Lecture: 8-1

- Prerequisites for this lecture are: 7-1, 7-2 and 7-3.

John Romero Programming Proverbs

- 7. “Use a development system that is superior to your target.”

- John Romero, “The Early Days of Id Software - John Romero @ WeAreDevelopers Conference 2017”

Sprites and collisions

- you will need to save these images in the same directory as the code:
  - gun.png ⟨`ball.png`⟩
  - arrow.png ⟨`ball.png`⟩
  - ball.png ⟨`ball.png`⟩

Sprites and collisions

- sprites are created and normally placed into a list
  - and referred to as a group
- you can then test for a collision between another sprite via:
- ```
  inter = spritecollide(foo, bar, dokill)
  ```
- `inter` is a list of all sprites from list `bar` which have collided with the single sprite `foo`
- the `dokill` parameter is either `True` or `False` and if it was `True` the the `kill` method is called for every sprite in the list `inter`
  ```

Sprites and collisions

```
for bomb in sprite.spritecollide(player, bombs, True):
    boom_sound.play()
```

- notice that this example tests whether a single sprite `player` has collided with any sprite in the `bombs` list

Managing collisions between two groups of sprites

- we can detect whether a collision occurs between two groups of sprites by using the following function:

- ```
  groupcollide(list1, list2, dokill1, dokill2)
  ```

- this function returns a dictionary
  - each key in the dictionary is a sprite in `list1` and its value is a list of sprites from `list2` with which it has collided
  - the `dokill1`, `dokill2` arguments determine whether the `kill` method should be called in `list1` or `list2`
  ```

Managing collisions between two groups of sprites

```
for alien in sprite.groupcollide(aliens, bullets, True, True).keys()
   boom_sound.play()
   kills += 1
```

- the code checks for the collisions between bullets and all the aliens
- in this case we only loop over the dictionary keys
  - but we could loop over the `values()` or `items()`
  - if we wanted to do something to the specific shots that collided with aliens

Managing collisions between two groups of sprites

- if we did loop over the `values()` we would be looping through lists that contain sprites

- note that the same sprite may even appear more than once in these different loops, since the same `bullet` could have collided against multiple `aliens`