Chapter 4: Coding Literacy Echoes in Black Lives

> Anything that comes quick and fast, leaves and goes just as early.
>
> – Rapsody, from the Netflix docuseries *Rapture*

> Sometimes you don't survive whole. You just survive in parts. But the grandeur of life is that attempt. It's not about that solution.
>
> – Toni Morrison

On a cold morning in late March 2017, adult learners gathered in the classroom and started the next day of their training with the stand-up ritual. Richard and Jessica asked each adult learner to share what they hoped to get out of tomorrow's tour of a company that offers help desk software to businesses. Tours of tech companies were opportunities to learn from professional software developers and managers and get a glimpse of shiny offices that mixed business with lavish perks. I would join the class on tour of a different company that overlooked downtown Sakowin. Just across the hallway on the other side of a wall was a break space with arcade games, billiards, a kitchen, cupboards full snacks and mini fridges with drinks. Touring a tech company was also a chance to impress someone working for the company. Networking was at the forefront of Alex's mind. He interrupted stand-up with a question about his "personality quirks." Alex told Richard and the rest of the class that he can be "blunt and frank … How do you deal with that when you network?" Richard, taking command of the front of the room, responded, "Practicing performance. Create a character. I know my background, my [body]size … Who you are depends on what you want people to see you as … You need to convince people." Later that day everyone was scheduled to practice elevator pitches, so Richard used that activity as an example of being who you need to be for clients and employers. "What gets you the job is communication," explained Richard. "Selling yourself. Everything else is just work."

He himself was quite familiar with putting on a performance to attract business. In the 1990s, Richard designed free websites for friends. As one of the few Black web designers in the country at that time, Richard wanted to get into the "white business world" where he knew lots of money awaited. When he lived in Indianapolis, Indiana, Richard hired a tax attorney who happened

to have connections to white corporate America. He also had a terrible website. In their first meeting, Richard offered his services to the tax attorney, pointing out in a joking and professional way that the attorney's website could use some work. The tax attorney made no excuses: he agreed the website was a mess. Finding common ground on their reality, Richard shifted the conversation. If the attorney helped with Richard's taxes for free, Richard would redesign the website in return. A month later, the tax attorney approached Richard with good news and bad news. The bad news: Richard owed taxes. The good news: He knew many other certificated public accountants who needed websites. Over the years, Richard's reputation grew such that clients came to him by referral. The elevator pitch, Richard reasoned, was a brief space to say what you need to say to get the job. "If you can take away people's pain, you make money."

But taking away pain with carefully designed websites and mobile applications was only part of the story. Richard knew well what he was asking Clearwater Academy adult learners to do: like him, they should enter white corporate America. That's where the money was. That was where social mobility was. That was where their digital literacies would best serve them. Richard admitted that there was a higher bar to climb when you're a person of color. He was blunt: "Show them that you can work with a team of white men because you are competing for jobs made for white men." That should not terrify them or push them away from the job market in the tech sector: they needed to be the face of the economy. Richard continued his lecture: "Services and goods need to be made for marginalized people and without those people in the industry, those services and good will never be made. If you avoid the company, nothing will change." Richard had to transform himself to stave off discrimination: He was a tall and large Black man raised in a Muslim family. The terrorist attack on New York City on September 11, 2001 made the cultural and political climate too hot for Richard. He changed his name to compete for white clients and navigate around their suspicion that a freelance designer with an Arabic name can't be trusted with their technology. And yet, at the same time, Richard and Jessica emphasized agency, control over the job market. There was no easy answer on how to be a professional. When you work with a client you must ask, "Who do you want to attract?" As professionals, Richard explained, that's a "decision you have to make for yourself."

Richard's lecture represents several themes germane to this chapter. First, Clearwater Academy's curriculum teaches coding literacy and employability skills, but Black adult learners can't filter out their racial identity, political affiliation, and how others perceive them. Their worth as coders gets bound up in social and political ideologies. Black adult learners must practice performative rhetoric to shape employers' perception of their race and literacy practices. This

theme recalls Chapter 3 where Black adult learners sensed they had to put on a show for majority-white tech companies, an additional burden to an already difficult job market. But a second theme emerges from the lesson: Their coding literacy practices afforded them many possibilities for the kind of professional they wanted to be. Black adult learners can navigate the world with the confidence that they have worth and dignity. Being Black and a coder can take you places. So there's the difficult mix of coding literacy on the job market: The realities of racism and needing to create a racial brand identity that does *and does not* center whiteness in the work life of Black adult learners.

In this chapter, I argue that while coding literacy itself does not help Black adult learners overcome sociomaterial and cultural barriers to social mobility, the coding literacy they learned—its practices and knowledge—continues to echo in their lives in other ways and promotes some sustainability. I contend that coding literacy education for work is a clarifying tool for Black adult learners. The unique qualities of coding literacy (the processes and knowledge they've learned mixed with their own assets) brings clarity to where else they can accumulate digital literacy and to the directions Black adult learners may take their labor and skills on their own terms. As a clarifying tool, computer programming promotes agency for Black adult learners in general. This clarification on literacy and labor comes with baptism by fire, however. As they begin their post-graduation journey on a job market that seems rife with potential jobs in software development, Black adult learners discover how racism persists in tech, how too slow the industry moves on hiring them, and how, when the industry does finally hire them, judgements on Black adult learners' value continue to be overlooked or ignored. Other personal circumstances determine how they move in the economy with such a prestigious literacy. Just as their worth as coders are being judged by tech, Black adult learners in this study make judgements about tech and computer programming. The institutions that structure and judge their worth provide clarity on their identities as literate workers and their position in the economy. These institutional and social structures shape how their literacies develop. The value of coding literacy begins to unravel for Black people. The pipeline to prestigious work and social mobility, so evident in discourse on vocational literacies, remains leaky for many. Some fall through these holes from the start or they never enter the pipeline. Participants *do* land in other pipelines, some of them they created themselves. Sometimes the tools of coding literacy echoes, and other times Black adult learners lean back on their own assets and communities. Where before, I would say, they had some cloudiness on what to do with literacy and work, after they experienced coding literacy, and after interacting with its sociocultural forces, they discover logical processes toward finding new possibilities.

I use post-graduation interviews as the basis for my argument. Built into my study design was an effort to keep track of what participants did after finishing Clearwater Academy. My longitudinal study was very small and modest, given the time constraints of my writing: check-ins three months and then six months after finishing Clearwater. I focused on these specific, quick turn-around updates for another reason: To put to the test the discourses of Clearwater Academy, and other computer code bootcamps that followed a similar model, that suggested adult learners *may* get a quick return on their investment with a paid internship or full-time work. As a qualitative researcher of literacy, I was less interested in the success of getting a job and the speed in which that job came and more interested in learning the ways adult learners develop their coding literacies further across different locations where others make judgements on Black adult learners' worth as human capital and where Black adult learners make similar judgements about themselves and the industry.

All participants agreed to do follow up interviews and gave me their email and phone numbers. However, as is common in longitudinal studies, most of the 12 participants were difficult to contact. Some I followed up with three times, such as Isaiah; many only once at the three-month mark but not again at the 6-month mark, like DeAndre. Rosie, I interviewed twice and nearly a year after she matriculated into Clearwater Academy. Thus, most interviews occurred throughout 2018 and into 2019. I'm fortunate that even three-month check-ins lead to rich details that could interlock with interviews that covered longer trajectories, creating as complete picture of their lives after taking up coding literacy practices as possible. I include one 2018 interview with a Black adult who had graduated in 2016, a year prior to the start of my study.

This chapter covers the post-graduation life of Black adult learners. I break them up into three groups of graduates with interlocking themes about coding literacy's influence on prospects for social mobility and navigating racism: Four stepped away from software development; two succeeded with paid internships but dropped out of software development by the end of their programs due to microaggressions and stereotype threats. This chapter uses interviews conducted with those participants three to twelve months after graduating from Clearwater Academy. I find that a year is too short, so I include two unique interviews: one interview I conducted with a full-stack developer who had graduate two years before my 2017 study. The second an interview with Zeus who had "disappeared" in 2018 and then unexpectedly returned to my attention on Snapchat five years later. My January 2023 interview with Zeus covered his post-graduation life since 2018. His longer trajectory was a welcome and rare find that highlights the downstream impact of a computer code bootcamps on one person's journey through the economy. In total, my interviews cover seven years of individual lives: 2016 – 2023.

Although just eight participants, the variety of exposures to these Black adult learners through interviews provide rich details on coding literacy's echoes in their professional and personal lives. In addition, by considering each type of career trajectory post-graduation, I show how they move from context to context, from work without computer programming to having no job at all, for short term literacy development and long-term literacy development.

I've designed a chart in this chapter that provides a snapshot of the variety of directions Black adult learners went after graduating. I include participants whom I did not formally interview but rather, curiously, happened to bump into during my day-to-day activities, like walking down the street, visiting a coffeeshop to write, or visiting a bar with friends. These are brief touchpoints that, although not super detailed with context, glean what happened to them after graduating. For example, I met Pierre in a downtown coffeeshop in summer 2018. He was behind the counter handing out drinks to customers. After we exchanged some pleasantries, I suggested we meet and catch up some time. Pierre's response has stayed with me all these years later: "I gotta go and finish making these drinks." In Chapter 3, Pierre had expressed concern he wasn't going to have enough experience with coding literacy to get a job, and in Chapter 2 I wrote that he had learned from a Clearwater Academy graduate that he needed to be patient—a software developer job will come soon. His position after graduating from Clearwater Academy six or seven months before had not deepened his place in tech industry, where opportunity and creativity abounded for others. The response to my suggestion that we chat reveals another layer for why I had not re-connected with all participants: shame. The idea had not occurred to me until 2023 when I interviewed Zeus. He was still in contact with Pierre, and Zeus speculated that maybe some people read my emails, text messages, and calls, but *they didn't want to pick up*. They had not done anything with computer programming, and they thought I was looking for that exact thing. "Some of them," Zeus said, "were probably sorry that they could not help you more." But I show in this chapter that it's not their fault. And that they have done better.

First, I discuss the precarity of employment and earnings potential for Black adult learners in the United States. That section begins at the high-level before zooming in on the job market for software development. Sociologists, I observe, agree with what literacy studies has long known about digital literacy and what its acquisition does and does not mean. But I draw a positive throughline to find a more nuanced take on what Black adult learners who have gained coding literacy do on the market and what they do for themselves. Then I show how Black adult learners navigate the job market; wrapped around these decisions are the judgements made about their worth as coders in tech, the judgements they conclude about themselves in relationship to those judgements, and even their own conclusions about computer programming itself.

