bitsy workshop
for nywf at nywf
by @haraiva

→ bit.ly/nywf-bitsy
Hello I'm Cecile and here are some fun facts about me:

I'm a Melbourne-based graphic designer/illustrator/comics artist/tiny game dev.

I am non-binary and my pronouns are they/them.

I like magpies and AFL.

This year I made two games so far, Novena and Touch Melbourne.

I am here to talk to you about Bitsy.

Website: cecile-richard.com
Twitter: @haraiva
Games: haraiva.itch.io
what’s bitsy?

“bitsy is a little editor for little games or worlds. The goal is to make it easy to make games where you can walk around and talk to people and be somewhere.”

– adam le doux, father of bitsy

what should i make?

you can make (almost) anything! short narrative games are the most common, but you could also make a vignette game, or a tiny adventure. some people even made a bitsy mmo!

for inspiration, you can take a look at the bitsy pieces twitter or claire morley’s cool examples of cool games at the bottom of her tutorial.

i like bitsy because

- it’s easy and satisfying to use.
- pixel art is (mostly) accessible and fun.
- it’s versatile while also having constraints.
- the community is cool
- there’s a bitsy jam every month!

bitsy has a growing, diverse, extremely supportive community, and is a genuinely exciting tool because of its versatility.
→ to bitsy editor
let’s start with the basics. once you have the editor open you can take a look at the various tools/panels. toggle your toolbar by clicking on the tools button next to the title field.

a room is a place or scene in your game world. you can make multiple rooms that you connect with each other using exits. a tile is a piece of scenery in your room. you can make your tile into a wall, making it impassable.

the avatar is the player character, which you control using the arrow keys, and you can interact with sprites by walking up to them.

an item is a sprite you can pick up. you can assign variables to your items if you want, but you don’t have to. an ending is an exit that ends the game. you can have multiple endings.
each tile and sprite’s canvas is a 8×8 grid. you can draw in the paint panel by clicking on the squares.

once you’re done making your avatar, you can click on animation to make a two-frame animation. edit your second frame and now your avatar is moving!

this process is the same for tiles and sprites.
the one thing that’s different in tiles is the **wall** button.

you can toggle this to make your tile **impassable**, which means your avatar can’t go through, or on it.

you can make as many tiles as you want.

also, you can only have one iteration of a given non-item sprite in a game, so if you want multiples of the same sprite, you have to duplicate them.

don’t overthink your drawings! in pixel art it’s important to get across the essence of whatever you’re drawing, but it’s also okay to keep it vague. vagueness can bring interesting results.

don’t forget to name your tiles and sprites! it’s helpful when you end up having a lot of them and you can easily search them using the **find** button.
Room design

Now that you have some sprites and tiles set up, it's time for some interior design. Let's put the tiles in your room!

You can find your tiles and sprites easily in the find drawing panel.

Place your tiles in your room. Click and drag on the room if you want to place a tile multiple times quickly.
**room design**

toggle the wall button in the room panel to see where your walls are.

click the play button to, well, play your game. that way you can check if your walls work, and you can also just get a feel of how to navigate your game.

once you have at least two rooms, you can connect them using exits.

---

**tips**

i generally roughly plan out my rooms on paper, then create them, adding placeholder tiles where i want my walls and exits to be.

there are some bitsy pieces that take place only in one room, but you can add as many rooms as you want, so you can go wild with them as well.
you’ve heard of “making an entrance”, let’s make an exit.

open the exits panel, and click the place new exit button. a mini-map of the room will appear with a dropdown list of all your rooms. select the room you want, and then click in the mini-map to create an exit destination.
while the exits panel is still open, click on its tile in the room panel to select it, and the mini-map will reappear so that you can click on a new tile for the destination, or drag it over to the desired tile.

exits are commonly used as doors between rooms, but I’ve seen them being used to make animations, show the passing of time, or making sprites appear out of nowhere.

if you’re placing them as doors, it’s important to keep continuity in mind, as to not confuse the player (unless that’s your aim). for example, it makes sense for the character to exit at the right-side and enter a new room from its left side.

always remember to test your game to see if your exits work as you want them to.

you can add exits going back to the previous room to make it two-way.

generally, you want to make it obvious where the exits in a room are, unless it’s part of your game to make it confusing.
now that we have most of the basics covered, let’s talk about **dialog**.

