

CODING *for* KIDS

*A guide to empowering children
through programming skills*



TYNKER™

Table of Contents

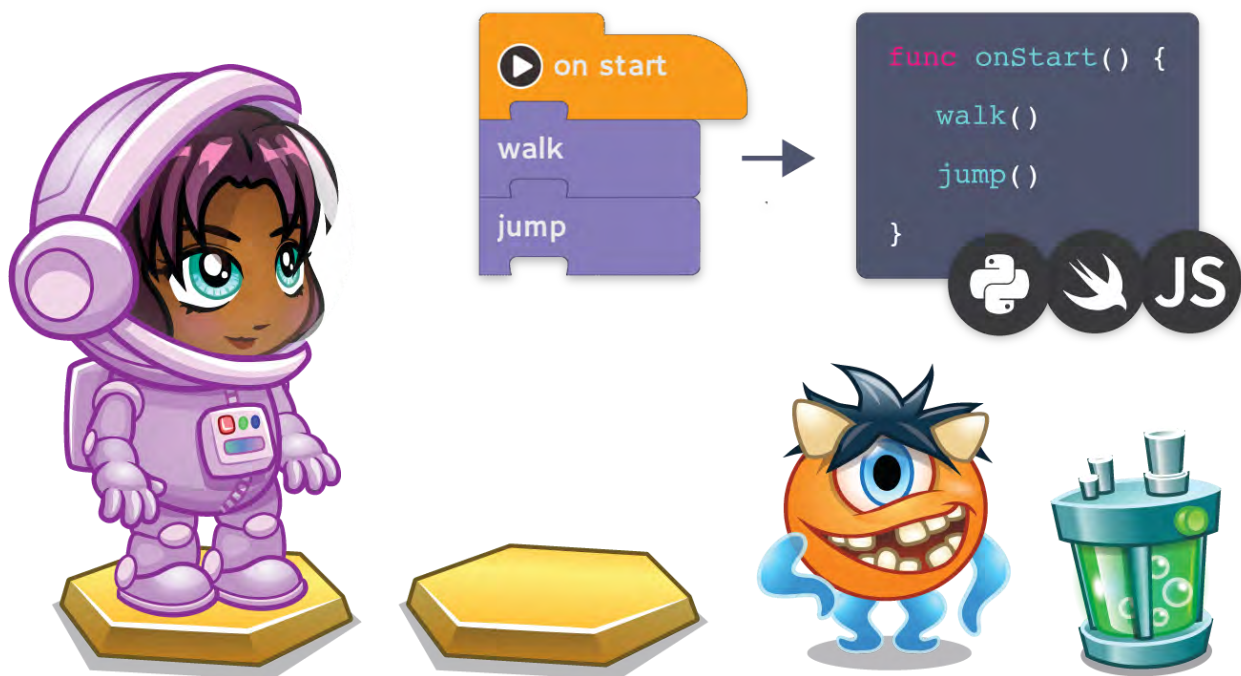
Intro to Coding	3
What Is Coding?	3
What Can My Child Do With Code?	4
Why Should My Child Learn to Code?	5
Coding Improves Your Child’s Academic Performance	5
Coding Develops Important Life Skills	6
Coding Prepares Your Child for the Future	7
Why Tynker?	8
Interest-Driven Learning	8
Game-Based Learning Experiences	9
Seamless Advancement to Text Coding	10
Tynker’s Global Community	11
Getting Started	12
Join Tynker for Free	12
Enroll in a Premium Plan	12
Taking A Deeper Dive	13
Begin Coding at Home	14
Learn How Coding Helps Kids Grow	24
Add Coding to Your School’s Curriculum	44



Intro to Coding

What Is Coding?

There's been a lot of discussion around coding lately, but it can be hard to figure out exactly what it means to code and how it plays a role in your child's future. Coding (or computer programming), is the process of providing instructions to a computer so it performs a specific task. You may have heard of popular text languages like Java, Python, or Ruby, but even kids can easily learn to code using a visual block language like Tynker!



Why is coding so important? Believe it or not, we rely on code in the technology we use every day – our mobile phones, thermostats, televisions, cars, and even the device you're using to read this wouldn't exist without code.

What Can My Child Do With Code?

In addition to the many practical and innovative uses for code in today's world, it is also a creative medium. With a coding education, your child can use their new skills to create almost anything they imagine!



Make Apps & Games



Create Animations



Mod Minecraft



Control LEGO®



Fly Drones



Explore STEM

Why Should My Child Learn to Code?

When it comes to preparing your child for the future, there are few better ways to do so than to help them learn to code. Coding can help your child develop academic skills applicable to any grade level, in addition to building critical life skills like organization, perseverance, and problem solving.



Coding Improves Your Child's Academic Performance

It's been proven that learning to code reinforces math skills, helping kids visualize abstract concepts and apply math to real-world situations. It also teaches logical communication, strengthening both verbal and written skills.

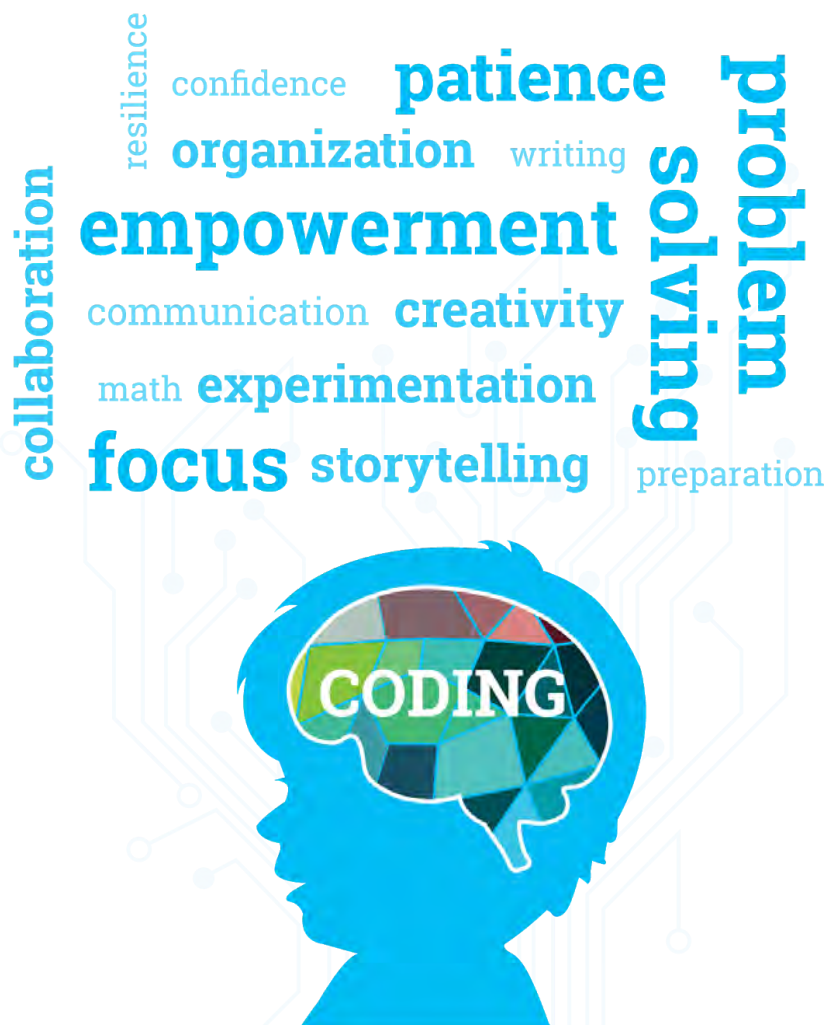
Learning to code means learning a new language!