Table 4.1 Job Placement and Education of Black Adult Learners in Order of Graduate Year

Participant	Year of Graduation	Job Placement	Education
Gerrard	2016	Full-stack developer	Certificate training in cyber security
Alex	Spring 2017	Director of Communications	---
Alice	Spring 2017	Credit union document designer	Seeking associate degree and IT certification
DeAndre	Spring 2017	Freelance web designer	Seeking associate degree and IT certification
Isaiah	Spring 2017	Web developer intern; stay-at-home artist and father	Completing associate degree
Kevin	Spring 2017	IT help desk	Seeking IT certification
Nadine	Left Clearwater April 2017	Reportedly houseless	---
Rosie	Spring 2017	Continuing retirement; community volunteer	---
Myra	Fall 2017	Certified nursing assistant	
Rania	Fall 2017	---	---
Pierre	Fall 2017	Barista	---
Zelda	Fall 2017	Web developer intern	---
Zeus	Fall 2017	U.S. Army IT specialist	Seeking cybersecurity certification

The Job Market and Shaping Black Coders' Literacy Development

What do Clearwater Academy Black adult learners enter after graduating? In Chapter 3, I highlighted hiring practices. Here I dive deeper into data from sociology that shows how institutional racism determines how Black adult learners experience the job market. In other words, I take the perspective of the job prospect, not the hiring manager assessing their value through coding literacy practice. Although computer code bootcamps, and other short-term programs, promise a significant rise in social mobility, the job market has been fraught with inequality between Black and white workers in the decades since the Civil Rights Movement. In the wake of the Civil Rights Act and

affirmative action, the Black unemployment rate since the 1970s has remained stubbornly high, even in moments of economic prosperity. With a college degree, or additional graduate degrees, employment fairs better for Black adult learners, although their earning potential still drags behind white people. Yet even controlling for educational attainment, the earning rates trails white workers (Rodgers, 2019). Experiences on the job market worsens for Black adult learners holding only a high school degree, which for many Black adult learners in this study have with some college education. The criminal justice system is more likely to cut off access to high-waged work, especially young Black men who commit non-violent drug offenses (Loprest et al., 2019). The examples above are just two puzzle pieces of a larger narrative that's difficult to pin down; there's no single way to explain racial inequality in employment and earnings, but that perhaps indicates how structural racism works: a massive web tangled in different ways across education, housing, healthcare, citizenship, and other institutions.

The job market for software developers suggests it can carve out a safe space for marginalized people—where talent and merit become the basis for employment and earnings. Across a variety of roles in the technology sector—data science, cybersecurity, database administration, web development, information research, and IT management – software development is projected to grow by 23 percent over the next decade (2022 – 2032). To place my 2017-2018 study in context, the US Bureau of Labor Statistics expected employment in software development to grow by 25.6 percent (Dubina et al., 2019) yet only 4% of Black coders will share in this latest projected growth; in addition, the same 2023 report shows further that the wage gap between Black and white developers persists in tech.

While race and gender discrimination limit the opportunities of marginalized people breaking into tech, geography adds another layer to the uneven distribution of highly coding-literate Black people. Tiffany Chow's work has been instrumental in helping me think about how place and space bind to coding literacy and social mobility. As I show later, for some Black adult learners, geography is a major factor in their success on the job market. Degree-holding computer science graduates focus their job search on "brain hubs"—cities that have a high concentration of tech companies and a dense pool of potential hires, such as San Francisco, California; Austin, Texas; Seattle, Washington; and Raleigh-Durham, North Carolina. Chow writes that these hubs "[confer] real advantages, such as job matching, career specialization, and higher salary, for those graduating with a computer science degree and who want to secure a job matching their degree" (2022, p. 4). However, tech companies and Silicon Valley do not always expect coders to come to them; they spread their enterprise throughout the country and create regional

offices to exploit regional coding literates. Extending these findings, Chow surveyed alumni from three Texas universities to discern how an educational institutions' proximity to a major tech hub city determined the successful hiring of Latina computer science graduates. Chow found that "alumni from universities near a major tech hub are more likely to earn higher wages in desirable job markets compared with graduates from a university located far from a dense tech hub" (2022, p. 14). The study shifts conversations on the value of coding literacy for social mobility: Whereas the original discourse on computer programming suggested that coding literacy alone will be necessary for the economy, geography adds an asterisk. Where did you study? What resources were available at that institution? How far away are those resources from a dense tech hub city?

Geography and its association with social mobility echoes a lot in my own study at Clearwater Academy. During an interview over summer 2017 between semesters, Jessica worried that Sakowin had "a mini brain drain." Sakowin had a vibrant startup culture, and Sakowin University, although not near a major tech hub, had social and material resources that could send its graduates to internships in, and eventually work full time for, Silicon Valley. So college graduates keen on honing their coding literacies could exploit the startup culture and then leverage both college coding literacy and small business experience for high-paying jobs (relative to cost of living, of course) in San Francisco or Austin. The brain drain is hard for tech companies in Sakowin, so they do need a backup pool of coding literates, people they know will stick around for a long time and not run to Silicon Valley. Clearwater Academy graduates were not going to go as far as a college graduate; that Black talent would stay in Sakowin. Clearwater Academy was designed with their reading of the tech job market. Clearwater Academy's adult learners didn't dream of making it big. Most, Jessica explained, wanted to go into gaming, which she called an "exclusive industry" all on its own. Fewer, in her experience, have dreamed of Facebook or Google. Richard reeled in these big dreams early and often in the program. He explained,

> The level you can compete at is not where you would be at for that kind of job, competing against people who have been in college for the last ten years to do that one specific thing they're applying for. Fourteen weeks is not gonna compare to that. That's a fact. It's just impossible. I feel like if I'm not honest about that, I'm setting them up to be dreamers. And I have realistic expectations about what they want to achieve and the amount time it takes to make it happen. Because if you think you can make it to Google next year, expect to have

> a mop and broom, unless you got the skills you learned outside of class.

The closest Sakowin had to a tech company on the same level as Google was a local healthcare software company. It was notorious for having low retention for new hires: the company often hired young college graduates from the local university and burned them out in two or three years by sending them across the country to consult with hospitals. They would make plenty of money, but their bodies and minds would be wrecked. Richard and Jessica knew no Clearwater Academy graduate would work for that company, at least not as a software developer.

If they weren't fresh out of college, they were recruited for their years' long experience in tech. Recruitment into a top-tier company was the most likely pathway for a Clearwater Academy graduate. In fall 2017, a white alumni visited class to speak about her extensive experience on the job market. She had graduated from one of the first classes of Clearwater Academy and worked her way through small companies in Sakowin. The accumulation of coding literacy experience increased her worth as a coder to the point she no longer needed to apply for jobs. Tech companies came to the graduate via LinkedIn hoping to recruit her. Her hard work at the bottom finally paid off. She was happy to announce to the class that she was moving to Austin, Texas to be a software developer for a big company. But there was plenty of work and plenty of money to be made in Sakowin; that's where Clearwater Academy graduates should think about. What I find then are multiple hurdles: not just race and gender and the sociocultural knowledge that comes with computer programming, but also job opportunities according to geographical location. All of these become sources of judgement for Black adult learners in this chapter and shape their attempts to obtain social mobility, just not according to the purposes of a computer code bootcamp like Clearwater Academy.

The uneven benefits from coding literacy would not be a shock for literacy scholars. For over two decades, New Literacies Studies has found that literacy is a weapon for structuring stratification among marginalized and dominant social identities (Street, 1984). Even as a material object that moves across borders, literacy participates in regulating how people migrate or build literacy practices among families (Leonard, 2017; Vieira, 2019). These findings hamper the (digital) literacy myth, or the persistent belief that literacy acquisition is necessary for economic development, democratic practice, upward social mobility, and other social markers of advancing toward progress and potential (Graff, 1979; Shapiro, 2015). As participants shared in Chapter 3, changes in policy and practice challenge institutional racism and produce opportunities, not the acquisition of coding literacy. What's left between these

two commonplaces? What ignites, animates, and excites literacy scholarship are the nuances of literacy practices, and not just the patterns that occur but it's vast differences, complicating the narratives that persist and simplify how we engage with literacy and how literacy engages with us.

Previous research on a more realistic hope of literacy in Black communities guides this chapter analysis. While literacy scholars have called for investigating how the literacy myth influences policy, pedagogy, and curricular design to reach overblown goals, that does not preclude that we should lose all hope in literacy. The Black version of the literacy myth paints a subtle difference. A historical study on Black literacy learning in the 19th century and early 20th century demonstrates that profound belief in literacy's power to liberate drove Black people to learn how to read and write (Bibbs, 2011). Those desires carry forward into coding literacy; as Black people interact with the social and material resources of computer programming, they demonstrate a "rhetorical practice of finding comfort in and celebration of" strategies for approaching technical and professional communication. These strategies point to different affective principles that guide Black adults' decisions to engage with those resources (Byrd, 2022, p. 299). The outcomes for Black adult learners in software development are indeed alarming, but they don't cover the entire story of the pivots they do make in life. Often "success," in the context of computer code bootcamps, means a person has become financially whole. On this note, I'm reminded of Toni Morrison's wisdom. In 2001 she joined The Connecticut Forum to discuss literature and race. Moderator Juan Williams asked one audience member's question: "How do you survive whole in a world where we're all victims of something?" Morrison replied, "Sometimes you don't survive whole. You just survive in parts. But the grandeur of life is that attempt. It's not about that solution" (CTFORUM, 2020). This final chapter makes that observation concrete: to document Black adult learners' survival in parts after graduating from Clearwater Academy, to focus on the attempt, and not any binary measure of success and failure.

The chapter charts paths of collective literacy development over Black adult learners' professional lives after Clearwater Academy. Although not using the theoretical tools or methods of lifespan writing from the outset, my attention to post-graduate life does echo with the mission of this writing research. Lifespan writing research conducts longitudinal studies on how writing develops using the lens of human development scholarship. Scholars study how individuals use "unique prior experiences and resources to identify, understand, and act in each new event, thereby further developing through the solving of new writing problems." Collecting "adequate situational data, we can see writing growth taking place as a response to social situations and demands, and formative of social relations and identities, which in turn provide

further opportunities for challenge and development" (Bazerman, 2018, p. 328). I owe my analysis to Deborah Brandt's insight on how roles in the workplace shape writing development for adults. Her re-analysis of interview data for a separate book—*The Rise of Writing*—shows how "Roles in some sense are opportunity structures for development and, as such, are one of the biggest sources of developmental variation and stratification." We cannot treat professional positions as neutral; rather a role's relationship with power in the hierarchy of work gets wrapped up in stereotypes of social identity and ultimately determine the duties organizations assign workers. "For writers from groups that are often negatively stereotyped or stigmatized," writes Brandt, "these expectations can register as differential treatment leading to a sense of heightened pressure" (Brandt, 2018, p. 254).

The chapter will highlight coding literacy development in the context of economic opportunity, especially regarding the roles Black adult learners vie for. A complex list of pivots occurs across these lives: fail to acquire, obtain but then consider quitting, and actually quit from the roles they had desired. It is the combination of locations, events, materials, racial identity, and communal and familial contexts that make coding literacy echo in other areas of their lives after brushing up against software development, however brief or, in many cases, never at all. What's revealed, again, are the survivals completed in part, and it is in this space we find a more dynamic practice for coding literacy than imagined from computer code bootcamps and software development in general.

Coding Literacy Decay

After graduation a journey begins, a journey to leverage what you learned from a computer code bootcamp into a job with the elites, the ones with hands on the levers of our digital ecosystem: software development. However, after graduating, the structures that support Black adult learners shift as well; all the time spent in computer code bootcamps, and the resources and people they had to keep them going through the bootcamp, end. They must return to inequality full time. But this time they take with them new hope that coding literacy will come just as quickly as the training to achieve a variety of desires. Rosie had taken medical retirement after her lupus diagnosis, so she had no intention of applying for an internship; she had wanted to use coding literacy to "work on my own. Now to have the tools that I needed to start up my own business, you know." However, in the transition over the summer of 2017 and the winter of 2018, Black adult learners found little immediate relief from the rush of poverty. You'll read more about adults like Rosie in this section where perception and expectation of coding literacy run into the realities of leveraging new human capital into work.

