Scheme of Work Overview

About this unit

This unit aims to teach students the fundamentals of games programming using Kodu, which is a visual game development environment.

Using Kodu students will understand how to build a world and program characters and objects before moving on to designing their own games.

Homework

Kodu is free to download from http://fuse.microsoft.com/kodu so students can use the software at home to make their own creations.

Language for Learning

World, character, object, program, page, instructions, sequence, design, evaluate

Resources

→ PowerPoint for each lesson outlining tasks
→ Kodu
→ Xbox game controllers (optional)
→ Help cards

Cross Curriculum

Numeracy – Logic and sequencing
Literacy – Story creation in own game

Support and Extension

Some students may benefit from working in pairs. Laminated help cards are also available.

Extension – Some students can explain what the code in their game actually does. Students can have a go at replicating old arcade games in Kodu

Other Information

Help Sheets

A number of help sheets are available on the shared area so students can understand the laws of Kodu and how to program etc.

Students are encouraged to access these whenever they are required.

Extension Material

A number of extension activities are available on the shared area and students are encouraged to use these both inside and outside of the classroom

Lesson	Resources
1	❑ Lesson 1 PowerPoint ❑ Self-Assessment Sheet ❑ Kodu

Learning

Objective(s):
→ Learn how to create a world and control a character using Kodu.

Outcome(s):
→ A completed world demonstrating different:
  ▪ Heights
  ▪ Types of terrain
  ▪ Varieties of colours
  ▪ Water/liquid
→ Character, controlled by the user that is able to collect objects.

Lesson Activities

Introduction to the Unit – explain what the unit is about, what it will involve and some example of games previously created in Kodu

Student:
✓ Write learning objective onto **Self-Assessment Sheet.**

Teacher:
→ Introduce what Kodu is and what it does, show example game.
→ Demonstrate how to make a world with different:
  ▪ Heights
  ▪ Types of terrain
  ▪ Varieties of colours
  ▪ Water/liquid
→ Demonstrate how to add objects (i.e. apples) to the world.

Student:
✓ Create a world of their own with a river through the middle.
✓ Add objects to be collected.

Teacher:
→ Demonstrate how to add a character to be controlled by the user.
→ Demonstrate how to make the character collect chosen objects.

Student:
✓ Add a character to the world and program it to be controlled by the user.
✓ Program the main character to collect objects by eating them.
✓ Use **Self Assessment Sheet** to assess their own progress.

Lesson	Resources
2	❑ Lesson 2 PowerPoint ❑ Self Assessment Sheet ❑ Kodu ❑ Lesson 2 – Starter activity (cut out)

Learning

Objective(s):
→ Learn how to add different types of paths to a world and control an enemy within a game.

Outcome(s):
→ A world that contains:
  ▪ Walls
  ▪ Bridge
→ Enemy that follows a set path and shoots at the player.
→ Player character that can fire when pressing a button.
→ *SOME students may also have adjusted the World Settings for the game.*

Lesson Activities

Student:
✓ Write learning objective onto **Self Assessment Sheet.**

Student <IN PAIRS>:
✓ Complete the starter activity by matching up Kodu's instructions to the correct boxes.

Teacher:
→ Go through answers to starter activity on the board.

Teacher:
→ Demonstrate how to add paths to a world to make
  ▪ Walls
  ▪ Bridges

Student:
✓ Add walls and a bridge to the world they created last lesson.

Teacher:
→ Demonstrate how to add a plain (invisible) path to be used for controlling a characters motion.
→ Demonstrate how to make the "enemy" character shoot at the user.

Student:
✓ Add a plain path to their world for the enemy to follow.
✓ Program the enemy to follow the path and shoot at the user.
✓ Program the users character to shoot when a certain button is pressed.
✓ *EXTENSION: Students who have completed all activities should be looking at the World Settings and experimenting with changing variables in here.*
✓ Use **Self Assessment Sheet** to assess their own progress.

Lesson	Resources
3	❑ Lesson 3 PowerPoint ❑ Kodu ❑ Self Assessment Sheet ❑ Small world containing apples and Kodu already created and controlled using Xbox controller (optional).

Learning

Objective(s):
→ Create a level system in Kodu and Set up a simple combat style game

Outcome(s):
→ A basic game in which the character moves between different worlds that act as different levels
→ Set up your characters so they aren't killed after being hit by one bullet
→ *SOME students may add Game End and Win screens to display at the end of the game.*
→ *SOME students will experiment in detail with other coding blocks that are available in Kodu*

Lesson Activities

Student:
✓ In groups create a mind map of all the different things you have learnt so far

Teacher:
→ Discussion about the type of skills and game that have been learnt so far. Discuss good and bad features

Student:
✓ Write learning objective onto **Self Assessment Sheet.**

Teacher:
→ Show video on how to create different levels and changing hit settings

Student:
✓ Rewatch the video and create a similar type of game to that in the video

Teacher:
→ Show the use of some of the different types of code block

Student:
✓ Play around with the different code blocks to see what is available that may become of use when creating own game in the coming weeks

Student:
✓ Use **Self Assessment Sheet** to assess their own progress.

Lesson	Resources
4	❑ Lesson 4 PowerPoint ❑ Kodu	❑ Self Assessment Sheet ❑ Small world containing apples and Kodu already created and controlled using Xbox controller (optional).

Learning

Objective(s):
→ Learn how to use different pages and add scoring to a game in Kodu.

