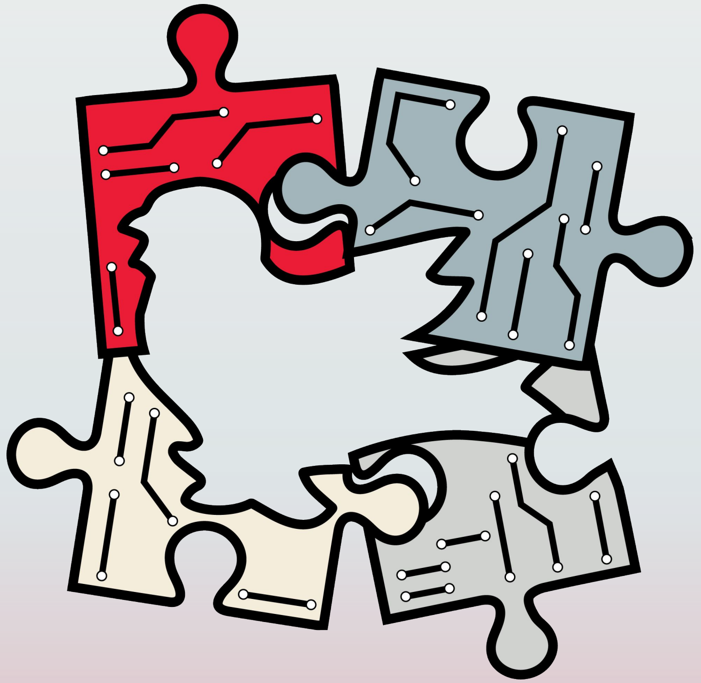
MCGILL PHYSICS

HACKATHON

Introduction to pygame

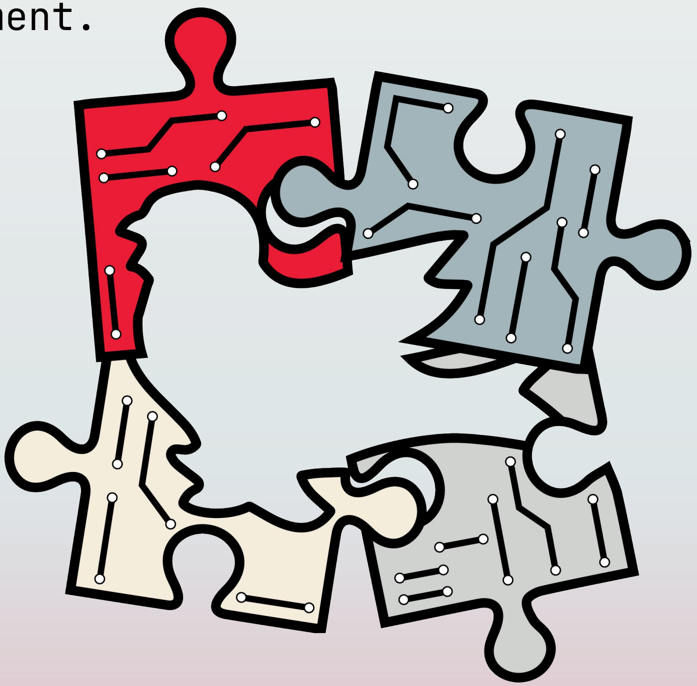


Simon Lavoie

February 5th, 2026

What is PyGame?

- NOT a game engine (no built-in physics).
- Python package.
- Provides *really* basic API for game development.
- Mostly places image on a screen.
- Renders on your CPU.
- Documentation found at https://www.pygame.org/docs/



