Reflection (What did you learn?)

I learned how to facilitate the production of raw materials and organize them in such a way that I could produce more complex house/apartment blocks and factories. Overall, I learned how to handle a semi-complex java project with multiple classes, drivers and testing files that all rely on each other. I had to make sure to keep my eclipse organized as to not mix up any files/code snippets.

How did this design incorporate future growth?

Using the base object class and type checks defeats the purpose of having a type system, and having to add a new class for every kind of type is slightly repetitive.

Did you change up the driver at all? If so, how?

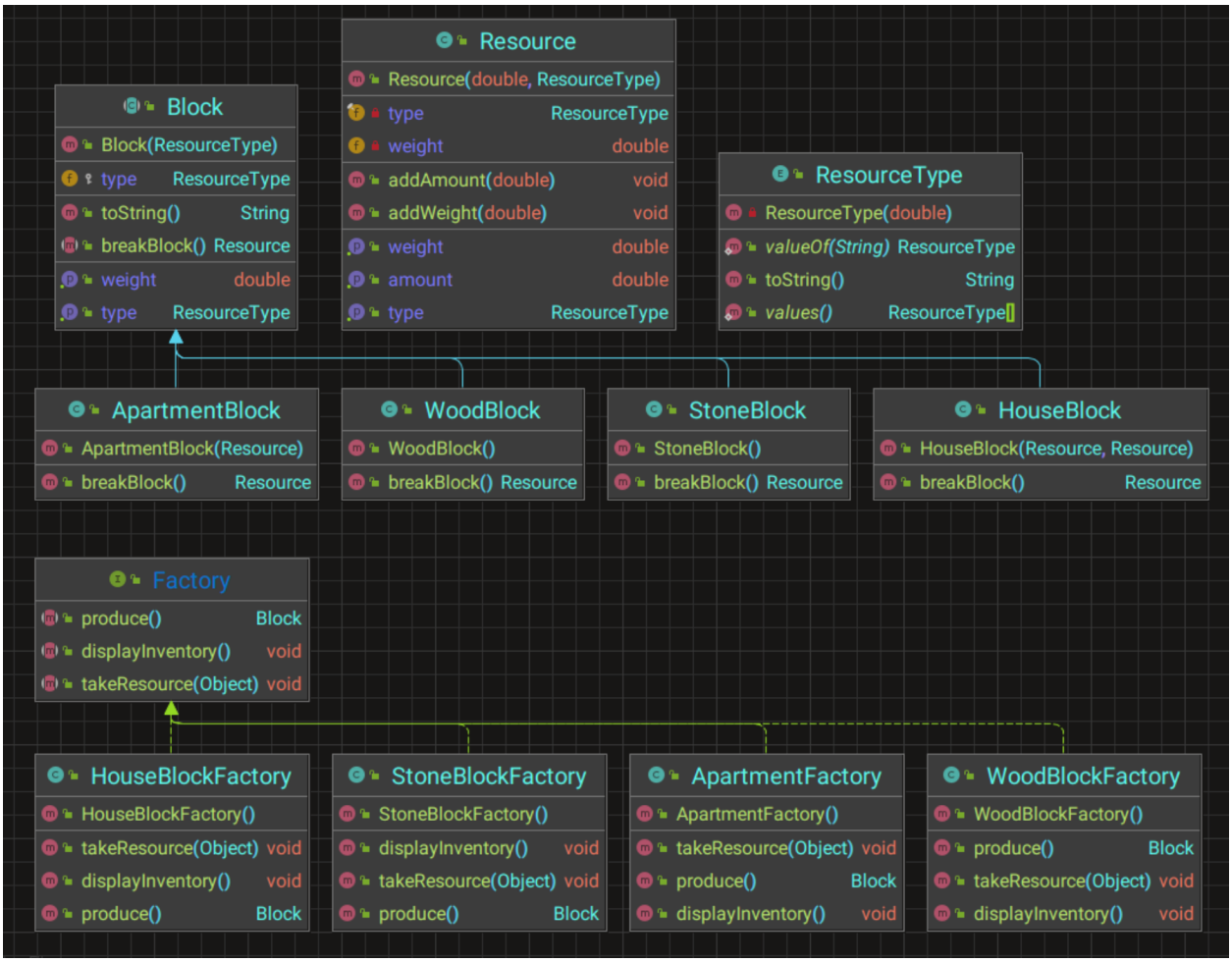
Yes, I've modified the driver to not produce infinite houses, but only three, so that I can use them to make an Apartment as my added extension which I also finished the execution for.

Extensions (What extensions are you requesting?)

While working on this, I discovered that I can bind values to enums, and thus refractored my code so that the weight of a resource type is part of the enum value itself. That change made my code much lighter and easier to work with and I'm proud of this discovery.

I've also added an Apartment resource type made up of the previously mentioned three houses from the drivef change, and added the needed `Const`ants for it. Finally, I tried to use JUnit's `assertThrows` method to test several exceptions within the same test case.

Your recitation UML diagram



Grading Statement (Based on the rubric, what grade do you feel you deserve? Be honest.)

I included all required parts for the base assignment, as well as adding several extensions including the apartment block and factory with a working main and modified driver to fit my vision for this assignment – I believe I should receive 100% for this assignment.