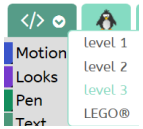
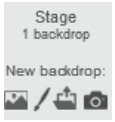
Year 3 Programming 1 Making my Program in Scratch (Core) inc RAG and PRIMM

Timing 6 sessions of approximately 45 minutes **Optional session 7 to reinforce or assess learning.**	**Children will** • Explore how to change sprites and backgrounds • Explore Scratch and discover what they can make their sprite do – emphasis on discovering what individual blocks do • Create a sequence to make a sprite move and turn • Think through an algorithm and 'RAG' to self-assess • Implement the algorithm as a dance program that includes sound • Plan and RAG an algorithm for a knock, knock joke • Create an animated story or an animated celebration card
e-safety links • I can protect my personal information when I do different things online. • I can post positive comments online.	**Objectives** **Programming** • I can break an open-ended problem up into smaller parts. • I can put programming commands into a sequence to achieve a specific outcome. • I can detect a problem in an algorithm which could result in unsuccessful programming. • I keep testing my program and can recognise when I need to debug it. • I can use repeat commands. • I can describe the algorithm I will need for a simple task.
Links to other learning **English**: writing instructions for your friend to follow. Create a short story. **PE/Dance:** create a simple dance sequence and teach it to your friend. Use this sequence to create your program in Scratch	
Resources Scratch Desktop version 3.0 Scratch story project Scratch 'celebration cards'	**Preparation** • Have Scratch Desktop downloaded to PCs / laptops from https://scratch.mit.edu/download . • You may also wish to have the I Am a Computer Programmer document to share with learners, to talk through the thinking and doing process of programming and a copy of algorithm rap. • Have Scratch dance support sheet available if you would like to use it in session 3. • Have a blank knock, knock algorithm sheet to record algorithm in session 4 and ask them to find a knock, knock joke as homework. • Ask technician to set a shot cut to Scratch story project used in session 5 or download project for children to use on your own network. • If you do session 7, set a link or download project for examples of celebration cards.

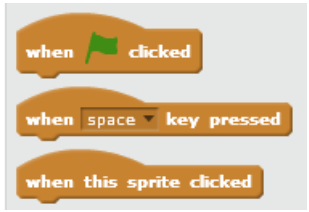
Thanks to Janet Ross (West Monkton Primary) for contributing to original WCP materials.

Year 3 Programming 1 Making my Program in Scratch (Core) inc RAG and PRIMM

	Expectations	Activity	Success Criteria

1	**Programming**
	
	I can put programming commands into a sequence to achieve a specific outcome.
		Make something happen
		Open Scratch. Children to work individually if possible.
		 
		• Ask children what experience they have had of Scratch. Tell the children that the cat is a sprite and there are lots of other sprites they can use. Focus the children on the new sprite icons. Allow 10-15 minutes exploration time. What can they discover?
		• Have a plenary for children to share what each new sprite icon allows. Can anyone see where you could change the background of the stage? Let children have 5 minutes to explore the stage background icons. Can they see that they work in a similar way to the sprite icons?
		 
		• Open a new project (File-New). Now focus on programming blocks.
		• What can they make the cat sprite do? What can they discover? Explore individually talking to your friends about what is happening, but stay with the one sprite. **Emphasise clicking on individual blocks to see what they do.** Don't rush children to connect blocks together. For each rule below encourage them to click on the blocks within the block area and then begin to drag them into the script area. Once one of the children discover the blocks will link together, begin to talk about sequences but maintain emphasis on always beginning by clicking on a new block by itself to see if you can discover what it does. (*Institute of Education at University College London refer to this as Direct Drive and provides an opportunity for all children to build their understanding of what a programming block can make happen*), Note that clicking on the top block of a sequence will run the sequence (a control block isn't required).
		  ○ Rule 1: Only use cat and blue (**motion**)
		  ○ Rule 2: Explore purple (**looks**) blocks as well
		  ○ Rule 3: Add in pink (**sound**) blocks. What can you do with only these blocks? (see session 3 for detailed work with sound blocks)
		  ○ When appropriate introduce **control** blocks (and **event** blocks if using version 2.0). Direct towards these six blocks rather than let the
			Gold: Can I select programming blocks for a specific purpose?
			
