PLEASE NOTE I WILL ONLY WRITE WHAT YOU NEED TO DO AND EXPLAIN WHAT IT IS, YOU NEED TO IMPLEMENT IT INTO YOUR PROJECT BY YOURSELF, FOR ANYTHING NEED HELP, COMMIT YOUR CODE TO YOUR GITHUB AND PING ME ON SKYPE ( dieptt2807 ) – PHONE ( 094.70.24.363 )

*DEADLINE: End of day* ***Thursday, 27th June 2019***

* We will add object to hold Texture and Scene in this week. First, open your project from last week, try to improve the existing code by split to as much functions as possible. Some note to do:
  + IMG\_Quit(); // this one is used to free IMG lib ( same as SDL\_Quit() )
  + function LoadFromFile:
    - use to load an image to a surface in SDL then create SDL\_Texture from it.
  + Instead of using x, y to asbtract the position, we should create a class to hold the position. In here we call Vector2D class
    - The position class should include x, y as int
    - Constructor default & Constructor non-default
* =======> Result: expected to load your image using **Vector2D** class, use function **LoadFromFile** in source.cpp, and use **IMG\_Quit** also in there.
* Next, based on it, we define a class to hold a Texture by ourself, instead of using SDL\_Texture which is belong to the library. You will notice the benefit of write this class is you don’t have to rewrite an load functions & clean up texture many time in each class we use Texture
  + class name: **Texture2D**
  + class properties:
    - SDL\_Renderer\* => pointer renderer, we need it because when we draw a texture, SDL require a renderer pass from source
    - SDL\_Texture\* => texture pointer, this is the main thing, the texture we will use to draw
    - width & height => every image will have width and height
  + class prototype functions:
    - Constructor default
    - Constructor non-default
    - Destructor => to delete our object Texture2D
    - Render => to draw texture
    - Get & Set => to set & get width height
    - LoadFromFile => to load image from file then assign it to texture pointer member of this class
* ===========> After you finish write this class, try to use it in source.cpp, to replace **SDL\_Texture\* g\_Texture** by **Texture2D\* g\_Texture.**
* Now I assume you can load your image by using Texture2D class, or even better, use keyboard to rorate, moving image by using Texture2D & Vector2D class wrote by yourself.

What we will do next is create something to hold a Scene. What is a Scene ? Scene is like a level, a Menu in your game. For example, About Menu ( About Scene ), Level1 Menu, Title Menu .... you will notice there are a lot of Scene needed for your full game. So we will create object to hold a Scene, and then create 2 scene ( intro & level1 ), each scene load their own background image, and the we change scene by press left-right arrow keyboard.

In order to do that, will have some classes like SceneIntro, SceneLevel1, SceneLevel2, SceneLevel ..., SceneTitle ... You will notice they are all Scene => we should find something similar between all scene, find common things between all those Scene.

Let’s say, all scene need:

* + - A Renderer ( to render anything to it )
    - A Render function
    - An Update funciton
* Create Class Scene:
  + properties:
    - SDL\_Renderer\* // it will be shared to its child => should be in protected, not private
  + functions:
    - Constructor default
    - Constructor non-default
    - Destructor
    - Render
    - Update
* Create class SceneIntro inherit from class Scene, this class is to show logo, contain all function & properties which parent let him to use ( in protected & public )
  + properties:
    - Texture2D\* // background texture
  + functions:
    - Constructor default
    - Constructor non-default
    - Destructor
    - Render
    - Update
* Create class SceneLevel1 inherit from class Scene, this class is to play the level 1 gameplay
  + properties:
    - Texture2D\* // background texture
  + functions:
    - Constructor default
    - Constructor non-default
    - Destructor
    - Render
    - Update
    - SetupLevel // later, everything we add to level 1 like mario, luigi, enemy... will be put here
* Now we already had object for hold data of SceneIntro & SceneLevel1, let’s use them in Source.cpp, then do something to change scene when press left-right arrow
* My suggestion is using something like SceneManager class, in here we init all the Scene needed, and just load the proper scene when needed.