Sprites and items are the only elements to which you can add dialog.

To edit dialog for each sprite you make, go to the paint panel and type into the **dialog** field.

Now click play and walk up to the sprite and it will talk to you!

tips

dialog is usually one of the last things I add to my game.

What I do is write down all of my dialog in a text editor and then copy/paste and edit that dialog in bitsy for each of my sprites.
with your paint panel still open, click on the dialog tool to open the advanced dialog panel.

there you can find different kinds of dialog sections (simple, list, conditional).

and text effects! you can change colours of your text or make it wavy, or shaky, or rainbow, or all of them at the same time!

list dialog is used if you want your sprite to say a lines in a sequence (say the lines only once), a cycle (say the lines once then repeat from the first line), and shuffle (say lines in a random order).

conditional dialog is used in association with item variables. for example a sprite will say a line only once you have picked up a specific item.
colour palettes

Set **colour palettes** for your game in the **colors panel**. You can make multiple colour palettes and apply them to different rooms by selecting the one you want in the dropdown menu in the **room tools**.

inventory

The **inventory panel** is the one you use for setting variables. Check out the **tutorials** if you want to know more about them.

game data

In the **game data panel**, you can find the text version of the game itself. You could tweak it in the editor, but I recommend pasting it in a text editor like **sublimetext** for that. This is also the data you need to use with hacks like **borks**.

settings

In the **settings panel** you can change the **background colour** of the HTML file of the game (so that it matches the background colour of your room), set the **game window size**, and change **dialog fonts**.

download game

In this panel you download your **game’s HTML file**. That’s the file you need for uploading to itch.io. You can also **import** a game from its HTML here.
useful tools

- **image to bitsy**: turn any image into a bitsy room.
- **pixilart**: online pixel art tool I use for making room backgrounds. Set the canvas to 128x128px, draw, save. Import your drawing to image to bitsy, get the code. Easy!
- **fontsy**: make a custom font for your bitsy’s in-game text.
- **spritesy**: sprite editor, mainly used for collaborations and multiple frame animations.
- **borksy**: tool for incorporating various bitsy hacks easily.
- **sublimetext**: for working on your html code outside of the bitsy editor.

tutorials

- **text tutorial**: this is a tutorial by Claire Morley. It’s very good and inspired me a lot for these slides.
- **video tutorial**: if you’re more of a video person, Dan Cox has got you covered.
- **variables tutorial**: a tutorial by Andrew Yolland about variables, which you may have noticed in inventory.
You can add music to your bitsy using the aforementioned borksy, but if it confuses you (understandable) or if you want just one track looping in your game, you can:

1. Place your audio file somewhere relative to your bitsy html file (in the zip if you’re uploading to itch.io). Name it music.mp3.

2. Open your bitsy html in sublimetext, copy-paste:
   ```html
   <audio loop="loop" autoplay="autoplay">
     <source src="music.mp3">
   </audio>
   ```
   in <!-- DOCUMENT BODY --> before the </body> tag.

This works only if you upload on itch.io. Test if your music implementation is working either inside your itch project by making a private upload/project or in a browser other than google chrome.
here are some examples of what you can make with bitsy. click through to play.

- Vitreous by Breogan Hackett
- Noodle Escape! by Mary T., Pete Ellison, Vince Bondy
- Forgotten Depths by Mark Wonnacott
- Bitsy Spider by Rabbitboots
- Silverybield Foss by Twisted Tree
- The Eviction Game by Raul Haagensen
- The Right Song Now.mp3 by Karina Popp
- Novena by Haraiva (me)
thanks

@haraiva

cecile-richard.com

thanks