

# Computer Science

**LO: I am learning  
to create my own  
game using video  
sensing.**



# Creating Plan, make, assess

## Creating video sensing game



<https://www.youtube.com/watch?v=8vHEqVdWn08>

Click on the link to see what a video sensing game is. I will show you Mr Viney's game when we are back in class! The programme is a slightly updated version but for now this video will give you an idea.





# Creating Plan, make, assess

What do you want your programme to do?

Creating  
Plan, make,  
assess

*Draw and label what your programme will look like (how will the sprites move, what will they say, what sounds will be used).*

Tinkering  
Try things out

*Edit your design with green pen to show how you improved your design.*



# Creating Plan, make, assess

## Example

Creating  
Plan, make,  
assess

*Draw and label what your programme will look like (how will the sprites move, what will they say, what sounds will be used).*

The purpose of the game is to keep the ball in the air.  
Use motion when there is contact with the sprite

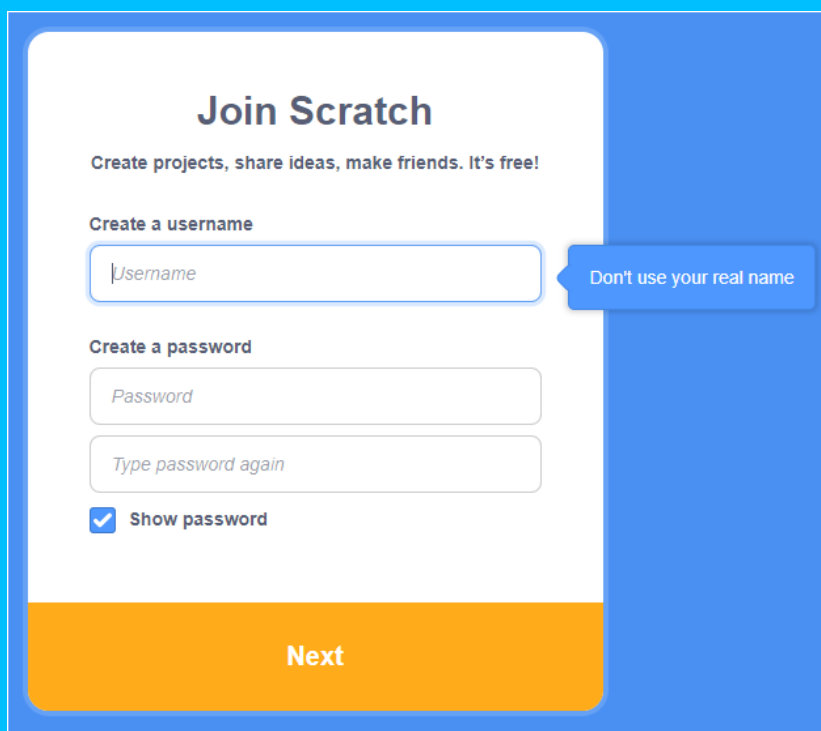
Tinkering  
Try things out

*Edit your design with green pen to show how you improved your design.*

# Computer Science

## Creating an account and logging in

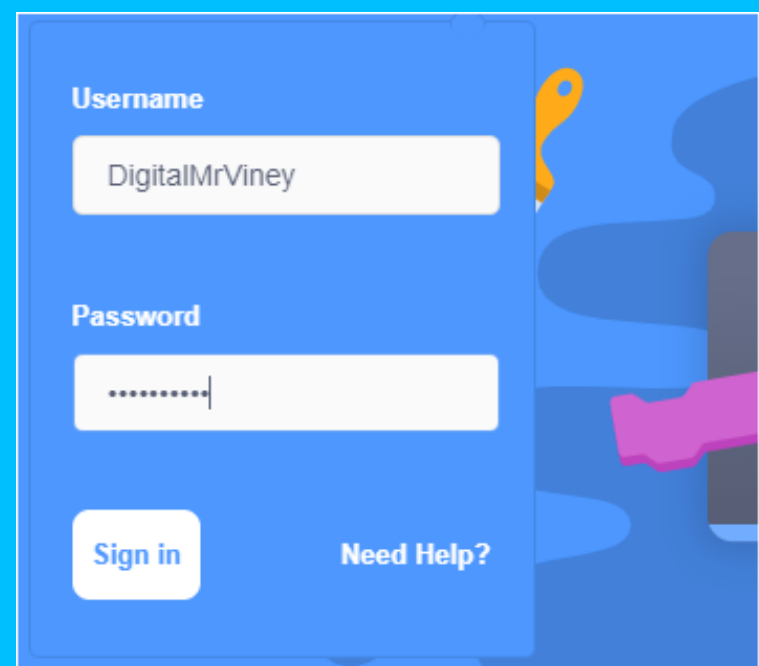
<https://scratch.mit.edu/join>



The image shows the 'Join Scratch' registration form. It has a white background with a blue border. The title 'Join Scratch' is at the top. Below it is the text 'Create projects, share ideas, make friends. It's free!'. Then, 'Create a username' is followed by a text input field with the placeholder 'Username'. A blue callout bubble next to the field says 'Don't use your real name'. Below that, 'Create a password' is followed by two text input fields: the first with the placeholder 'Password' and the second with 'Type password again'. A checkbox labeled 'Show password' is checked. At the bottom is a large orange button labeled 'Next'.

1. Create an account by making up a username and password

2. You will need to create a username and password that you can remember!



The image shows the Scratch login form. It has a blue background with a white box for the form. The title 'Username' is above a text input field containing 'DigitalMrViney'. Below that, 'Password' is above a text input field with masked characters '.....'. At the bottom left is a white button labeled 'Sign in'. At the bottom right is a link labeled 'Need Help?'. On the right side of the form, there is a cartoon character of a yellow figure with a blue tie and a pink figure with a blue tie.

