



Topic 1 – Graphic Design and Manipulation

Year 9 builds on knowledge and skills from Year 7 & 8 and introduced more advanced coding, graphics and design skills. Work in Year 9 is based around working to a client brief and undertaking creative projects as well as creating, re-using, revise and re-purposing digitally artefacts .

1. What makes good design? – What makes an image visually appealing? You will look at examples of both good and bad design to come up with a set of “key design rules” that you will follow through this unit.

2. Introduction to Draw Plus – Learn how to use Serif DrawPlus – a professional graphics design package. Learn how to use advanced tools to both create and manipulate graphics to produce professional looking final products.

3. The importance of planning – Learn why the planning stage is often the most time consuming and import part of a project. If you fail to prepare, you prepare to fail! Learn about mind maps, mood boards and visualisation diagrams.

4. Get creative – Using the skills and knowledge learnt so far, along with a suitable plan and following the golden design rules, complete a short project based on a real world client brief. This will give you valuable experience of working to a deadline and working alongside a client to make their ideas a reality.

5. Review and evaluate – How happy is the client with what has been produced? How do you know? Being able to critically evaluate both your own and other’s work is a key skill to producing the most professionally looking final pieces. Learn how to both take and receive creative feedback in order to produce the best possible final outcomes. Your work will go through several iterations before being “perfect”



Year 9

Half Term 1

Half Term 2

7-14. Design, Develop and Test a fully functioning game in Construct 2 – This is the start of their extended creative project and assessment for this unit. Working to a client brief, design, plan and create a fully working product. Combine all of the skills and knowledge learned so far in this and the previous units to create a professional final product that will satisfy the requirements of the client brief. The final product will be full tested by both the author and gameplay tested by users to inform areas for improvement.

6. Experimentation time – Time to experiment! Game design involves a lot of experimentation time and that there is more often than not more than one way of doing things. Time is allocated to make mistakes, identify what went wrong and find creative and inventive solutions.

4. Variables – Explore the use of variables in coding and more specifically in game design. Learn how to create variable to control different element of games such as health, time, ammo. Arithmetic operations are introduced at this point to build more complex code into projects.

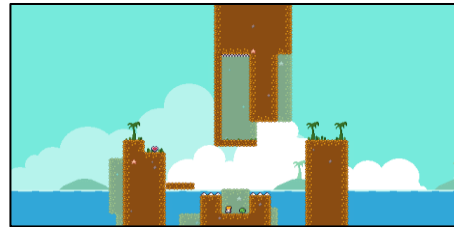
Half Term 3

Half Term 4

5. Gameplay mechanics – What is it that makes games fun to play? What keeps players coming back for more? Explore game mechanics by looking at some classic examples and than design, plan and implement game play mechanics to add to individual projects.

3. Pseudocode – Introduction to the concept of pseudocode – talking about code in plain English. Use this skill to debug example code and then fix problems in provided coding in examples. Linked to this is the idea of decomposition- breaking down coding problems in to small chunks as well as debugging code.

2. Events and actions – Understand how coding in Construct 2 uses a hierarchical structure and a “top down” approach. Explore the concept of “events” and “actions” and the “if this, then that” element of coding. Write code to collect an item when the player collects it and add to a scoring system by learning about both global and instance variables.



Topic 3 – Python programming

1. Python warmup – In this lesson, you’ll learn to write programs that display messages, receive keyboard input, and use arithmetic expressions in assignment statements. You’ll explore using if-elif-else statements to control program flow. Additionally, you’ll practice locating and correcting syntax errors and creating and accessing list items. These skills will build your programming foundation for future lessons.

2. Playlist – Here you will cover common list operations: adding, removing, or modifying items, and locating or counting occurrences. Learners tackle short challenges, identifying and applying relevant list operations to complete tasks to further improve your coding skills.



Half Term 5

3. In a while, crocodile – This lesson teaches you how to use iteration (while statements) to control program flow. You’ll also learn to perform common operations on lists and individual items, as well as operations on strings and individual characters. These skills are essential for effective programming.

4-5. The famous for – In this lesson, you will use a for-loop to iterate over list items. You will study various examples to understand its syntax, use, and mechanics before applying this knowledge to similar tasks. Activities include iterating over lists of real-world textual and numerical data, requiring you to recall and apply knowledge from previous lessons.

Half Term 6

6-10. Make a thing – In this lesson, you will choose from several meaningful mini-projects to apply your acquired knowledge and skills. Each project includes an introduction for context, a detailed description of the task, and clues to help create a solution. You will select one mini-project to complete for your final assessment.

