Questions and Exercises

These questions and exercises is an opportunity to see what you've learnt from the lecture as well as practice the new things we've been talking about. In other words, these questions and exercises are completely optional but it's recomended to do them. In the end of the document you will find the answers to the questions as well as possible solutions to the exercises, note that one can solve an exercise in different ways. There will also be some suggestions about what one could code if one want to continue with some more advanced things. These suggestions will not come with a possible solution and might include things that haven't been covered in the lecture.

Question 1
If you have an entity and a model what do you need to do to connect them together to allow the entity to be rendered like the model?

Question 2
To make a model one has to work with a ModelBase, ModelRenderers and ModelBoxes. What's the relation between these?

Question 3
What do you need to do to render the textures of a model properly?

Exercise
Create an entity with a model so it looks like a blocky turtle: 4 legs, a tail, a head and a shell. Make sure that its textures are being applied properly.

Further explorations
Make the droid from the lecture or the turtle from the exercise spin around. You shouldn't animate the model to achieve this.

Answers and solutions

Answer to Question 1
First of all you need to create a Render class, this will to some translations, binding the texture and then telling the model to render itself. When this has been created one will simply have to register a new render object together with the entity class. That should be done in the client proxy. It can look a bit like the following:

RenderingRegistry.registerEntityRenderingHandler(EntityDroid.class, new RenderDroid());

Answer to Question 2
The ModelBase is the actual model, to create your model you'll make a subclass of the ModelBase.

*By **V**idar **Swe**nning*

The model consists of multiple parts and when you want to render your model you'll have to render all those parts. Those parts are ModelRenderers and can be moved around, rotated and set tp use different parts of the texture. Each ModelRender consists of one or more ModelBox. These boxes are the cuboids that we'll see as the model in the end.

Answer to Question 3
First of all you'll have to load and bind the texture from the render class. Then, in the model you'll have to define which size your texture has. If it is 64 pixels wide and 32 high you won't have to define that. The width and the height of the texture has to be a multiple of 2. The last step in the code is to tell the ModelRenderers where they should be looking for their texture inside the texture image. This is done by giving them an x – and y coordinate in the constructor call. The final step is to make a proper texture file to use.

Possible solution to Exercise
https://dl.dropboxusercontent.com/u/46486053/TurtleSolution1.zip