E-Safety: Make sure you use a strong password and do not use your real name.

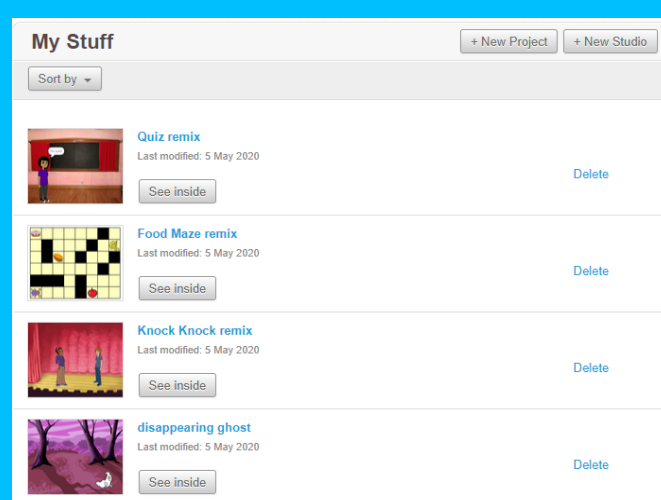
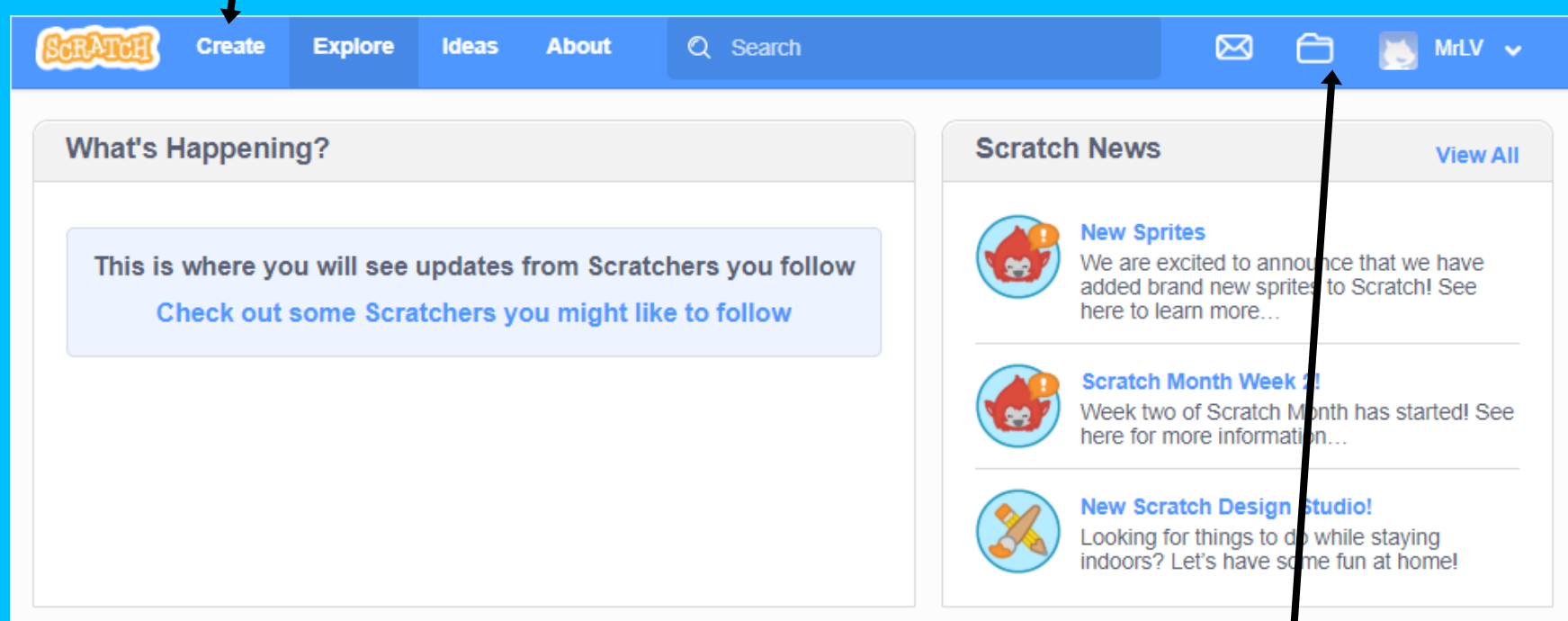