			Silver: Can I use a sequence of steps to make things happen?
			
			Bronze: Can I tell you what different blocks make happen?

Thanks to Janet Ross (West Monkton Primary) for contributing to original WCP materials.

		children explore all the control and event blocks at this stage.  and  • While they are exploring, introduce and give opportunities for the children to use the words: sprite, background, blocks and stage. • Open a new project. Children choose background and sprite. Use all they have learnt to make something surprising happen to their sprite. • Children show friends the project they have created. They could move around to look at one another's screens and think about what they have done. • What have you seen that has interested you? Give the children the opportunity to make changes to their program based on what they have learnt from others. • What have children learned about Scratch? *That blocks will make things happen, that blocks can be dragged into the script area, that blocks can be connected together, that green flag or a key press can make project start*. **Make sure children recognise that individual blocks make something happen and that sequences can make more than one thing happen** – an event block isn't always required.	
2	**Programming** I can put programming commands into a sequence to achieve a specific outcome.	**Change how I look and move** • Review what children have learned about Scratch. How did we tell the sprite to do what we wanted? (*Clicked on an individual programming block or created sequences of instructions using programming blocks.)* • How could we make the cat walk across the stage? Get children to physically walk and think about what they are doing or get the class to watch one person walk – moving along, changing legs.	Gold: Can I make a sequence to keep the cat jumping and walking backwards and forward across the stage? Silver: Can I create a sequence to make the cat keep walking backwards and forwards across the

	I can use repeat commands.	• Give the children a three block challenge which scaffolds the children to make the cat walk. They must use the cat sprite. Tell them they can try changing the numbers in the blocks. They can use a block more than once, but they can only use these three blocks.  • What can they make happen? Direct children to click on the next costume block. What can they see happen? You could show the children the different costumes (appearances) by clicking on the costumes tab above the block area. • Encourage the children to make changes to the numbers inside the blocks to see what difference this makes. **Optional** if you think children are ready: *You could use the phrase' vary the numbers' to prepare children for later understanding of variables.*  • Children could make a short sequence and keep clicking the top block  • They could make a long sequence. • Ask children what pattern they can see? The same thing keeps repeating so what block could we use to make the sequence of blocks shorter? – yellow **repeat** block. • It would also be useful to get the children to add the blue '**if on edge bounce'** block at the top of the sequence to stop the cat disappearing off the side of the stage. This block needs to go inside a repeat so that the cat keeps checking whether it is on the edge.	stage? Bronze: Can I make the cat walk across the stage?

- What could make the cat keep moving all the time? Use the '**forever**' block. What will make the cat stop moving? *Red stop icon next to green flag.*
- We can stop the cat from turning upside down by adding the 'Set rotation style left-right' block at the start of the sequence.

- Extension: Can you make sprite jump across the stage instead of walk? What happens when you jump up and down. Has anyone discovered the block that lets the cat point in different directions to move? Add the '**point in direction block'** to the challenge. Get children to click on the down arrow to see the options.

- **What have we learnt? How did we learn it?** Encourage children to think about how their friends may have helped them and how they have learnt from mistakes they may have made.

