Endless Runner

Introduction and Sprites

Introduction

- Problem driven
  - Still looking at problems and solutions
  - Expanding this to include strategies for tackling large problems.
- **Starting first Unity game!**
- Concepts
  - Sprites
  - Transformation

Today's Problems

- How can I create an endless runner game?
  - The problem is too BIG!
- How to add a player Sprite in Unity?
  - Sprites
  - Unity image formates and importing
  - Slicing and the sprite editor
- How to I position and orientate my sprite?
  - 2D coordinates
  - Transformation

Problem: How can I create an endless runner game?

Motivation

- Looking to develop an endless runner game. Here the principle character runs at a constant pace jumping over or sliding under obstacles which are randomly generated in their path. If these are not avoided the player character dies.
- Controls
  - W - Jump
  - S - Slide

Solution(s)

- Create a 2D Game in Unity with the features described above:
  - User controlled character
  - Appears to run at a constant speed
    - While staying at the same point on screen
  - Randomly generated obstacles move towards the player
    - Contact results in player character death
- This is a BIG problem
  - How can we tackle this?

Problem Solving Techniques

- Experienced developers have a mental toolkit
  - Solutions to problems (or parts of problems) they've seen before
- They also have techniques for solving large / difficult problems
  - Divide and conquer
  - Simplest problem first
  - Change the problem

Problem Solving Techniques (cont.)

Divide and Conquer

- Split big problems is to smaller ones
  - Small problems are easier to solve
  - Build confidence
  - Learn more about the problem

Simplest Problem First

- Having broken the problem into smaller more manageable ones
- Tackle the easiest one first
    - Build confidence
    - Learn more about the problem

Change the problem

- If the problem is complex or difficult
- Change the problem (make it simpler) or solve a similar problem
  - Explore similar problems/solutions
  - Learn more about the problem
  - Build confidence

How to add a player Sprite in Unity?

Motivation

- We need a image to represent the runner (the players avatar)

Solution

- A sprite is used to represent the player
  Wait ... what's a sprite?
- Sprites can imported into Unity
  - See worksheet (Player Sprite)
- Sprites can be added to a scene
  - See worksheet (Player Sprite)

What is a Sprite?

Sprite

- A sprite is a 2D image or animation (more than one image) we can move/rotate on screen
- Can be downloaded or created

From Learning Unity 2D Development by Example

What is a Sprite? (cont.)

Sprite Sheet

- An image made up of multiple other images
  - May be whole objects (like a film reel)
  - Can be component parts of something animated by the game (Unity).

From Learning Unity 2D Development by Example

From Learning Unity 2D Development by Example

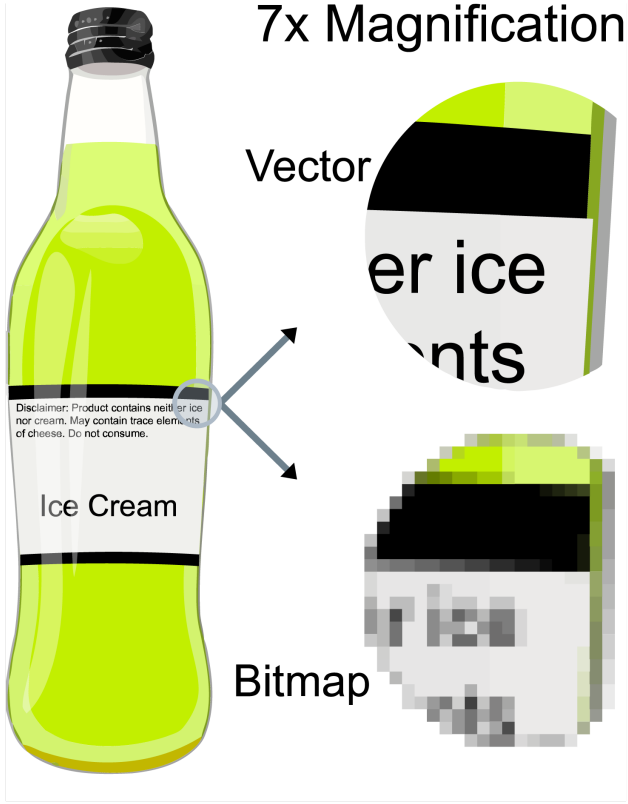
Images in Unity?

