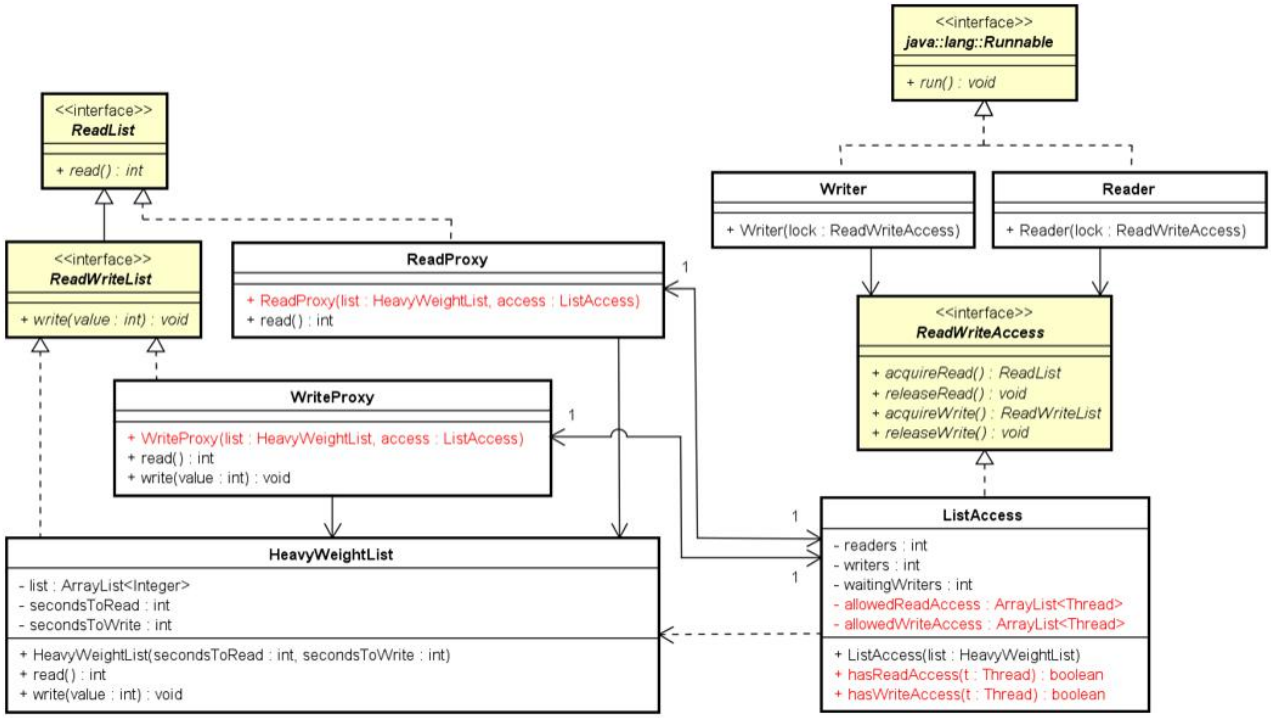
Exercise: Readers-Writers with access to a model object

Re-implement the Readers-Writers problem exercise with the `HeavyWeightList`. This time using a read and write proxy.



Part 1 (ListAccess without Proxy)

Update class `ListAccess` to include two `ArrayList`'s storing Thread objects.
In method `acquireRead` add `Thread.currentThread()` to the list (if it is not already there) and in `releaseRead` remove it from the list again. The same idea for write access in method `acquireWrite` and `releaseWrite`. Implement two boolean methods to return if a given Thread is on the list.

Part 2

Implement class `ReadProxy`. In method `read`, throw an `IllegalStateException` if the calling thread (i.e. `Thread.currentThread()`) is not on the list (calling the `hasReadAccess` method in class `ListAccess`). If it is on the list then call the `read` method in `HeavyWeightList`)

Part 3

Implement class `WriteProxy`. In methods `write` and `read`, throw an `IllegalStateException` if the calling thread (i.e. `Thread.currentThread()`) is not on the list (calling the `hasWriteAccess` method in class `ListAccess`). If it is on the list then call the methods in `HeavyWeightList`)

Part 4 (ListAccess including Proxy)

Declare two Proxy instance variables (types `ReadProxy` and `WriteProxy`) and initialize the two Proxy objects in the constructor.
In the two `acquire` methods return the Proxy instead of `HeavyWeightList`. You can now remove the `HeavyWeightList` instance variable.

Part 5

Run the `main` method