Coding instills qualities like creativity that help kids perform better in school. When they code, kids learn through experimentation and strengthen their brains, allowing them to find creative solutions to problems.

Coding Develops Important Life Skills

Coding is a basic literacy in the digital age. It's important for your child to understand and be able to innovate with the technology around them. As your child writes more complicated code, they'll naturally develop life skills like focus and organization. It also develops resilience – when kids code, they learn that it's ok to fail and improve. There's no better way to build perseverance than working through challenges like debugging code!

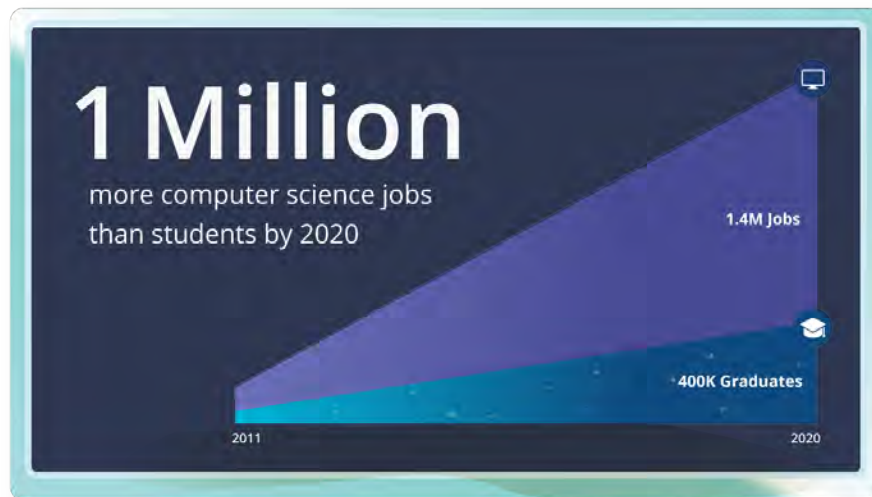
Kids also feel empowered to make a difference when they code – many kids in Tynker's global community use the platform to spread messages of tolerance and kindness. Parents have even reported that their kids develop more confidence as they learn to problem-solve through coding.



Coding Prepares Your Child for the Future

In today's rapidly evolving digital world, it's more important than ever that your child has the skills they need to adapt and succeed – and coding is a big part of that. Jobs are quickly becoming automated, and half of today's highest-paying jobs require some sort of coding knowledge. By 2020, there will be 1.4 million computer science-related jobs but only 400,000 computer science graduates to fill them, according to the Bureau of Labor Statistics.

At the very least, kids today must be familiar with basic coding concepts in order to prepare for the job market's demands. And like learning a second language, learning code is best done at a young age.



\$98,260

Median salary of
a Software Developer

***"I think basically every job in the future is going to
have to require some part of coding!"***

– Scout, Student in Australia

Why Tynker?

Our award-winning creative computing platform helps kids develop computational thinking and programming skills in a fun, intuitive, and imaginative way. As they're guided through interactive game-based courses, kids quickly learn fundamental programming concepts. With Tynker, your child can apply their coding skills as they build games, tell stories, mod Minecraft, create apps, control drones and robots, and more! We even offer a parent dashboard where you can follow your child's success and share their creations.

Interest-Driven Learning

Tynker's scaffolded curriculum is organized around interest-driven learning. It's a simple philosophy that means kids who already love to play with LEGO® or Barbie® will be more inclined (and genuinely excited) by the chance to integrate Tynker with those interests, expanding their potential to play as they learn.



Kids begin to code using Tynker's block-based visual language, which helps them recognize patterns and master programming concepts like sequencing, loops, conditional logic, and algorithmic thinking.

From there, they can flex their creativity by animating their own games and telling stories with code. Kids who'd like to dive deeper into concepts from the classroom will enjoy creating projects with our STEM tutorials and puzzles. Our interactive

notebooks make it fun for independent makers to learn JavaScript and Python at their own pace.

For more hands-on learners, our drone and robot programming courses are the perfect way to apply coding to the world around them. With the Tynker app, kids can program their own mini-drone to fly patterns, perform flips and stunts, and even transport objects. And with LEGO® WeDo 2.0, kids can use programming to bring their LEGO® creations to life!

Game-Based Learning Experiences

Tynker's self-guided courses are made up of individual lessons designed to create a fun and captivating learning experience. Lessons build upon each other to ensure concept mastery, and guidance from interactive tutorials empowers kids to learn independently and at their own pace.



As they learn, kids build mini-games, solve puzzles, create coding projects, complete daily missions, earn exciting badges, and unlock new characters. This is why kids love learning with Tynker – even though they're mastering important programming concepts, they feel like they're just playing a game!

Seamless Advancement to Text Coding



Tynker introduces kids to coding with simple visual blocks. This technique allows young makers to learn the fundamentals of programming and create incredible projects without the frustrations of syntax. Whenever they're ready, kids can start experimenting in those same block-based activities by switching between visual and text code blocks. Once kids become familiar with programming basics and syntax, they can move to full text programming with Swift, JavaScript, and Python.

Tynker's Global Community

Tynker's award-winning platform is used by over 80,000 schools and 60 million kids, spanning more than 150 countries. Global partners include brands like Apple, Microsoft, Mattel, PBS, Sylvan Learning, and more.



80,000

Schools have coded with Tynker



60 Million

Students have used Tynker to learn to code

Getting Started

Tynker's game-based learning environment makes it easy for your child to learn to code! Your child will begin by using Tynker's visual blocks to learn fundamental programming concepts through self-guided game-based activities, then graduate to text coding languages like JavaScript, Python, and Swift as they gain confidence in their new abilities.



Join Tynker for Free

Create a free Tynker account to give your child access to fun, free coding activities. Access to our safe, and secure tools is free, forever.

- Get started with free coding puzzles
- Code simple projects using tutorials
- Create Minecraft skins and mods
- Safe + Secure

[DOWNLOAD THE TYNKER APP](#)

Enroll in a Premium Plan

Master skills five times faster with award-winning courses and tutorials. Watch kids learn to code and build amazing things.

