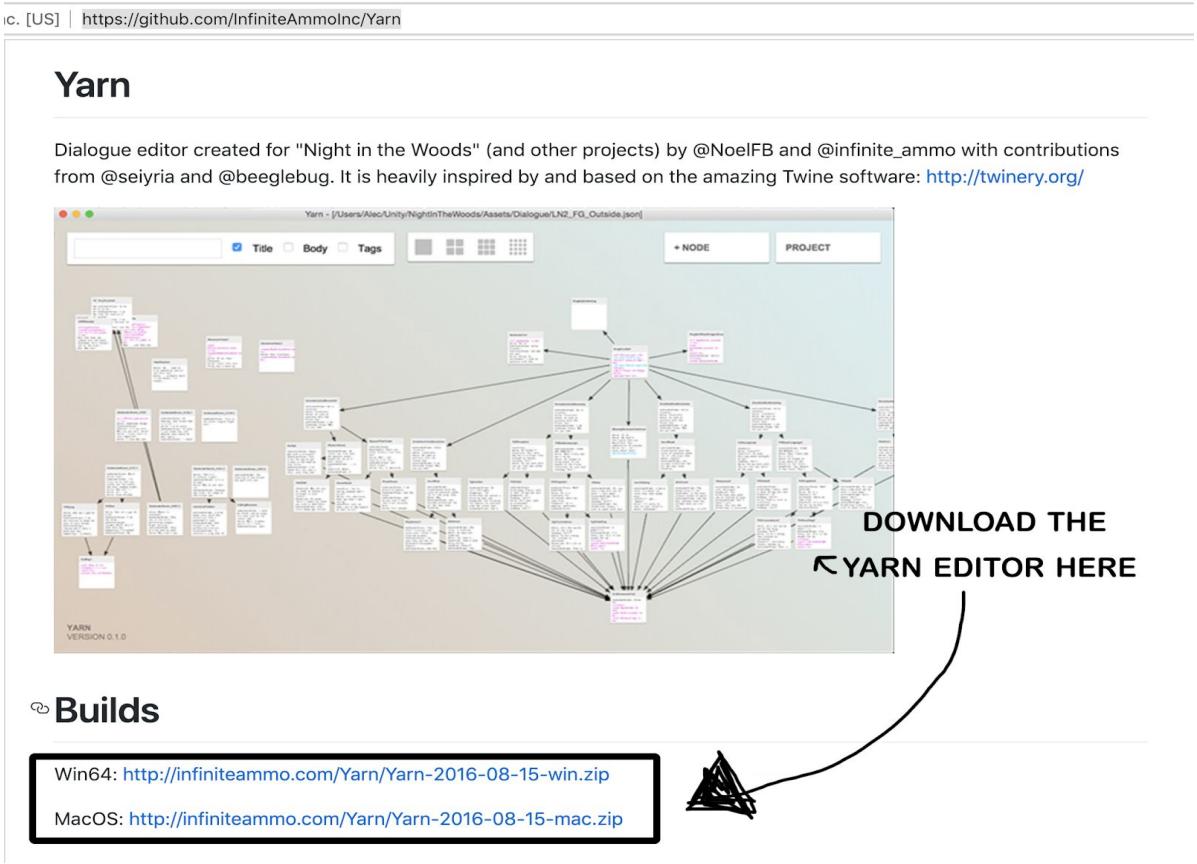
Starting Guide to Yarn

[Starting Guide to Yarn](#)
- [Yarn Basics](#)
- [When to Change Nodes](#)
- [Variables, Commands and Game Team Instructions within Yarn:](#)
- [Else / If Statements](#)
- [Saving](#)
- [Writing Yarn in a Plain txt Doc:](#)
- [Extra Example:](#)
- [Validating your Yarn](#)

