Pygame Framework

- Object-Oriented Analsis and Design
  - (CSCI 4448/6448)
- Ryan Stout
- Jon Meador
- Marcus Seeland
- Peter Robinson

Introduction

- Pygame is a framework which allows for easy manipulation of common game elements like:
  - Backgrounds
  - Sprites
  - Actions and Movements
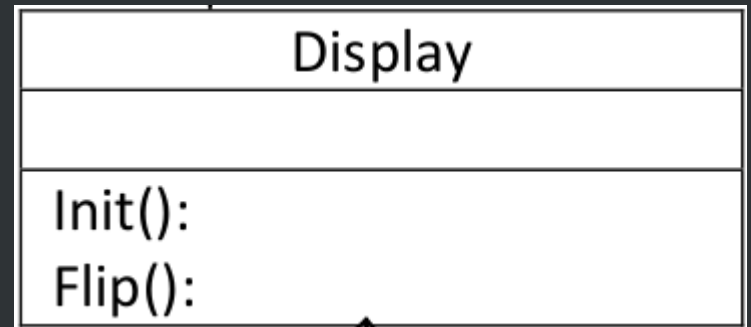 - Sounds and Music
  - Event Handling

Main Services

- Display
- Surface
- Sprite
- Group
- Rect
- Mixer
- Pygame

Display

- Display is what you see and the actual call to show the image on the screen

  - Init

    - Initializes the display variables

  - Flip

    - Replaces the current screen by the off screen buffer

Display
Init(): Flip():

Surface

- Surface is one of the primary services that you interact with when using pygame. Surface is used to represent an image or combination of images on the screen.

Surface
height: int
width: int
blit(Surface,(int,int)): Rect
set_colorkey((int,int,int)):

Sprite

- Sprite is also used often in game development to represent the actors

- It is a base class to represent game objects

- It comes with a great set of built in methods which take care of common interactions with the game environment

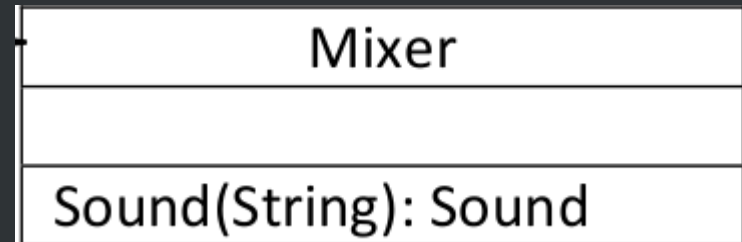| sprite |
| --- |
| |
| add(*groups):
remove(*groups):
update(*args)
kill():
alive():bool |

Group

- Group is a container class for sprites.

- It is very helpful for manipulating large sets of sprites

- It will define more specific behaviors for groups of sprites.

Group
add(*sprite):
remove(*sprite):
has(*sprite)
draw(Surface)
update(*args)

Mixer

- Mixer is the way the programmer implements sound effects and music within the game.

- It has all the methods used to control the sounds in the game.

Mixer
Sound(String): Sound

Rect

- Rect is the lower level representation of objects like sprites.

- It defines and controls specific areas on the screens

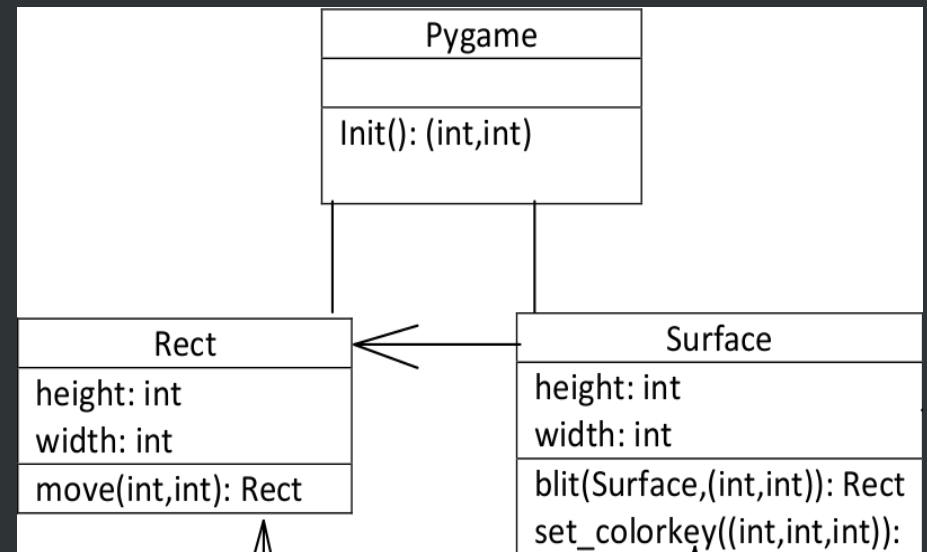| Rect |
| --- |
| height: int |
| width: int |
| move(int,int): Rect |