- 1,500+ fun learning activities
- 18 online courses
- 10 iPad courses
- Private Minecraft server
- Student progress tracking
- Affordable & flexible plans

[EXPLORE COURSES](#)



Taking a Deeper Dive

Tips for Encouraging Coding at Home	14
Make Screen Time Productive	15
Explore Tynker’s Maker Community	18
Get Inspired by These Kid Coders	21
How Coding Helps Kids Improve in Other Subjects	24
Improve Writing Abilities	25
Strengthen Math Performance	28
Develop Creativity	34
Build Focus and Organization	37
Start Early: Girls Who “Make” Choose STEM	37
Invest in Your Child’s Future	41
Add Coding to Your School’s Curriculum	44
Try an Hour of Code	45
Build a Makerspace for Hands-On Learning	46
Hear from a Tynker Educator: Why Our CS Programs Are Working	47

Begin Coding at Home

The best way to help your child learn to code is to treat it like any other extracurricular activity – make a habit out of it! Building a habit can take a little time, but once your child integrates coding into their daily schedule, you will see their learning accelerate and their engagement skyrocket.



Check out the following articles for ways to make screen time more productive, discover some fun (and free!) coding activities that you can do at home with your child, and get inspired by other makers in Tynker's global community.

Make Screen Time Productive

If you're concerned about the amount of screen time your kids are engaging in, you're not alone. Millions of parents worry that kids spend too much of their lives glued to screens, and for good reason – according to the BBC, the average child spends over 6 hours each day looking at a screen.

But we'll let you in on a little secret: you can give in to your child's request for screen time by leveraging that time with uplifting, engaging, and creative activities! It's a trend on the rise, and it's easier than you might think.



Blend Learning and Fun

Leveraging screen time doesn't mean allowing your children to play only strictly educational games. They don't need to be drilling math facts or learning history – the key is to find ways to keep them thinking and creating. Let them discover that learning is fun!

Try swapping simple, distracting games for tablet art apps, games that involve solving mysteries or puzzles, and coding! Encourage your kids to find inspiration in the world around them as they make art on their tablets. When your kids look for a game to play, help them find a game that uses problem-solving skills.

“He actually gets to use the skills that I like to see him work on, like sharing artistic talent, music, art, as well as math.” – Lauren, Featured Maker Noah’s mother

Try having your kids program connected toys like drones or robots. They’re hugely popular for good reason – kids get to use screens to create, then see the physical effects of their hard work and creativity in real-time. It’s entertaining, of course, but it also provides a great feedback loop!

Distinguish Between Playing and Making

We all love a little bit of mindless screen time now and then – social media is to you as gaming is to kids – but kids have an immense capability to learn and create that needs to be explored. Creativity is one of the most important skills your children can develop – in fact, a recent Forbes article cites creativity as the most important skill for a future of AI!

A good rule of thumb for leveraging your child’s screen time is to find ways to focus on “making” rather than simply “playing.” When kids are playing, they’re following predetermined paths. When they’re making, kids pave those paths.

Here’s an example: your child is playing a game, clicking and dragging, following a predetermined path. She’s enjoying herself and relaxing, but she’s not as engaged as she could or should be.

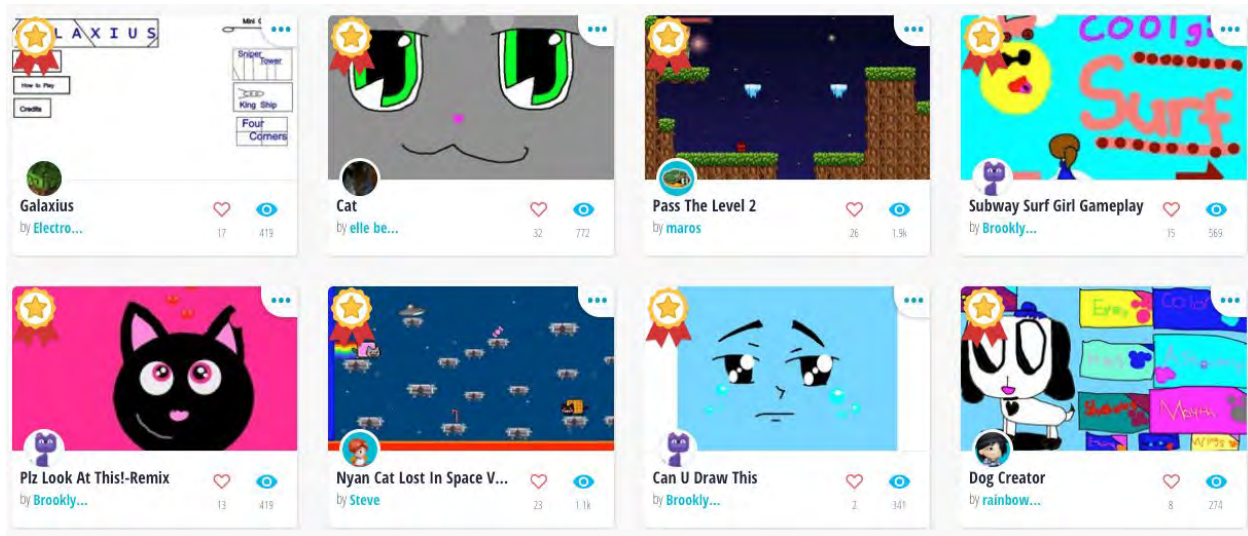
When she switches to building her own game, she’s interacting with the platform in a new way. She’s entertained by the prospect of a challenge, and pushes herself to learn new things in order to figure out how to create what she wants to make.

“My husband especially is very anti-screen time, so he doesn’t like the kids just watching TV – but this kind of activity is something [Max] can do where he’s actually being productive. There’s a difference between passively just absorbing something and then actively engaging with it.” – Stephanie, Featured Maker Max’s mother

Explore Tynker's Maker Community

What happens when more than 60 million kids share coding projects with one another in a supportive and collaborative space? A whole lot of fun – and loads of teaching, sharing, and supporting!

When asked about their favorite Tynker feature, kids list things like the ease of coding with blocks, the ability to draw, and – almost unanimously – the Tynker community!



We are so excited by the way the Tynker community has taken off. We provide the framework and support for the community, but all the trends, ideas, and teaching happening in the community is self-perpetuated. It's constantly changing and improving, but one thing stays constant: it's always inspiring!

"I like how when you look at the Tynker community there are so many projects that are all so different, but they're all made with the same software. You look at the code and there's just so many opportunities for what you can make – it's proof that everyone can come up with a different thing. From the same building blocks, you can make all the ideas that you have!" – Featured Maker Scout

They Teach Each Other

Not only does the Tynker community motivate and inspire kids to code, but it also helps them learn! When we ask kids how they learned to use Tynker, they often cite the Tynker community. Kids look at the code in community projects in order to learn how to code elements of their own games or projects in Tynker. We've even noticed kids making "How to Use Tynker" tutorials as projects in Tynker!

"I'm trying to figure make a joystick instead of the button system that I have," Mark said, "but I haven't figured out how to do that yet. I've been looking at other projects to try to figure out how to do it." – Featured Maker Mark

They Share Ideas

Using Tynker's tools, kids have embraced the ability to truly create anything in their imagination! Kids use Tynker to share what's on their mind, teach others skills (like how to draw a turtle!) make games to share, and even create educational math activities.