To summarize, leveraging coding literacy comes too slow and without much guarantee that it will fit into their lives. Participants deny or put away coding literacy just as, or because of, different tech companies or companies with tech departments denies them. I confirm what literacy studies knows about the limits of coding literacy and its institutions with more nuanced details and Black perspectives on the process. In other words, while Black adult learners do keep themselves humble and go after the lowliest of positions in software development (or do not seek them at all), they still do not appear to be valued and they, in turn, flounder to find value in coding. The multiple ways Black adult learners let coding literacy die out or try to keep alive outside the job market I call *coding literacy decay*. It represents the majority of ways coding literacy did not pull through on its promises. But what's valuable is what happens *after* the decay has happened.

Black Computer Code Bootcamp Graduates on the Job Market

I first take the job market as an important location filled with moments of judgement on Black adult learners' coding literacies. Elevator pitches and mock job interviews were hallmarks of Clearwater Academy's curriculum, where adult learners prepared and then received feedback for their performance from mentors working in the industry. Along with résumé and cover letter writing (also receiving feedback from industry tutors) Black adult learners would be well-poised to perform, to use the limited space they had to, in the words of Richard described at the start of this chapter, communicate they can takeaway someone's pain with computer programming. By the start of my study, Clearwater Academy left adult learners to find and apply for jobs, with the bootcamps offering letters of recommendation for their best adult learners. This already created different levels of stratification among graduates. Of course, the approach also affirmed that Black adult learners had agency over the kind of professional they wanted to be. For some participants, they found the job market undesirable because it proved how little coding literacy could respond to their economic and familial needs. Their perspective and experience on the job hunt left them re-thinking the definition and boundaries of their newfound skills and competencies. These experiences demonstrate that coding literacy, unlike the range of other writing opportunities available to Black people, loses status without a favorable job market. It's too tied to careers, too specialized for labor. The structures that reward coding literacy, supposedly on merit alone, fail these Black adult learners.

DeAndre graduated from the spring 2017 class. While he didn't regret spending three and a half months learning coding, DeAndre found his personal life depleted. He explained his post-graduation experiences to me via

video on Google Meet. On my screen, DeAndre leaned back on his couch and smoked a blunt, which, he said, was the only way to get through the interview with me. "It was a very stressful thing. Like that was stressful, I ain't going to lie to you," he explained. "Me doing that, ain't going to work, that was stressful. So stressful." He worked as a dishwater for a local pizza restaurant throughout the semester, but a month before graduation, DeAndre lost that job. He was physically tired from working nights and mentally tired in the day from coding. Something had to give in that last stretch. Free from the constraints of Clearwater Academy, DeAndre needed a job. He had fallen behind on rent and other bills, so the promise of a paid internship—which usually came soon after graduation—was the solution. The instructors connected DeAndre with one internship; he interviewed for the position but never got called back, never got hired. Turns out, DeAndre had not put all his hope into one position. While interviewing for the internship he used his background in food service to get a job with a Southern Country-themed restaurant as a cook. He would have preferred not to work there because the patrons were "some strange motherfuckers." He continued, "Don't get me started on what they order. Just no. No. Um, no. I ain't going to lie to you—there's some strange stuff that happened one day. The man said he want a double bacon cheeseburger. No bun." In other words, I learned, the customer wanted all the ingredients for the cheeseburger outside of the bun, defeating the purpose of even calling the food a burger. Money was in tech, DeAndre reasoned, that's where he wanted to be, yet that money wasn't going to come any time soon. Money from strange customers with strange food orders was his only immediate option for relief.

When I finally caught up with Kevin in May 2018, I learned he had more luck on the job market than DeAndre in his home state of Arizona. He had not intended to apply for jobs in software development; initially, Kevin registered for another computer code bootcamp to learn Swift, Apple's computer programming language for all of its iOS products. He had settled into Sakowin after just three months of living there and was ready to dive deeper into coding literacy. But Kevin felt a nagging bit of homesickness. His mother kept telling him all was well, but that did not alleviate his uneasy feelings. Kevin trusted his gut: he withdrew from the upskilling opportunity, packed up his stuff, and returned home to where he found his house and family—his mom, adult cousins, aunt, and uncle, and two dogs—in chaos. Looking at Kevin on web cam with his afro sticking up in the air like Fredrick Douglass in his old self-portraits, I would not have thought the return home was so fraught with emotional and mental pain. "Like that's what I came home to," Kevin said. "My car's messed up. Yeah, my house is a mess. My mom is sicker than ever. My house is dirty. I mean, I was just like, 'What is this?'" Kevin spent several

weeks getting his house back in order, including trying to scrape together money to get his truck fixed.

Like DeAndre, Kevin returned to the old job he had since before moving to attend Clearwater Academy: cooking for generous and kind managers, ones who had supported him to leave and come back should things not pan out. He was cashing in on their kindness. But cooking wasn't where he wanted to be. Kevin had worked hard learning coding literacy, and to just go back to working as a cook, left him depressed. Kevin did get some encouragement when he applied for entry-level software developer jobs. He got several interviews, and he put his personality and employability skills to use: Kevin, already gregarious from what I could tell, talked to one hiring manager for thirty minutes, no questions related to the job, just conversation. But when employers moved from being a colleague to being a good coder, Kevin hit walls every time. He recalled a consistent theme for why every company didn't want to hire him. "I didn't have any backend, you know, any real projects under my belt," he recalled. "'They were just kinda like, 'You know, you seem like a great person, but we need you to have more experience.' That's literally what every coding job told me. 'We need you to have a little more experience; get some more projects under your belt and then come back in 4 months when you can re-apply.'"

Kevin was getting a preview into what was at stake for many companies; software is partly about transforming business solutions into code, and code can stretch around the world many times to ease people's pain. Hence, one employer told Kevin that he would have only proven himself to be a potential liability if they let him touch code that was the bedrock of their clients' lives. Projects in themselves would not provide access to a prestigious job as a junior software developer, however, but rather the ability to understand a problem from multiple angles and then program the solution from beginning to end. What Kevin would have to show is not a polished web app or ecommerce website, but rather the potential to solve a problem using computer programming. He had not developed enough projects to demonstrate how his literacy practices fit with these standards. Here's a pattern of needing to learn more beyond the computer code bootcamp. More experience, more knowledge to possess these jobs "made for white people." Kevin felt defeated, saying to me, "And it kind of like—well, with everything going on with my mom, with everything going on in the house, me going right back to the cooking job, it made me depressed. I was just like, 'Dude. I'm just stuck in a rut because I'm right back where I started.' Like literally [Clearwater] was for nothing at that point."

Back in Sakowin, Isaiah had achieved the dream after he graduated in spring 2017: he interviewed for and then accepted a paid three-month internship with a company that provided support and services for credit unions. He

would work in backend programming – the coding that manages stuff users don't see, like data and security. However, at the end of his internship, Isaiah had learned very little backend programming. The reasons I will explore later in this chapter, but I bring his experiences on the software development job market at this point in the chapter because he, like DeAndre and Kevin, desired full-time work in the industry. Even with more experience on the job than his classmates, Isaiah had trouble leveraging coding literacy into social mobility. Isaiah had a baby boy coming in September 2017, and at the behest of human resources he was beginning to leave the internship two weeks before it officially ended. He wasted no time trying to get back into computer programming. Isaiah made significant progress when an insurance company interviewed him for a developer position. Before the interview he studied coding for the technical portion. "But even I went out of my way to read some of their goals and to say why I wanted to be a part of that, "Isaiah explained over the video call. "I did everything that I should've for the interview." Isaiah thought the interview went well; he followed up on next steps with the hiring manager over several emails over several days. "Till one day she hit me up," explained Isaiah, "and said, 'Hey, we're going through a hiring freeze. All of the managers right now are going through doing a lot of financial planning for this year. And really budgeting this year. If anything changes within the next six months, we'll let you know.' And so that kinda sucked. I was really looking forward to that." Isaiah applied to a digital asset management company who later called him at work and asked for an interview the following day. He had little time to prepare, so again Isaiah used his lunch hour to study coding for the technical portion, but he was sure he didn't study hard enough. They did not call him back for another interview.

These three Black men accepted the promise of coding literacy—that the training they had could be directly applied to a labor market hungry for coder, *Black* coders. They had no aspiration for Google or Facebook—they applied for entry level work, and even that judged their worth as workers as less than qualified compared to others. One reason for this may be the lack of projects that represent conceptualizing business solutions for business ideas. I mention this above with Kevin, but Isaiah extends it further when he reflected on his training in Clearwater Academy and in the internship. "I feel like it would've been more beneficial to have strictly three months, was like, working on just coding. So HTML, CSS, Bootstrap, JavaScript. Those four things, and just relatively work on projects from scratch in there more so than Wordpress," he concluded. "Because like if I were to have my projects … The way I look at it, towards the end I kinda of looking at, if I had that programming experience making my own portfolio from scratch, I probably been able to show businesses I'm applying for I got stuff really ready." The learning to learn

model that Clearwater Academy adopted from Google helped Black adult learners develop problem-solving processes and some critical knowledge of the limits of computation, yet, from Isaiah and Kevin experience on the job market, each felt they had not spent enough time translating those competencies into coding projects. Portfolios of their best work, even if rudimentary in execution, would have proved their ability to take a problem and develop a solution from beginning to end. This knowledge, they suggest, was the missing link for fulfilling their job prospects.

Struggling to Find a Place in the Labor Market with Coding

Without much direction on where else coding could go, Black adult learners re-defined coding literacy's status and gave more value to other literacies and educational institutions. This section shows how coding literacy detached from the job market flounders in the literacy ecology of Black adult learners. DeAndre and Isaiah attempted to keep computer programming alive in their repertoire and looked forward to possibilities of finding other ways of getting into tech later. However, they had a shaky relationship with the nature of coding itself that suggested some hinderance to their fully leveraging its power.

DeAndre took a break from coding after he graduated from Clearwater Academy, because he had needed time to work and get caught up on bills. But, frankly, he welcomed the break from coding. Learning languages in Clearwater Academy was a mental challenge, because DeAndre had to learn multiple languages all at once. "I was … I ain't going to lie to you, when I was doing that, it was hectic. My head hurt. I didn't have enough weed, either," DeAndre explained. I asked him, and other participants, what coding was like to him. His response was influenced by his experiences learning coding in Clearwater Academy as well as his trying to learn post-graduation, but they nevertheless clarified what learning coding looked like in his mind: whenever DeAndre saw code, he would think, "Ah shit," which means "I'm finna prepare for a brain cramp. … Just think about this though: Building a whole website, you use at least 5 to 6 languages. Don't your brain hurt just thinking about that?… Because, remember, you have to put everything in the right spot. Everything has to be in the exact ordering with the exact lettering or it's not going to work. It's a lot of work. It has to be perfect."

