ONE PAGE TO START · EVERYTHING ELSE CAN WAIT

Start Here

Your first week of coding, in four short sessions. No experience needed, nothing to install, nothing to buy.

Print this. Do week one. Then open the site and tick the box.

The First Fifteen Minutes

Do this once, together, before session one.

Open a browser and go to **scratch.mit.edu**. Click **Join Scratch** at the top. It asks for a username, a password, and a parent's email address — use yours, because the confirmation email comes to you. Pick a username that is *not* his real full name; this is a good first conversation about what you put on the internet and why.

Once you are signed in, go to **Account settings** and set comments to require your approval. Scratch is a heavily moderated place and about as safe as the internet gets for children, but it is still a comment section, and two minutes now saves a conversation later.

Then click **Create**. A cat appears on a white square. That is it — that is the whole setup. Nothing to download, nothing to pay for.

> **YOUR JOB DURING THE SESSIONS**
>
> Sit near him. Do not touch the keyboard. When he gets stuck — and he will, in about four minutes — ask him what he expected to happen and what actually happened. Those two questions solve most of it. If you fix it for him, you have taught him that being stuck means fetching an adult.

Week One: Sequence

The whole idea of the week, in one sentence: **a program is a list of instructions, carried out in order, exactly as written.** The computer is not clever. It does not work out what you meant. This sounds obvious and it is the thing beginners take longest to truly believe.

SESSION 1 · ABOUT 30 MINUTES

Make the cat do something

1. Find the blue **Motion** blocks on the left. Drag `move 10 steps` into the big empty area in the middle.
2. Click the block. The cat moves. Click it again. It moves again.
3. Change the 10 to 100. Click it again. Notice what changed.
4. Find the purple **Looks** blocks. Drag `say Hello! for 2 seconds` underneath so it snaps on.
5. Click the stack. Both things happen, in order, top to bottom.
6. Now swap the two blocks around. **Before clicking**, say out loud what you think will happen. Then click.

DONE WHEN he can make the cat move and speak, and he predicted correctly what swapping the blocks would do.

SESSION 2 · ABOUT 30 MINUTES

Sound, colour, and the green flag

1. From **Sound**, add `start sound Meow`. From **Looks**, add `change color effect by 25`.
2. Build a stack of five or six blocks that do something silly. Silly is the correct goal here.
3. Now go to **Events** and drag `when green flag clicked` to the very top of the stack.
4. Click the green flag above the stage. The whole thing runs on its own.
5. Ask him: why does that block have to go on top and not in the middle?

DONE WHEN pressing the green flag runs his program without him clicking on the blocks.

SESSION 3 · ABOUT 30 MINUTES

Choose a character and a place

1. Bottom right, click **Choose a Sprite**. Pick anything that is not the cat.
2. Next to it, click **Choose a Backdrop**. Pick a place.
3. Give the new sprite its own stack of blocks. Notice that each sprite has its own separate script.
4. Delete the cat if he wants. Things can be removed. Nothing here is precious.
5. Rename the project at the top of the page. Give it a real title.

DONE WHEN there is a named project with a character he chose, in a place he chose.

SESSION 4 · ABOUT 40 MINUTES

Build the Name Badge

The first real project. Roughly ten blocks. He builds it; you watch.

1. The sprite starts at the left edge of the stage.
2. When the green flag is clicked, it glides to the middle.
3. It says his name for two seconds.
4. It plays a sound.
5. It grows bigger.

DONE WHEN he presses the green flag and the whole thing plays through by itself — then he shows it to somebody who is not you. A sibling, a grandparent on a video call, anyone.

THEN, AND THIS BIT MATTERS

Open his site, find **Week 1**, and let him tick the box himself. Press **Save my progress** and bookmark the page. That is the first square filled in on a chart with fifty-two of them, and watching it fill is a surprising amount of the motivation over the next year.

What Happens Next

So you know where this is going, and can say so when he asks.

Week two is loops, and it is where most children get properly hooked, because eight blocks produce something that genuinely looks beautiful. Weeks three to nine build up events, conditions, variables, coordinates, clones and lists — which is, quietly, the entire vocabulary of programming. By week ten he designs and builds his own game from nothing and puts it online.

Then Python, then websites, then the tools professionals actually use. But none of that is your problem this week. This week is a cat, a green flag, and a name badge.

If it goes wrong

He finds it babyish. Fair — he is twelve. Tell him the truth: in Scratch an idea takes four minutes to test, in a typed language the same idea takes forty because half the time goes on spelling. Ten weeks here means testing a hundred ideas about how programs work. It is the fastest road to Python, not a detour around it. He is allowed to find it easy; easy means go faster, not skip.

He gets frustrated and wants to stop. Stop. Genuinely stop, mid-session, and go and do something else. Pushing through is how a child learns that coding feels bad. Come back the next day and start with something he already knows how to do, so the last feeling in his body is competence rather than defeat.

Nothing happens for two weeks. Normal. Do not restart and do not have a conversation about commitment. Sit down together, rebuild something small he already made, get one thing working on screen inside half an hour, and carry on from where you were.

EVERY FRIDAY — ASK THESE THREE

1. **What did you build?**
2. **What problem did you solve?**
3. **Can you show me?**

Then praise the **process**, not the brain. “You stayed with that bug for twenty minutes and you got it” — never “you’re so smart.”
And never take the keyboard.

ONE HONEST EXPECTATION

He will not love every session. Some weeks he will be brilliant and some weeks he will not want to open the laptop. That is what learning anything looks like from the inside, and it is not a sign that this was the wrong idea. The only thing that actually matters over twelve months is that you keep showing up, keep asking to see what he made, and keep being visibly interested in it. The curriculum is a support. You are the reason it works.