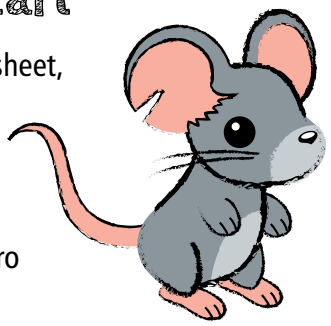
Cheese Chase 2

- Keyboard input
- Sprite movement
- Collision detection

60 mins

1. Before you start



- Download the first worksheet, "Pumpkin Smash", and follow the instructions for downloading the Thonny Python IDE and installing the Pygame Zero plugins.

2. Getting set up

- Create a new folder called "02 Cheese Chase"
- Within your new folder, create another new folder called "images"
- Download the graphics pack from this link: www.mr.langford.co/downloads/pgz-02.zip
- Un-zip the downloaded archive, then copy the images into your new images folder

3. Starting your program

This is the basis of our game program. It creates the game window, loads the sprite graphics for the cheese, cat, and mouse, and places them on screen.

```
import pgzrun
import random

# Generate random coordinates

# Draw screen
def draw():
    screen.clear()
    screen.fill((255, 255, 255))
    cheese.draw()
    mouse.draw()
    cat.draw()

# Set screen size
WIDTH = 800
HEIGHT = 600

# Create sprites
cheese = Actor('cheese')
mouse = Actor('mouse_r')
cat = Actor('cat_l')

# Start
pgzrun.go()
```

Import the Pygame Zero and Random libraries

The indents are important! Make sure yours are the same as shown. Press the TAB key to indent code blocks

Set the background colour to white.

Create three sprites: the cheese, the mouse, and the cat.

Make sure the graphics files are in your images folder as described in Step 2.

Start the program. This must always be your final line

How can you change the colour of the background?

Investigate using RGB (red, green, and blue) colour values to change the colour of the background area

After each step, save and run your program to ensure it works properly so far

4. Random positioning function

In the Pumpkin Smash game, we used the random library to generate x and y coordinates for the pumpkin. This time, we will be using it to generate coordinates for all three of our sprites.

Rather than repeat code for each of our sprites, it is more efficient to put the algorithm into a function and call it each time it is required.

Find the `# Generate random coordinates` comment, and add the following code to it:

Each function generates a random number between zero and the width or height of the window

Find this comment and add the rest of the code

```
# Generate random coordinates
def randPosX():
    posX = random.randint(0, WIDTH)
    return posX

def randPosY():
    posY = random.randint(0, HEIGHT)
    return posY
```

Return the random number back to the command which called it

5. Calling the random positioning functions

A function will not do anything until we call it.

Return to the main part of the program, and add the following three lines.

```
# Create sprites
cheese = Actor('cheese')
mouse = Actor('mouse_r')
cat = Actor('cat_l')
```

Existing code. Add your new underneath it

```
cheese.pos = randPosX(), randPosY()
mouse.pos = randPosX(), randPosY()
cat.pos = randPosX(), randPosY()
```

6. Moving the mouse

Within the `update()` function, add the following to move the mouse sprite using the keyboard's arrow keys.

```
# Keyboard controls
if keyboard.up:
    mouse.y -= 2
elif keyboard.down:
    mouse.y += 2
elif keyboard.left:
    mouse.x -= 2
elif keyboard.right:
    mouse.x += 2
```

The mouse sprite's x or y position is increased or decreased by 2, depending on which arrow key is pressed.

7. Chasing the mouse

Also within the `update()` function, add this code to move the cat towards the mouse sprite's position.

```
# Make cat follow mouse
if cat.x > mouse.x:
    cat.x -= 1
if cat.x < mouse.x:
    cat.x += 1
if cat.y > mouse.y:
    cat.y -= 1
if cat.y < mouse.y:
    cat.y += 1
```

The mouse sprite's position is checked, and the cat sprite's position is changed to move towards it

8. Detecting collision

When two sprites collide, it is called a **collision**. Pygame Zero has built-in collision detection.

Add the following code to the `update()` function:

If the mouse and cheese sprite collide, generate a new position for the cheese

```
# Collision detection
if mouse.colliderect(cheese):
    cheese.pos = randPosX(), randPosY()

if mouse.colliderect(cat):
    print("The cat got you! Game over!")
    exit()
```

If the mouse and the cat sprites collide, it's game over and the program ends!