Even though coding was a mental challenge, DeAndre still enjoyed it, so much so that he flipped Clearwater Academy's philosophy for learning to code, from practicing all at once to practicing with one computer programming language at a time. He stayed connected with coding via websites like FreeCodeCamp, which offered hundreds of exercises in his list of languages: PHP, SQL, Python, and Swift. He always kept the tab to FreeCodeCamp open on his

browser, so when he opened his laptop, it was the first thing he saw. This tactic was to keep him motivated to learn by seeing he was in the middle of an exercise that needed to be finished, like how some writers leave sentences unfinished to keep them writing and from falling into writer's block. At the time of our interview, DeAndre was about to embark on a new approach to staying connected to coding: just make projects and break things. "I'm finna just start doing a whole bunch of shit—just making random websites that way I can just like learn it more," he said. "Fuck up, fix it, and then find some new shit … You never know."

DeAndre had attempted to turn his devotions to coding literacy into real world opportunities. With the labor market in software development cut off from him, at least for the time being, DeAndre followed Richard's advice to offer freelance work to friends and family. A cousin in Chicago asked DeAndre to develop a custom website to host porn. Months ago, DeAndre learned that freelance was partly about knowing your values and who your work should align with; DeAndre needed practice, so creating an adult website was no problem. However, two months into the project, nothing happened. DeAndre said he just made the shell of the website. "I never put anything up there. … It's just right there. Just sitting there. I just made a basic shell; I haven't put any code into it," DeAndre said. He was waiting for his cousin to send the videos but "they're lazy and they never want to do anything." DeAndre didn't find much value in charging money for a shell of a website; that was easy labor. When he got the videos and started coding the design around them and creating their functionality, then DeAndre would start charging money. Until then, the project sat on his computer unused. DeAndre needed coding literacy to go somewhere tangible, to leap off the training playground and into real projects. But work was scarce in the space outside of the labor market; he didn't seem to leverage what resources he had into opportunities. Yet he still believed in the power of coding. Once he got going with freelance, he would turn that experience into working for a company fulltime. DeAndre said he needed to do research on jobs and what company would be the best fit for him. But that would be a year-long endeavor. His plan was to take the experience from working on the job back into freelance work. Be his own independent worker, making his own money for his family.

I interviewed Isaiah the most consistently every couple of months from early summer to early Fall of 2017, so from interview-to-interview Isaiah's narrative demonstrated the slow digression of his confidence in coding literacy for social mobility. For example, in the second interview, Isaiah had kept his options open to both a tech job and a non-tech job. "Well right now my predicament is, at the end of the day, I would like to be in IT, but if the job I get is not IT, that's what I got to do. Because I got to provide," Isaiah reasoned. "At the end of the day, I just need to get a job. If that job is IT, that's great. If not,

well, oh well. I need a job." A mentor in user experience design at the internship connected him to one job at the local university, and Isaiah had heard of a family member getting a different job at the university that paid seventeen dollars an hour. However, by our third and final interview, failure on the job market and awful experiences at his internship left Isaiah unmotivated and disconnected with the tech industry. Between our second and third interview, he had done a lot of thinking and declared tech wasn't what he wanted to do with his life. "I was rattled because a lot of places didn't want me because I didn't know certain things that I was supposed to gain from [the company] and I didn't," he explained to me. "So I kinda stopped coding for a while. I still know how to code; I still mess with it sometimes but as far as like do I still code? No." He did not lose faith in what coding literacy may do for other Black people: grant them power and a seat at the table of shaping the digital landscape. It was just that he no longer saw himself at that table.

Isaiah started to play to his strengths as a graphic designer and artist, the literacies he started out with before attending Clearwater Academy. He practiced graphic design in high school and later completed one academic year of general education courses at a local community college. His advisors told him he wouldn't start the graphic arts and design program for two or three years; staying in the program for that long didn't make sense to him at first. Isaiah switched from college to the short-term training at Clearwater Academy. Having bathed in the fires of software development, Isaiah wanted to take up college again, but for information technology systems, and then combine coding literacy with graphic design. This was a better, clearer pathway for him. Unlike DeAndre, however, Isaiah had little faith in freelance as a starting option; his experience as an artist informed his perspective on the drawback of independent work, explaining, that "a lot of people that I know aren't willing to pay the amount that really is for a website." It's hard to find anyone willing to pay more than a hundred dollars. "Let's be real," Isaiah continued. "Whatcha going to do? Like I spent this whole time doing this website for me from start to scratch, I mean from start to finish, and all you willing to pay is a hundred dollars? I don't want to be in freelance."

Instead, he would make computer programming work for himself, as an independent small business owner taking care of his newborn son. Isaiah would draw on his passion for fashion and art and then develop a website that would sell his work and showcase designs for a clothing line. While freelance was at the whim of temperamental clients, coding his passion into an ecommerce website and portfolio would allow him to "grow my wealth" and "be able to get a house one day. I want to be able to drive a nice car that doesn't freakin' breakdown. … I mean now that I have a son and I definitely know as this point and time if I'm not seeing results, yeah, I can work on getting to

where I want to be…." I note the difference in tone from discussing the job market as a coder with disdain and disappointment with more hope and determination when Isaiah discussed how he could redirect coding toward his passion. Isaiah's worth may have been denied by powerful tech companies, but he found a clarification on where he could place himself in the economy and, more important, the value he could bring to his family, especially his son. Isaiah imagined himself visiting Japan and owning a car that doesn't breakdown, but his true standard of success was creating a better life for his son through graphic design, art, and computer programming.

Isaiah, Kevin, and DeAndre's experiences on the job market demonstrates the precarious work one must do to "get in." They're narratives about failing to get in or, in the case of Isaiah, staying in software development recalls Clearwater Academy's curriculum. Isaiah, for example, had the hindsight of his internship to realize that perhaps project-based work for three-months would make him more valuable as a Black coder competing for jobs made for white men. At the same time, however, it suggests the standards, by which judgements are made in the job interview; the other documents that described their qualifications did enough to get them an interview, but the coding literacy at a computer code bootcamp had not met their standards. Standards for Black coders from a computer code bootcamp would need to shift. What other assets are at play here that's not clear in the coding literacies they do have? Could separate questions that probe for problem-solving skills, and not just looking at the project itself, become a new direction for assessing their worth? I come back to this question about standards and job candidates' coding literacy assessment in the context of on-the-job experiences from Isaiah, Zelda, and Gerrard. For now, I want to underscore that once coding literacy gets disassociated from the job market, these participants resubscribe computer programming to something for themselves, as independent entrepreneurs, for example. So that's what happens to coding literacy on the job market and what it does for some Black adult learners who attempt to go into that pipeline. However, another place coding literacy goes once disconnected from the job market is a metaphorical closet or safe. Or it's thrown out from a Black adult's life completely.

"I Just Haven't Found a Home for That Skill"

In the section above, I noted that financial and social circumstances shaped Black adult learners' experiences on the job market. For example, Isaiah's expecting his newborn son in September accelerated his effort to get a job after the internship; knowing he had a family to take care of led him to accept that even a non-tech job that pays well would be fine. Those familial and financial

needs shaped how other Black adult learners came to value coding literacy or transform its value in a way that was more realistic and met their needs. In this section, I describe what else happens when coding literacy becomes unmoored from the economy; what readers will discover here is that a prestigious literacy like computer programming that fails to move fast enough and produce results that change the consequences of racism for these Black adult learners become a new resource in their lives as workers and community members. I highlight what happens outside the job market effort, adding another layer to why and, later, how Black adult learners develop in Black tech ecosystems: in these new environments for coding, they assert agency over their literacy development and drive toward other avenues that don't directly benefit software development and its deep ties to our digital ecosystems.

Alice judged that the structures of assessing and welcoming her human capital weren't ideal. While she could probably get a paid three to six-month internship immediately, companies made no promises that they would promote her to full-time permanent employment. She could have applied for a different job, leverage the internship into full time work elsewhere, but Alice had no interest in staying on the job market path for nearly a year. She had two daughters and a husband to financially support; Alice needed stable, long-term work that offered no surprises but only consistency. After graduating from Clearwater Academy, she did two things: first, a nice vacation with her family; second, a return to banking where Alice had spent much of her professional career. Clearwater Academy's contribution was getting Alice access to touring the same company Isaiah would later intern for. After graduation, Alice connected with a high school friend who happened to work as a hiring manager for the financial company, and she helped Alice get an interview via referral. Although Alice did not go into software development, she did discover that knowing some computer programming gave her an advantage in the interview process. Alice mentioned that she knew JavaScript, and that put her ahead of every other applicant. "And she [the hiring manager] was like, 'Wait what did you just say? JavaScript? You know JavaScript?'" Alice explained. "And I was like, 'Yeah.' And she was like, 'Oh well some of our forms, we have to use JavaScript and someone who knows that would be helpful.' And I was like, 'Well, yeah'—You know, I keep up with—I do Code Academy every now and then. Keep up. But yeah, I was like, 'I know that.' And so she was like, 'Oh.' So that was like a turning point in the interview. Like it was going well already. I could tell but then it was kinda like, you know, she stopped. 'Wait.'" That was the most Clearwater Academy and coding literacy added to Alice's worth.

Recall that Kevin had tried and failed to make any moves on software development and ended up returning to cooking. His managers could tell how

unhappy Kevin was, so they suggested he quit and figure out what pathways would fulfill him. After Kevin found a replacement cook, he hung out with friends, took time to reflect, and slowly felt reconnected with himself. Kevin ended up landing a job repairing hardware for a major computing retail store. Like Alice, knowing computer programming launched him ahead of other job candidates. During the interview he brought up Clearwater Academy. "I was telling them about the coding, and they were like, 'Cool.'"

Both Kevin and Alice admitted that although computer programming helped them get hired in jobs adjacent to coding, they themselves found little reason to use their knowledge. Kevin said that "maybe in a month I write, like, twenty lines of code. I'm like, '"Maybe this can fix it."' 'Oh, yeah, cool!'" Otherwise, he spent his working hours fixing hardware and software, never digging into the ghost beneath the machine. When I returned for another interview with Alice three months later, the story was still the same. Although management told her that the company used JavaScript to design documents, Alice herself never had to code. Instead, her main responsibility was taking requests for custom financial documents from U.S. credit unions, such as credit card applications, mortgage loan applications, membership registration, opening a checking and savings account, and credit card applications. Her company designed these documents and maintained their compliance. Alice was one of the first steps in the process of creating these financial documents: She took requests from clients via phone or email. That involved identifying the type of document they wanted and how it should be customized to their company brand and policy. "And then you send it over to compliance analysts and consultants who like review the language, the department that has to be in there to be complaint, yadda, yadda," Alice explained to me, "and then they would send it to a document specialist who would actually create the document, add the logo, send it out via paper or electronic format." Alice and Kevin then found themselves adjacent to coding, sitting beside coding, but never in a position to shape digital technologies, or the design of documents, in the ways others do. This meant they were removed from the more prestigious areas of the knowledge economy. Better positioned socially but not at the table of technology design. Still coding literacy echoes or reverberates to help them obtain some level of social mobility.

