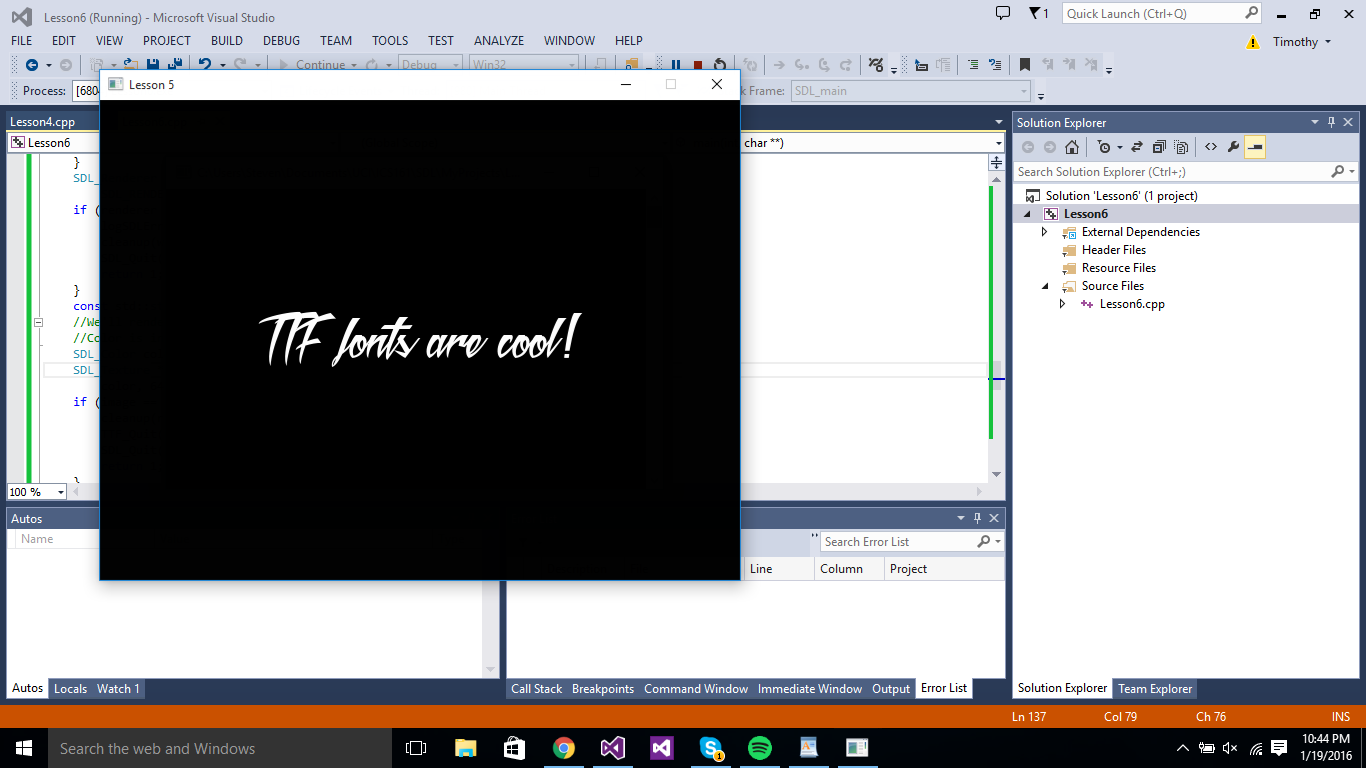
Timothy Pranoto  
38964311  
ICS161 – HW2

//From Lesson4.cpp  
//Our event structure

SDL\_Event e;  
bool quit = false;  
while (!quit){  
 --x;  
 while (SDL\_PollEvent(&e)){  
 if (e.type == SDL\_QUIT){  
 quit = true;  
 }  
 if (e.type == SDL\_KEYDOWN){  
 if (e.key.keysym.sym == SDLK\_RIGHT)  
 x+=4;  
 //quit = true;  
 }  
 if (e.type == SDL\_MOUSEBUTTONDOWN){  
 quit = true;  
 }  
 }  
 //Render the scene  
 SDL\_RenderClear(renderer);  
 renderTexture(image, renderer, x, y);  
 SDL\_RenderPresent(renderer);  
}

//From Lesson5.cpp  
int main(int, char\*\*)  
{  
 if (SDL\_Init(SDL\_INIT\_EVERYTHING) != 0){  
 logSDLError(std::cout, "SDL\_Init");  
 return 1;  
 }  
 if ((IMG\_Init(IMG\_INIT\_PNG) & IMG\_INIT\_PNG) != IMG\_INIT\_PNG){  
 logSDLError(std::cout, "IMG\_Init");  
 SDL\_Quit();  
 return 1;  
 }  
 SDL\_Window \*window = SDL\_CreateWindow("Lesson 5", 100, 100, SCREEN\_WIDTH,  
 SCREEN\_HEIGHT, SDL\_WINDOW\_SHOWN);  
 if (window == nullptr){  
 logSDLError(std::cout, "CreateWindow");  
 SDL\_Quit();  
 return 1;  
 }  
 SDL\_Renderer \*renderer = SDL\_CreateRenderer(window, -1,  
 SDL\_RENDERER\_ACCELERATED | SDL\_RENDERER\_PRESENTVSYNC);  
 if (renderer == nullptr){  
 logSDLError(std::cout, "CreateRenderer");  
 cleanup(window);  
 SDL\_Quit();  
 return 1;  
 }  
 const std::string resPath = getResourcePath("Lesson5");  
 SDL\_Texture \*image = loadTexture(resPath + "image.png", renderer);  
 //Make sure they both loaded ok  
 if (image == nullptr){  
 cleanup(image, renderer, window);  
 IMG\_Quit();  
 SDL\_Quit();  
 return 1;  
 }  
 //iW and iH are the clip width and height  
 //We'll be drawing only clips so get a center position for the w/h of a clip  
 int iW = 100, iH = 100;  
 int x = SCREEN\_WIDTH / 2 - iW / 2;  
 int y = SCREEN\_HEIGHT / 2 - iH / 2;  
 //Setup the clips for our image  
 SDL\_Rect clips[4];  
 for (int i = 0; i < 4; ++i){  
 clips[i].x = i / 2 \* iW;  
 clips[i].y = i % 2 \* iH;  
 clips[i].w = iW;  
 clips[i].h = iH;  
 }  
 //Specify a default clip to start with  
 int useClip = 0;  
 //Our event structure  
 SDL\_Event e;  
 bool quit = false;  
 while (!quit){  
 while (SDL\_PollEvent(&e)){  
 if (e.type == SDL\_QUIT)  
 quit = true;  
 //Use number input to select which clip should be drawn  
 if (e.type == SDL\_KEYDOWN){  
 switch (e.key.keysym.sym){  
 case SDLK\_1:  
 useClip = 0;  
 break;  
 case SDLK\_2:  
 useClip = 1;  
 break;  
 case SDLK\_3:  
 useClip = 2;  
 break;  
 case SDLK\_4:  
 useClip = 3;  
 break;  
 case SDLK\_ESCAPE:  
 quit = true;  
 break;  
 default:  
 break;  
 }  
 }  
 }  
 //Render the scene  
 SDL\_RenderClear(renderer);  
 renderTexture(image, renderer, x, y, &clips[useClip]);  
 SDL\_RenderPresent(renderer);  
 }  
 cleanup(image, renderer, window);  
 IMG\_Quit();  
 SDL\_Quit();  
}

From Lesson 6  
<http://www.1001freefonts.com/mad_rats.font>  
mad rats ttf  


From LazyFoo22