Year 3 Programming 1 Making my Program in Scratch (Core) inc RAG and PRIMM

3	**Programming** I can describe the algorithm I will need for a simple task. I can use repeat commands. I can keep testing my program and recognise when I need to debug it. **e-Safety** I can post positive comments online.	**Change my sound** • Review learning from last session. Remind children of the word algorithm. Can anyone tell you what this means? This is when we think through what we are going to do before actually doing the programming. What will we need to do to achieve an outcome? It is the sequence of actions which we can then make happen using the correct programming blocks. You could use the algorithm rap with the class.  • Ask children how we can add a sound to our sprite? Which colour blocks will we use? Take suggestions then model selecting the sprite, clicking Sounds tab and choosing a sound from the library. You can only add a sound to your program if it has been added to the list of sounds under the sound tab. The cat has one sound – the meow – but other sounds can be added from the library or you can record your own sound by selecting the microphone icon if you have a microphone in your laptop or PC or have a microphone attached to it. • Check children have discovered **play note** block, **play drum**. • You may want to check they have discovered they can add a sound from a folder on their device with the folder icon. • Tell the children they can select a sprite and practise adding sounds (recorded or imported). Only allow them a maximum of 2 minutes to select their sprite. Can they make the sprite move and talk (if their device has a microphone)? Otherwise they could make their sprite move with different sound effects. **Make me dance** • Explain that now we are going to combine sound and images to make a dancing sprite. A support sheet is available if you would like to see some possibilities. • What is our algorithm to create a dance? Act it out – do a short sequence of moves the children could program in Scratch eg ballet dance side to side and up and down OR break-dancer, jump one side then other side, bounce up and down. Ask the class to describe the moves - this is the algorithm. Write / draw the algorithm – sequence of moves as a class or in pairs.	Gold: Can I tell you the algorithm I will use to create a dance to a beat and implement this as a program which includes changing costumes? Silver: Can I tell you the algorithm I will use to make the cat sprite dance to a beat and say something? Bronze: Can I create a program to create a dance to a beat?

Thanks to Janet Ross (West Monkton Primary) for contributing to original WCP materials.

- Move right
- Play sound
- Move left
- Play sound
- Move up
- Play sound
- Move down
- Play sound

- Tell the class they are going to implement the algorithm as a program. They need to add ballerina or break-dancer 1 sprite to a project and delete the cat. Support the children to RAG their design / algorithm as a self-assessment to see what knowledge and skills they already have that they can apply. You may need to model this if the class are using the idea of 'ragging' for the first time.

RAG: Self-assessment / Formative assessment Support Poster

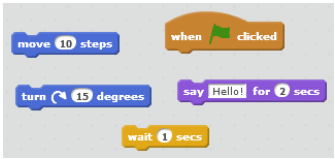
- Photocopy a class agreed **algorithm** or have children working in pairs or individually with their own algorithm
- Ask children to **RAG** the algorithm, (highlighters or underline with coloured pencils).
  - Red: 'I do not know which blocks to use to make this happen … yet.'
  - Yellow: 'I think I know what to do to make this happen but I am not sure. I am happy to have a go.'
  - Green: 'I know which blocks to use to make this happen.'
- Use the self-assessment to **identify** children that may need:
  - Red: additional direct teaching
  - Yellow: Reinforcement
  - Green: confident to work independently

Challenge the children to implement this as a program

- Additional support that may be required:
  - How will you change the direction of movement? Use the **point in direction** block or use **move** 20 steps, move -20 steps or **glide to**.
  - What block can we use where there is pattern in dance where a move happens again and again? ***Repeat*** *block*
  - What other moves will you add to your dance?
- Remember use of 'set rotation style left-right' to stop the dancer tipping upside down.
- Remember to keep testing your program and debugging where there are mistakes or you want to improve it.
- Let the children program the algorithm.
- Stop the children to look at the costumes available for each sprite. How could these be incorporated into the dance? Remind children of the purple look costume blocks. Let them experiment with the difference between **switch costume to** and **next costume** blocks. What moves do the costumes let us add to our algorithm? eg jump or plie. Add moves into algorithm and then challenge the children to add to their program.

- We can also make the sprite say something with a speech bubble. How will we add this? *From the Looks blocks, drag the 'say' block and join it to the top of the stack. Left-clicking the text window allows you to type whatever you would like the sprite to say. Type in "I want to dance!" Then drag another say block and join it to the bottom of the stack and type in "I love dancing!"*
- What will add interest to the appearance of the dance? *The sprite could be on a stage.* Let the children select a background for the dance.
- How could you start the dance? Green flag, press a key, click on sprite (*event blocks*)

Thanks to Janet Ross (West Monkton Primary) for contributing to original WCP materials.