Raster versus Vector

- Raster aka bitmap (supported)
  - Made up of bits that translate into pixels defined according to a grid of pixel elements
  - Becomes pixelated at high magnification
- Vector (not-supported)
  - No grid
  - Use a mathematical formula that defines points and paths that connect to form an image.
  - Clear when magnified.

Raster versus Vector (cont.)

Example



From Wikipedia : Note .png copy in notes.

Images in Unity? (cont.)

Supported Types

- PSD
- TIFF
- JPG
- TGA
- PNG
- GIF
- BMP
- IFF
- PICT

Recommend PNG

- This supports transparency (usually denoted by a checkerboard background)
- Uses Lossless compression (see here) which does not lose quality when the size (on disk) is reduced.

How to import images?

- Imported into the Assets folder
    - See worksheet (Player Sprite)

Working with Sprites

Slicing

- Cutting the sprite sheet into multiple sprites
- Unity
    - Provides tools for Manual and Automatic slicing
        - See worksheet (Player Sprite)

Packing and the Sprite Atlas

- Use a single texture file for all images
- Sprites are packed into a texture atlas
- Draw call batching used to improve performance

Transformations

Position

- x,y and z
- We're looking down the z.

Orientation

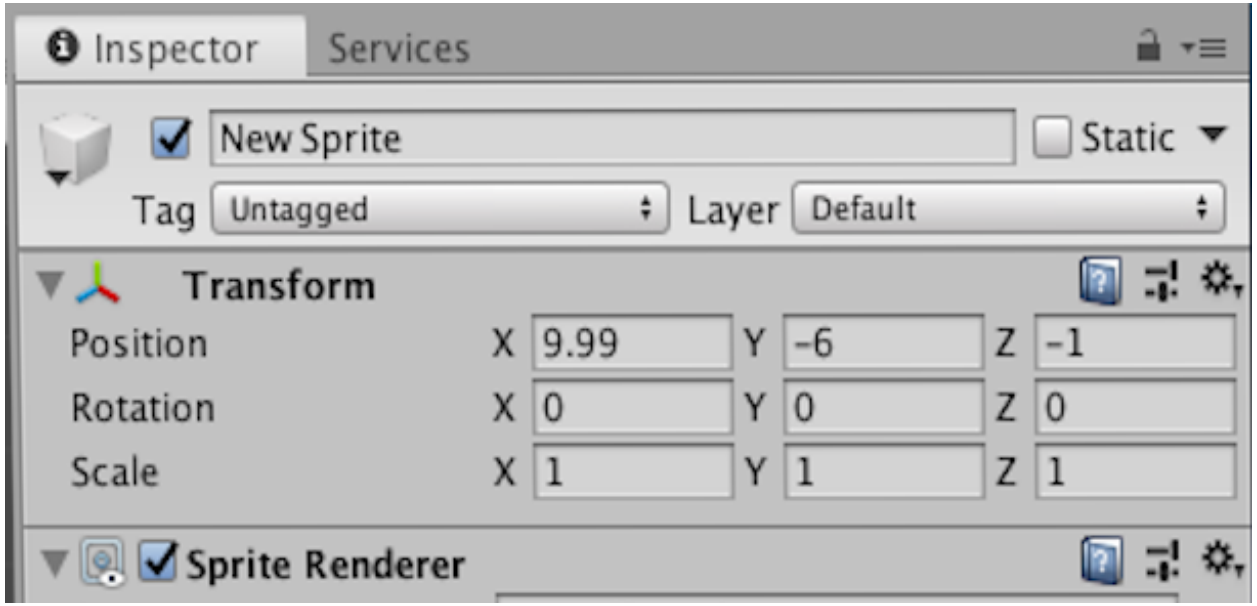
- Rotation around x, y and z axis
- Can switch between object and world centre

Scale

- Change the object size in any direction

Sprites Transform Component

- Unity is component orientated
  - The components present determine what can be done with the object
  - We can add/remove most components
  - Some cannot be removed (e.g. Transform from sprite)



Summary

Sprites

- Essentially images in our scene
- Imported as assets in Unity
- Have a transform component

Transformation

- Move the object from the origin of the world
- Rotate the object from its current orientation
- Scale the object from its original size

Next Week...

- Animating the Player