Pygame

- Pygame basically is only used for the Init() method which initializes all of the imported pygame modules.

- It is also capable of handling errors.

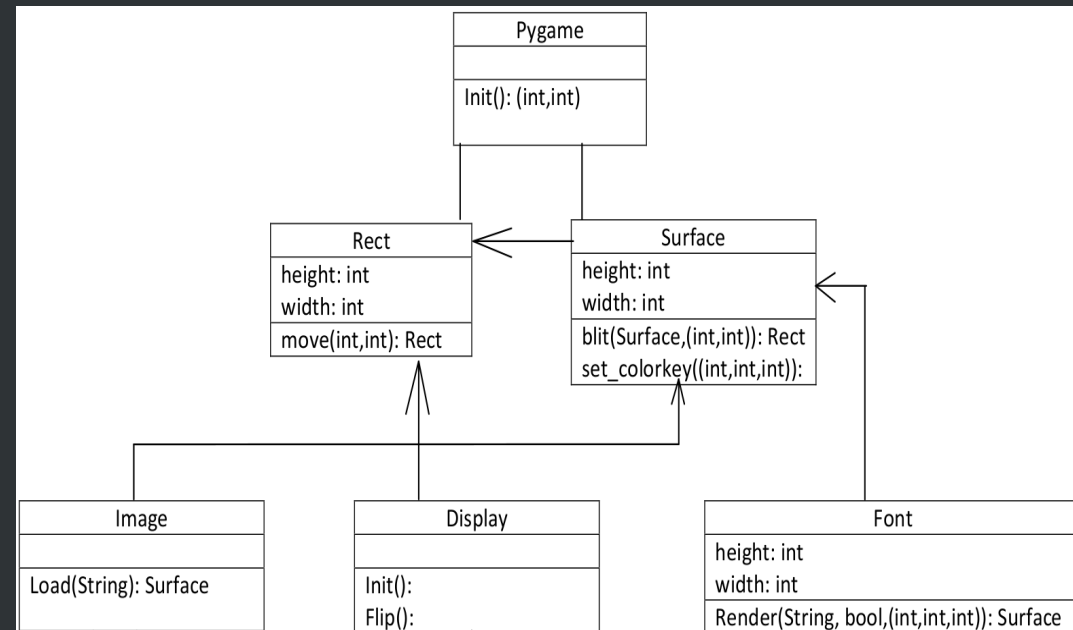| Pygame |
| --- |
| |
| Init(): (int,int) |

Main Concepts

- Pygame is the top-level package which is broken up by several submodules which perform the various services.