Introduction  
to the platform

# Computer Science

Where to go to create your own  
programmes

Your profile and log out



Where your work is saved

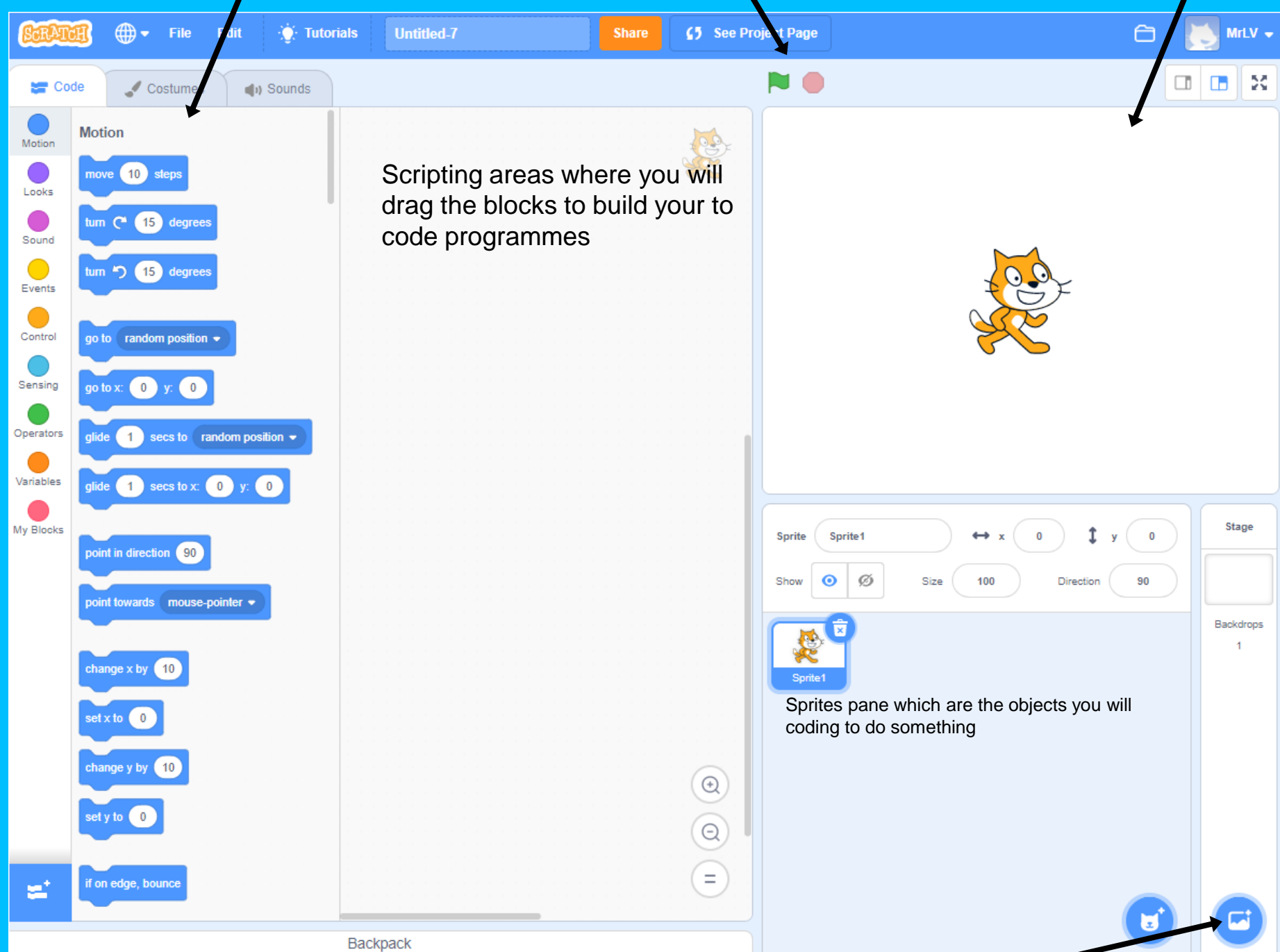
Create

# Computer Science

These are command BLOCKS which are grouped to make them easy to find.

How to start and stop your code

Staging area where your coding will come to life!



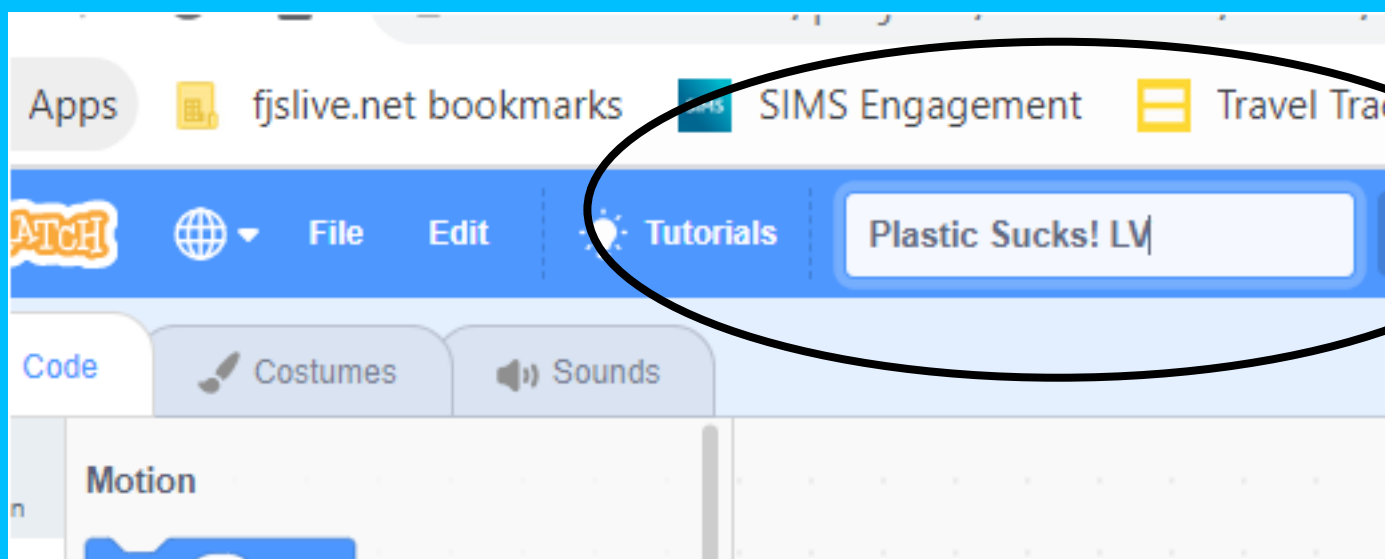
Make your code look great with a background (backdrop)



Name your  creation

# Computer Science

Change the name of the project



E-Safety: Make sure you do not use your whole name, just initials please.



