



Questions and Answers from our iPrimary and iLowerSecondary Computing webinar

Hosted on 24/6/20 by Danny Nichols

What is the best software to use for programming for KS1 and KS2?

The best software to use for KS1 would be 'Scratch Jnr' or 'Daisy the Dinosaur'. It would also be best to start with some of the games on 'Hour of Code', there are familiar characters available, such as *Frozen* and *Star Wars*.

Links; [Scratch Jnr](#), [Daisy the Dinosaur](#), [Hour of Code](#) ([Frozen](#), [Star Wars](#))

The best software to use for KS2 would be Scratch, use [this](#) as a starting point.

Links; [Scratch](#)

What is Scratch?

It's an environment for writing and running programs. You don't need to know the actual programming language; you simply snap the blocks together to produce code.

You can find out more about scratch [here](#), plus it is totally free!

Does Scratch run on Android?

Scratch 3.0 is the latest version and it will run in a browser, so it will run fine on tablets. On an iPad it will run in Safari, on Android tablets it is best to use Chrome.

Are there any alternatives to using Scratch? As we only have tablets available in class.

Scratch 3.0 is the latest version and it will run in a browser, so it will run fine on tablets. On an iPad it will run in Safari, on Android tablets it is best to use Chrome.

As alternatives, you could try Espresso Coding or Purple Mash 2Code, both of which will run via a browser so will run on anything. These are not free and do require a subscription.

There is also [Hopscotch](#), which is an app for iPads, this is initially free but there are in-app purchases.

Is there a way that I can link Scratch programming with Visual Basic for KS3?

It is a good way to get students thinking about the process of coding via Scratch and understanding this and then moving onto VB.

For lower grades, how do we teach them clearly (they really understand the instruction) by online how to draw 2D figures?

Start by not using the computer, which may be difficult in distance learning, but asking them to be on camera when they start this exercise may help.

Can they work out how to draw a triangle, square etc. using squared paper, pencil, ruler etc. Then discuss with them the algorithm (instruction) they need to use and you as a teacher will be able to determine how much detail they will need, keep it simple. From that you could then ask them to do the same thing in Scratch or similar on a computer.

Is a textbook necessary for this subject? And are the resources on ActiveLearn supportive enough for teachers?

A textbook whilst is not entirely necessary for teaching computing, a resource can give plenty of ideas and activities to teachers and students and can provide structure to a class especially for those teachers new to teaching and new to teaching computing.

ActiveLearn [Primary](#) and Active Learn [Secondary](#) provides all the supporting curriculum documentation that teachers will need to effectively teach computing. Resources include, lesson plans, learning objectives, schemes of work, overarching curricula documentation and assessments. Giving both flexibility and structure to teachers teaching computing at primary and lower secondary.

If you would like more information on ActiveLearn and how to access this documentation, please do contact Stacey Roberts, Stacey.roberts@pearson.com for more information.

Is there a way we can integrate computing with the contents of other subjects?

Yes, lots. In fact, it's best to integrate computing as much as you can with other subject to make it feel relevant to your students. For example, look at using spreadsheets in Maths, creating videos in Science, Podcasts in History, Coding in Maths (creating function machines), the list is endless. The unit plans on Active Learn also contextualise computing and its uses and effects on everyday life.

Are we going to introduce all the terminologies for this age group?

iPrimary and iLowerSecondary Computing covers ages 5-14, and all key terminology/vocabulary that students need to know are included in the lesson plans on ActiveLearn. We would not have been able go through all terminology in the webinar, so key terminology/vocabulary was picked out to give you an idea of what is included.

Will social distancing in schools have a big impact on teaching computing, specifically the hands-on offline activities?

You will be able to use collaboration tools such as Google for Edu to collaborate on online projects. Scratch programs can be saved and shared online so students can share what they're doing and work together on code.

In the classroom you may have issues with sharing laptops, students will need to work more independently, but they can still collaborate online. Computers and tablets (and roamers etc) will need to be cleaned before reuse by another student.

How often do the students need to use a computer/laptop during the computing class?

Students do not need to be on a computer/laptop during every computer lesson, they need to learn theory and the impacts of computing as well as the physical computing lessons. The lesson plans included on ActiveLearn state when students should be at a computer. For example, in Year 8 during the spring term, students will be required to be at a computer for 5 out of the 10 lessons. p

Which version of Office did you use for your PowerPoint presentation, which translated audio into subtitles?

The latest update to Office 365 now allows for subtitles during a presentation, please see this [link](#) for more information.

Where can we get scratch/python training?

There are many resources to help you get to grips with how to use Scratch and Python, [Resources for Educators - Scratch](#) and there is a wealth of information on how to use Python by simply googling 'learn python'.