"It's not just coding, but it's also giving them an opportunity to learn other topics, and share what they know about other topics, which I thought was fantastic." – Guru, Featured Maker Yaamini's father

It's inspiring to see that when given freedom to make whatever they want, many kids opt to educate about the environment or spread messages of respect. A few months ago, one maker started a trend around projects about respect and encouraging others to be more kind. Other kids caught on quickly!

They Support Each Other

After kids' submitted projects are approved and published, they garner likes and views from other users. The praise kids get from their peers is exciting and validating!

“When she posted her first coded game, she was amazed by the number of views she garnered within a few days and over a week. She couldn’t believe that her game was that popular, and the view of her first game encouraged her to continue coding other games.” – Tynker Parent

A love for coding (and a common understanding of its difficulties) bonds the community together! They’ve all experienced the satisfaction of solving a complicated problem and the frustrations along the way.

“Almost everyone on Tyner has made their own program, so I think that people kind of appreciate everyone else’s hard work more.” – Tynker User

A supportive community is an asset to kids learning to code. The ability to see what other young makers are up to, give and receive positive feedback, and share their message is important; it’s a way for kids to teach and learn concepts like kindness as well as skills like animation (or how to make a pancake!).

Get Inspired by These Kid Coders

Hailing from all around the world, our Featured Makers begin their coding journeys in many different ways! They discover us through enthusiastic teachers, supportive parents, or all on their own; we discover them as we approve each project for the Tynker community. Each week we share the profile of a child whose projects we're especially impressed by. Some create projects that are artistically stunning and others write complex code – and all of the makers we feature inspire us with their hard work and dedication. Will your child be next?

Our Featured Makers all have big dreams and the determination to achieve them – catch a glimpse of some incredible kids who are coding up a storm!



Kindness Advocate: Layla, 6th grade, California

Kids spread kindness in the Tynker community; one of the first to code projects about kindness and respect was ten-year-old Layla. She was inspired at school and chose Tynker as the platform to spread her message. She told us, “My school had a program about being leaders, so I wanted other people to be leaders and show respect to others. I started posting projects about respect and other people did it too.” Not only does Layla create projects advocating for a better world, but she codes projects full of tips for fellow Tynkerers!



Future CS PhD: Aidan, 5th grade, South Carolina

Tenacity and passion are great characteristics for young coders, and it turns out eleven-year-old Aidan is chock-full of both! During our interview, he showed us a huge LEGO project he's working on and said, “This one has 2793 pieces. I was building it all of yesterday, and I only got two-thirds of the way done.

I was working all day since I woke up!" Aidan has big dreams – he plans on a PhD in CS! In his words, "I want to do something to do with Computer Science. Thanks to Tynker, I have a head start!"



Creative Storyteller: Grace, 3rd grade, Scotland

When Grace grows up, she'd like to become either a doctor or a coder! In the meantime, she's practicing by creating amazing stories on Tynker. "I like to code because it's really fun to make your own game and decide what will happen in it," she says. "I like when other people like my projects. It inspires me if someone has a better game than me because it means I can look at it and see what they've done and if I can do anything different with my game." We're so excited that

Grace has found support for her passions in the Tynker community!



Strong Girl in STEM: Scout, 8th grade, Australia

Every girl deserves to feel welcome and included in the STEM world, and fourteen-year-old Scout is helping to craft a future in which women in STEM is the norm. Scout is involved in a club supporting coding for girls in her community, has taught herself

Python, and has a deep understanding of the importance of learning coding. According to her, "I definitely think that lots of people should try to get involved – it's so important! At my school, there's not much coding, so people should try to take advantage of all the opportunities that are given."



Budding Game Designer: Anthony, 5th grade, Illinois

Every Tynker user has their own reason to code and can relate coding to their dreams for their futures, whether those dreams include CS or not. For Anthony, Computer Science gives him the tools to follow his passion – gaming! He aspires to open a massive gaming center his father used to own, and even wants to design games himself. According to Anthony, “I think every kid in the world should code! It’s so fun, and it gets your brain thinking. Why not have no school and just code all day? Same thing, right? It gets your brain thinking!”

When you give your children and students a bit of inspiration and access to tools that teach, you provide them with a world of possibilities. How will the kids in your life use Tynker to express creativity and follow their dreams? We can’t wait to see what they come up with!

Learn How Coding Helps Kids Grow

You've already read a little bit about how your child benefits from learning to code – you know that it can improve your child's academic performance, develop important life skills, and prepare your child for the future.

Read through this next section for a deeper look at the ways coding helps your child as they learn and grow. We cover how coding helps kids write better, earn better math grades, develop creativity, and improve focus and organization. You'll also learn why now is the time for your child to begin programming, how computer science prepares your child for the future, and why girls who create with code excel in STEM fields. There are so many reasons to help your child learn to code!



Improve Writing Abilities

Developing strong writing skills – especially when paired with technical abilities like coding – all but guarantees your child success in school and beyond. But did you know that writing and coding actually go hand in hand? When they learn to code and create digital storytelling projects, children acquire skills that improve their writing, and they have fun in the process.



Coding is a New Medium for Imaginative Storytelling

A writer's tools for telling a story include words and sentences. Coders have access to a more open-ended medium, including pictures, music, and animation in addition to words. The flexibility of programming even allows children to make their story react to audience input.

“I really like creating stories, writing – so I thought if I could code it and make pictures, it’d be even better.” – Featured Tynker Maker Grace

Writing a script in a story-based game forces kids to think through the exact details and consequences of how their characters act. They can’t be vague – they have to hone their ideas, an important skill that takes practice.

“3rd graders created stories with dialogue and lively characters in Tynker. They were so caught up in creating and narrating their stories, they didn’t even realize they were coding, writing, and crafting a compelling storyline at the same time.” – Kathy Bottaro, Digital Learning Program Coordinator, Sioux City Community School District, Iowa

Coding Reduces the “Blank Page” Syndrome

Creating a story-based game requires narrative pacing, compelling storylines, engaging dialogue, and an understanding of the audience. In short, it requires the same skills that some children struggle with when their English teacher hands out a creative writing assignment.

The difference is that staring at a blank sheet of paper often evokes panic, but programming offers multiple starting points.

When they code, kids start with one character, then experiment with dialogue, movement, and interactions. They build from there by adding other actors, scenes, and interactions. The program starts at the child’s point of interest and evolves to a final product through a process of experimentation and iteration. When coding, there is no “blank page,” only discrete problems to be solved.

Coding Teaches the Value of Concision

When kids first start coding, it takes them five lines of code to program a character to move across the screen. As they learn more programming concepts, like loops and conditional statements, they can condense that code to two lines. Children

learn that the goal of coding – or of writing – is to leverage the tools at their disposal in the most powerful way possible to express ideas efficiently and directly.