# Creating Plan, make, assess

1. Select the "video sensing" extension.

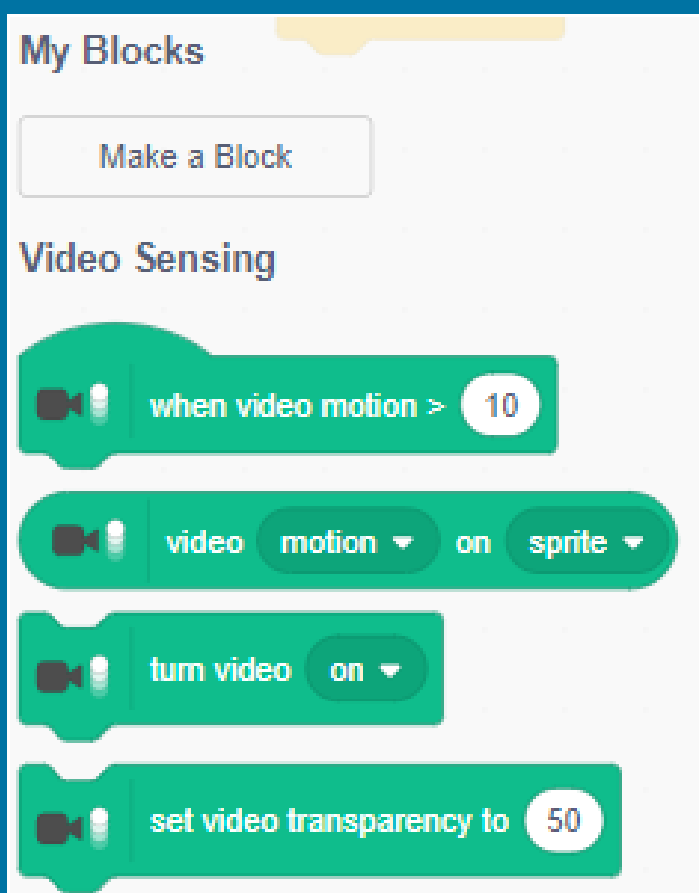
The screenshot shows the Scratch interface. On the left, the 'My Blocks' section is visible, with a red box highlighting the 'Video Sensing' icon. A red arrow points from this icon to the 'Choose an Extension' dialog box. The dialog box displays several extensions: Music, Pen, Video Sensing (highlighted with a red box), Text to Speech, Translate, and Makey Makey. The 'Video Sensing' extension is described as 'Sense motion with the camera.' and 'Requires' a camera.

2. "video sensing" is added to the bottom of the command blocks

The screenshot shows the 'My Blocks' section in Scratch. The 'Video Sensing' extension is added to the bottom of the command blocks. The blocks are: 'when video motion > 10', 'video motion on sprite', 'turn video on', and 'set video transparency to 50'.

# Creating Plan, make, assess

Video motion: This sets how sensitive the movements are. It goes from 0 where there is no movement on camera to 100 where the video senses a lot of movement.



Video motion on sprite: Sense the motion on the sprite so when the sprite is touched it moves.

Turn video on/ off: Command to turn the webcam on/off

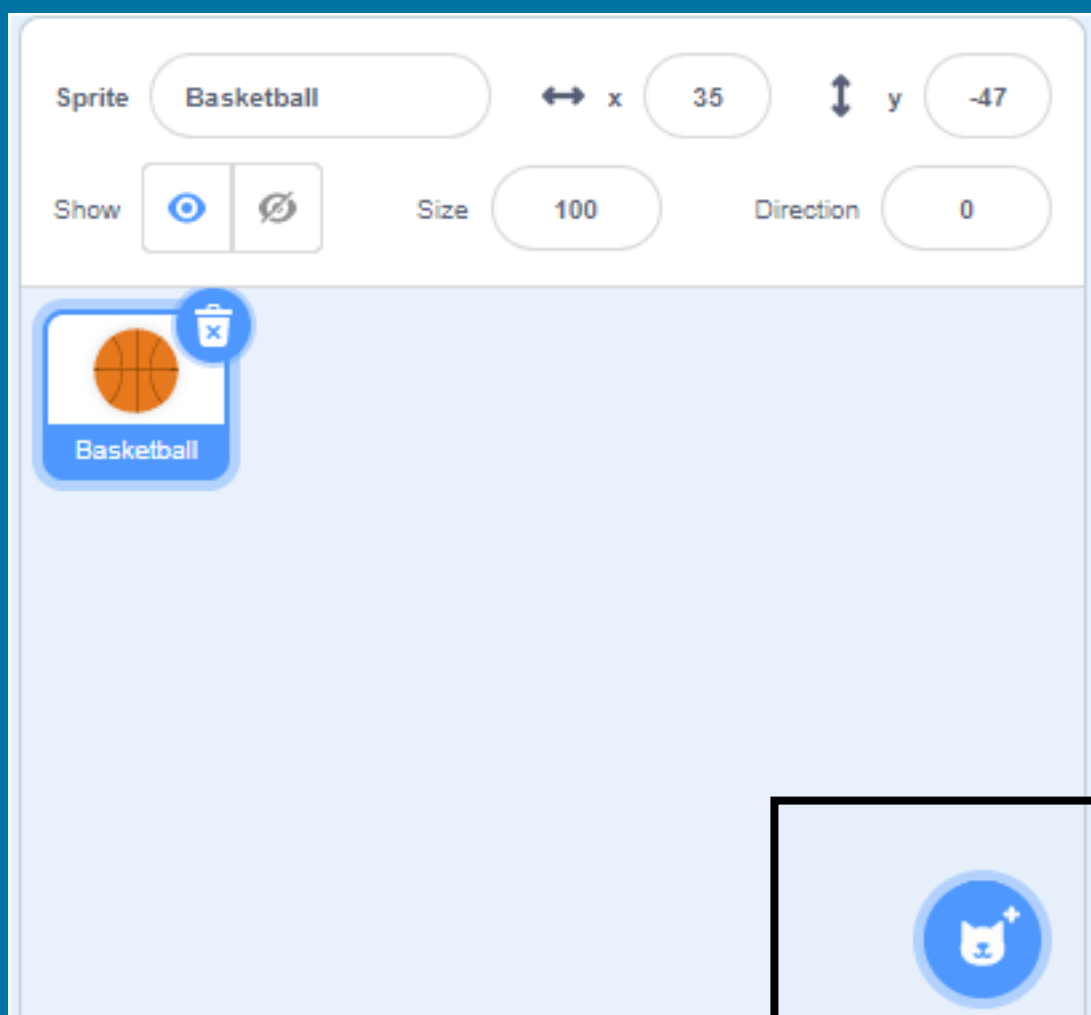
Transparency: The backdrop is set to white (0 = 100% transparent, 100 = white backdrop)

