

Coding into the future: Why students should start coding today



Introduction

Have you ever wondered about your favorite games, apps, and websites? How do they work? Who created them? Well, I have the answer to this question. Behind every digital experience lies something magical. But this magic is created by humans themselves. The magic we are going to discuss in this article is *code*. In the world around us – from our phones and websites to our cars and even smartwatches – code is working quietly behind the scenes.

All around us, people are creating innovative things with just a computer and a bit of imagination. They're building games, making apps, creating websites, and solving real-life problems. It's not just adults now; even students from every part of the world are joining in this amazing experience. Learning something new doesn't have to wait until you're older. Why not start today? Today, everyone curious and brave enough to try can start exploring, building, and sharing ideas in ways that weren't possible before. And behind all is a skill that helps turn ideas into something real.



What is coding?

This is a very common question everyone asks. We should know what coding is before we dive deeper into it, right? Coding, also known as programming, is the process of writing instructions that can tell a computer what to do. These instructions are written using special languages called *programming languages*, such as Python, JavaScript, or C++. Each language has its own set of rules, like grammar in spoken languages. Imagine you're giving directions to a robot. You say "Move forward," and the robot follows as you gave the right command. In coding, you write that instruction using a programming language like Python or JavaScript.



Why should students learn to code from an early age?

The answer is pretty simple. Here are 9 benefits of coding:

1. Imagine building your own video game, creating an app that helps your friend stay organized, or automating boring homework tasks with a few clever clicks. That's what learning to code empowers you to do—it turns imagination into action.
2. Coding helps sharpen your problem-solving skills, boosts creativity, and teaches you how to think logically. It's more than just screens and numbers; it's about understanding how things work and figuring out how to make them work better. Students who learn to code often find new ways to express themselves and discover talents they never knew they had.
3. It improves focus and patience. Coding teaches students to slow down, think carefully, and stay committed. Debugging errors and solving problems require persistence, helping students develop patience and learn that giving up too soon won't get results.
4. Coding also enhances communication skills. When students explain how their code works, they learn to talk clearly and in order. Coding helps them organize their

thoughts, describe steps simply, and share ideas in a way that others can understand—just like giving good instructions.

5. Coding opens global opportunities. Coding is used all over the world, so learning it helps students connect with people from different countries. Whether it's working on shared projects or joining online communities, it shows students how their skills can reach far beyond the classroom.
6. It turns play into creation. Instead of just playing games, kids can learn how games are built—and even make their own! They understand what's happening behind the scenes and discover how fun learning can be when it feels like creating something exciting.
7. It builds tech-savvy awareness. By learning how programs and apps work, students get smarter about using technology. They can spot scams, protect their privacy, and choose safe tools. It helps them be confident and careful while using the internet.
8. Coding encourages self-learning. Coding often sparks curiosity. Students start asking, “What happens if I change this?” or “Can I build something new?” This kind of thinking helps them learn on their own, try new ideas, and feel proud of solving problems without help.
9. Finally, guess what? The skills you gain aren't just for tech careers—they're helpful in science, art, business, and even music. Coding isn't just for programmers; it's for innovators, creators, and curious minds.

Real-Life Example

1. Iddris Sandu – Tech Prodigy at 13

At just **13 years old**, Iddris Sandu developed an **app for his high school** that helped students navigate classrooms and access resources. But that was just the beginning. By **age 15**, he was **consulting for Google**, and later collaborated with **Kanye West and Nipsey Hussle** to create tech-driven community spaces. His work blends **coding, design, and activism**, and he's become a symbol of youth innovation in tech.

Why Everyone Knows About Him

- Featured in major outlets like *Forbes*, *CNN*, and *TechCrunch*
- Recognized for bridging culture and technology
- Advocates for tech literacy among underrepresented youth

2. Tanmay Bakshi – AI Prodigy at 13

At just 13, Tanmay Bakshi had already built *AskTanmay*, the world's first IBM Watson-powered natural language Q&A system, and was working on AI models to help people with neurological disorders communicate through brainwave signals. He began coding at age 5, and by 9, launched his first app on the Apple App Store. His mission is bold and empathetic: using artificial intelligence to improve lives, especially in healthcare and education.

Why Everyone Knows About Him

- Became a Google Developer Expert and IBM Watson advocate before turning 15
- Featured on platforms like TEDx, The Wall Street Journal, and Times of India
- Advocates for ethical AI and tech for social good
- Authored books and runs a YouTube channel teaching programming concepts
Tanmay's journey shows how coding can amplify human potential—bridging tech with empathy and impact.

3. Emma Yang – Tech Trailblazer by 12

Driven by her grandmother's Alzheimer's diagnosis, Emma Yang created *Timeless* at age 12—an app that uses facial recognition to help patients remember loved ones and stay connected. She began coding at 6, merging compassion with innovation in a way few others her age could. Emma's vision wasn't just to build tech—it was to restore human connection through it.

Why Everyone Knows About Her

- Became the youngest MIT Solver and won the Women Startup Challenge at Google
- Featured in Forbes, TEDx, Talks at Google, and World Mobile Congress
- Created an app that tackles Alzheimer's with facial recognition
- Advocates for girls in STEM and teaches computational thinking through her own camp

Challenges and How to Beat Them

Learning to code can sometimes feel like solving a tricky puzzle with missing pieces. Students may face bugs (errors in the code), confusing instructions, or moments when things just don't work as expected. It's normal to feel stuck—but that's where growth begins. The secret is not giving up. Coding teaches patience by showing that every mistake is a lesson, not a failure. When things break, students learn to slow down, review their steps, and fix the problem piece by piece. It also helps to ask for help—from teachers, friends, or online forums—because no one solves everything alone. Over time, students become more confident. They learn that setbacks are part of creating something great. Coding rewards persistence, and with each small success, learners take one step closer to building something amazing.

Seeing the lines of code, you might be a little frightened, but eventually, you'll overcome that fear. When my father used to code, I used to watch and say how scary it looked, and that I would never learn to code. I was just frightened at that time. Eventually, I managed to make myself pick up that computer and start to write a few lines of code. And now I'm obsessed with programming. I even created projects that are over 300 lines long. And all of this started from a girl who was so afraid of programming that she wouldn't even go near the computer. Thus, just believe in yourself and you'll be very proud of yourself one day.

Conclusion

Coding is more than a technical skill—it's a mindset. It teaches students how to break down problems, think critically, and approach challenges from new angles. At first, lines of code can look like an unfamiliar language, but with time and patience, students learn to turn those lines into meaningful instructions, dynamic programs, and even creative digital projects. Every error they encounter, every bug they fix, and every breakthrough they celebrate contributes to a deeper understanding—not just of computers, but of how to learn and persevere.

In school and beyond, coding unlocks opportunities. It connects with fields like science, art, finance, medicine, and engineering, giving students the power to innovate in whatever path they choose. But its true value goes even deeper: coding builds grit. It shows learners that failure isn't something to fear—it's something to work through. It encourages curiosity, rewards experimentation, and helps students discover how powerful their own minds can be when they stick with something hard.

Most importantly, coding gives students the chance to be creators—not just consumers—of technology. In a world increasingly shaped by software, learning to code means taking part in that shaping. From building apps that solve real-world problems to writing programs that bring their ideas to life, students gain more than just a skill—they gain a voice in the digital future.

So, if you're just starting out or struggling along the way, know this: every coder begins with confusion. But every step forward is a win. Stay curious, be resilient, and keep writing code—you're building something amazing with every keystroke.