These are the kids who will write a 650-word college application essay that gets them noticed.

Coding Teaches Planning and Organizing Skills

Programming and writing follow a similar process. When children start a coding project, they plan out the different functions they will need and how these functions will work together to make the project work. Likewise, to write an essay, they must organize their ideas into paragraphs and understand how the paragraphs fit together.

“Coding helps develop the organizational skills required for good writing. I encourage my students to plan their writing assignments by breaking down their topic, selecting the evidence they need, and sequencing their points in a compelling way. Some of the same skills are required when planning a video game. The more rigorous they are as they divide a large problem into components and organize tasks, the more successful they will be.” – Lucinda Ray, Educator and Writing Instructor

The interactive nature of programming makes challenging subjects more accessible. Whether your child loves to write or needs a bit of help, coding is a fun activity that will supplement their Language Arts education.

Strengthen Math Performance

The conventional belief has always been that kids interested in coding should develop strong math skills. However, it turns out the reverse may also be true: coding can help children build math skills and make learning math more engaging and fun.



In the three years that Casita Center, a magnet elementary school in Southern California, has been teaching coding with Tynker, they've seen considerable improvements in their students' math scores, outperforming virtually all California schools with similar demographics. Jenny Chien, the school's STEM specialist, believes this improvement is due to the effectiveness of their CS program. When kids learn to code, they develop key skills like problem solving and practice algorithmic and computational thinking – and when they learn to code with Tynker, they have fun at the same time, so they're more likely to stay engaged with the material. These broad skill sets and ways of breaking down and analyzing problems translate across the curriculum and are particularly helpful when it comes to math.

Whether kids are learning to code at school or at home, you may just see an impact on their overall academic performance! Here are a few ways that coding helps kids learn math:

Coding helps kids visualize abstract concepts

Grasping abstract math concepts can be a challenge to many kids and put them off the subject entirely. Parents, teachers, and technology specialists are using Tynker to help children visualize abstract math concepts.

“One of the most common cross curricular benefits of computer programming is that the kids have an easier time learning math skills,” says Michelle Lagos, a computer science teacher who uses Tynker in elementary classes at the American School in Tegucigalpa, Honduras. “When they have to work on long division, it is easier for them to visualize the numbers now instead of counting with their fingers. They visualize the equation and think of the best way to solve it.” Lagos reports that she has “seen kids in many grades improve their math skills” by using Tynker to learn coding.

Jesse Thorstad, Technology Coordinator for the Fergus Falls Public Schools district in Minnesota, has had a similar experience. “Tynker provides kids with a concrete example of the power of decimal places,” he says. “When studying decimals in math, the students experience a heartwarming ‘Ah-ha!’ moment when they see how moving a decimal block of code can affect the objects on the screen tenfold.”

Kids explore the real-world applications of math concepts

Repairing spaceships or saving puppies with Tynker is a great way for a child to see concrete applications of math strategies. Tynker parent Sri Ramakrishnan points out that kids develop stronger math skills when applying concepts in a real-world context: “The computational thinking involved in computer programming involves logic, organizing and analyzing data, and breaking a problem into smaller and more manageable parts. Much of this is also required when solving math problems!”

Math can be used in creative coding projects

Kids who use Tynker see how math is inherently creative – 10-year-old Jacob Myers, a big math buff who regularly competes in math contests, uses Tynker to make math art with spirals and triangles! Kids can also complete activities like Pattern Maker and Spin Draw to learn how to create art with coding and math.

Coding teaches problem-solving skills

Coding is a real-world way to teach mathematical thinking. When students create or debug a program, they practice solving problems. Math teachers find that Tynker’s beginning lessons are a great way to teach pattern identification as well. Teachers can assign activities like Multiplication Escape or Analog Clock and encourage students to find solutions with math.

Coding makes math more fun

“My kids ask to program with Tynker because they enjoy it,” says Jennifer Apy, the parent of a 15-year-old, an 11-year-old, and an 8-year-old. “Without realizing it, my kids are identifying attributes and grouping variables, applying conditional logic, developing algorithmic functions, and calculating angles within geometric shapes. But most of all, they are patiently articulating hypotheses to solve problems, and boldly applying trial-and-error experimentation, strategies required by any field of study. And this is in addition to some of the coding that requires real math – to correctly calculate wait times, set score counters, calculate points, and time interactions between characters in their games.”

“If kids realize they are using math when programming Tynker games,” Apy says, “it could actually build their confidence with math and show them that mathematical thinking can be cool.”

Math is cool? What could be better than that?

Develop Creativity

Casey stares at his computer screen, carefully calculating his next move. As part of a school science project to create a simulation of the Earth's tides, he has spent the better part of the hour trying to animate a moon orbiting the earth, a series of commands that is proving more complex than he had anticipated. But with every iteration and tweak, the determined sixth grader finds himself inching closer to his vision. Finally, he inputs -10 degrees on the X coordinate, 21 on the Y, and hits enter. He grins in giddy satisfaction as he watches his moon makes a perfect circle around the earth.



Casey's story is one of many that illustrates how the process of learning to code encourages one of the most important skills we can teach our kids: creativity. We're all born with it. As kids, we embrace imaginative play, we ask questions, paint colorful pictures, and build elaborate things with our blocks, but somewhere along the way our capacity for creative thinking diminishes. It's not because we lack the

“creative gene” – we just haven’t reinforced it. Creativity is a skill that can be developed at home and in our schools through the cultivation of three qualities:

1. An experimenter’s mindset
2. Whole brain thinking
3. An innate desire to be a creator (and not just a consumer)

1. Programming Teaches Kids to Experiment

Creative thinking begins with a questioning mindset. It can be taught by encouraging kids to experiment, explore their ideas, question their assumptions, make mistakes and learn from them. Thomas Edison was a master of this type of thinking. He tested thousands of materials and processes before creating the first working light bulb. “I have successfully discovered 1,000 ways to NOT make a light bulb,” he famously said. With programming, kids are exposed to this process of experimentation. They start by learning a handful of commands to do simple tasks, and with each successful result, they slowly gain the confidence try new and more ambitious things, things that force them to question each decision and ask “What if I tried X?” Testing their assumptions in a live environment frequently results in errors and bugs, giving kids the opportunity to find a workable solution. With practice, kids gain a proficiency in their technical and hypothesizing skills, allowing them to move onto solving increasingly complex problems, and eventually building programs completely on their own.

2. Programming Strengthens Whole Brain Thinking

Each side of the brain is said to control different parts of thinking and information processing. The left hemisphere is typically associated with logical, technical, and analytical thinking, whereas the right hemisphere is associated with imagination, artistic, intuitive thinking. We tend to think of creativity as a right-brain function, but the most creative thinkers and problem solvers can effectively engage both hemispheres. This idea of marrying “art with science” is what Steve Jobs built Apple on, and it’s this kind of “whole brain” thinking that teachers have been embracing in the classroom by promoting active, project based learning, using everything from

3D printers to sewing machines to encourage kids to create, design and build things. Learning programming with a platform like Tynker is particularly powerful because it requires kids to use their technical skills (to build the program) in combination with their artistic and storytelling skills (to design a program that is visually compelling and fun).