<https://www.youtube.com/watch?v=8vHEqVdWn08>



# Creating Plan, make, assess

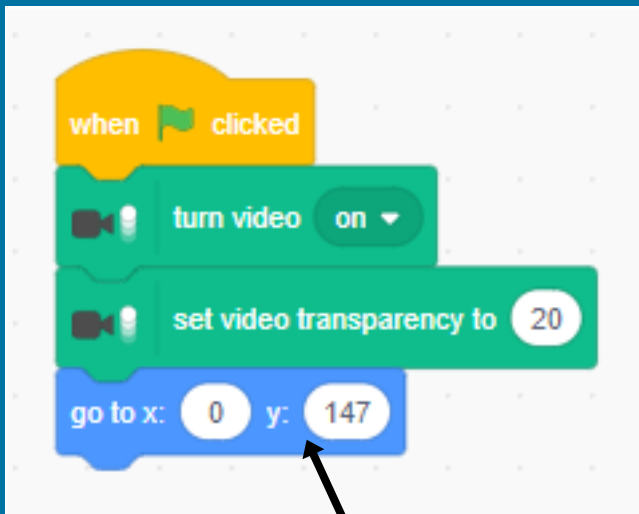
1. Add the sprites you need (you may want to create your own better sprites!). Place them on the backdrop.



Use the sprite search to select a basketball

Turn video sensing on

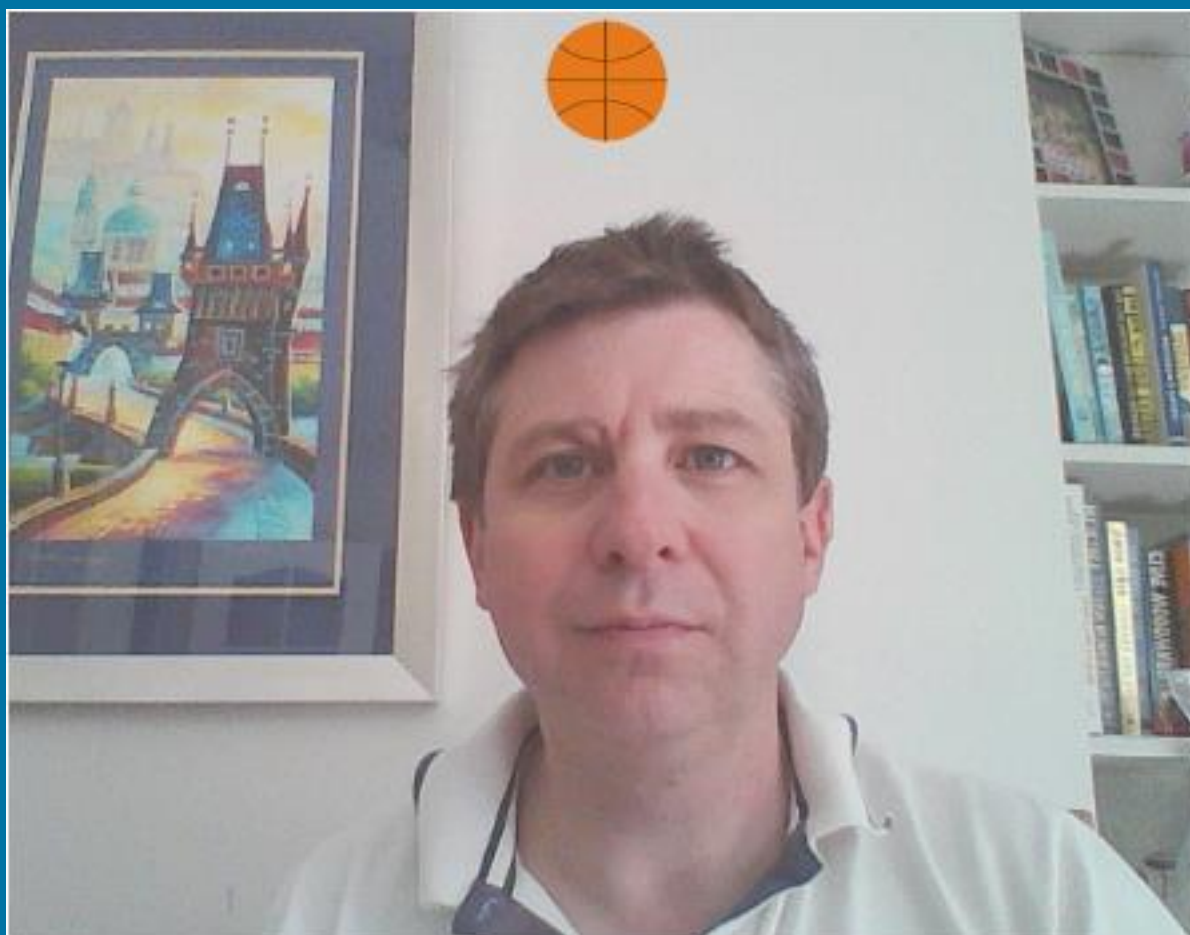
# Creating Plan, make, assess



1). Use the video sensing blocks to turn the video on.

