Top Down Game Scroller Part 1

Introduction

Today we will start with a Top-Down Scroller Game Engine.

We want to be able to build a scrolling world where our hero can explore!

You can take this engine and build cool games with it. We can improve it some more in a few weeks. Our hero's world is populated with trees, he can move left, right, forward and the world is as big as we like.

Lots of games can be done with a top down view – for example:
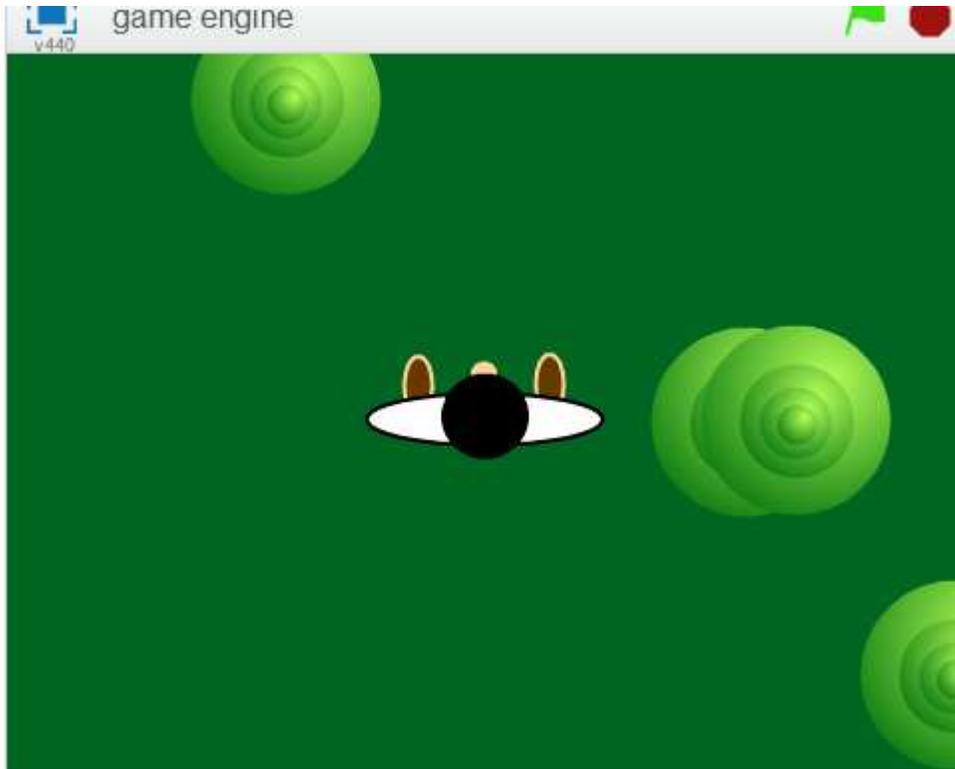
⇨ Adventure games like Zelda (shown below)

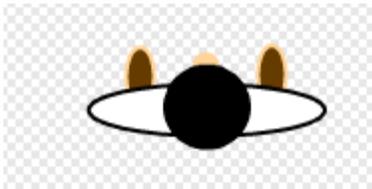⇨ Car driving games like Mario Kart
⇨ Flying games
⇨ Space/Scrolling shooting games

We are going to make and engine for something like a Zelda game.

Instructions:

1. Draw a character sprite. As this is a birds eye view, you can draw the top of his head as I did above. You could also draw it orthogonally like the character lync in the "Zelda" games

2. Next draw something that exists in the world – we started with a tree as this world will have plenty of trees. Again, have fun doing your own design. It can be nice to do it orthogannally so you can see a bit of the trunk .. I didn't below.

3. Finally pick a solid background colour for now. Later we might see if we can do proper moving background like grass, fields, etc but for now it'll be easier if everything is solid.

4. In this game we are imagining that the world is much bigger than the 480 x 360 of the scratch grid. As we move around, the character doesn't go anywhere, but the trees will come to meet him!

5. Player code: Moving the player just changes some variables – thes variables are the x and the y of the player in this large map.

```
when right arrow key pressed
change player map x by speed

when left arrow key pressed
change player map x by -1 * speed

when up arrow key pressed
change player map y by speed

when down arrow key pressed
change player map y by -1 * speed
```

```
when [flag] clicked
set player map x to 0
set player map y to 0
set speed to 10
go to x: 0 y: 0
point in direction 90
```

6. Tree Code: Each tree starts as a clone and figures out its X and Y coordinate from the position that the player was in. When the clone is created it gets a random position in the big world from -1000 to 1000 in the X and Y – this part of the world is like a "forest".

```
when [flag] clicked
set my_width to 95
set my_height to 95
hide
repeat 200
  create clone of myself
```

7. This creates 200 tree clones which might be a bit slow on your laptop! We can look in future weeks to see if we can figure out a more efficient way to draw the trees. If it's slow, change it to a lower number like 10.

8. Next we added code to run whenever a sprits starts as a clone.

9. Next we added some code to hide the trees whenever they are out of sight!

It's not exactly Zelda, but we can look at improving it in future weeks!