Year 8 – Computing

For spring term 2, pupils are using KODU Game lab to develop key programming skills. The software is free and will need to be downloaded and installed on home computer.

**Worksheet 3: Navigation and pathing**

Open up one of your previous worlds from last lesson. If you don’t have a world of your own then you can use “Lesson 3 - Navigation Methods”.

Do not start a blank world, the idea of this worksheet is to practise coding navigation techniques.

1. Add some manual navigation to your main character so that it moves around the landscape when the arrow keys are pressed
2. Add at least one automatic method of navigation to your game such as:
   1. When an enemy character “sees” Kodu it starts to move slowly towards him
   2. Enemy character starts the game wandering around the map slowly
3. Add a path to the game to form a circuit around your island
4. Set the colour of the path to red
5. Add another character and program him to move on the red path

**Note:** If you get stuck examine the code from “Lesson 3 – Navigation Methods”.

**Challenge Tasks**

Try the following.

1. Add a line of code so that if the “Shift” key is pressed along with the arrow keys then the player gets a speed boost and moves “Quickly”
2. Add another line of code for the character who is moving on the path so that if it sees Kodu nearby then it will start to follow Kodu
3. Add another line of code for the same character so that if it no longer is able to see Kodu nearby then it should move back on the path

**Stretch & Challenge Tasks**

1. There is actually a much better way to do tasks 6-8 above using the concept of **“Pages”**. Pages will be covered in a later lesson, but if you have got this far you might like to experiment!

**Worksheet 4: Clones vs Creatables**

Open up one of your previous worlds from the last lesson. If you don’t have a world of your own then you can use **Lesson 4 – Clones vs Creatables V2**. However, ideally by now you should be building on your own world each lesson and slowly developing your own game.

1. Do you understand the big difference between clones and creatables? Turn to a partner and explain it. What are the main advantages of turning an object into a creatable over simply copying and pasting it?
2. By either adapting one of your existing characters or by adding a new one, turn one of them into a creatable.
3. The trick here is to find a good reason to use a creatable. Simply doing it for no reason will not make sense. Show your creatable to the teacher or a partner and explain what you are using it for.

**Challenge Tasks**

Try the following.

1. Often when using a creatable to produce armies of enemies in a game you will find they all end up moving in a very fixed and artificial way.
2. Creatables **do not** have to all behave in exactly the same way even if they share the exact same code! Can you think of a way to program your creatable so that all the individuals you have created move differently?

**Hint: Wander**

**Stretch & Challenge Tasks**

1. Creatables can be used in many other powerful ways, e.g.
   1. Every time the Space bar is pressed your character fires out a special object
   2. Every 5 seconds another enemy spawns in the game
   3. Every time your character crosses a check point a power up or booster coin appears
2. A lot of the code needed to make the above situations occur has not yet been covered, but you can experiment to see whether you can figure any of them out.

**Hint: You will need to use concepts such as “Timers”, “Launching” and   
“On-Land”**.

**Worksheet 5: Pages and selection**

Open up one of your previous worlds from last lesson. If you don’t have a world of your own then you can use “Lesson 5 – Melvin the Turtle”. However, ideally by now you should be building on your own world each lesson and slowly developing your own game.

1. Do you understand what the term **Selection** means? Turn to a partner and explain it. How is **Selection** different from **Sequence**?
2. What are the three methods of **Selection** you can use in Kodu?
3. Look through the existing code you have in your game. Are there any sections of your code you could improve by using “Indenting”?
4. Program a simple “Page” change into your game just so you can get it working. Something like **If** Kodu bumps into an apple **Then** switch to page 2. On page 2 set it so that you “win” the game.
5. Test that your page changing code works. Do you win the game when you bump into the apple? If not, why not?

**Stretch and challenge Tasks**

1. Try to change one of your characters so that its behaviour changes in some way based on what is going on around it in the game. (Like, for example, the fish getting close to the turtle and becoming angry.)
2. Pages can be combined with shifting camera angles to add some very dramatic effects to your game. For example switching to first person perspective when a power-up is collected. We have not gone over how to set or change camera angles using code, but see if you can figure it out.
3. Attach the change in camera angle to a page change.

**Worksheet 6: Game depth and complexity**

Open up one of your previous worlds from last lesson. If you don’t have a world of your own then you can load up any of the previous worlds you have been using in the past lessons.

1. Think about the features you have been exploring today. In the list below, the green (starred) ones are demonstrated in the sample game you have just looked at.

**Shooting\* Holding Expressing**

**Communication Health\* Grabbing**

**Scoring\* Special effects Colour winning\***

**Music Power ups\* Lighting effects**

**Levels On-Water / On-Land Camera angles**

**Timers\* Winning & Losing\* Giving**

**Launching\* Camouflage**

1. Adapt your existing game world by adding in at least two of the above features.
2. If you are finding it hard, then explore the code from the sample game “Lesson 6 – Game Depth and Complexity” some more. Then try to copy the ideas to your own game.

**Stretch and challenge tasks**

1. Try to add a layer of depth and challenge to your game. Can you implement power-ups or power-downs?
2. Can you make these power-ups and power-down effects only temporary? You will have to use a timer and pages for this.
3. Add a way for your game to be “Won” and also a way for it to be “Lost”.
4. Make sure that the instructions to the player at the start of the game tell them what they need to do in order to win.
5. Challenge a partner to play your game. What did they think? Ask them for suggestions for improvements and try to make those changes.