2). Set the video transparency to 20 (tinker with this to see how it changes the video view).

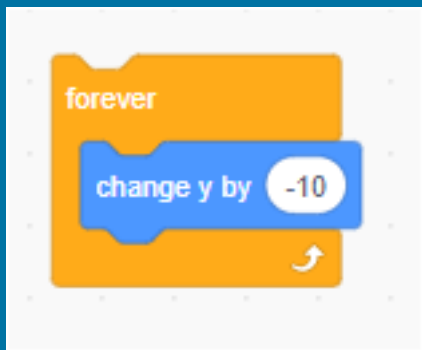
3). Position your sprite on the back drop.



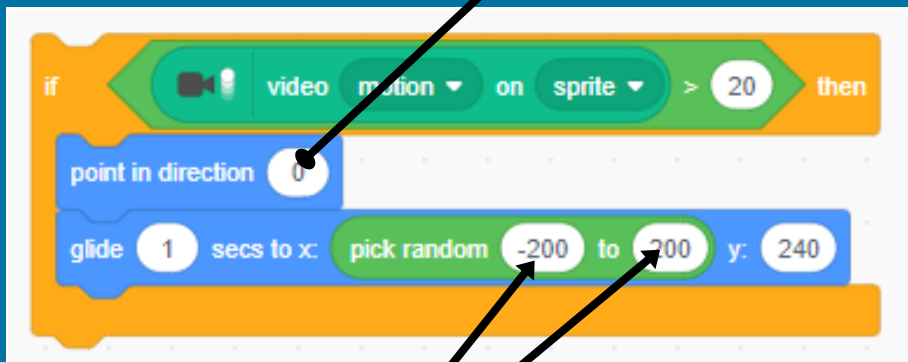
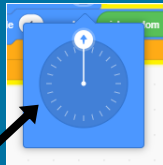


# Creating Plan, make, assess

Describe what the code is doing.



Makes the sprite fall 10 steps (y axis) continually (for ever)



When some motion (level 20) is detected on the sprite it will move in direction 0 (upwards).

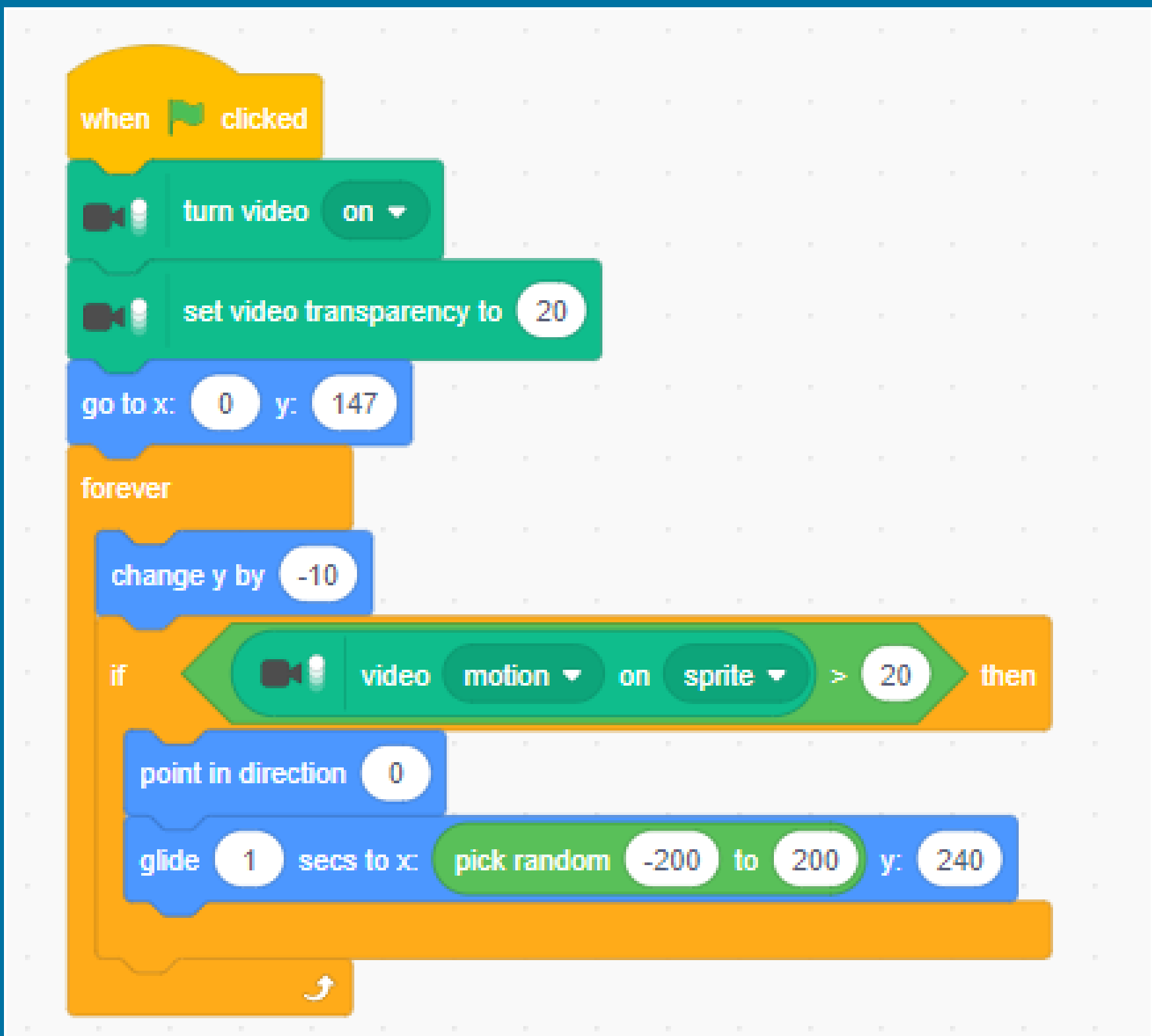
Glide to the top (y axis) to a random place on the x axis (horizontal).

