

# Twine: Programming for Language Arts

---

Clarissa Littler

Updated as of March 15, 2019

## What'll we learn!

- What Twine is
- How to make a simple interactive story
- How to incorporate programming gradually into story writing
- Ideas for incorporating interactive fiction collaboratively with language arts classes

## How we'll do it!

- A brief introduction to Twine and key concepts
- We'll spend a little bit of hands-on time and try to make something neat
- We'll do some brain storming and talk about how to get Twine into the classroom
- We'll discuss how to deploy and share Twine stories

# What's a Twine?

Twine is a a mature, open-source, system for programming  
*interactive fiction*

## What's interactive fiction?

If you're a smart-aleck, you might say “isn't all fiction interactive?”

Absolutely!

*Ergodic*, or *path-dependent*, fiction might be more accurate

...but it requires explaining

## A brief glossary

**Passage** An individual scene/one square in the story editor

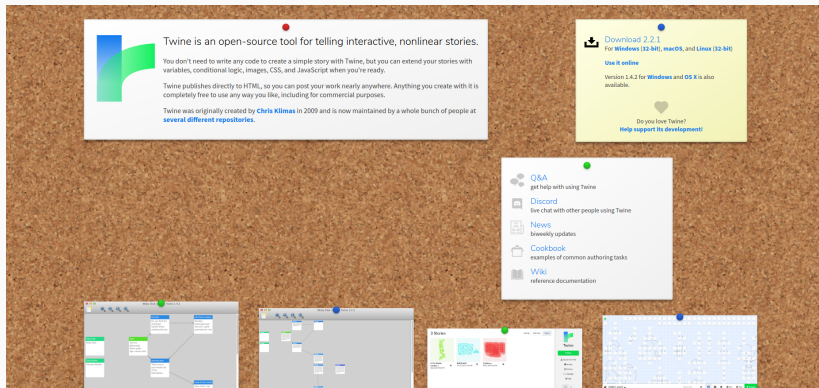
**Link** A way to move between passages

**Hook** Text that's between single square-brackets

# Getting started!

`http://twinery.org`


Navigate to here and click on “Use it online”




# Starting your story

## Click on the “+ Story”


0 Stories


Sort By Edit Date Name 


There are no stories saved in Twine right now. To get started, you can either create a new story or import an existing one from a file.


  
Twine  
  


+ Story

 Import From File

 Archive

 Formats

 Language

 Help



# Basic story

Editing a scene, or *passage*, is done by double-clicking on a passage



Making new scenes with *links*

```
[[Text to next passage]]
```

```
[[Text you want to appear->name of passage]]
```

## How do you play it?

Eventually you'll publish your story as an HTML file

But for now you'll just hit the *play* button

## That's the basics!...or maybe the minimum

With just those two concepts you can absolutely make a real branching story

Let's try it! Ready...set...go!

## Where's the programming?

This is neat, but where's the *coding*?

Twine has multiple ways to add code: *story formats*

We'll stick with the beginner-friendly default: *Harlowe*

# Variables

```
(set: $heroName to  
  (prompt: "What is your name?", "Hiro Protagonist"))
```

```
Greetings, $heroName!
```

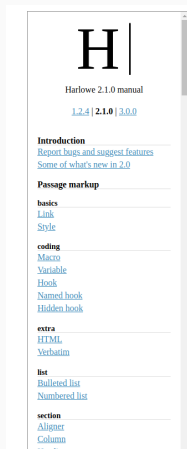
# Conditionals

```
(if: $hasFlashlight is true)[You shine your flashlight:  
    you see the entrance to the ancient Chicken Cathedral]  
(else:)[The dark clucks at you]
```

```
(set: $thingy to (a: "this", "is", "an", "array"))  
(for: each _item, ...$thingy)[_item ]
```



http://twine2.neocities.org



## Introduction

### Report bugs and suggest features

If you think you've found a bug to report in Harlowe, want to make a feature suggestion, or wish to see what future features are already planned, simply visit [the project's issues page](#).

### Some of what's new in 2.0

A lot of features have been added to Harlowe 2.0, many of which are designed to shorten existing code idioms or make certain workarounds unnecessary. The changes to existing features you should first familiarise yourself with are:

- The default Harlowe colour scheme is now white text on black, in adherence to SugarCube and Sugarcane. You can change it back to white using the instructions below.
- Expressions like `$a < 4` and `5` will now be interpreted as `$a < 4` and `it < 5` instead of always producing an error.
- Using `is` with comparison operators, like `$a is < 3`, is now valid.
- Changes can be attached to hooks with `whitespace` between them - `{(if: $coverBlown) [Run!]}` is now valid.
- Changes can be attached to named hooks - `{(if: true) [moths$[Several\ moths!]}]` is now valid.
- Changes can be added together using `+` while attaching them to a hook - `{(font: 'Shatter')+(text-style: 'outline') [CRASH!]}` is now valid.
- The default CSS has been changed such that the story's `font` must be overridden on `tw-story` rather than `html` (for consistency with other CSS properties).

The following new features also deserve your attention.

- The built-in `?page`, `?passage`, `?sidebar` and `?link` hooks
- Hidden hooks, and the `(show)` and `(hidden)` command macros

## Is Twine just text?

Absolutely not!

You can easily include images by including the raw HTML into the story

```

```

You can also include music and video this way!

## Complex clicks

```
On the table you see a [letter]<letter|  
(click: ?letter)[  
My dear friend,
```

```
I hope this letter finds you well.
```

```
If you pass this along to three friends...  
]
```

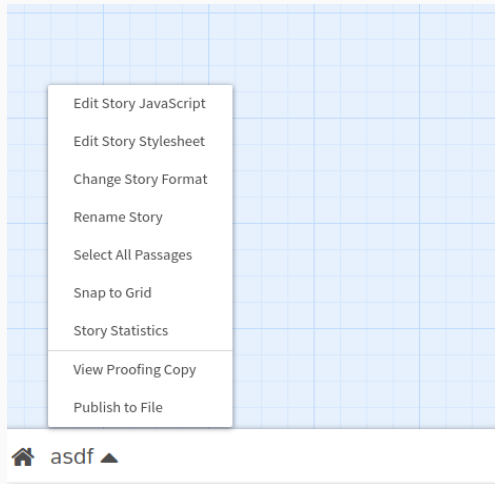
Let's just put together an example right now

## Try adding something to your story!

See what you can try adding to your story with another few minutes of work.

# Publishing games

When you're done you can publish your story by going to  
“Publish to file”



# Publishing games

## philome.la: free twine hosting

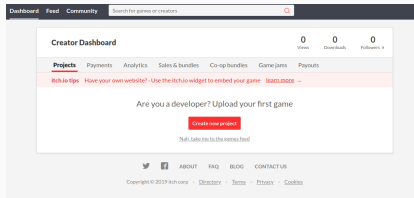
1  
Login with Twitter

2  
Upload your built .html file

Drop your file here

3  
Name your game

Publish!



## How to get Language Arts classes involved

It's brainstormin' time!



## Lots of things we couldn't cover!

- Data maps
- Building inventory systems
- Full rpg mechanics
- CSS
- JavaScript

- <http://twine2.neocities.org/>
- <https://twinery.org/cookbook/>
- <https://clarissalittler.github.io/interactive-fiction/TwineDemo.html>
- <https://github.com/clarissalittler/interactive-fiction>