

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 recommended 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 one model and one renderer for entities and one for items that you want the model to be used for, how do you do it?

Question 2

An itemrenderer can render the item in different locations, for example when it's in the inventory and when it's dropped on the ground. How would you tell the itemrenderer to only work when the item is equipped (and therefore make it render the default way in the inventory and on the ground)?

Question 3

When changing the block's bounds depending on its state (for instance its metadata) we'll have to tell it to update the state for the collision and the item rendering. Why is that?

Exercise 1

Make the turtle from previous exercises render as an item as well.

Exercise 2

Make a block that looks like a chair when placed in the world.

Further explorations

Give the chair a proper collision and selection box, so you only can click it when you're actually aiming on the rendered parts.

Answers and solutions

Answer to Question 1

What you should do is tell both the renderers to render the model but give them slightly different information, like angles and positions etc. The entity renderer should get the rotations and other things from the entity itself, in this way the entity can modify them and cause the model to animate.

By Vidar Swenning

The item renderer should give the model some default angles and positions or alternatively some values depending on the itemstack, the item version don't usually animate. The render method of the model doesn't have to be the one that you can override from ModelBase.

Answer to Question 2

The emthod `handleRenderType` should return true when the itemrenderer wants to render a specific item. You can simply return true for the equiped types and false otherwise. It could look something like this

```
public boolean handleRenderType(ItemStack item, ItemRenderType type) {  
    return type == ItemRenderType.EQUIPPED ||  
           type == ItemRenderType.EQUIPPED_FIRST_PERSON;  
}
```

Answer to Question 3

Since there's only one block of each type we need to make sure that the bounds of that block is the correct one when we calculate the collision box or the rendering size. If we don't make sure to update it before we will just use the bounds that was used latest, which can be from any block for any reason which simply means that we don't know exactly what value we'll get. That value will likely be wrong.

Possible solution to Exercise 1

<https://dl.dropboxusercontent.com/u/46486053/TurtleSolution3.zip>

Possible solution to Exercise 2

<https://dl.dropboxusercontent.com/u/46486053/ChairSolution.zip>