Coding literacy sometimes went unused or was locked up. This isn't a metaphor. Quite literally the resources Clearwater Academy provided sat on a computer unused. Rosie found Clearwater Academy's resources for entrepreneurship the most valuable for her life. However, after she graduated, Rosie returned to the family duties she had been doing before attending the computer code bootcamp: taking care of her mom and her brother who had stage four cancer. Each day or week could look different. "So I might need to call to

make appointments for my mom," Rosie explained. "Take her to appointments or to the store. Just caregiving in general. You just don't know what today is going to bring." She also had to make sure her schedule would be clear for her brother. For example, once he had scheduled a last-minute procedure and needed someone to drive him home. "It's just like, 'Okay, I didn't have anything specifically at that time, so I can do it.' But you just don't know what's gonna come at you in the course of a day," Rosie explained. Because Rosie spent most days caregiving for her family, she "haven't had the opportunity to utilize those skills yet." She said the same thing again nearly a year after starting the computer code bootcamp: "I haven't had the opportunity to open up that door yet." Rosie knew where to go in the event she did have an opportunity to open that door, explaining to me, "Well I do know where the resources are so in the future when I get here in my vault. So I do know the websites and I took a lot of notes that I keep hold of." Rosie used her experience to write content for a lupus support group's website, but coding sat in a vault waiting to be used.

Kevin loved his new job in IT, but not finding a whole lot of use for coding literacy, he also put computer programming away. The problem was that he couldn't imagine how else it could be used, something not really explored at Clearwater Academy, which was hyper-focused on coding literacy for work. He put the problem to me like this: "Yeah, I just haven't found a home for that skill. I would like to do something with it. I just haven't found a home for that skill yet." Kevin knew he could find jobs—he knew that experience well. He could investigate Meetups around town, too. So he understood coding could go into many places. But what made the most sense for him, where he was in life, was a puzzle to work on. "But the thing is," said Kevin, "when I think about coding it's just like, 'What do I want to do with this?' I just really haven't found a place for it. You know, like my other skills, I have a place for them. I'm like, 'Okay, I can do this. I can do this with this skill.'" Kevin shows that computer programming is one of many literacy practices he possesses; some literacies have clear locations for their uses, but computer programming still lingers in the air, or sits on his computer, waiting to be used. By the time I interviewed him, Kevin was getting back into coding, revisiting some of the lessons he had completed back at Clearwater Academy. Perhaps through practice he could discover where coding could go.

This section covered two narratives: what happens when coding literacy practices are fixed to the job market, as Clearwater Academy intended, and what happens when coding literacy breaks away from that purpose. A mixed story becomes clear on how structures for work still create barriers for participants, sometimes getting in the way of coding being a response to the consequences of racism. Still, coding literacy lingers like a specter, a resource to leverage into other kinds of work. In the final subsection on coding literacy

decay and the ways it happens, I turn attention to the joy of having not doing coding at all. On this note, Black adult learners pivot to other literacies and institutions for social mobility.

"Definitely Not What I Want to Do as My Career"

Although framed as necessary to the economy and for social mobility, arguments appealing to anyone who loves tech and needs money, Black adult learners had not bought into the idea after graduation. In some ways, Black adult learners were happy to *not* be coding at all, finding the practice more trouble than it was worth. Alice, for example, knew a few weeks into Clearwater Academy that she didn't like computer programming. I wanted to unpack further this fissure, how Alice had joy for learning tech but not at the level of coding. She explained that coding was like "building a Lego. Lots of tiny different pieces. You kinda trial and error to see if they're gonna fit. Um, you know, there's lots of different ways to do things. like you're building a pirate ship and it has ten thousand pieces and really hard instructions and they're not really good instructions, so it's kinda, you know, random little pictures and it doesn't have any words. …" Alice couldn't quite pinpoint why she didn't want to code. She could manage coding just fine, although it was still a struggle to do in Clearwater Academy, and Alice wasn't sure if struggling with the specifics of coding turned her away from software development as a career. What Alic did know for sure was that when she practiced coding it was "definitely not what I want to do as my career. That part of, you know, learning computers just didn't interest me as much."

She did find design appealing. In fact, Alice tried to transform the design skills she learned in Clearwater Academy into work; she found one job, but the pay was abysmal and the benefits subpar. Another adjacent area of interest to coding was IT; for that, she was already in the process of doing by attending a local community college and majoring in Information Systems. Navigating college while working full time with a family of four was not easy, as she constantly struggled for time to study. For example, she had to retake writing classes because they didn't transfer from her participation in Journey, the humanities course for returning adults offered through the local university. "I'm taking one this semester," explained Alice, "and it's a lot of work. I mean the readings usually take like three hours alone. It's just—I have to dedicate a lot of time, and I struggle with…Time is something I do not have a lot of." Her trying to find time to study was reminiscent of Clearwater Academy, where she had to find time to study after putting her children to bed. She visited libraries to complete work, and even briefly checked out books that were too expensive to buy and didn't pay much if she had tried to sell them back.

Nevertheless, the payoff was there: In fall 2017, she finished with a 4.0 GPA, and the hardware class was her favorite. Other courses offered clear career pathways in help desk and information systems. She attended a fair where instructors representing each area of the IT program set up tables. "And I actually talked to them to try to narrow down what exactly—and that's what I came up with. Because I was gonna do like network security, but they said that one was super specific," said Alice. One of the instructors said that network security was more challenging than IT help desk, "so I was like 'Eh I wanna stay broad because I can do a lot of things.'" Sitting down to code wasn't super interesting to her but in IT Alice could combine her love for people with technology. Computer programming can have impacts not immediately felt or seen but with IT she could connect with people directly and note the influence right then and there.

It's clear now that Alice had gone completely away from coding for career. Like leveraging coding to get a non-coding job in banking, Alice was actively ideating other ways she could launch herself into IT. She had eyed that department in her current job from touring the company in spring 2017 and ever since had been plotting how to spring out of her current position. In other words, during our interview Alice frequently imagined how she could use every ounce of her literacy in document requests as leverage into IT. From what little she knew of JavaScript to the CompTIA A+ certification to taking apart CPUs and error handling, she could learn while studying for her associates degree. Alice was prepared to make moves, and she wasn't going to hide that desire from management: "So the goal is, I am gonna talk to my manager eventually, but I wanna kind of talk to her about the degree path I'm taking, working, tuition reimbursement conversations. And then let her also kinda know that I would like to make a strategic transition to IT at some point. So that's kind of—this is kinda my foot in the door kind of position." She thought the combination of literacies—not coding literacy alone—produced value in the economy and would, she hoped, lead to not only social mobility, but the benefits that sometimes come with that upward move in class: flexible work from home so Alice could be more present for her two daughters.

Zelda graduated from the Fall 2017 class. A self-proclaimed nerd and mother of one daughter, Zelda was the only participant other than Isaiah to get an internship. For our first interview, we met in a diner downtown. Her daughter—about one and half years old—nursed a foam egg as we chatted over lunch. During the internship, Zelda was actively trying to learn new programming languages, something that was fun like Python, Ruby, or Swift. She found it boring, so the search for a programming language that was fun was her only way to stay in software development. Her status as a web developer intern even attracted software developer men on dating apps. Zelda possessed "geek capital,"

which drew in social resources that could help her get ahead in the industry (Twine, 2022). For example, an Indian coder reached out on one of the apps, and after getting to know Zelda, was thoroughly impressed that a Black woman was in a coding position. He offered himself as a resource for learning the computer programming language Ruby; Zelda hated him, but she found men like him useful for upskilling and networking. She used other men she met online to learn more about their jobs: "What do they do? 'Oh! I want to have that job! Tell me more.'" Zelda was less interested in dating and more interested in how those men could help her get ahead in software development. Even in the first interview, at the start of her internship, Zelda really found data analytics at her the internship more interesting. At the diner, Zelda tried to keep her daughter on track with eating her food. She said bluntly, "It's just kind of boring to me. I mean I can read it. I don't like coding; I don't like code. I don't want to be a coder, unless I can learn a language that I like." She loved doing data analytics for the company because she loved numbers. Working with numbers in an Excel sheet was "relaxing; it's calming. It's like smoking weed … Making sure all of them are equal; I say doing finance because I appreciate numbers that have meaning behind them. I mean in front of them. I mean, it depends on which currency." That was in spring 2018. Later in July 2018, I met Zelda again, this time in a park by a lake. Her daughter wanted to chase after ducks waddling near the shore; little cousins played on the jungle gym nearby. As we caught up about how her internship concluded, Zelda had not found a fun programming language worth learning, so she had traded it for something in data analytics and a new area of interest: design. "I like designing," Zelda told me. "Like that was the thing I fell in love with at my job. Like I guess I could give credit to. Is that I found my niche." Finding her niche: it's one of many experiences with coding literacy—its processes and social structures—that teach Black adult learners where they fit in the world, to clarify what niche or literacies give value to Black well-being and sense of self, eliding the white software world and its dire need for diverse perspectives on technology design.

What I've described in the first half of this chapter are the multiple ways coding literacy decays; that is, the relationship Black adult learners have with computer programming when on the job market has conflict and pressure. These locations of judgement extend or retract their literacy development. Through direct experience with coding, Black adult learners can rethink or revise their position in the economy and what they truly want. They deny coding literacy's power because coding literacy has less value than they had thought; they give power to other literacies, other institutions, many of them educational or legacies with proven results. A re-assessment of their ecology of literacies ensues and they find a way to push further, to upskill, to easily flow in the economy.

Agreeing to Participate in Disruptive Whiteness

I have traced the variety of contours coding literacy takes when disconnected from the job market. How does labor, race, and coding literacy converge to help one achieve the purpose of learning computer programming—an internship? In this section, I detail the internship experiences of three Black adult learners, the only three who got an internship in my study. One participant—Gerrard—had graduated two years before my study and was the only one who made a career of their training. He was a full stack developer working for an insurance company at the time of our interview. While some readers note the quantitative data of success (or failure) to get a job, I'm more focused on the narratives that detail the conditions of their work, what those narratives bring to light that can help me think about coding literacy in relationship to labor and race, and what that relationship may indicate about emerging technologies in the work lives of Black adult learners. Plus, coding adjacent jobs complicates what counts as a desired outcome from a computer code bootcamp's training. Whiteness can disrupt Black people's literacy practices in the workplace; while they focus on doing good work, their colleagues note their race and supposedly limited knowledge of coding due to their status as graduates from computer code bootcamps. I start with the rhetorical value of internships for Clearwater Academy and tech companies in general. With this context, I highlight incidents from the internship experiences of Isaiah, Zelda, and Gerrard.

Internships are the bread and butter of Clearwater Academy marketing. To be placed in an internship, adult learners must complete all coursework requirements over the fourteen weeks of intensive training in computer programming and employability skills for graduation, pass a drug test, and show they can behave professionally in timeliness, email etiquette, and business attire. Graduates must still interview for the position, and they only get one chance, as DeAndre in the previous section experienced; if they fail to impress, that's it. No other opportunities to do an internship. Clearwater Academy graduates could be placed among several companies in tech or in a software development department within a company. Internships through Clearwater Academy were often paid full time work for three to six months (sometimes companies would check in with interns at three months and decide to extend their time to a full six months). They did not have the benefits typically afforded to permanent employees, like health insurance or a 401(k). According to Clearwater Academy's policies, internships reinforce the skills learned in the bootcamp. But there's an economic goal: if Black adult learners do well enough, they'll get hired on permanently or the internship supervisor at the company may write a strong letter of recommendation for future job searches.

