



6 SITES TO SEE

LOOKING FOR DIGITAL RESOURCES TO HELP YOU DELIVER THE NEW COMPUTER SCIENCE CURRICULUM AT KS3? DAVID MORGAN IS ON HAND TO HELP...

1. Light Bot (light-bot.com)

My go-to for the basics of programming, Light-bot is a nice little flash game that, through simple tasks, teaches the basics of sequencing, loops and procedures; and all without the need to write a single line of code.

It's not scary. All we get is the little light-bot, the maze he's trapped in, and an occasional hint or tip to aid us on our merry way. The drag-and-drop blocks are more simplistic than Scratch and all we are doing is giving commands like 'forward', 'backward' and 'light', but it's more than enough to teach the concepts in a very accessible manner.

The best part is that it's also a bit of fun, and there's nothing like competitive students trying to out-think each other to produce the most efficient solution. Also available as an app for your tablet.

2. Computer Science Unplugged (csunplugged.org)

It's important to remember that Computer Science isn't all about programming. Computer Science Unplugged is a great resource that provides a fantastic set of resources and activities to teach the wider concepts of computing without needing a computer.

These activities are exciting, varied and can even be done outside. Not only does the site help with ideas for teaching concepts that aren't as accessible as others, but it provides a fantastic opportunity to deliver these skills in a unique and fun way.

It has everything from 'teaching binary numbers' to the ever-exciting 'sorting algorithms' and all presented, for free, in a straightforward manner. I've had great fun getting out and about, and 'unplugged' makes a big difference to the learners' ability to remember the concepts.

3. Code Avengers (codeavengers.com)

This is the nicest online course to teach students the fundamentals of programming. The key to it is giving students enough exercises to practice the skills they have learned. If you're planning on marking all of those by hand then good luck; I'd much rather the real time feedback system and pre-prepared exercises of Code Avengers do that for me.

Code Avengers provides students with a feast



ABOUT THE AUTHOR



David Morgan (@lessonhacker) is a moderately internet-famous teacher who excels at pedagogy and everything digital; he is also the author of *Stop Getting I.T. Wrong!* (lessonhacks.com, £9.99)

of exercises that self-mark, but the most powerful thing here is the analytics provided by the website, where you can see the progress of your students: from the amount of attempts they took per exercise, to the accuracy of their answer and the length of time they have used in answering each question. What this gives us is an embarrassment of rich data that we can use to tailor our teaching to the students' strengths and weaknesses.

4. Play my Code (playmycode.com)

If you can get over the inevitable jokes about 'forking' then Play my Code is a brilliant place to build students' love of programming from the very start. What we're asking here is to take someone's existing code and edit it to make changes to a game. These changes can then be saved as 'forks' of the original just in the same way that open-source programming would.

What's great about this is that it has a massive amount of differentiation built in – students may choose to find aspects of the code related to scoring and make themselves look like gaming gods by getting points galore, alternatively some of your G&T students may extend the game: building new levels, modes or fixing bugs.

All of this is done in your browser and there's a huge selection of games already provided.

5. Screenr (screenr.com)

Screenr has completely changed the way that I demonstrate or teach anything that directly uses the computer.

It's just a website with a browser plugin allows you to log in, click record, and quickly create a screen recording to demonstrate an

example of the skill you're discussing. Once that's done it lets you easily share and distribute the recording all without having to bother with YouTube and filtering.

This tool does what a lot of others do, but turns the process of screencasting into a couple of clicks and a few minutes' work, so lends itself to short, sharp explanations and very little faffing about. Brilliant for small programming or software skills, and much better than spending hours creating a worksheet for what is just a few clicks and taps.

6. code.org (code.org)

If there's one resource you should go and plunder then it's definitely code.org. Aside from its rich collection of links to all the websites and resources trying to teach students to program, it has an online course that uses computing celebrities, popular games and genuine excitement to deliver a course in the basics of Computer Science.

We're talking learning programming through making your own version of Flappy Bird, sequencing with the Angry Birds and all dotted amongst their own version of the unplugged activities. This could quite easily form an entire term or two of work for KS3 giving students a broad and entertaining overview of this wonderful subject of ours.

Students can even sign up and the system will track and report their progress to you. What more do you need?

HANDS-ON LEARNING **ollie**

The makers of Sphero have taken connected play to the next level with Ollie, combining the technology that Sphero 2.0 is known for with innovative controls and integrated gameplay for a whole new experience.

Engineered for high performance driving and tricks, Ollie drives up to 14 MPH, and provides complete control and agility to execute tricks including spin, drift, jump and flip moves. In addition to this, through apps, Ollie can provide automatic feedback based on the difficulty of the manoeuvres and tricks performed. Programming apps are also available.