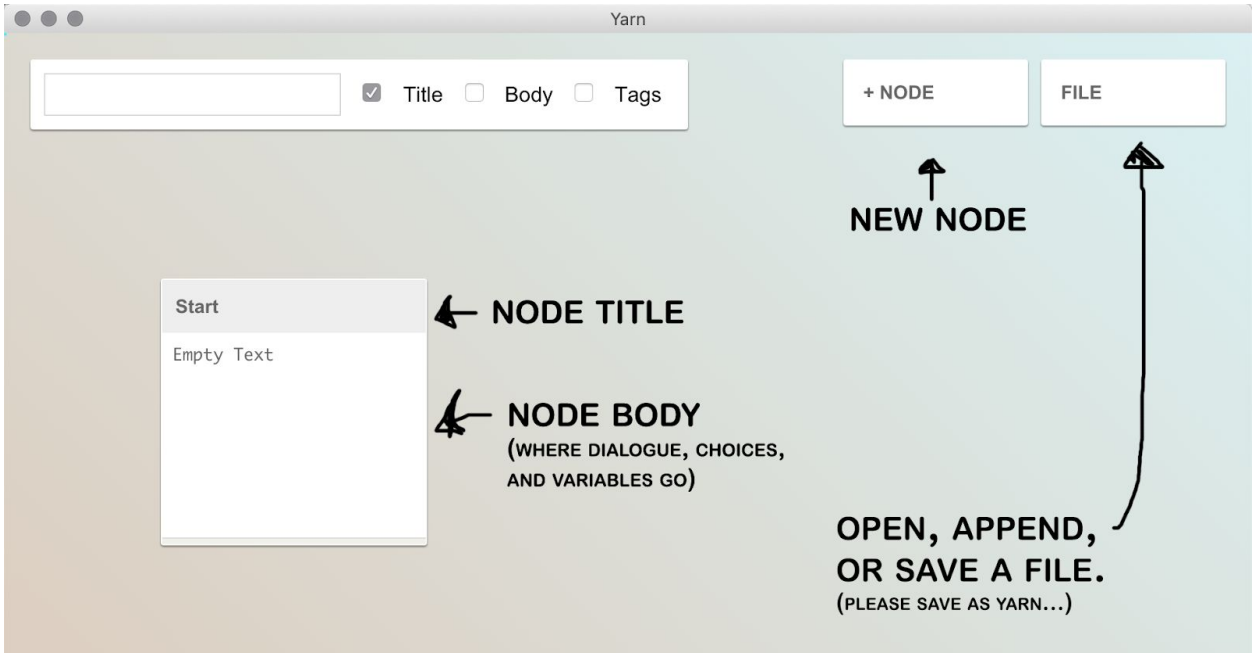
[Using Yarn in Unity](#)
- [Variables in Yarnspinner](#)
- [All the Links](#)

Yarn Basics

https://github.com/InfiniteAmmoInc/Yarn



The **yarn editor** is the IDE [integrated development environment] for yarn. It is a visual way to write nodes and to see how each node connects to each other. However, **you do not need to use the editor to write the script**. It is just a convenient way to view connections and titles If the text-based method is confusing. Yarn also now has a web-based editor too!



A screenshot of the Yarn editor with labels
*file can be saved as .yarn or .json, but I think .json is more work to import

This is the basic structure of the editor. Important notes:

- **A node can have multiple characters speaking in the Body**

```
Character1: Hey.

Character2: Oh! Hi.

Character2: You snuck up on me.
```

    - Each line that you write is displayed on a new line (ie, the first line plays, then the player clicks, then the second line replaces it)
- **Changing a node looks like this:**

```
[[Words to describe choice|nameOfNewNode]]
```

    - The words before | represent the dialogue that appears as the choice, like “That sounds like a good idea!” “Go Left” or “Pick up the can of tuna”
    - The thing after the | is the node you want the choice to lead to.
    - You cannot have a space before or after |nameOfNewNode
        - The visual editor will color this, but you can always check by running your code
- To visit a node without a player choice, omit any text and the |:

```
[[nameOfNewNode]]
```

- **Node Titles**
    - Each node should have a unique title to avoid confusion
    - If you are using a character’s name, use “Name.Start” or “Name.Anger” instead of “Name” for clarity
    - Node names cannot contain spaces
- **Comments**

Use two slashes “//” for games team instructions

When to Change Nodes

- when the player makes an important choice
- a conversation ends.
- When the player talks to different characters

- You want multiple choices to lead to a certain point

Variables, Commands and Game Team Instructions within Yarn:

https://github.com/thesecretlab/YarnSpinner/blob/master/Documentation/YarnSpinner-Dialogue/General-Usage.md

- All variables need to be prefixed by $
- These variables end up stored in ExampleVariableStorage
- Unity can read Yarn variables, but changing them in unity does not change them in yarn.