Internships may be mutually beneficial to the reputations of the computer code bootcamp and the tech company. The adult learner contributes to real-world work in the company, often at a moment in the development process where their mistakes won't hurt the product overall, but they also become a witness to the strength of their training in college and a witness to the company's great work culture. For Clearwater Academy, the success of a candidate strengthens ties between the bootcamp and potential tech sponsors and employers. At the time of my study, Clearwater Academy was just rebuilding its reputation in the community. The first few classes of graduates worked poorly on the job, and as a result, Clearwater lost multiple relationships with tech companies. Thanks to Richard and Jessica's curriculum, however, the bootcamp had solidified itself in the world of tech again. An internship means to benefit all parties. Internships in software development place Black people close to power and status; by showing their worth to tech companies, they can receive in return social connections and material resources necessary to obtain social mobility. This process – coding literacy to advance human capital, laboring in a company to gain resources and then a well-paying job – makes up the pipeline for diversifying the tech industry.

An internship may reinforce what's learned in the bootcamp but what exactly it reinforces remains uncertain. As Richard explained to the spring 2017 class, these are internships made for white men; the whiteness of tech hold the pipeline together and even flows through the pipe itself. Black interns must swim in that whiteness to be successful. Black adult learners must *perform* according to the expectations of the workplace. And they have been prepared to follow proper professional email etiquette, professional attire, and timeliness is to code the experience of work as a white middle-class experience. In other words, the tech company as sponsor of participants' coding literacy have little reason to interrogate their expectations and cultures more closely and are even less certain of how to adapt to Black adult learners with computer code bootcamp experience. I submit that coding literacy gets reinforced in the context of race and racism, and whiteness as a reinforced tool inside the internship disrupts the power of coding literacy to enact real change for Black lives. I document race's complexity in Isaiah, Zelda, and Gerrard's internship experiences and show how the overemphasis on whiteness as a marker of success hides the assets they bring to technology design. Their tech companies had several missed opportunities to learn from these Black adult learners. In the grander scheme of this chapter's argument, the three participants' stories demonstrate how they are squeezed between the consequences of racism in the world and the consequences of racism within the companies meant to help them. This section underscores how coding literacy works at best as echoes in Black lives, but not the music they dance to.

Taken together, the three participants with internships cover a variety of contexts, from small startups to nationally known financial companies. Gerrard graduated in 2015 and completed an internship with a startup company that created custom software for other businesses; his primary focus was backend programming. Isaiah graduated in spring 2017 and did a 6-month internship with the same company Alice would later work for. Like Gerrard, Isaiah worked with a team of backend programmers. Finally, Zelda graduated in fall 2017 and was hired on to complete a 6-month internship with a mid-sized company that developed software platforms for clinical researchers. Although varying in size and service, each team of developers and managers participants worked for brought whiteness to their relationship and vice versa. I document how whiteness disrupts the coding literacy practices Black adult learners attempted to learn.

Gerrard

Isaiah, Zelda, and Gerrard worked in majority white tech companies, as expected given the city they lived in and how racism pushed resources for computer programming to white men. However, they weren't prepared for the immense pressure of being a Black coder in these workplaces. That feeling disrupted coding practices left and right for participants. Gerrard was already a full stack developer when we spoke at a coffee shop downtown. The journey to that valuable and prestigious status was not easy when he first started out as an intern. In Clearwater Academy, Gerrard enjoyed pair programming alongside a diverse group of adult learners. He recalled that those early days of the computer code bootcamp "felt a little more, I guess, like, I didn't feel like I was out of my bubble, so to speak. … But Clearwater—it's just great because so many folks of color just trying to learn how to code. Helping each other out. I think it was a great atmosphere. A great learning atmosphere." He had no trouble asking for help from peers or his instructors; he had no trouble learning coding literacy.

But the mental weight of working in an all-white internship left him bereft and closed off from his co-workers. Stereotype threat settled into his mind immediately during the first three months of the internship. Coined by psychologists Claude Steele and Joshua Aronson, Black people, and other identity groups, experience stereotype threat when they fear "confirming, as a self-characteristic, a negative stereotype about one's group" (Steele & Aronson, 1995, p. 797). Steele and Aronson's research focused on schooling and tests, but here was stereotype threat in the meritocracy of coding. The mental distress can arise in multiple ways, but for Gerrard, possessing "solo status" in the workplace shaped how he would interact with others (Sekaquaptewa

& Thompson, 2003, p. 73). When he worked at the small startup, Gerrard felt his coding had to be perfect, or he would just attract his white co-workers' suspicions. "Because of the whole stigma of being Black," Gerrard explained to me, "and you had to work two times better, you know what I mean? And that's not always true. Not everyone's great. I was trying to fit that narrative, you know. I was trying to fit those shoes." Asking questions, Gerrard thought, confirmed the negative stereotype that Black people are lazy and aren't smart was in asking questions. While asking questions helped him succeed in Clearwater, questions in a real-world job only revealed how much he didn't know and how much of a mistake the white tech world made allowing a Black man to walk their lands. Having solo status only heightened that negative stereotype; he represented Black people, and he needed to show they were good coders. Gerrard thought too much about race and racism, and thus felt more outside of himself, like "a piece of me was out there in the ether. Gone. Trying to get it back. Try to reel it back in." Self-doubt as a Black coder followed him for months.

To be fair, Gerrard's co-workers and managers did nothing to activate this negative stereotype; they merely existed doing their day-to-day work. Gerrard explained they could tell he behaved as if he was "tiptoeing around [them] every day. I didn't want to come off like [that] because in my head I'm always thinking about stereotypes and things like that. And I don't want to come off as this Black dude." These white workers had no idea how to mitigate the problem of whiteness and stereotype threat; their only solution, it seemed, was leaving Gerrard to figure out how to work through these negative feelings. His coding was on par; in fact, the internship sent him to another computer code bootcamp so he could learn how algorithms supported backend programming. Gerrard had to convince himself to stop caring what others thought about his being a Black coder: Gerrard redirected his energy into achieving his personal goals, to say to himself that racism was, and here I cite the words of Toni Morrison, a disease white people had to figure out.

A long road trip to a conference in Texas was also helpful: it was just Gerrard and his team in a car, so he had no choice but to get to know them. But the scenario confirms existing research on how to counteract stereotype threat: persisting to participate in a group to eventually be seen (Cohen & Steele, 2002), which Gerrard's co-workers honestly wanted. Gerrard had many wins after his three-month ordeal with stereotype threat. For example, the startup later hired Gerrard full-time at the end of his six-month internship. He said he found himself doing more design work for the company during "Christmas time." In 2017, *Star Wars: The Force Awakens* was in theatres. Taking inspiration from Chewbacca, Gerrard redesigned the company mascot with

a Wookie hat and bandolier. The managers loved his idea and printed the custom design on their holiday cards, which attracted the attention of conference organizers down in Florida. They tapped Gerrard to design a one-page conference program. That job attracted another conference; this one was local and focused on inviting software developers who were experts in a specific computer programming language. Gerrard designed and coded their website with another Clearwater Academy alumni. Gerrard had made such inroads with his work that he even did a keynote at that same conference. He would later depart from the small startup to work frontend design for a major insurance company; that workplace had an extensive plan for inclusive practices, had hired diverse coders and other staff, and even had a record of hiring some Clearwater Academy graduates. Transformation within the company even before Gerrard arrived on the scene made that work environment more comfortable and responsive to his race and expertise in computer programming.

Gerrard's experience with stereotype threat made him think about the mental health of other Black coders. While he had a terrible time breaking through the mental and emotional strain of whiteness in the small startup, he admitted that Black coders have worse experience in California. I joked that since he was a full stack developer, Gerrard could try for Silicon Valley. He worried, however, that Black people were leaving tech fast and programs like Clearwater Academy helped plug in leaks along the tech pipeline. While Gerrard appreciated any diversity and inclusion effort from tech companies, he mused, "if you aren't really investing in mental health of these Black and Brown developers, you really have a problem." Gerrard recalled the suicide of thirty-one-year-old Joseph Thomas, a software developer who had worked in Uber's notoriously known workplace culture. During his five-month employment for the ride share company, Thomas worked long hours and slowly his personality began to change. Working under the intense strain of Uber, Thomas later died by suicide. The family sued Uber for contributing to Thomas' death, but the company claimed he had not worked for them long enough to claim benefits for psychiatric injuries under California law (Morse, 2017). "He was just extremely pressured," Gerrard said as he recounted the story. "He felt marginalized. And just … I can only imagine. If he was feeling what I was feeling it must've been that times 1000. And he had two sons, a wife, all that. And I don't want that to happen." So crucial was mental health to the productivity of Black coders, Gerrard suggested that the company and Black coders invest in therapy to make it through. Although the suggestion made sense, I could not help but think that perhaps therapy, even if as an add-on to health benefits from a tech company, was more of a Band-Aid than a treatment for whiteness. I'm not necessarily making a critique of Gerrard's suggestions but rather nodding toward Sara Ahmed's (2012) observation about diversity and

inclusion practices in universities: administration invests in DEI not to address whiteness but so that no one *finds* whiteness, so that it is no longer visible. Providing therapy puts the onus of addressing whiteness in tech on Black coders. It's an additional burden on marginalized coders, occupying their cognitive capacity to think about themselves and coding literacy at the same time. Whiteness stays intact.

Gerrard focused on the realities of working in white tech companies. Although therapy would be a worthy investment for Black coders and their employers, he knew from his early struggles in the internship that he needed to carve out a space for himself and other Black people. "I was like, 'Man, when you're older you have to create your own stuff,'" Gerrard recalled during our interview together. While he enjoyed his status as a full stack developer, he knew real control over tech was in venture capital. He would leverage his position and financial status to fund other Black-owned tech companies, to invest in properties that those Black techies could use for themselves, separate from the expectations of whiteness. Living the dream DeAndre imagined for himself, Gerrard hoped he would not work for someone else in five years: "Right now I'm just learning the tricks of the trade. Understanding the business and just teaching myself as I go along. And I want to bring other folks of color in as well." Just as he learned about the toil of whiteness on Black coders' health through his and others' experiences, Gerrard noticed how Silicon Valley was also a detriment to the local housing market; he again dreamed of an alternative that moved away from the Silicon Valley-model of business toward the growing tech market in Africa. That's where the safe space for Black coders was. In the future, Gerrard wanted to visit to learn where could contribute and invest. Ultimately, Gerrard argued, tech should be used to better communities: "You can't make an app for everything," he reasoned. "You can't. You can't. Invest in community. Invest in the materials. Work with the environment instead of against it."

Despite the very rough start to his internship, Gerrard broke through whiteness and broke into the tech industry. Navigating the whiteness of tech put a significant toll on his coding literacy practice and even put at risk his value as a literate worker. And yet looking within himself and his goals and caring less and less about the needs of whiteness, helped Gerrard re-discover his worth to himself and to his employers. But his coding literacy experience and the business structures working in and around it clarified his next ambitious goal: to take tech culture's emphasis on independence and self-sufficiency and turn it into a vision for Black coding literacy practices; those practices include gathering the expertise of Black people into one space, investing Black dollars into Black talent and tech, and create a Black tech ecosystem that promoted affordable housing and environmental justice.

