# Implementing the Duck Behaviors…

With this design, other types of objects can reuse our fly and quack behaviors because these behaviors are no longer hidden away in our Ducks!

And we can add new behaviors without modifying any of our existing behavior classes or touching any of the Ducks that use flying behaviors.

BOTTOM LINE:

We get all the benefits of REUSE without all the baggage that comes along with inheritance.

The key is that a Duck will now delegate its flying and quacking behavior, instead of using quacking and flying methods defined in the Duck class (or subclass).