3. Programming Gives Kids the Confidence to Create

Like learning a sport or a musical instrument, cultivating creativity requires hard work and practice. For kids, if the work is confusing, monotonous, or the end goal unappealing, the desire to practice weakens. They need to be in an environment that builds confidence and instills in them a genuine desire to create. Kids pick up on technology with shocking ease, so giving them a basic knowledge of programming on a coding platform that is fun and easy to use is one of the best ways they can spend time in practice and actually enjoy the process. Learning programming on the right platform, one that is structured, engaging and well paced, puts kids on the path to fluency in the language and logic of programming, and ultimately gives them a springboard to create – to not just play the games that they love, but to create the games they love to play.

Nurture Creativity Through Programming

Learning to code is very much like learning a new language – it gives kids a fluency not just in technology, but also in the language of creativity. Maria Klawe, mathematician, computer scientist and president of Harvey Mudd believes that “coding is today’s language of creativity. All our children deserve a chance to become creators instead of consumers of computer science.” It doesn’t mean they’ll all grow up to be computer programmers. Programming is part of the development of a valuable technical and creative skill set that will grow with them into adulthood, enabling them to thrive in our ever growing digital world. It’s creativity that lays the foundation for innovation, ingenuity and leadership because it represents the ability to connect existing ideas with new solutions, approaches and concepts. And we owe it to our curious and imaginative kids to give them the tools to be the creative thinkers and problem solvers of the next generation.

Build Focus and Organization

Soft skills are a popular notion in the business world, and they encompass qualities like leadership, communication, and perseverance. Although they may be difficult to measure, soft skills are vastly important for children to learn. As founder and CEO Krishna Vedati told the BBC, “Our goal is not to create programmers, but to offer coding as a life skill.”

Focus and organization are two soft skills that are the key to, well, everything! In a world where it’s increasingly harder to focus, everyone could use a boost – especially kids. Coding is a great example of an activity that requires focus and organization, but more than that, it’s a fantastic way to develop those skills.



Decreasing Attention Spans

It's no secret that the distractions we all face impact our ability to focus, and kids are no exception to that. Between shows on TV, games on phones, and other distractions, there's a lot of opportunities for kids to lose focus. The instant gratification found in these activities can make it difficult to focus, which consequently makes it difficult to be organized.

*"You speak to young kids nowadays and because of all this stimulus everywhere – billboards and movies and everything going on – the attention, the focus, just isn't there."
– Alain, Featured Maker Elisha's father*

Why Organization and Focus are Key to Coding

Organization goes hand in hand with focus, especially when it comes to writing complicated code. In order to think through code logically, it needs to be organized in an intuitive way, and vice versa. The more complicated the code, the more a programmer has to organize, incorporating elements like data structures to streamline the code.

*"To me, it's a way of organizing ideas, structural thinking, and logical thinking. It's challenging because kids are forced to think about how to solve a problem." – Santiago,
Featured Maker Nico's father*

Video game designer James Hague reiterated this idea on his blog, writing, "To a great extent the act of coding is one of organization." Focus and organization are key to writing good code, but can coding help develop these skills? It's possible!

How Learning to Code Develops Focus and Organization

Programming can help establish the focus and organization it relies on, especially when working with a platform like Tynker. The goal-oriented nature of an exciting project promotes focus and organization in kids. Kids are motivated to finish a

project because it's exciting, so they push themselves to focus, organize, and make it happen!

"I like to code because it is sometimes complex. It's like when you're working up a hill that has jewels at the top. It's hard to get up it, but when you get to the top you're really proud and you think it's awesome." – Featured Maker Anthony

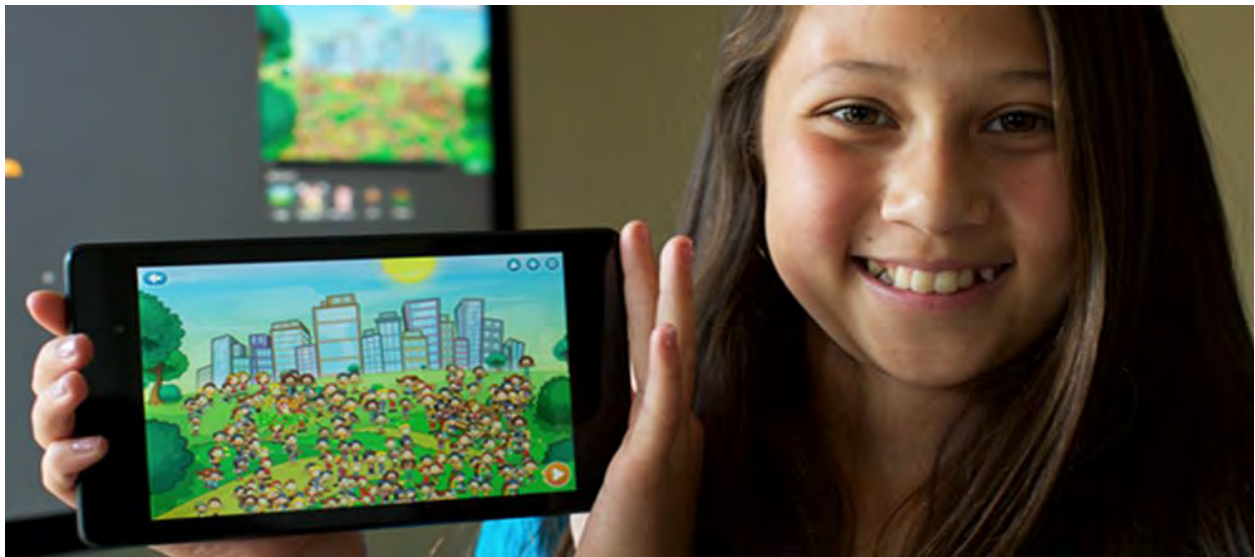
The logical nature of programming – identifying a problem, thinking through steps, and then implementing a solution – encourages organization of thought and sustained focus. A paper written by Roy D. Pea and D. Midian Kurland argues that, "In explicitly teaching the computer to do something, it is contended that you learn more about your own thinking." The focus and organization needed to problem solve from start to finish can be difficult to maintain, but it isn't impossible to develop.

As with any other activity, coding improves with practice – as do the skills accompanying it. Inside and outside the classroom, applied to coding or homework, the focus and organization learned through programming will help kids in any endeavor!



Start Early: Girls Who “Make” Choose STEM

The growing shortage of women in computer science and engineering is a hot topic these days. Fortunately, the future looks bright for the next generation of girls, according to a new study by Intel. Their research finds that “girls that make, design and create things using technology may develop a stronger interest and greater skills in computer science and engineering.”