Isaiah

Isaiah learned he would have a baby in February 2017, just a few days before he started attending Clearwater Academy. His successful interview for an internship with a financial services company—one of the largest in the country—in May 2017 was a godsend. With his foot in the door of tech, Isaiah could build on his frontend design experience by learning backend programming, and, maybe, do well enough to get a permanent position or a different full-time job somewhere else. Such was Clearwater Academy's goal, although it was never promised or guaranteed. Isaiah had gone to high school with Gerrard; he was following in his footsteps from two years behind. However, unlike Gerrard, whiteness disrupted Isaiah's coding literacy from beginning to end. He enjoyed working with his software development team; he was the only Black intern out of seventy-two in the program. Isaiah remembered that there were about six Asian or Asian American undergraduates with the others being white. Yet he enjoyed their company and attending events with them. Isaiah even made friends with one undergraduate. They arranged to meet each day for lunch and started a little rivalry over the best Starburst flavor: cherry or orange. "We essentially clicked," Isaiah explained to me during our third and final interview, "and started scheduling things after that with each other. And it was just really dope how that just happened. And so like I said [the company] would be fine if not for that." Isaiah referred to his squabbles with management, specifically his assigned peer mentor, Sarah, and the peer mentor's product manager, Tonya. These two represented whiteness' tight grip on coding literacy practices in the company.

During his job interview, Isaiah acknowledged that he knew little about backend programming, but he was motivated to learn its languages and become a full stack developer. Isaiah's worth as human capital was learning on the job and learning quickly. He remembered exactly what they told him in response: "This is what they told me: 'You will be learning backend programming. You will be getting training in backend.'" Tonya assigned Sarah to mentor Isaiah, to teach him the programming language C# (pronounced "C-sharp) while he completed small projects for the company. The first two weeks went well: Sarah extended Isaiah's coding literacy practices through one-on-one meetings to discuss C#, instances, and barrier methods (lines of code that instruct a software program to not execute a result until other lines of executed code meet a specific criterion, or "the barrier"). She was also responsive to Isaiah when he was confused. For example, variables and objects threw him off because they seemed similar to each other. What made them distinct? She gave him reliable videos and articles to view and read from home, and then the next day Isaiah met with Sarah to dissect a specific concept from

the materials that he didn't understand. Some tutorial videos were harder to follow because Isaiah had not set up his integrated development environment (programs in which you can write and execute lines of code) on his computer. So he couldn't write the same code from the video. "She would explain it to me," Isaiah said. "I would get it. She would be like, 'You get it?' I was like, 'Yeah.' So I was picking up on it right now. Cool." After those two weeks, the peer mentorship stopped. Sarah had a family emergency out in another state and was gone for a week. Left without a mentor, Isaiah tried to keep his coding literacy going by learning C# on his own while completing smaller projects in frontend design. But the language was too hard, and the only other person he could turn to for help was a senior developer with a tight schedule.

Something changed when Sarah returned the following week: she scheduled one-on-one meetings less frequently, and Isaiah later learned from another intern that Sarah was meeting him instead of Isaiah. Her time and labor switched from a Black computer code bootcamp graduate to a white undergraduate from a prestigious computer science program. What she thought about Isaiah's coding literacy practices became known when Tonya, the program manager, requested to see him for a meeting: "Yeah your skills aren't up to par," Tonya had explained to Isaiah. He recalled that she thought he "wasn't articulate enough as the other interns. And things like that. And what frustrated me about that was that she based that on what my peer mentor said." The racial microaggression on his intelligence was a slap in the face and unfair. Suddenly Isaiah found himself being judged according to the standards set for the other white interns, undergraduates who had cut their teeth on learning multiple programming languages at the same time and steeped in computer science theory. The critique had multiple layers: not only did Isaiah not know backend programming, but it also didn't seem like he was working hard enough, that he didn't go home and study.

Isaiah had significant reasons to pushback against these ideas. First, he yearned to learn and ask questions about computer programming from professional experts. Eventually, Sarah did invite Isaiah to join meetings with the other intern. The topic was objects and classes in C#. But this invitation wasn't to help him strengthen his coding literacy, as had been the case before the family emergency. The meeting itself was a microaggression. From the moment the meeting began, Sarah said, "Look, I was actually skeptical of inviting you to our meeting because I didn't think you'd understand anything relative to the subject. I don't know if I can dumb it down anymore for you." Isaiah pushed back: he wasn't stupid. He could learn computer programming without "dumbing down" the teaching. Isaiah just needed his mentor to strategically teach the concepts in a way he could understand. Sarah, he thought, needed to find a different standard of assessment and instructional method

that would match the experiences of a computer code bootcamp graduate, not a senior college student. Case in point: the intern later helped Isaiah understand C# better than Sarah. Second, studying at home was an ongoing domestic problem for Isaiah. His work ethic outside of the office and the consequences of his hard work ethic was like poetry to my ears. His words are worth quoting at length:

> It's getting to the point, Antonio, where me and my girl have arguments. Huge arguments. Because I come home. I code. I wake up. Go to work. I code. I go home. I code. I code until I'm ready to go to sleep. And they tell me I don't go home and study? I don't go home and practice? That's not fair. You can't tell me that unless you hack my computer and watch me at night. You have no right to tell me that. None.

Tonya layered on one more racial microaggression: "She told me I wasn't articulate enough as the other interns … I don't know. I didn't like that. I want to pull the race card but that borderline racist, for you to tell me that I'm not articulate enough and predominantly most of the interns are white. …" Isaiah had learned the unfortunate assets of code switching as a necessary move in all-white space in high school. He admitted that he came from a "background of thugs"—Black low-income communities raised in disinvested neighborhoods—and it was Gerrard who helped Isaiah realize that background was an affront to white people. While Gerrard picked apart Isaiah's language, he learned from another mutual friend and a mentor how to speak and show himself "business ready." From sophomore year into junior year, Isaiah taught himself how to code-switch. That was, again, an unfortunate necessary skill for community college where he was often in majority white spaces with classmates and professors. "I had to speak formally," said Isaiah. "There was no, 'Yeah aright, I got that. Aight, you can catch me on the flip.' There was none of that; it was like, 'Yeah I'm going to go speak to Gregory because we have to talk about how we can implement this project or how we can set up the gallery.' I switched up quick. Where I could've been 'Aight, I got ta speak to Greg real quick. We need to get this project on the go.' Difference in the way I say things. And they tell me I don't speak articulate." Isaiah was honest about his shortcomings, from not knowing computer programming to making sure he communicated clearly. Sometimes he would catch himself when talking to team members, other interns, or managers. He took responsibility for what they considered his flaws. It was management who did not do the same. Despite these experiences and efforts to code switch, Isaiah felt the weight of stereotype threat on his mental health and emotional well-being; he had become too self-conscious of his language around management: "I'm

that much more conscious. And then every time I speak, I either stumble on my words or stutter." If he messed up talking during a meeting with the two people who had said he wasn't articulate, Isaiah would have to run a formula on the consequences in his head: "She can or she won't, or she will or she won't hold that against you. What are you doing? Are you going to be fluent or are you going to stumble?" Whiteness was getting in the way of his coding literacy (articulating his knowledge of computer programming and asking the right questions) and relationships with more powerful people. His rhetorical performance as a Black body worthy of the rewards given for his labor faltered during the internship. Coding literacy was constrained.

What about support against these racial microaggressions? While some co-workers, including a senior user experience and interface designer, agreed that racism was in play, Isaiah needed help from Clearwater Academy. The computer code bootcamp's internship policy states that if there are issues, a coordinator from Clearwater Academy could mediate. Isaiah didn't trust the policy, however. He ran calculations in his head, considered all the factors that would make everyone point to him as the problem. First, he was an intern; he could make a case to management about his peer mentor refusing to teach him, especially when they had promised that Isaiah would learn backend programming. But that was a full-time employee versus an intern. Who had real power? Perhaps to the coding literate who had proven themselves as worthy contributor to the labor of the financial tech sector. The one with mountains of experience teaching other interns. Second, Isaiah felt that the balance of power wasn't tipped in his favor. He worried that the professional ties between Clearwater Academy and the tech sponsor would diminish his own reputation. Jessica, the employability skills instructor, who would be the mediator between Isaiah and the internship program, had a favorable professional relationship with Tonya. She admired the product manager, and apparently previous intern experiences went well. "If I go talk to Jessica about Tonya, and some of the things she's doing wrong, she's gonna be like, 'Well what are you going to do to improve?' Like I mean I'm trying to tell you I haven't even got the chance to prove that because they're not even trying to give me that."

As the summer waned into July, Isaiah was getting closer to September, the month of his son's expected birth. He wanted to know if he would receive another three months on the internship, so that way Isaiah could start looking for a new job. The peer mentor created new conditions that contradicted the requirements established by management. For example, dressing in proper attire and creating projects using JavaScript from scratch. Although the company used JQuery—a library of pre-made JavaScript code that makes coding faster and more efficient—the manager insisted Isaiah use JavaScript, and then submit the project by mid-July. At the time he received the assignment,

that would be due in just two weeks. Moreover, Sarah did not give more instructions beyond "Impress me." Isaiah was frustrated. First, he had to complete some kind of project without clear guidelines and then do so while completing work that mattered to the company. He appreciated the opportunity to show his ability to develop a solution and execute the solution through computer programming, but the request felt unreasonable and undoable within the timeline given. Management told Isaiah to not follow these requirements from his mentor, but the damage done from the peer mentor remained: Isaiah had not been given a fair chance to show how his skills and competencies as a coder made him valuable human capital to the labor of the company, and the digital ecosystem itself. Human resources requested that Isaiah leave the internship two weeks early and receive his pay for those two weeks. They said he didn't know enough backend programming and that perhaps there was a miscommunication in department's expectations three months before. Isaiah politely made it known that the expectation was clear: he would be learning backend programming. They had failed him. Exhausted by the internship, Isaiah took the request and returned to his newborn son.

Both racial microaggressions—that he didn't work hard enough, and he didn't talk about coding, or talk in general, like white interns—showed that Isaiah's willingness to learn backend programming wasn't an asset; it was a burden and a liability. But it wasn't computer programming itself; it was that he also didn't embody the behaviors, practices, and language of whiteness that seemed to be a prerequisite for workplace culture and collaboration. The assets he brought with him—code switching and an intense work ethic that even upset his partner—were hidden underneath expectations that perhaps didn't match with computer code bootcamp graduates or at least were too rigid to adapt to Isaiah's specific background: a Black coder with no backend programming knowledge. This internship experience—using coding in the context it was meant to be used—helped him consider new opportunities with other kinds of literacy practices for himself.

Zelda

Stereotype threat in all-white spaces suggest the high cost of performing whiteness or not in the internship. However, whiteness hovers like a specter even if you do not feel the burden of stereotype threat in a tech workplace. Zelda discussed her work as a web developer and data analyst in the marketing department with confidence and poise. She joined a company that provided management systems for clinical software in January 2018 as a five-month intern and then got an extension to stay through August. We were chatting at a lake in a park at the end of her internship, watching her two-year old daughter

try to play with ducks. We had last spoken in March before then. Remembering Isaiah's harrowing tale from last year, I asked Zelda why she got an extension. Zelda responded with amusement: "You know my deadlines? I'm a cool person? People fucking like me?" Yes, she had met her deadlines, she had done the work the company wanted her to do to meet their standards. Her immediate supervisor also pulled for Zelda's extension, when the company wanted to end the internship because the supervisor was due to have a baby soon.