Else / If Statements

```
<<if not visited("Jada.Hello")>>
    [[Have you done anything cool recently?|Jada.Hello]]
<<endif>>
[[See you later.|Jada.Exit]]
```

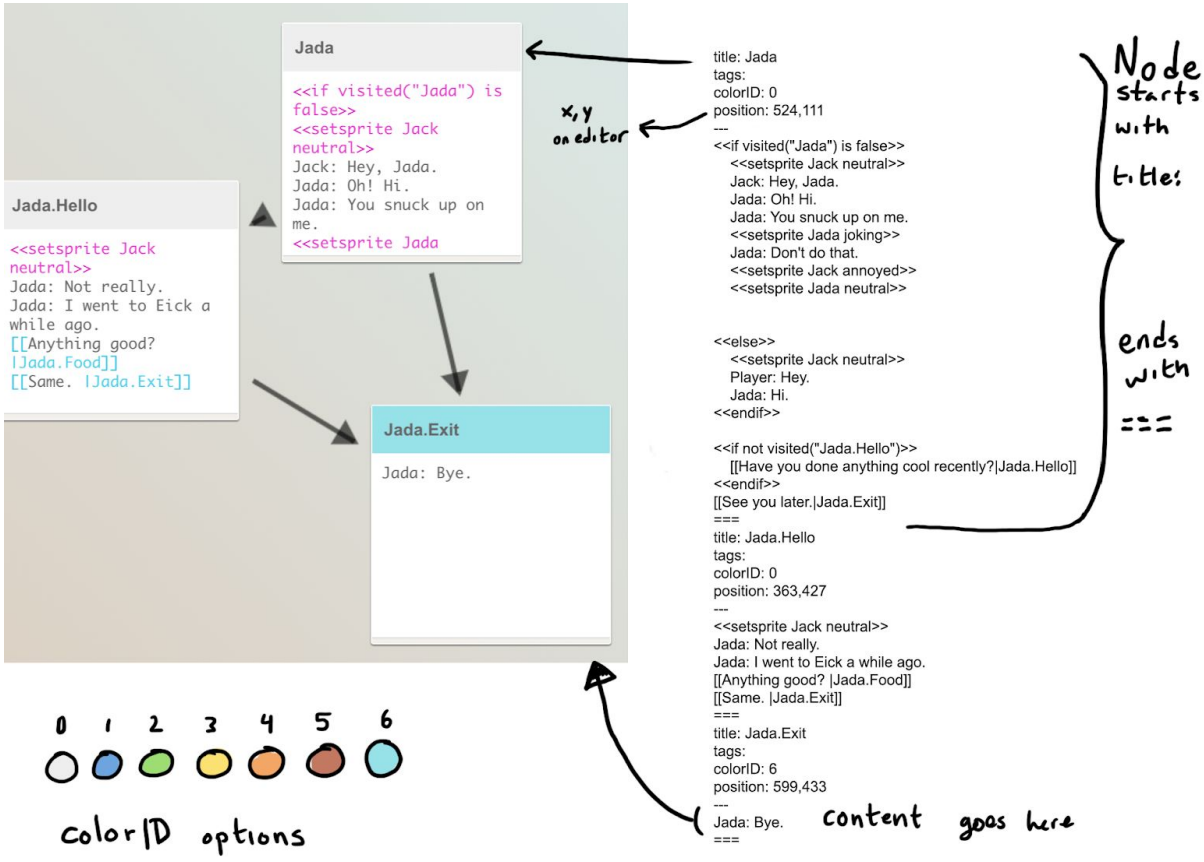
An example of an if statement

- In this example, Yarn is checking if the function visited (which checks to see if a node (Jada.Hello) has already been seen by the player) is true.
- You need to use the <<>> format
- <<endif>> is like your {} - you need it for the if statement to work correctly

Saving

- Keep the .yarn.txt or .json
- Avoid using spaces - party_start, partyStart, or party-start instead of “party start”

Writing Yarn in a Plain txt Doc:



An example of the written Yarn script compared to how it appears in the editor

title: Node

tags:

colorID: 0

position: X,Y

Name1: Hey, Jada.

