SHOULD WE TEACH PROGRAMMING OR TEACH THE *IDEA* OF PROGRAMMING? - AN INTERVIEW WITH DR. DAMIEN KEE

STEM educator Dr. Damien Kee is a passionate educational technologies advocate who specialises in bringing technology concepts to teachers and educators around the world. With a focus on robotics, programming, and electronics, Damien works to educate teachers on the benefits and relative ease with which technology can be embedded into their daily classroom activities.

We recently caught up with Damien to talk about programming and coding in education, and here's what he had to say. (For a more in-depth interview, listen to our podcast with Damien here.)

Arduino Education: Hi Damien, thanks for chatting with us today! Firstly, we'd love to find out a little more about you - so let's get straight to the point by asking you to tell us something interesting about yourself that not many people know.
Damien Kee: [*Laughing*] I'm well known for my robotics and all the educational robotics systems I've dabbled in, but something that people don't really know about me is that I play violin. In fact, I've played violin for the last 30 years, on and off. None of it is professional but I've played in a community orchestra and a local theater group.

AE: That's very cool! Now down to business; what are the common myths about your profession you'd like to set straight?
DK: I think the common myth is that robotics is difficult. And I won't say that it's easy. But with just a little bit of thinking and a little bit of planning, it's amazing what we can create. To

outside people, it may look complicated, but at the end of the day, it's a series of blocks we put together, and a bit of code we write. So that's something I'd like people to wrap their head around - it's really not as difficult as you think it might be.

AE: Moving on to coding and programming in education; how do you think it has evolved over the last few years?
DK: One of the big things where it's evolved is in teacher support. Twenty years ago, when people were learning coding it was from teachers who were originally in the IT field and had moved across to become educators. So they had very specialized knowledge, not only on how to program, but also on how to teach programming. The biggest shift we've seen in the last 10 years is the increase in resources, both in print resources and web resources, but also in professional development, to teach teachers how to teach programming.

AE: That’s very true, and it’s something we try to provide to teachers as well. What would you say to teachers so they know the impact programming could have on their curriculum and lessons?
DK: When we teach programming, it often comes down to two things: do you teach programming? Or do you teach the *ideas* of programming? I'm very much in that second camp, that it's about the ideas of programming. I firmly believe that learning how to program teaches these ideas of computational thinking, where it's about solving problems. If children learn that, they can solve different problems, not just software problems, but any problem that they come across in life. I think one of the things we're getting much better at is showing kids that it's not about becoming a software engineer, it's about a way of thinking, and coding is a great way of teaching that way of thinking.

AE: How do teachers bring programming to their curriculum, if they're not programming teachers themselves?
DK: It’s about looking and seeing what you're doing in class. So perhaps you just need to start integrating programming with other subjects you're doing, and you have to draw on what kids are passionate about. For example, if your kids are interested in horses, you could design an automated horse barn which has doors that open and close at set times, and a thermostat to keep the temperature nice and comfortable. This weaves in some electronics and programming to help kids realize the things that they want to create - it’s very much about wrapping a context around it. Trying to teach kids how to program with no end goal is hard, as programming can be quite dry and boring if there's no reason for it. So as teachers, we're always trying to find ways to wrap different contexts around things to give them a little more relevance.

AE: Do you have any other tips for how to help break down the barriers and get started with programming?
DK: A lot of times, I see teachers who are trying to kind of take on a little bit too much, and projects tend to fall over. So instead, it’s important to just keep it simple. And don't expect to know everything! What we want is for teachers to be able to know where to find the information. There’s an amazing range of expertise out in the world. And as a teacher, you

need to be comfortable saying to the kids, look, I don't know the answer to your question. But I'm going to find out from someone who *does* know the answer.

AE: Great advice! Final question - what's one thing that you wish you'd known when you began your career?
DK: I really wish I'd known that electronics sets were out there and that it's not hard to get your hands on them. And it's not hard to just tinker. And it's okay to tinker, and to make mistakes. It's okay to make crazy projects that don't benefit anything!

INSPIRED? HERE'S WHAT YOU NEED TO GET STARTED WITH ARDUINO EDUCATION

The *Arduino Student Kit* can be used for both home learning and classroom teaching. It will help you get started quickly and easily with robotics, electronics, and coding.

***Find your country's distributor*, or *buy the kit online*.**