Outcome(s):
→ A basic game in which the character gains points by eating green apples and loses health and changes pages when eating brown apples.
→ *SOME students may add Game End and Win screens to display at the end of the game.*

Lesson Activities

Student:
✓ Write learning objective onto **Self Assessment Sheet.**

Teacher:
→ Show example world and demonstrate how to change the colour of objects to create green and brown apples.

Student:
✓ Create a small, flat new world and add Kodu.
✓ Add three green apples and three brown apples to the world.
✓ Program Kodu so he can be controlled with the Xbox controller and eats green apples if he touches them.

Teacher:
→ Show the use of different pages and how to get to them.
→ Demonstrate how to switch to page 2 after eating a brown apple
   to make Kodu turn green and say he doesn't feel well.
→ Demonstrate who to switch back to page 1 after a 5 second delay.

Student:
✓ Program Kodu so when he eats a brown apple he switches to page 2.
✓ Program Page 2 to switch back to Page 1 after 5 seconds while doing the following:
   ▪ Changing colour to green
   ▪ Saying he doesn't feel well

Teacher:
✓ Demonstrate how to add a score to go up when Kodu eats green
   apples and a hit metre to show damage when he eats brown apples

Student:
✓ Change your program so the score goes up when Kodu eats a green apple.
✓ Add a hit metre to show damage to Kodu, this should go down by 2 points when he eats a brown apple.
✓ Make the score increase when Kodu eats green apples.
✓ *EXTENSION: Add Game End and Win screens to display when Kodu loses all health or collects all of the green apples.*
✓ Use **Self Assessment Sheet** to assess their own progress.

Lesson	Resources
5	❏ Lesson 5 PowerPoint ❏ Kodu	❏ Self Assessment Sheet ❏ Example of a "Space Invaders" game created in Kodu ❏ Game Design Templates - Homework

Learning

Objective(s):
→ Learn how to use creatables to make clones of an object.

Outcome(s):
✓ Completed a Space Invaders style.game
✓ Students will experiment in detail with other coding blocks that are available in Kodu and explain how they could be used in game situation

Lesson Activities

Student:
✓ Write learning objective onto **Self Assessment Sheet.**

Teacher:
→ Demonstrate how to make a character a "creatable" and created every five seconds.
→ Discuss reasons why this method is better than copying & pasting multiple characters
→ Demonstrate where to find the world settings to change the lighting effects.

Student:
✓ Students can begin making their own "Space Invaders" game
✓ Students do not need to create land, just "space decoration", an enemy and a player character.
✓ Use **Self Assessment Sheet** to assess their own progress.

Homework:
✓ Students plan out the game they would like to create for assessment

.

Lesson	Resources
6	❑ Lesson 6 PowerPoint ❑ Self Assessment Sheet ❑ Kodu ❑ Help sheets

Learning

Objective(s):
→ Understand the points to be considered when designing a computer game.

Outcome(s):
✓ Completed Game Design Template
✓ The start of own game created in Kodu.

Lesson Activities

Student:
✓ Write learning objective onto **Self Assessment Sheet.**

Teacher:
→ Lead class discussion and mindmapping activity (writing on PowerPoint slide) to discuss good game examples and what makes them a good game.

Student:
✓ Students make any necessary changes to Game Design Template Document.
✓ Students begin developing their game in Kodu once their designs have been checked by the teacher.
✓ Use **Self Assessment Sheet** to assess their own progress.

Lesson	Resources
7-9	❏ Lesson 7 PowerPoint ❏ Self Assessment Sheet ❏ Kodu

Learning

Objective(s):
→ Learn how to make a game in Kodu from a design.

Outcome(s):
→ Completed game created in Kodu that matches design.
✓ Completed peer assessment sheets.

Lesson Activities

Student:
✓ Write learning objective onto **Self Assessment Sheet.**

Teacher:
→ Recap on what they did in previous lesson.

Student:
✓ Continue to develop their game assessment making use of their game plan.

Teacher:
→ Help individual students with their game production.

Student:
✓ Use **Peer Assessment Sheet** to evaluate their partners game.
✓ Use **Self Assessment Sheet** to assess their own progress.
.

LEVELS AND PROGRESS

Excellent	Secure	Developing
• Gameplay is realistic and is suitable for purpose and audience	• Gameplay is good with some unsuitable elements	• Gameplay is basic and may not be suitable for audience
• Handed in all documentation	• Handed in all documentation	• Handed in most of the documentation
• Made use of complex coding blocks (E.g. pages, timer)	• Made use of a range of advanced coding blocks (E.g. score, health)	• Made use of basic coding blocks (E.g. move, shoot)
• Create efficient sequences of instructions including the use of subroutines	• Annotated code is clearly explained	• Annotated code shows little understanding
• Annotated code is clearly explained and in great detail	• Evidence supplied is to a suitable standard	• Minimum requirements not met
• Evidence supplied is professionally presented	• Use criteria to evaluate the quality of solutions, identifying improvements and refining their work.	• Evaluate the quality and success of their solutions.
• Use criteria and feedback to improve the effectiveness and efficiency of solutions.	• Create precise and accurate sequences of instructions.	• Devise and refine sequences of instructions.
• Design own criteria for a Kodu test table and test final game	• Change variables within models and explain the impact.	• Use models to explore relationships between inputs and outputs and explain how the models work.