box is present on the screen*. You cannot force dialogue to pause without a dialogue box on screen (so we made a custom function)

Why to use

- Use for pauses in the *middle* of dialogue.
- Currently, Dialogue Manager automatically pauses briefly after every period. Use [wait] for pauses after other punctuation, if it makes sense.

# Custom Functions

## Interrupt Dialogue:

How to use

- do EventBus.interrupt\_dialogue.emit("X")
- Put this line *before* the specific line you want to interrupt
- X is the number of seconds to wait after the dialogue line finishes *before* the next line interrupts it. It *must* be a String.
- You *cannot* chain interrupts together.

Why to use

- Use any time you want a character to interrupt another character.

## Custom Wait:

How to use

- do CutsceneManager.wait(X)
- Put this line wherever you want the dialogue to wait X number of seconds

Why to use

- This causes the dialogue to wait *without* a dialogue box present on the screen.
- Use this for waiting *in between* two lines of dialogue

## Fade:

How to use:

- do CutsceneManager.fade\_to\_black()
- do CutsceneManager.fade\_from\_black()
- They work exactly as intended, taking a few seconds to transition the screen
- Note: Dialogue boxes appear *below* the black screen, and thus are not visible

## Music Fade:

How to use:

- do! MusicManager.\_fade\_out()
- do! MusicManager.\_fade\_in()
- Always include the '!' for the 'do'
- They work exactly as intended, taking a few seconds to fade in or out the music

## **Play Sound:**

- do CutsceneManager.play\_sound("Sound")
- The String can be a keyword that is established in the "play\_sound" function OR a file path to a desired sound
- You may also specify a volume and pitch scale for the sound
  - Example: do CutsceneManager.play\_sound("Sound", vol, pitch)
  - "vol" and "pitch" are float variables