

# CS 250

## Mouse and Keyboard Input with SDL

Spring 2016

# Linker Warning

SDL\_MouseKeyboardLab Property Pages

Configuration: Active(Debug) Platform: Active(Win32)

- Common Properties
- Configuration Properties
  - General
  - Debugging
  - VC++ Directories
  - C/C++
  - Linker
    - General
    - Input
    - Manifest File
    - Debugging
    - System
    - Optimization

Additional Dependencies	SDL2_mixer.lib;SDL2.lib;SDL2main.lib;SDL2_g
Ignore All Default Libraries	
Ignore Specific Default Libraries	<b>msvcrt.lib;%(IgnoreSpecificDefaultLibraries)</b>
Module Definition File	
Add Module to Assembly	
Embed Managed Resource File	
Force Symbol References	
Delay Loaded DLLs	
Assembly Link Resource	

**msvcrt.lib**

# Topics

---

- Sprites (images)
- Mouse Input
- Keyboard Input
- <http://lazyfoo.net/tutorials/SDL/index.php>

# Sprites

---

- Add `SDL2_image.lib` as a dependency
- `#include "SDL_image.h"`
- Image formats
  - BMP, GIF, JPEG, LBM, PCX, PNG, PNM, TGA, TIFF, WEBP, XCF, XPM, XV
- [https://www.libsdl.org/projects/SDL\\_image/](https://www.libsdl.org/projects/SDL_image/)

# Code – create sprite

```
// load file
SDL_Rect rect;
SDL_Texture *pTexture = IMG_LoadTexture(pRenderer,
                                       "images\\logo.png");

// find the width and height of the image
SDL_QueryTexture(pTexture, NULL, NULL, &rect.w, &rect.h);

//set the location of the image
rect.x = 100;
rect.y = 100;

SDL_DestroyTexture(pTexture); // after the animation loop
```

# Code - render

---

```
// render the image on to the screen.  
SDL_RenderCopy(pRenderer, pTexture, NULL, &rect);  
  
// rect contains the location to render at
```

# Event Driven Programming

---

- Wait for an event (mouse click, keyboard press, etc) to occur and react to that event.
- Inside of the animation loop

# Grab a Mouse Click

---

```
SDL_Event sdlEvent;
int x, y;
if (SDL_PollEvent(&sdlEvent) != 0)
{
    if (SDL_MOUSEBUTTONDOWN == sdlEvent.type)
    {
        SDL_GetMouseState(&x, &y);
    }
}
```

[https://wiki.libsdl.org/SDL\\_EventType](https://wiki.libsdl.org/SDL_EventType)

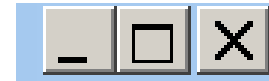


# Get a Keyboard Event

---

```
if (SDL_PollEvent(&sd1Event) != 0)
{
    // https://wiki.libsdl.org/SDL\_Scancode
    if (SDL_KEYDOWN == sd1Event.type )
    {
        switch (sd1Event.key.keysym.scancode)
        {
            case SDL_SCANCODE_UP: // up arrow
                rect.y -= 2;
                break;
        }
    }
}
```

# Quit



```
if (SDL_PollEvent(&sd1Event) != 0)
{
    if (SDL_QUIT == sd1Event.type)
    {
        // cause the animation loop
        // to stop.
        bIsRunning = false;
    }
}
```