First Steps

- Installed via pip

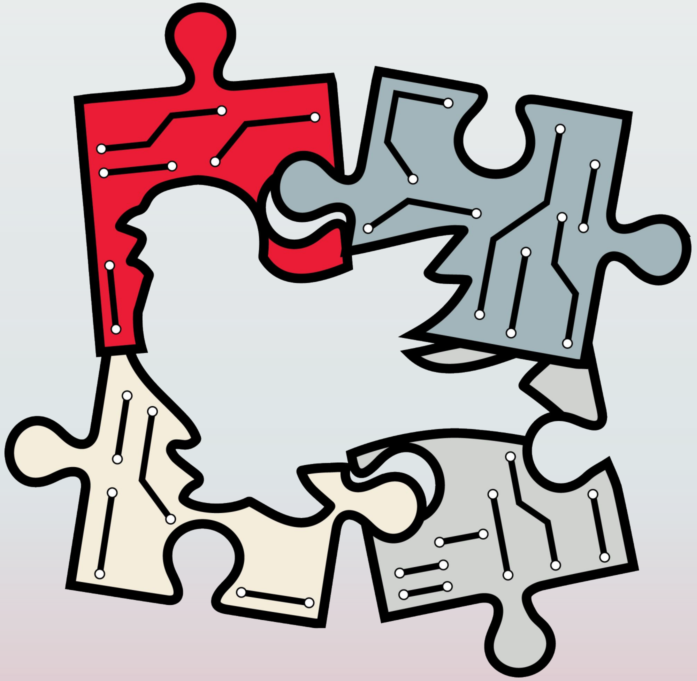
```
$ pip install pygame-ce
```

- Let’s create a window!

```
import pygame

pygame.init()
screen = pygame.display.set_mode((1280, 720))
```

- Now try to keep this window open...



MCGILL PHYSICS
HACKATHON

Simon Lavoie

The Game Loop

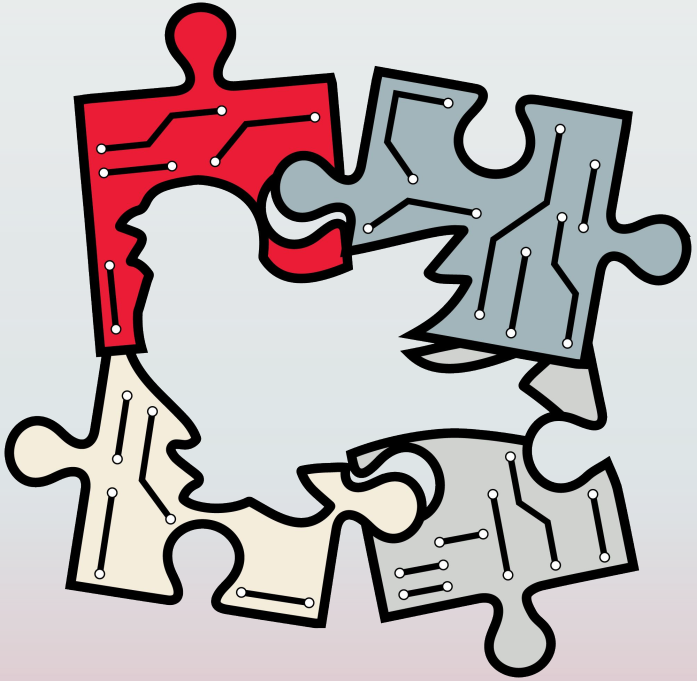
- This is a loop that runs continuously until the user quits the program.

```python
import pygame

pygame.init()
screen = pygame.display.set_mode((1280, 720))
running = True

while running:
    pass
```

- The above keeps the window open, but now it cannot close...



MCGILL PHYSICS HACKATHON

Simon Lavoie

Events

- Events listen in on certain inputs from the OS.