Name2: Oh! Hi.

Name3: How are you?

===

Yarn Commands

To use a Yarn Command open and close with “<<” and “>>”

Extra Example:

https://docs.google.com/document/d/1vqrvPEKtyjJ8r6td-7ZoX0uuViRYDJyZSpG3yrBqu7s/edit?usp=sharing

title: Start
tags:
colorID: 0
position: 755,-3

A: I am a character.
B: So am I.

->this is a player option that doesn’t change nodes

A: This is dialogue again!

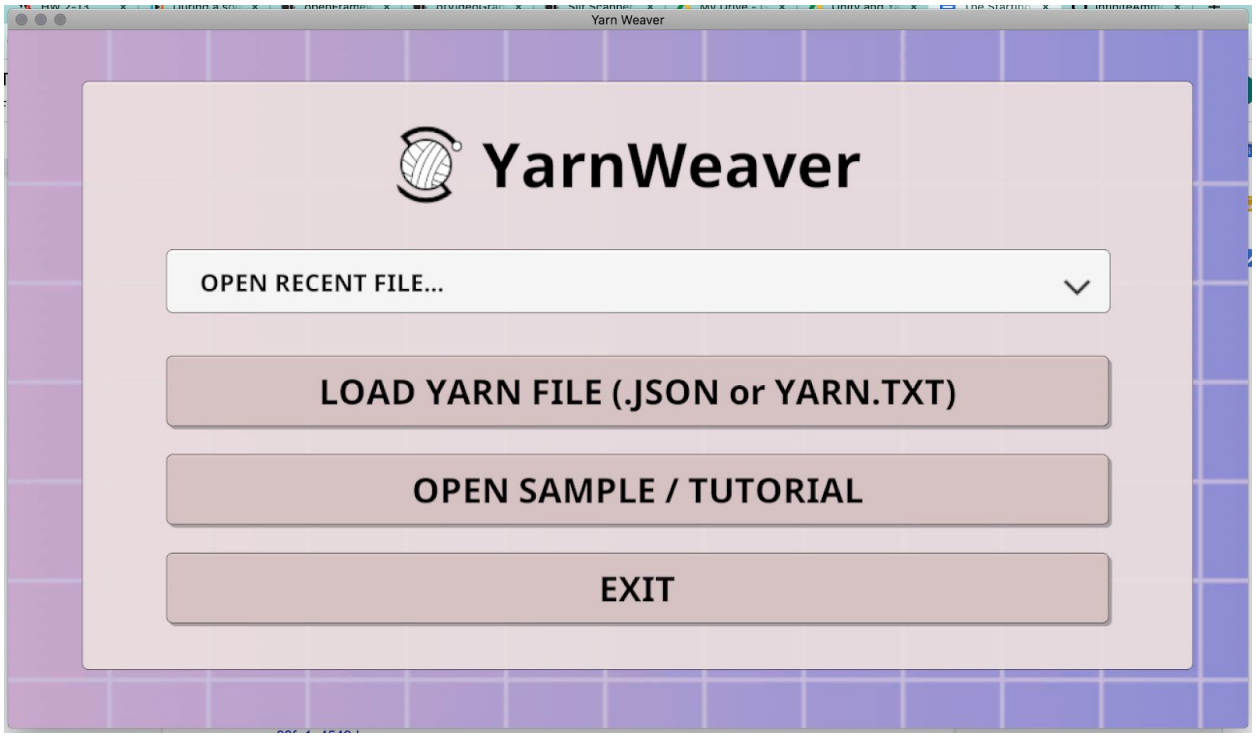
[[this is your first node choice |choice1]]

[[this is your second! |choice2]]
===
title: choice1
tags:
colorID: 0
position: 562,246
---
word
===
title: choice2
tags:
colorID: 0
position: 935,249

Empty Text

===

Validating your Yarn



https://www.blog.radiator.debacle.us/2017/09/writing-stories-dialogue-for-unity.html

https://github.com/radiatoryang/Yarn-Weaver/releases

Yarn Weaver is a tool made to test Yarn Dialogue. Opening a file runs it as dialogue, and it lets you know if something is going wrong!