- Rect and Surface are examples of classes which create these submodules.
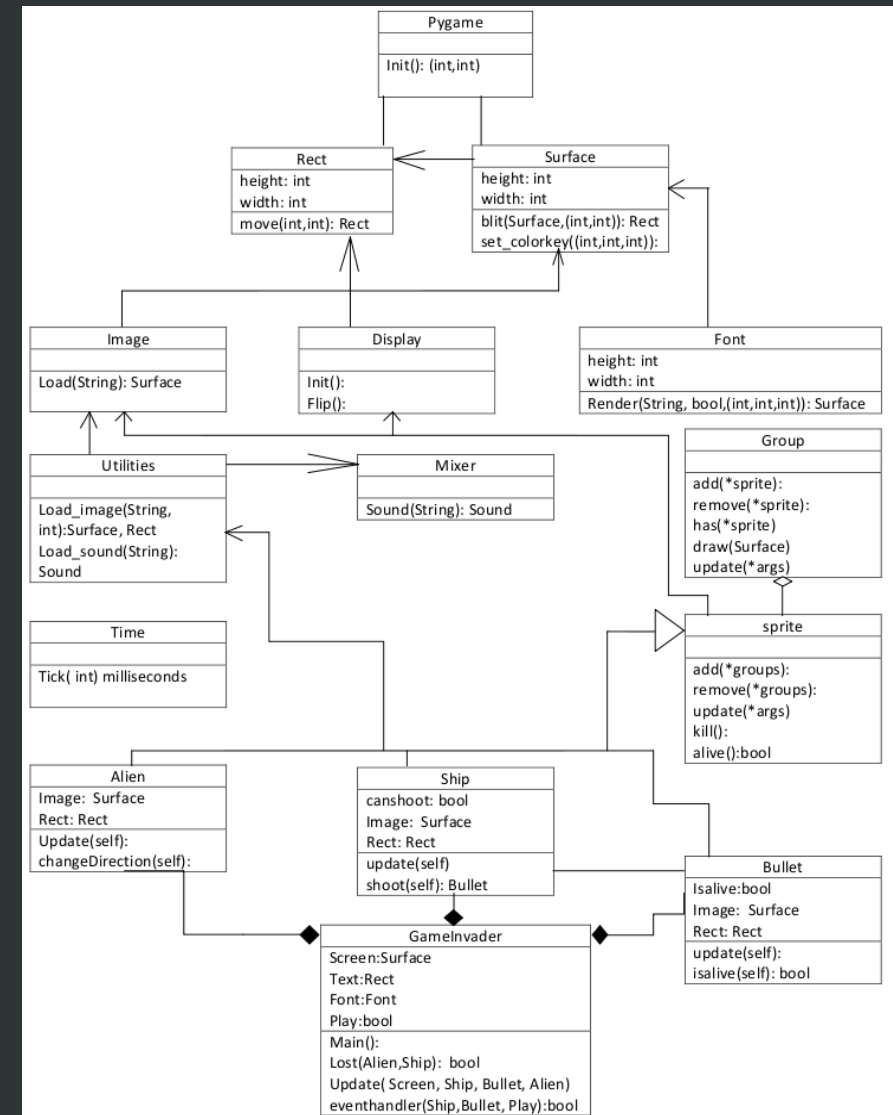
Main Concepts (Cont)

- There are several classes like Display and Image and Font which create submodules of the lower-level submodules like Rect and Surface
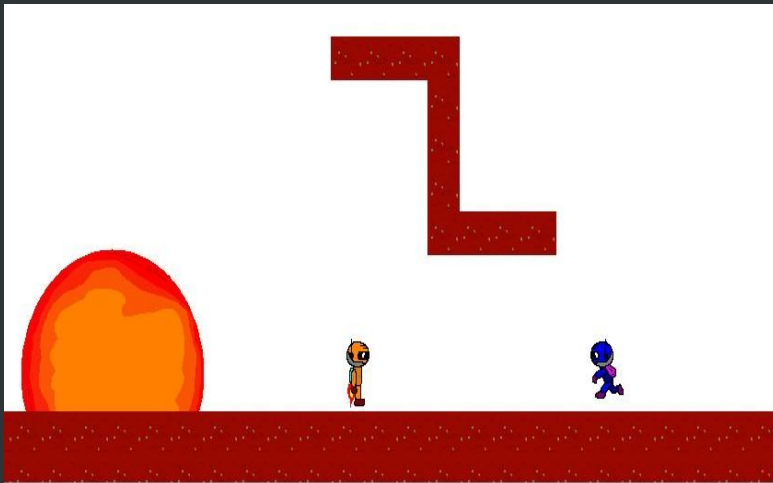
Main Concepts (Cont)

- This is how the Invader Game demo is constructed using the Pygame framework

Demos

- Here are some of the games we created using PyGame!

Any Questions?