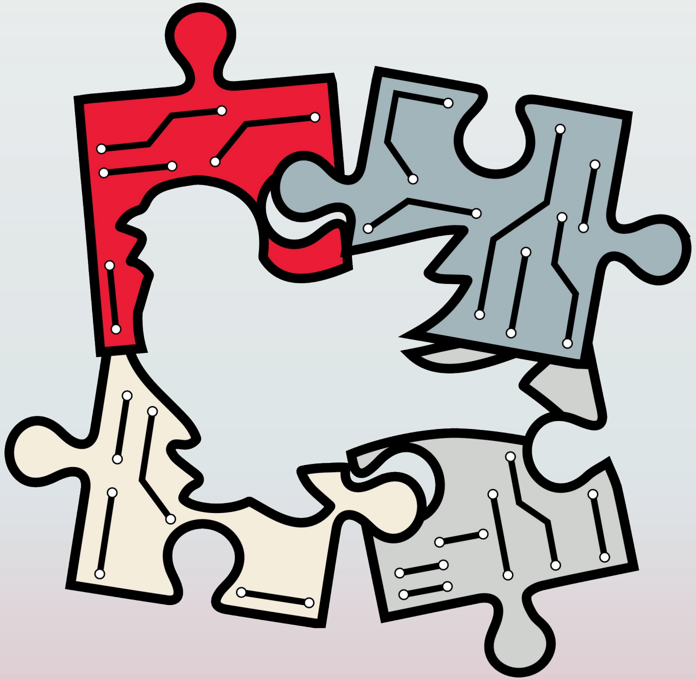
```python
import pygame

pygame.init()
screen = pygame.display.set_mode((1280, 720))
running = True

while running:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = False

pygame.quit()
```

Now let’s draw something to the screen!



MCGILL PHYSICS HACKATHON

Simon Lavoie

Drawing Images to Screen

```
image = pygame.image.load("Image.png").convert()
while running:
    screen.blit(image, (30, 30))
    pygame.display.flip()
```

