CSI 5342 Assignment 13.1

By: Matthew Tuan and Robbie Dutton

After setting up our initial Class diagram (as seen below), we began adding in constraints to ensure our model behaves as expected.

A screenshot of a computer

AI-generated content may be incorrect.

The first constraints we added were the basic SSD and DSD role constraints:  
A screenshot of a computer screen

AI-generated content may be incorrect.

We confirmed these invariants with two example script files InvalidSSDTest.x and InvalidDSDTest.x:

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Next we add in additional invariants for to check the SSD Permissions constraint:

A close-up of a sign

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

For the operation Session::CheckSession(), we needed to find a way to make sure that a user who is in a session, has access to perform operation op, on object obj. To do this we had to make a few updates.   
  
The first, we needed to update the Descendants function on Role class, to return a list of all of its role descendants. Then, update the Session::CheckSession() method. To test this, we created a ValidSessionCheckSession.x file, ran that, and then opened the OCL window and ran s1.CheckSession(o1, op1). This returned true, which meant this session has access to that object with that operation.

A computer screen shot of a computer

AI-generated content may be incorrect.

Next, we developed models that would make this false. To do that we created an InvalidSessionCheck.x file, that did not associate the role to the object. We got false!

A screenshot of a computer

AI-generated content may be incorrect.