		• Share some of the dances with the class, asking them to comment with two stars and a wish. Model commenting positively and including an idea for improvement.	
4	**Programming** I can break an open-ended problem up into smaller parts. I can keep testing my program and recognise when I need to debug it. I can detect a problem in an algorithm which could result in unsuccessful programming.	**Create a Knock, Knock joke** • Tell the children that they are going to make two sprites do different things at the same time. • Give the children a five block challenge. What can they make happen with these five blocks? Tell them they can change the numbers in the blocks. They can use a block more than once, but they can only use these five blocks.  • Ask the children to add a second sprite to their project. Can they use the same five blocks to make the two sprites do something together? They need to select the sprite in the sprite area. This is then the sprite they are programming. Each sprite will have its own sequence of programming blocks. • What is the purpose of the 'When green flag clicked' programming blocked? *Try to run the program without the block. Each sequence will run when you click on the top block, but they won't run together. Make sure children understand the purpose of the 'green flag' is to make both program sequences run at the same time. They both start with the same event block.* • Choose some children to explain what they have done and you show on the board. Has everything they have done been successful? If not, do they know why not? • Tell the children they are going to animate a knock, knock joke: choose a joke from http://www.funology.com/knock-knock-jokes . You could set this as homework. • Act it out: In real life, enact two people telling a knock, knock joke. Who says which part?	Gold: Can I program my own animated joke with two characters that move around, continually trying out the programming and debugging any mistakes? Silver: Can I program my own joke with two selected characters, continually trying out the programming and debugging any mistakes? Bronze: Can I program two characters to tell a joke?

Thanks to Janet Ross (West Monkton Primary) for contributing to original WCP materials.

		Person one: Knock, knock. Person two: Who's there? Person one: Harry. Person two: Harry who? Person one: Harry up, it's cold out here! • Ask the children to talk to each other to describe the algorithm of telling a joke. • Model the algorithm on handout sheet to the class. Give pairs the partly completed algorithm sheet to complete. • **Ask children to 'RAG' their algorithm as per the RAG: Self-assessment, formative assessment in session 3 above.** • Open a new project. Choose two sprites. Can you program the sprites to tell a knock, knock joke? Remind them of the looks block 'say XXXX for XX sec'. (Some children may decide to record their own voice. If they choose this way they will need to record the correct part of each line of the joke for each sprite. It is best to avoid this option unless the children choosing it are confident with Scratch.) • Tell children it will be important for them to keep trying out their programming as they go along. Keep reminding them of this. • Leave children to work at telling the joke, intervening as required and continually reminding them to test out their programming. Prompt children to use when green flag clicked to start each sprite. Some may discover other ways but using the green flag will start both sprites off at the same time. Tell children to add start sprites together in their algorithm handout. • Children may need to consider the 'wait for xxx' control block so that one sprite waits for the other to finish before they tell their part of the joke. Acting out telling the joke again can support children to recognise why they need this. What is person 2 doing while person 1 is talking? *They are waiting.* Tell children to add wait to their algorithm handout.	

Thanks to Janet Ross (West Monkton Primary) for contributing to original WCP materials.

		• Early finishers can be prompted to add a background and appropriate movement to their sprites. • Ask children to save projects. View and discuss projects with 2 stars and a wish. Talk about the way in which the algorithm has been implemented by different children. Talk to the children about the need to make changes to the algorithm as well as to the program to get the right outcome.	
5	**Programming** I can describe the algorithm I will need for a simple task. **e-Safety** I know that anything I share online can be seen by others. I can post positive comments online.	**Make an interactive scene** • Review what we have learned about Scratch so far – using blocks, wait, repeat and forever commands, changing backgrounds and sprites, changing costumes, adding sound. • Explain to the class that we are going to design and program our own interactive scene using Scratch (could link with a class text e.g. fairy tales). **Predict** • Interactive story. Look at example with the class. Can the children 'read' the programming blocks for the Princess? Read together as a class. Ask the children to predict what happens. **Run** • Run the sequence for the princess by clicking on the top block in the sequence (NOT the green flag at the top of the screen). Does the princess do what you predicted? **Predict** • Ask the children to work in pairs to read the programming blocks for the knight and for the dragon. Can they predict what each will do? • Take feedback about what the children predict for each sprite. Can they predict what will happen in the scene when the program is run for all the sprites together?	Gold: Can I plan an interactive story with more than one sprite that interact with each other? Silver: Can I plan an interactive story with more than one sprite? Bronze: Can I plan an interactive story with one sprite?