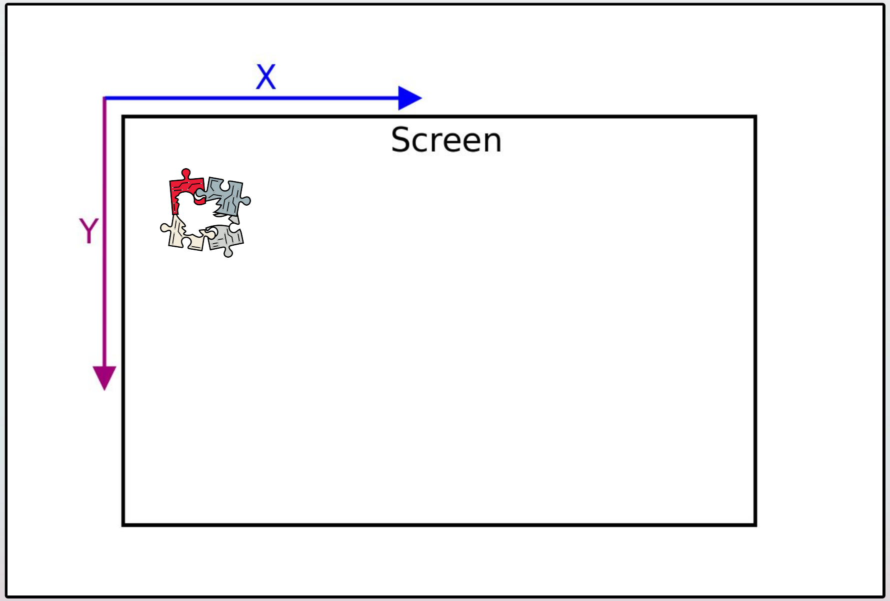
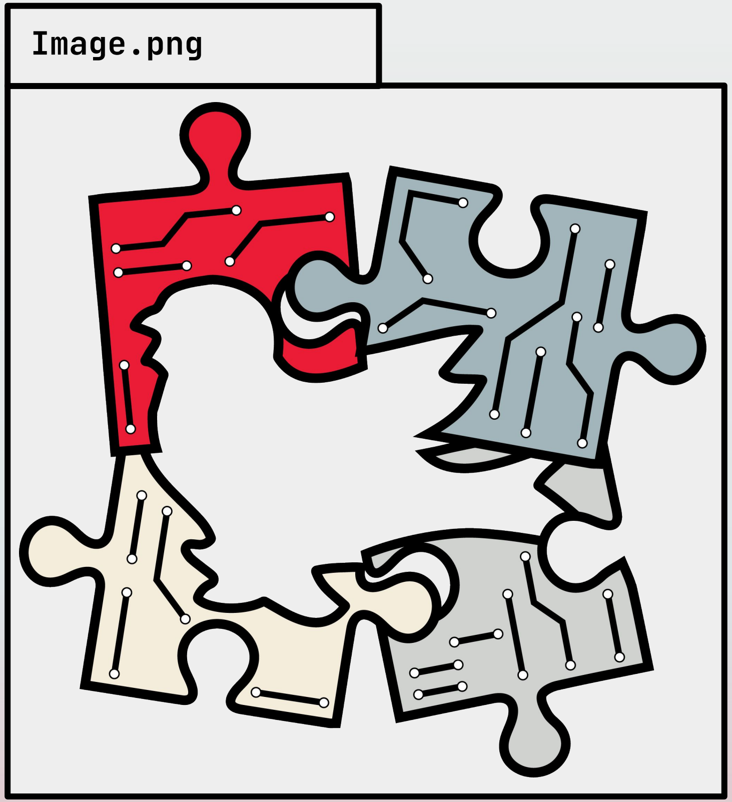
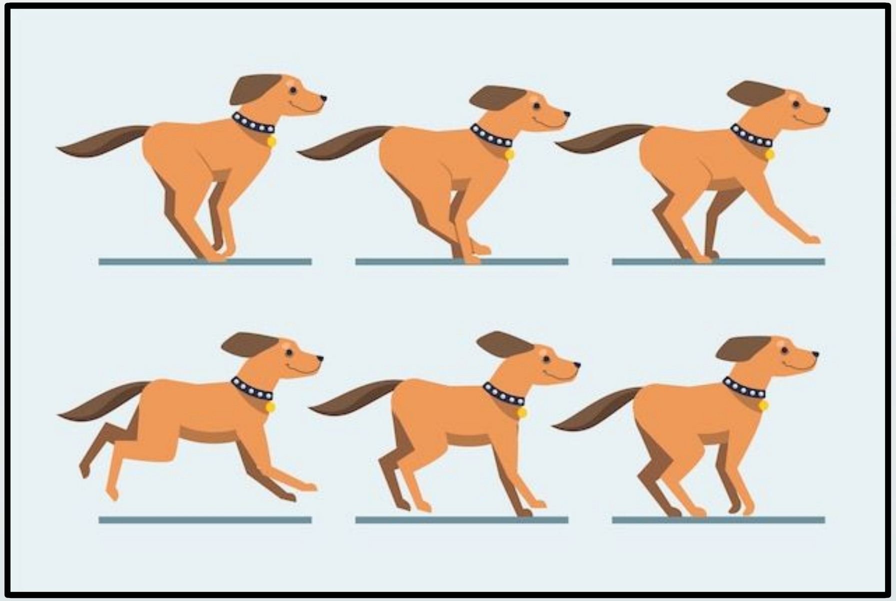