The popularity of “DIY” and the Maker Movement are creating a generation of girls and women who show just as much interest as boys in making, inventing, and solving problems through technology. According to Intel, 1 in 4 teens and tweens have already made things with technology and 7 in 10 are interested in learning to make things with electronics.

The key to getting more girls interested in STEM is to expose them to making. Whether it be through designing, building, coding, or the arts, making provides them with a pathway to learn and explore new ideas and technologies.

Girls Become Makers with Tynker

Tynker is a popular coding platform for girls because of its fun and easy approach, as well as the variety of projects and tools available to inspire their broad and creative interests. Girls have created millions of Tynker projects, such as greeting cards, music videos, comic cartoons, digital stories, quiz games, drawing tools, music makers, classic arcade games, platformers, geometric designs, Minecraft mods and skins, and science fair projects.

Let's hear from a few who have enjoyed "making" with Tynker.



Carly – "I like Tynker because you can build whatever you want. You can make a video game or a dancing video. I always try to build new things so I can show my friends."

Emma – "It's really fun to see what you can do. With Tynker you can make a game and if there's something you don't like about it you can change it and make it the way you like."





Kaela – “I like Tynker because you get to use your imagination to make something awesome. It’s like drawing and painting and writing all mixed into one.”

Quinn – “When I do Tynker it’s just fun. I don’t get stressed out, but I know I’m learning something.”



Haley – “Tynker is a lot faster and simpler. I once built a game in another language and it took me about 3 days. In Tynker it takes about 30 minutes to make a good game.”

Grace – “Unlike other coding languages, with Tynker you don’t have to memorize lines of difficult code. You just put the blocks together.”



Kami – “Tynker is easy and simple. It’s like a sentence, you just fill in the blocks you need to make it do what you want and it’s easy to understand.”

On a Mission to Inspire More Makers

Girls and women use technology more than ever. It’s time for more of them to create it, too.

It’s easy to become a maker with code. How do we get more kids (and girls) started? By making it fun (and teaching them that there’s a rich history of women in STEM careers)! Let’s inspire more young makers to create with code and get more girls and boys to choose STEM.

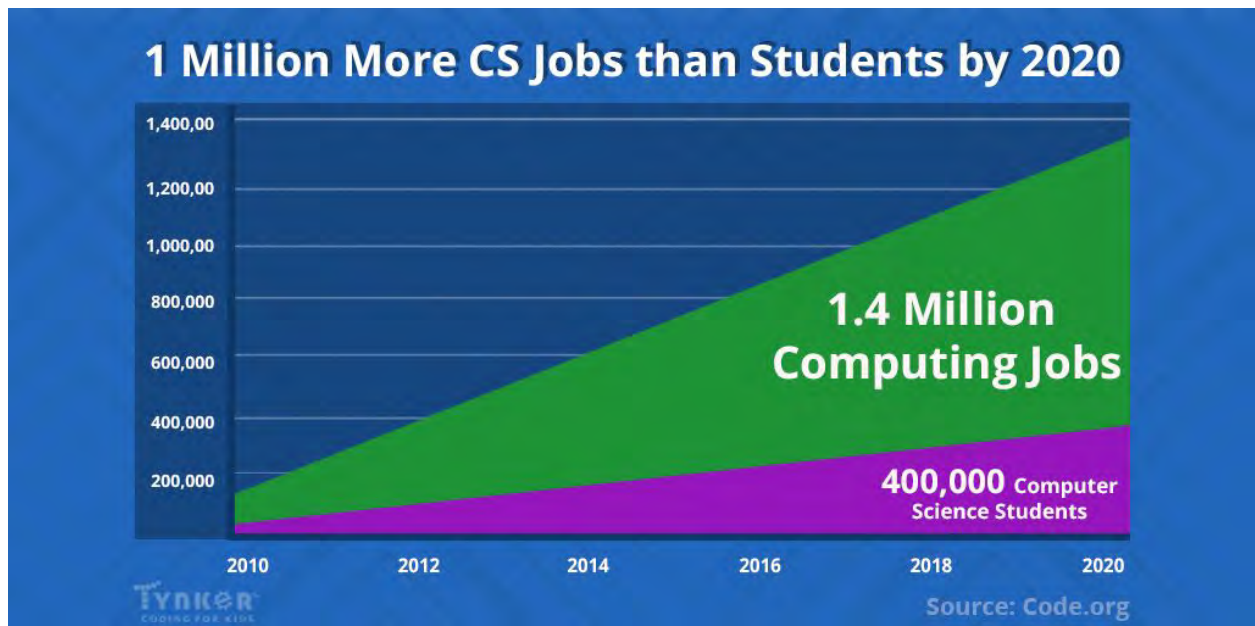
Invest in Your Child's Future

In today's economy, we are in urgent need of people with coding skills to meet the demands of a burgeoning tech industry that isn't going to be shrinking anytime soon.

That's why introducing children to coding is a crucial investment in their future. Coding is a skill of the 21st century, and with the rapid technological advancement of the modern age, it has quickly become as necessary a skill as reading and arithmetic that is important for a wide range of professions.

1. Coding skills are in high demand.

The tech industry is in constant need of workers. And it's not just coders or computer science majors – they need graphic designers, software developers, computer engineers, linguists, mathematicians. The jobs are not only plentiful, but they're also lucrative.



For example, graphic designers, whose skills can range from artistic ability to web design, make an average of \$51,640 a year. While linguists can come from a wide range of concentrations, being a computational linguist can earn you about \$91,307 a year. And computer hardware engineers can easily make an average annual salary of \$111,730.

Working in any of these fields, and being able to code using JavaScript, Python, or any common computer programming language, is a great step toward a secure and stable job.

2. Prepare your kids for high school and beyond.

Like a foreign language, coding skills are best learned early. Once kids are fluent in the type of thinking required to break down and solve a coding problem, the transition to any coding language is relatively straightforward. Tynker even transitions kids from block-based coding to JavaScript and Python with fun, gamified courses. The coding skills that Tynker teaches lay the foundation that students need to succeed in high school and beyond.

One of the biggest obstacles to succeeding in high school or college-level computer science classes is lack of confidence in tackling the difficult, unfamiliar material. Earlier exposure is the best solution – being introduced to coding at an early age makes it easier to learn the harder, more technical aspects of computer science in high school and college. In fact, a study by Google and Gallup shows that early exposure is one of the most important ways we can shrink the gender gap in STEM, as it boosts confidence in kids, especially young girls, while they're still interested in technology.

3. Coding is a lifelong skill.

Even if your child wants to do something outside of computer science when they grow up, their coding skills will prove useful across fields – coding teaches problem solving, organization, math, storytelling, designing, and more. The beauty of coding

is that it comes in handy for different aspects of life and allows kids to express themselves creatively.



To get your child interested, show them what coding allows them to make. Coding allows them to do anything from writing stories and building video games to making Minecraft mods and designing animations. And, of course, it's fun! Tynker provides the easiest and most enjoyable path to learning how to code – kids have so much fun that they hardly realize they're learning at all.