The x and y axis goes from -240 to 240



# Creating Plan, make, assess

Put the code together like this...



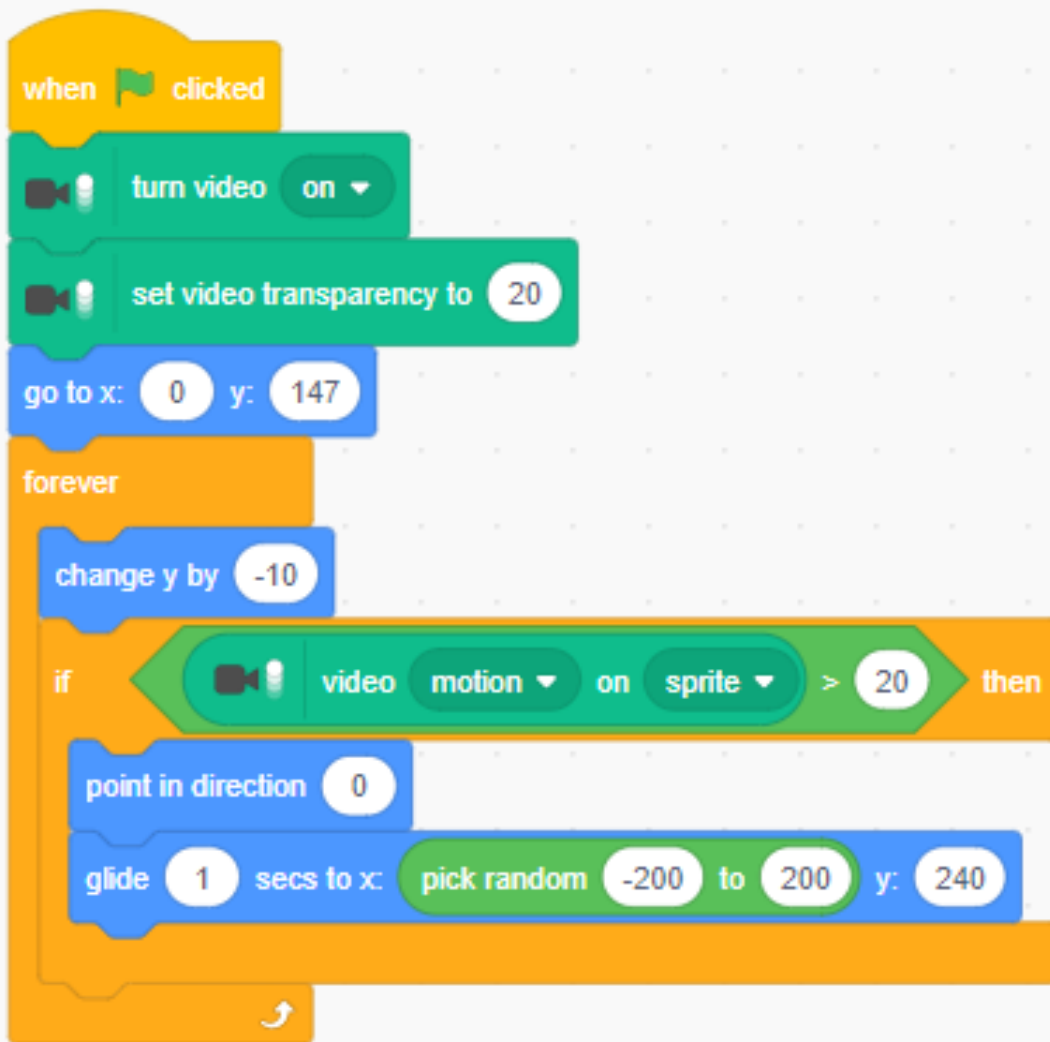
Play your keepy uppy game!



# Tinkering

## Try things out

Let's review our code.  
What do you think?



What can you change to see how the  
game could be improved?



# Tinkering

## Try things out

How can you improve the game?



Add a bounce sound  
Change colour



Add more sprites using the same code



Make sure you test your code EVERY TIME you make a change!  
Use the stop button to reset the programme







# Tinkering

## Try things out



Add a bounce sound  
Change colour

```
when clicked
  turn video on
  set video transparency to 20
  go to x: 0 y: 147
  forever
    change y by -10
    if video motion on sprite > 20 then
      start sound basketball bounce
      change color effect by 25
      point in direction 0
      glide 1 secs to x: pick random -200 to 200 y: 240
```