Unlike Gerrard and Isaiah, Zelda felt no stereotype threat. She did feel out of place, less so because she was the only Black woman in the small startup and more because Zelda had a hard time connecting with people who mostly lived privileged and safe lives. Most co-workers were young and had college degrees, but they didn't make mistakes or do stupid illegal things. Only her manager made what Zelda considered a mistake: having a baby at fifteen. Zelda herself gave birth to her daughter as an adult, but she could still relate to her manager. Both had "fucked up" at some point in life. That's the reality of life, and for Zelda that meant her manager had cultural and social capital. She knew what was up. However, Zelda carried a different kind of capital: proximity to whiteness. She was visibly Black, but her daughter was visibly white. Just having a picture of her daughter on her desk would change Zelda's white co-workers' disposition. "I don't know," said Zelda. "I put a picture up and when people see it, like, 'Oh my God, is that your daughter? So cute!' And then they start talking to me. More often than they were." When Zelda brought her daughter to a company picnic, co-workers could not resist seeing and talking to her. Zelda made note that this attraction to her wasn't a symptom of a white tech workplace; it was a marker of whiteness in general. If Zelda went with her Black friends to the farmer's market, white people would hesitate to have a conversation; show up at the market with her daughter, and white people suddenly have the agency to cross racial lines and talk to Zelda.

I would not have believed Zelda had I not witness the phenomenon as I interviewed her for the first time in March at a diner. At one point, our server stopped and lingered at the table conversing with Zelda's daughter. She agreed with my analysis, that her white-passing baby made Zelda "safe." She found a different word to describe the behavior: "It's like a level of—there's this level of *acceptance*. Whatever. I don't know." Although coding literacy and all the employability skills that come with it make the workplace, well, work, whiteness tends to seek itself out in a culture of meritocracy, to find comfort, even when found on a Black person. It seeks a mirror image that's replicable. That's one reason Zelda settled into an all-white space without much push from stereotype threat on her coding. And Zelda was happy to take advantage of the position her daughter seemingly gave to her if that meant she could get ahead in the internship. By the time we finished our first interview at the diner, the

only challenge Zelda faced was trying to make computer programming fun. She was on a mission to find a new fun language among the abundance of languages the internship already had her use, such as HTML, CSS, JavaScript, JQuery, PHP, SQL, and SAS.

However, when we met again in August, at the end of Zelda's internship, fissures had grown and spread throughout the internship. It began with multiple people quitting the company. Before her supervisor went on maternity leave, the lead designer quit. "He was telling me like why he left," Zelda said. "He was telling me how much he hated our VP [Vice President]. I was like, 'Oh she's not too bad.' I didn't think she was too bad." And then another person left, a Clearwater Academy graduate that was in the same cohort as Zelda; this graduate also disliked the vice president because she "Doesn't give credit where credits due, overworks people, doesn't have any respect, takes the credit of everybody's work or something." This last bit of advice resonated with Zelda, as she slowly noticed these problems. Strike one was a problem that she had not thought about, even though it was an egregious slight on her coding literacy and labor. The company was going to host a conference for medical professionals; Zelda took on the task to design the website and the speakers' webpage. The vice president assembled the team, including Zelda's manager, to look at the page in-progress and give Zelda feedback. But Zelda wasn't invited to the meeting. Her manager was pissed but Zelda thought nothing of it.

Zelda did receive feedback after the website went live. However, the comment had less to do with the coding she did. The team loved the videos embedded on the website, but Zelda had nothing to do with the video itself: Zelda recalled, "They were like, 'Oh the content on this video was amazing. Whoever put this together is great!' And I'm like, "That's so funny because it was just on the spring conference site. It's the same video." Zelda had simply made the video stand out more by moving the play button to the top of the page. The content may have popped, but Zelda made it so. She did push for more comments that would give more insight on her web design skill. She emailed a few co-workers. Only one responded. This person said the video was great and the website itself followed accessibility guidelines well. She asked where the web design needed improvement. "And after I said that he's like 'Well I really wish the speakers were on the agenda.'" Zelda paused and began to spiral. None of the speakers were listed on the webpage; she had thought something happened to the code and the problem was on her. But then Zelda realized that the content wasn't her responsibility; someone else had not assigned speakers to the schedule. Zelda told the co-worker that the content would be published, but in the back of her mind, Zelda thought, "Damn really? Like that's what you want to say? Like—they're not linking with each other? But like the first two sessions have their speaker already

linked. So you scroll down to see that the rest weren't linked with each other yet. Like 'What the fuck?"" Zelda had positioned herself as a strong coder, despite finding the process and practice boring. Nevertheless, even full stack developers need code reviews and user testing on web designs. So her labor and time had been overlooked, and the web content needed to complete the website was in the labor and time of others who had not attended the peer review meetings.

Then strike two happened, and the problems started to become clear to Zelda. The story went like this: Zelda had to work remotely because her daughter was sick, so she and her co-worker Anna joined a team meeting via phone. She had liked the vice president up until this meeting. For the entire hour, the vice president never once acknowledged Zelda's presence; she did acknowledge and speak to Anna but did not even tell the team Zelda was on the call. And who joined the call was clearly available: you just go to webinar and see who is present, who is muted, everything. And Zelda was no stranger to the vice president: everyone knew everyone in the small company, and Zelda had been "nice as fuck" to the vice president from the first day of starting the internship.

The final strike came at the end of June, a month before Zelda' s internship concluded. Her manager was about to take maternity leave on June 28th. She wanted to know Zelda's plans for the coming July 4th weekend; Zelda had planned to travel but being an intern, she got all the pay with none of the benefits like getting a bonus or paid time off (PTO). Thus, she planned to do some remote work in her spare time. And so began the process of regulating her labor and coding literacy through bureaucracy: can Zelda work remotely as an intern? She and her manager understood that this was a given benefit, but the vice president caught wind of the idea. She denied the request to work from home. This opportunity, she argued, was for anyone on medical leave. Zelda had a chance to double down on that policy then, as she had gotten in a car accident several days before. The doctors diagnosed her with a concussion and with a stress disorder. She could still code but from home would help, and Zelda could still get paid. Although her manager argued for paid time off and even shared Zelda's medical reasons, the vice president still denied the request.

A slight on Zelda's labor; her time and labor wasn't acknowledged by her vice president. To be a valuable worker, you must feel valued. Your work must be acknowledged. Your name must be called. Grace must be given when policies on labor and health care overlap. Zelda may have navigated whiteness well, but other forces mitigated her worth and coding literacy practice. The barrage of slights on her labor and assets diminished Zelda's motivation to keep working for the final month of the internship. What was the point of giving respect when respect was not given back? Zelda found that they had

less care for everyone else; that is, if goals were met, people like the vice president had little care for the individual professional fulfillment of employees. The strikes coalesced into unbearable working conditions after Zelda's first manager left; without full support from her, Zelda was left adrift in a sea of cynicism, after her old manager had kept the respect flowing from upper management to herself. Without much support, and having her labor slighted multiple times, Zelda lost motivation to keep meeting deadlines. In the final month of the internship, Zelda planned to do the bare minimum and do that minimum quickly so she could do nothing for the rest of the workday. And still get paid. Zelda was prepared to make up excuses: "I can just tell my manager 'I was trying to work on this, trying to work on it but people keep asking me to do other shit. Like—So it didn't get done. Oops my bad.'" She had done what she was supposed to do. Finished all tasks. There was nothing left to do but get the final paycheck and go on her planned vacation.

Zelda had abided by the whiteness of her internship; with the help of her daughter and her own work ethic, she upskilled in coding literacy and increased her worth as a coder. Zelda's labor brought profit and reputation to an already established startup. However, doing everything "right" had not demonstrated her value to management. From ignoring her labor to withholding benefits that she could use for medical care; Zelda found the structures around coding literacy unwelcoming and unprofessional. In some ways, the company exploited her labor to their benefit without acknowledging how they gave value to Zelda.

Conclusion

Black adult learners encounter inequalities on two fronts after graduating from Clearwater Academy. The consequences of racism, in particular poverty, and being shut out from designing digital technologies. They hope the invitation to work in tech suggests a new direction in the history of the tech industry; that more Black people can get a seat at the table. However, the experiences of Black adult learners from Clearwater suggest otherwise. A systemic reprieve on the other side of coding literacy doesn't exist. That means for industry and computer science education, structures of inequality remain rigid. Participants feel the tight squeeze of inequality in life and inequality in the job market they hope to break into. Coding literacy locked down into typical design philosophies and expectations suggest that policies and practices—social and culturally informed policies—around who codes and why they code hardly promotes any justice-informed recruitment or change in design technologies from white spaces. While Black participants in this study believed in the power of coding literacy, and in some way still held onto the imagined possibilities

of their lives not only achieving social mobility but also using technology to claim independence in the labor market, coding literacy's close tie to a specific area of tech hindered literacy development at that point in their lives. If the idea was to extend their literacy development, and thus their place in the economy for financial gain, new standards in computer science and tech would recognize the value Black people do have that may not look like the standards or traditionally considered for assessment. Easing known microaggressions and equalities removes one wall for Black adult learners.

But in the meantime, their discourse about the job market and hopeful imaginations of a Black tech ecosystem—not a pipeline—suggest there's a better way to disperse Black people across many sectors of technology; although it sounds counterintuitive to the need for more Black coders engaged with designing technologies, we may think of coalition building that deploys multiple kinds of Black technical knowledge, from computer programming to design to hardware to project management, in private industry but especially in community engagement. Rather than a single pipeline from start to finish, consider the relationship between race, labor, and coding literacy as a network of large and small pipes transporting knowledge and literacy practices throughout the digital ecosystem. Once participants realized their sour position in the job market, they imagined other possibilities—they began ideating and acting on their existing knowledge to better position themselves. Legacy institutions such as community colleges and their information technology programs seemed more attractive than computer programming. Coding literacy joins an ecology of literacy repertories; it becomes an option, should an option arise, or a spirit to summon in the final stage of an interview battle, or it sits collecting dust as the tools of coding literacy in 2017 are supplemented by artificial intelligence in 2025, a technology Richard had no way of teaching to his adult learners.

These experiences paint a more nuanced picture of literacy development through the labor of tech companies. While they do confirm previous studies and reports that show how damaging local white tech companies can be to Black adult learners, they also show the ways coding literacy still echoes or resonates in their lives long after the computer code bootcamp concludes. Clarity has come to my mind multiple times while analyzing the data and writing this book. And the more I wrote and thought the more I gained clarity on how coding literacy rewards agency and independence. Kevin's big takeaway, for example, was not coding literacy or finding a job in software development; through that failure he noticed other skills and competencies: learning how to solve problems in his workplace as a hardware repair specialist and in his own family life. And then the pivot toward social services for children, a call back to his volunteering for Black LGBTQ+ youth in Sakowin. While

Black adult learners lose in the tech industry, they gain more for their personal lives and future educational pursues. Clearwater Academy's gift to participants far outweigh any job a graduate finds. Even Gerrard confirmed this idea: shifting from learning coding with women and BIPOC adults and then reinforcing his learning in an all-white internship only taught him to move above and beyond what white tech offered, beyond what white tech needed. He wanted to cultivate a world for the Black diaspora, to undo or avoid the consequences of tech capitalism and its policy of exploiting poor laborers in Congo. Although coding literacy had less value in many participants' lives, they did agree coding literacy had value to Black communities in general but not for the kinds of digital lives computer code bootcamps suggested in their own discourse.