Most importantly, the ability to code transforms kids from passive consumers into innovative creators, with eyes that see every piece of technology as more than just a toy but as a problem to solve and an opportunity to create.

Add Coding to Your School's Curriculum

Coding at home is a great start, but most of us want to see computer science taught to our children at school, too! Unfortunately, there's a big disparity between what we would like as parents and the reality of the our education system today.

According to a Gallup poll, while 90% of parents want their child to study computer science, only 40% of American schools teach computer programming.

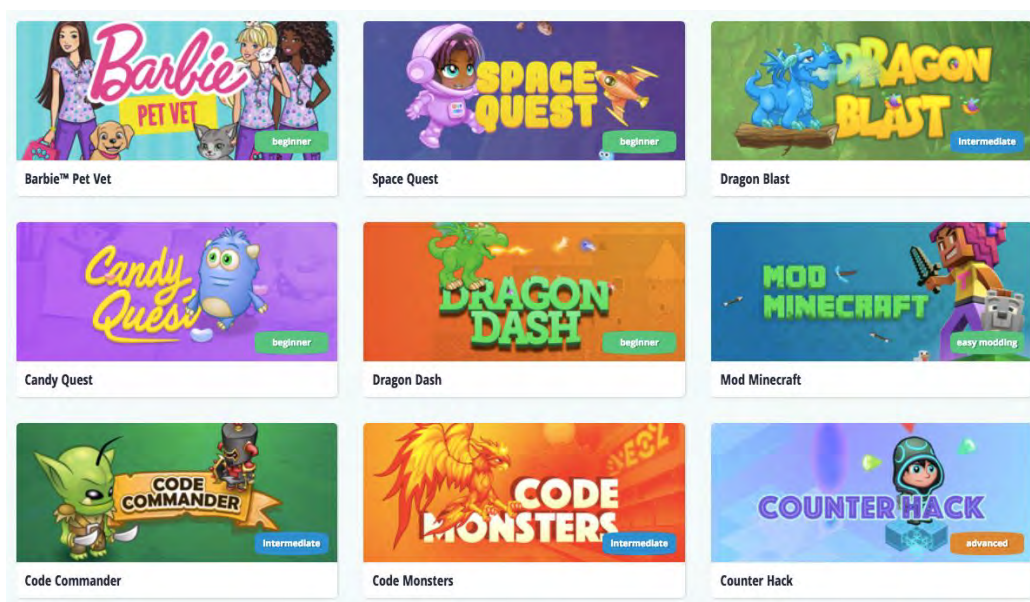
The movement to bring coding to schools has can be led by parents like you! In this section, we introduce some computer science programs that have been successful in schools. These programs include Hour of Code, a global movement to encourage students to learn programming, as well as makerspaces equipped with robots, drones, and more. Read up to see how you can empower your child's school to add coding to the curriculum!



Try an Hour of Code

If your school is new to coding, there are plenty of free resources to get started – even for teachers without a computer science background! One quick way to get started is with Tynker’s Hour of Code tutorials. Hour of Code is a global movement that reaches millions of students in more than 180 countries. An initiative designed to get more kids engaged in computer science, Hour of Code traditionally takes place during Computer Science Education Week in December – but your child’s school can run their own Hour of Code at any time!

You can help organize an Hour of Code for your child’s school or your community with the help of our coding activities. We created our Hour of Code tutorials with a view to inspire and motivate beginning programmers – children who know nothing about coding and are perhaps wary of it. Kids find it easy to drag and drop our visual code blocks to create simple games, stories, animations, and more.



Running an Hour of Code is a great way to demystify the concept of programming, provide early successes, and share the positive impact of a computer science education. And when your child’s class is ready to move beyond an Hour of Code, be sure to check out our full coding curriculum for schools!

Build a Makerspace for Hands-On Learning

Kids enjoy (and benefit from!) hands-on learning. Schools can set up a makerspace – a flexible learning space separate from the main teaching area – where kids write programs to control drones, robots, and more. Tynker integrates with a variety of Parrot minidrones, Sphero robots, and the LEGO WeDo 2.0 to make it easy to equip a makerspace for your child’s classroom.



Setting up a makerspace gives tactile learners an opportunity to see their code play out in front of their eyes!

Hear from a Tynker Educator: Why Our CS Programs Are Working

By STEM Coordinator Jenny Chien

Did you know that just one in ten schools nationwide currently are teaching computer science (CS) classes? Private companies, the federal government, and states and districts have all invested to make these classes more available. As CS classes and programs expand, we can learn some lessons from those that are already in place.



As a STEM specialist at a dual magnet elementary school, Casita Center, in Vista Unified School District, I'm here to tell you what is working in the schools and districts where CS classes are already up and running. A big part of my job is to work with other teachers and coordinators to design and implement curriculum to get elementary school students to learn computer science (CS) – a robust part of the STEM universe.

As background, the Vista Unified School District has 29 schools and serves about 23,000 students in San Diego County, California. Like many districts, Vista Unified has Title I schools, high proportions of second language students, and has made STEM education a priority.

The CS program we designed has done exactly what we hoped it would. In a pilot program at Casita Center, the computer science implementation is a factor that has boosted math test scores and enhanced other key skills including problem solving and critical thinking. Three years in, we were one of the top achieving schools in the state in mathematics compared to other schools with similar demographics – and California is pretty big, pretty diverse state.

So, what do we know? What are we doing right? And how did we get here?



The results have been so clear that our program is now being used as a elementary model for all of Vista Unified School District. We've started the process of training teachers on how our model works and how to get the most out of all the program we utilize to meet the needs of students in Computer Science.

Our first 18 teachers from 15 schools met in August and, as hourly teachers with full plates, they were hesitant about the ability to learn and deploy the new methods in ways that would have real benefits to their students. But once they saw firsthand that they didn't need to be computer experts to teach computer skills, they started to see the value in the game-based and project-based learning approach. Instead of being the wealth of knowledge with Computer Science, they are learning alongside the students as lead learners.

We're now in the process of a strategic roll out to other schools and getting our new teachers up to speed on issues such as publishing and how to start coding projects in an open inquiry model.

Going on the journey of finding and using classroom technology to teach high-tech, job-prep skills – and actually seeing it work – has been very rewarding. It's also taught me a thing or two about how other teachers or district leaders can succeed with similar goals.

We also know enough to offer some tips to districts and educators seeking to start or boost their own CS programs.

Among the most important lessons I learned is the value of real time vendor support. Due diligence is essential. Don't just take a "we offer support" as an answer. Try it – challenge it.

In CS specifically, there is a need to converge and diverge with professional development. Good program design requires troubleshooting both in and out of the classroom. Educators need to feel and see the support. Tynker is quick to help troubleshoot technical issues as well as provide ideas with how to meet the needs of our students in coding.

I'm still very optimistic for our CS program as we work to scale it. So far, the results have been inspiring and have underscored the value of infusing tech and tech-based learning in a K-12 curriculum. So explore plenty of options and find a partner that's right for the needs of your school(s) and your students. It makes a tremendous difference.