Add the start sound

Change colour of sprite

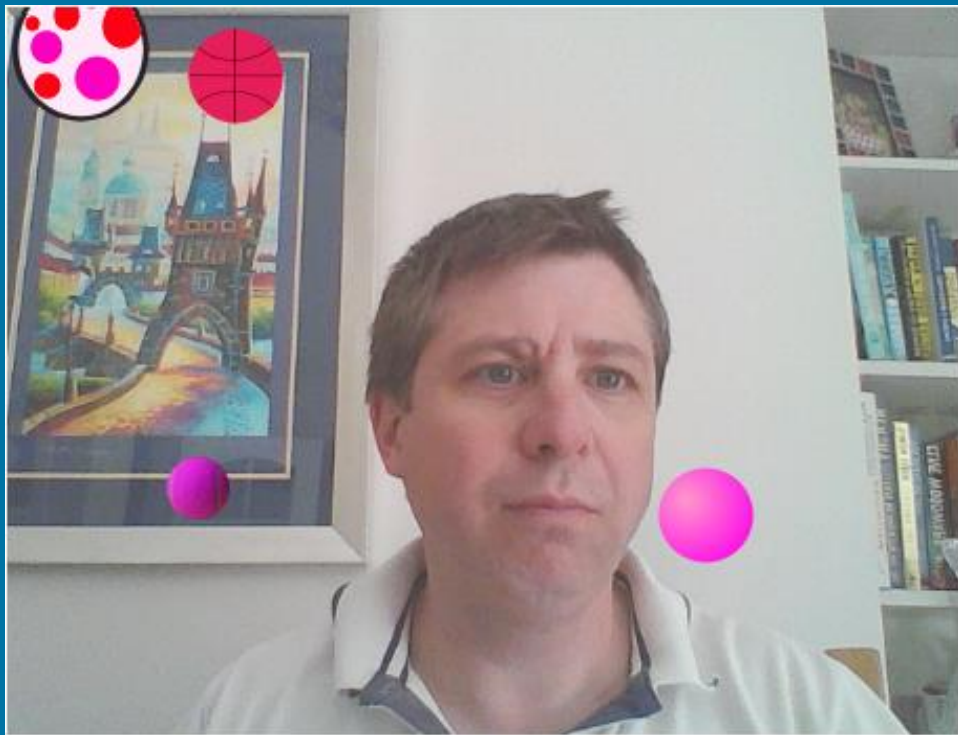


# Tinkering

## Try things out

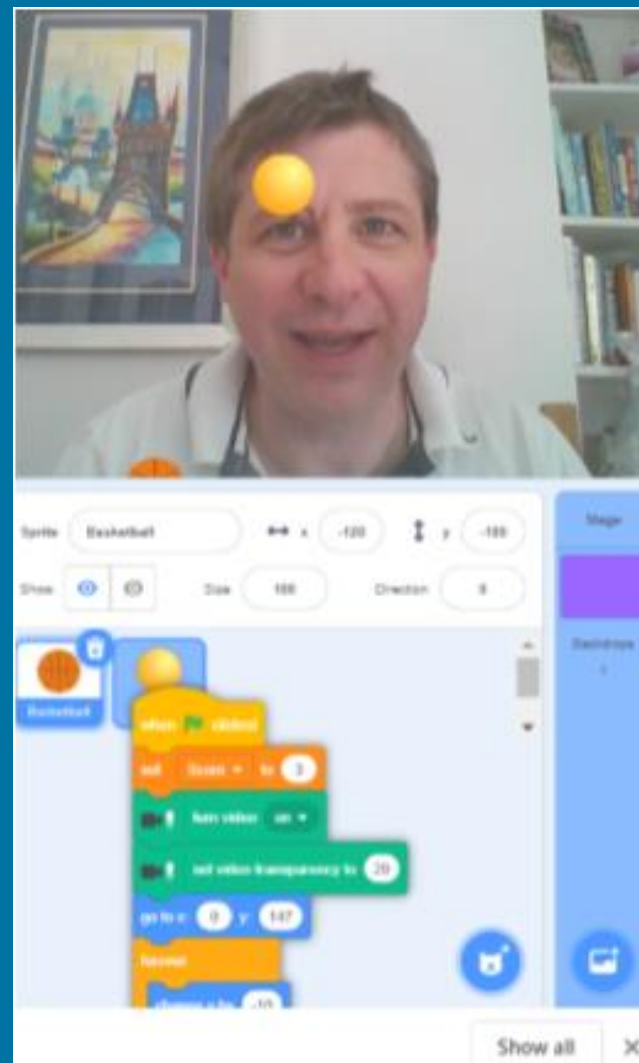


Add more sprites using the same code



1). Add new sprites

2). Drag and drop the code on to the sprite in the sprite area







# Tinkering

## Try things out

Use green pen to annotate your design with the new ideas for your programme.

Creating  
Plan, make, assess

*Draw and label what your programme will look like (how will the sprites move, what will they say, what sounds will be used).*

The purpose of the game is to keep the ball in the air.  
Use motion when there is contact with the sprite.  
Use a score variable.  
Use sound and change the colour of the sprite to make the game more interesting.

Tinkering  
Try things out

*Edit your design with green pen to show how you improved your design.*

# De-bugging

Identify and correct mistakes, evaluate

What have you had to de-bug today?

What skills do you need to "de-bug"?

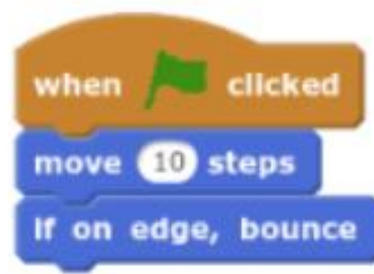


# De-bugging

## Identify and correct mistakes, evaluate

What's wrong with this code and how would you correct it?

Here is some code for a ball sprite:



Why will this code not result in the ball constantly bouncing from one side of the screen to the other?



- ☐ The 'if on edge, bounce' block is not needed.
- ☐ The code needs to be within a repeat block.
- ☐ The code needs to be within a forever block.
- ☐ It needs another 'move 10 steps' block after the 'if on edge bounce' block.

The code needs a forever block because the ball would only move 10 steps then stop.



Saving your project

# Computer Science

You do not need to save your project but if you are able to take a screen shot of your game, I would like to see it!