I would like to say thank you to all of you on this call. I am so happy to see all these familiar faces during this scary and precarious time.

So the schedule for the workshop will be a very baby lecture and then we will all be coding in HTML and CSS in these pre-made Glitch web pages linked in the Google Spreadsheet next to your names. So maybe while I’m talking y’all can just open them up and make sure you have a working link. I’ll explain more about how these pages work later but each of us will end up with something that looks like this -> a web page with a bunch of text and colors and formatting. Ok so before I begin does anyone have any questions!

1. COMPUTATION AS METAPHOR

In leading up to earlier versions of this workshop, I had been thinking a lot about the computation and metaphor. More specifically how the metaphor of the desktop was claimed along with the personal computer as “for everyone” but really it was for everyone who was already familiar with the system of the office.

I am perpetually perplexed that nearly everything I do as a programmer is somehow removed from what my computer is physically doing and instead I have been given sets of metaphors to remember and understand how to do those things within this visual interface of the desktop.

This desktop metaphor ensured that computers require programming, as we know it today, to be a set of extremely abstract processes. So, when I’m coding something, I’m often coding on top of someone else’s code, in someone else’s software, on top of the software that makes up my operating system which is permanently baked into the proprietary silicon chip of my Apple computer.

2. ONES AND ZEROES

Computers at their core consist of a complex series of on and off switches, ones and zeroes. This means that everything we do on our computers today is already abstracting away those complex patterns of numbers in order for us to use them in the way that we do.

These abstractions allow us to do things like writing text in a file, without thinking about how or even if the computer will be able to do this. I think it is important to note that since the early days of computing computer time has always been more important than human time. When programmers were writing code for computers the size of rooms the time it took to run the program was really expensive, so people were always writing code that was most efficiently read by the computer itself not by the people who were making the programs. Hence the situation we’re in now where code is written in often illegible jargon and minified into unreadable blobs of text.

3. DESKTOP

Going back to this idea of the desktop metaphor. The idea of a file inside a folder on a desktop...and then there's the trashcan. I’m always wondering, when you put something in the trash, does it ever really go away? And what is the file really? And what is the folder?

**Melanie Hoff** unpacks some of these questions in a class that they teach called Peer To Peer Folder Poetry. Even though all files at their core are a series of numbers ~ the organization and re-making of those files can be a poetic act. A computer scientist friend once told me that folders aren't actually anything. They don't take up \*any\* space on your computer.

These days I often reflect on something **Kameelah Janan Rasheed** once said in a class she taught, how analogies can be a poverty of language. In step with this provocation I am always wondering how the metaphors of files and folders limited our imagination for what computing can be? If not for the baked in interface of the desktop what might we put on a screen emitting light?

Or to think about it in a different way, what would it be like if we could hear all of the computation happening inside our laptops as we typed out an email?

4. ILLUSION OF POWER

This is all just to interrogate the fact that there has been a significant shift from physical programming, the act of physically adjusting switches or weaving software into core rope memory to abstracted programming, building software on top of software.

Since folders and files, which if we looked closely at them don’t resemble anything close to a piece of paper inside another folded piece of paper, I’m left wondering does the way that we use our computers also require some kind of illusion of control over our computers. The key word here being illusion.

5. PROGRAMMERS

Programmers, through the creation of complex frameworks and algorithmic systems go through so many hurdles trying to get the computer to do what they want it to do. When the computer finally obeys their commands, they are granted with this feeling of immense gratification and power. I’ve felt this feeling of gratification as a programmer after hours and hours of trying to debug something I wrote. It’s a cyclical command and response from you to your computer. My provocation here is not to call for an undoing of this cycle of command and response (although I’m interested in that) but to become aware of what is happening in between. So instead of humans controlling computers and computers controlling humans I am interested in how computers can be between, not over or under, the people that use them.

6. IN A VACUUM

This is a video of my grandma writing an email to someone. It makes me wonder what it would be like if programmers had to press the keys of their keyboards so gently that it required them to move extremely slow and therefore with care?

And so, all of these questions have led me to care deeply about hand coding. Hand coding is the process of coding that bridges the gap between you as the programmer and your browser as the programmed. In other words, it is a way to code which is not dependent on someone else’s software or framework or library, instead it is the process of writing in the inherent language of the browser.

So I’m kind of lying to you here because your browser is software but hand coding in the software of the browser is what I like to call closer to the metal of your computer then doing something like making a Wordpress website which is software on a big server depending on fragile code written by tons and tons of people and then minified into HTML CSS and Javascript. Not that this is bad or good but if we are interested in questioning relationships around intimacy, transparency, encoding and decoding, file systems, and proprietaryness then doing something like hand coding might be a step towards new questions about our relationships to these things.

[OPEN CODE SOCIETIES GLITCH PAGE IN CHROME]

So your browser is software which translates HTML and CSS into web pages. Your browser is the performer of the program you are writing.

The page we are looking at is from this past January when I led this workshop at Code Societies at The School for Poetic Computation.

The code you will write in this workshop will be very particular and you will write out each character slowly and by hand, there will be very little copying or pasting. We’ll move as slow as we want to and I will also be writing all the code out with you on my screen. If looking at my screen share and working in the browser becomes too unwieldy I’ve also linked my example page in the   
Google Doc so you can follow along there.

Ok so You will also do this on each other pages. The intention here is to perhaps erode this feeling of ultimate control that you have over your computer's system (and in this special virtual edition, your web page) and to move carefully around someone else's. So, in this session will be in discussion with each other not necessarily in a verbal way. Instead we’re going to be in conversation through the things we are writing on each other’s pages.

I am in the camp of learning by doing so instead of me going on about HTML structure and CSS syntax we are just going to write code in these languages and then immediately see what they are doing in the browser. I will be typing out the code with you in my example page so if you ever get lost or stuck you can ask me a question or check my page as a reference.

So the round robin part works like this, we’re going to start at our own pages, I’m going to give you a prompt, we’ll all code along with what I’m doing on the screen, you’ll respond to the prompt in your own way, then after a like 3 minutes I’ll ask you to stop typing and close your web page. We will all meet back in the spreadsheet. I will shuffle the links and then you will click the new link next to your name. If you're not finished don't worry we'll try to take on something else I learned in a class taught by **Kameelah**: which is to embrace the unfinished sentence.

So we will do this switching of pages like 5-6 times depending on time and then go back to our original pages. The intention is to have a bunch of beautiful hand coded pages at the end that we can share with each other. Tonight I will compile them all into one singular web page that you can share or go back to months or maybe even years from now! All of this can be as anonymous as you want it to be just know that whatever you write will be seen and responded to by someone else at this virtual table. Which brings me to the final thing I want to say before we start coding: Please **engage with each other’s writing from a place of RESPECT,** any content (writing, images etc) that is racist, sexist, homophobic, or in any other way discriminatory will not be tolerated. I’d also like us all to be extra aware that we are all experiencing this fucked up time in extremely different ways and its super important that we are all carefully listening and responding to each other as we move through these prompts.

OK! Are there any questions before we switch over to coding?