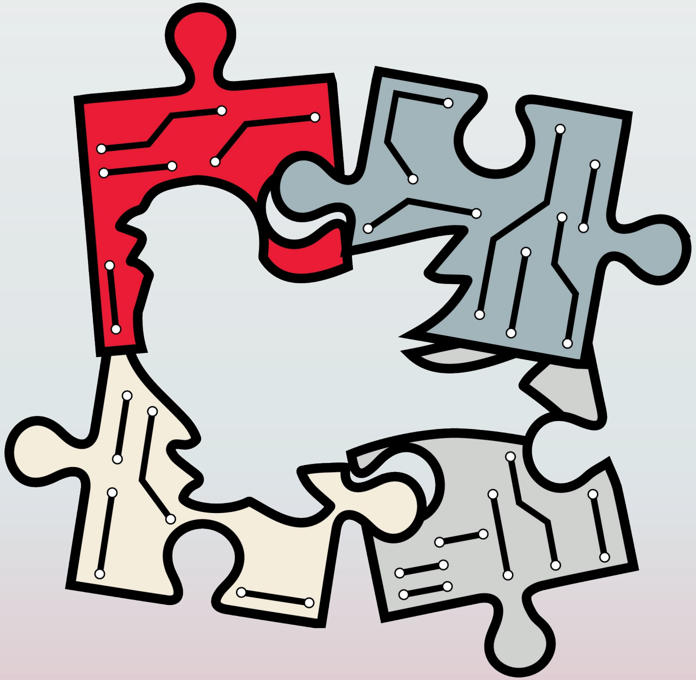
Image.png



Moving Pictures



- Each game loop iteration is a frame!



Timing

- Let’s *loosely* cap each loop iteration at 60 fps.

```python
import pygame

pygame.init()
...
clock = pygame.time.Clock()
dt = 0

while running:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = False
...
    dt = clock.tick(60) / 1000
pygame.quit()
```

- 60 fps = 16.67 ms/frame
- clock.tick(60) returns the difference in time since the last time it was called.
- Say your iteration loop took 4ms to render, then clock.tick(60) will wait the extra 16.67 - 4 = 12.67 ms before looping.
- If it took longer than 16.67ms, it does not need to wait.
- This means you can use dt for physics calculations!

Input

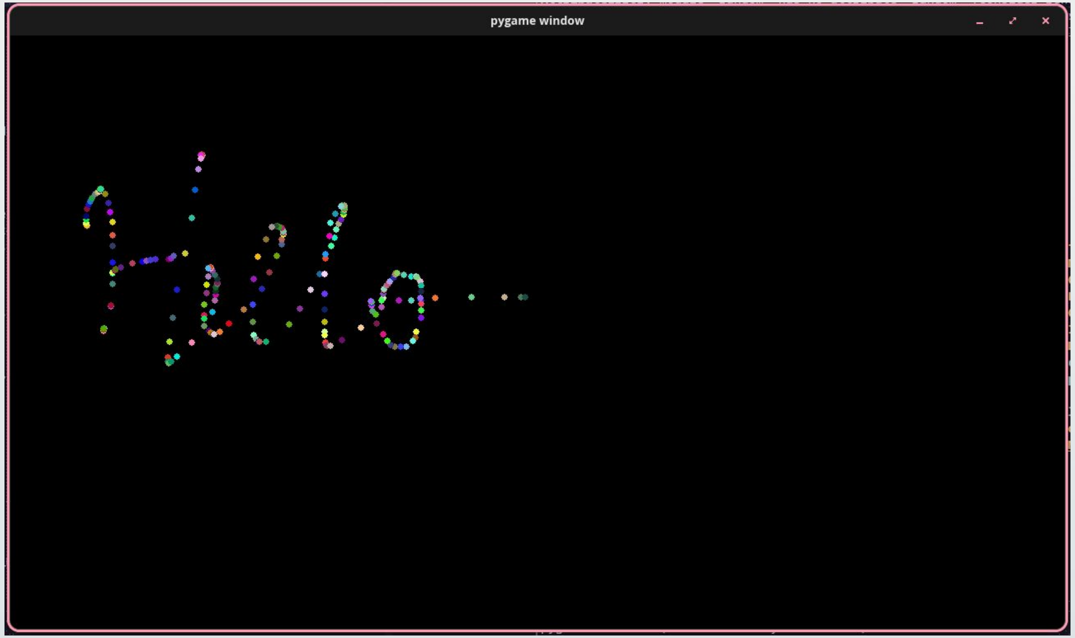
```python
pygame.mouse.get_pressed(3)
pygame.mouse.get_pos()
pygame.mouse.get_rel()
```

```python
pygame.mouse.get_pressed()
```

Name	Key
K_1	1
K_2	2
K_3	3
K_a	a
K_b	b
K_c	c
.	.
.	.
.	.

Simon Lavoie

Putting these together



Let me demo my falling confetti “game”