Thanks to Janet Ross (West Monkton Primary) for contributing to original WCP materials.

		Run • Provide the children with a link to the project online or download for them to run on your local network. Ask them to run the program for each sprite by clicking on the top block in the sequence. Next get them to run the program by clicking on the green flag at the top of the screen. • Feedback: Were their predictions correct? What job does the 'when the green flag is clicked' block do? (*Will start the blocks for all the sprites working at the same time.)* **Investigate** • Give children time to change values for movement, waits and repeats. Encourage them to talk in their pairs about what each change makes happen. **Modify** • Ask each pair to make changes to the 'story'. Can they make the dragon fly? What else might the knight or the princess do? • Let the children have time to show their 'story' to other pairs. • Make a list of success criteria for the 'story' – this can include the idea of what could make it better (keep this to things that you know are achieveable!) **Make** • What story would you like to create? Give the children time to design a scene. Make a note of the type of background and number of sprites wanted before start. Emphasise that they must keep it simple. They can always add extra bits add to their story later. • What is your algorithm? What do you want to happen? Draw a story board to show changes- Look at Events blocks and think about when changes are going to happen. Note events on story board (you may wish to make links with storytelling and speech marks in English). • **Ask children to 'RAG' their algorithm as per the RAG: Self-assessment,**	

		formative assessment in session 3 above. • Share plan with a partner and explain to them how you are going to construct your algorithm as a program using programming blocks to program your sprites and to achieve your algorithm. Encourage children to help each other where they have red or yellow highlights/underlines on their plan.	
6	**Programming** I keep testing my program and recognise when I need to debug it **e-Safety** I can protect my personal information when I do different things online. I can post positive comments online.	**Make and share an interactive story** • Remind children of the success criteria you agreed. • Let children talk through their algorithm planning sheets with a partner to remind themselves of what each sprite is going to do. Are sprites going to move? Speak? Change costume? What else could they do? • Tell the children they now need to implement their algorithm as a program. Children to animate sprites. Remind them that if they have more than one sprite, they will need to click on each sprite to program it in turn but they can copy blocks from one sprite to another if they want them to do the same thing. (Drag the sequence on top of the sprite you want to do the same thing.) • Independently open project and program each sprite, considering the need to test as you program in order to make sure it's doing what you expect. • The projects can be saved on your local drive and children can look at each other's. • **Optional:** If created in Scratch 2.0, you could upload projects to the Scratch community. You can upload these with a class log in rather than give children individual logs ins at this stage. • Make sure you talk through e-Safety considerations with the children: Who will see these? Is there any personal information they may have included? You could add hyperlinks for the story on your class webpage or blog or in Class Dojo class story **OR** you can choose **EMBED** option to add within a web page on the school website. Children will then be able to show them to friends and family at home. • You could prompt other teachers with a Scratch log in, or older children in the	Gold: Can I program more than one sprite to interact with another? Silver: Can I program more than one sprite? Bronze: Can I program my sprite?

Thanks to Janet Ross (West Monkton Primary) for contributing to original WCP materials.

		school to comment on the work in the Scratch community. You can delete any inappropriate comments. You could model this with the children. Let them decide if the comment is okay to keep, should it be deleted or should the person be reported.	
7	**Reinforce the learning that has taken place OR use as an assessment activity:** The final activity of creating an interactive scene can be repeated focussing on an interactive card. This could be Harvest, Halloween or Christmas. Children could look at examples in the online Scratch community. • Log into Scratch online community with a class log in. Don't tell the children your password as you will be adding comments. Ask the children why it would be important to not let others know your password if you are going to be commenting on another person's work. *You can make sure your comments are always positive but if someone logged in as you they could say something nasty that could result in you being banned from the website.* • Ask the children for comments for examples you have identified. Remind them of the two stars and a wish that you used to comment on the dances in session 3. What feedback shall we add for the people who created these projects? • Give children time to design their own 'celebration card'. Use the RAG strategy to identify confidence of the children and any further support they may require. • Let children create their cards and share them as detailed in session 7.		