This is also done in Merino! It was made by the same person.

Using Yarn in Unity

https://docs.google.com/presentation/d/1o22Uoshvt5EKkrquSf70RAgJqF0OyhblEhG4GSxQmH4/edit?usp=sharing

Variables in Yarnspinner

May 23rd, 2018

Pinned by @zaeche

Aura Dominator 7:42 PM

Hello. This is just a question incase I'm working with variables inside yarn spinner. I can declare a new variable by doing this right?

<<set $b1 to true>>

Unity will see that as a Boolean right?

mcjones 7:49 PM

that will save $b1 as a bool into the variable storage, Unity can then talk to the variable storage to get that back out as a bool

Aura Dominator 8:02 PM

Oh ok. So in a separate script can I change $b1?

mcjones 8:04 PM

yeah inside unity you can talk to the variable storage system in there, so if you are going off the example Unity script on GitHub on line 461 is where we've defined an example variable storage so you could use that as your starting point

erm sorry nope my brain is thinking stupid thoughts, https://github.com/thesecretlab/YarnSpinner/blob/master/Unity/Assets/YarnSpinner/Examples/DemoScripts/ExampleVariableStorage.cs is the file I was thinking of

> GitHub
> **thesecretlab/YarnSpinner**
> YarnSpinner - Yarn Spinner is a C# library for interactive dialogue in games!

Pinned by @zaeche

Untitled

```
<<set $random to 0>>
<<randgen DialogueRunner 3>>
<<elseif $random is 1>>
lne 1 goes here
<<elseif $random is 2>>
line 2 goes here
<<elseif $random is 3>>
line 3 goes here
<<else>>
rand still at 0.  something broke.
<<endif>>
```

and this for the randgen function:

I appreciate the help! 😉

Untitled

```
/// Generate a Random Number
        [YarnCommand("randgen")]
        public void RandGen(string totalOptions)
        {
            int optionsValue;
            if (int.TryParse(totalOptions, out optionsValue))
            {
                //Debug.Log("Parse Complete.  Total Options: " + optionsValue);
                int randomReturn = Random.Range(1, optionsValue + 1);
                Debug.Log("randomReturn: " + randomReturn);

                gameObject.GetComponent<ExampleVariableStorage>().SetValue("$random", new Yarn.Value(randomReturn));
            }
            else
            {
                Debug.Log("Parse not possible");
            }
        }
```

All the Links

https://github.com/YarnSpinnerTool/YarnEditor
The yarn spinner editor!

https://github.com/YarnSpinnerTool/YarnSpinner/blob/master/Documentation/YarnSpinner-Dialogue/Yarn-Syntax.md
Yarn's Guide to Writing in the yarn editor! Includes online stuff!

https://github.com/YarnSpinnerTool/YarnSpinner/blob/master/Documentation/YarnSpinner-Dialogue/README.md
Yarnspinner's guide to writing Yarn Dialogue

https://github.com/YarnSpinnerTool/YarnSpinner/blob/master/Documentation/YarnSpinner-Unity/YarnSpinner-with-Unity-StepByStep.md
Yarnspinner's primer for using Yarn + Yarnspinner!

https://github.com/YarnSpinnerTool/YarnSpinner/blob/master/Documentation/YarnSpinner-Dialogue/Complex-Dialogue-Tutorial.md
This is Yarnspinner's tutorial on how to build their complex tutorial!

https://www.secretlab.com.au/blog/2017/11/14/how-night-in-the-woods-uses-yarn-spinner
How A Night in The Woods uses Yarnspinner