

## Adapter Design Pattern

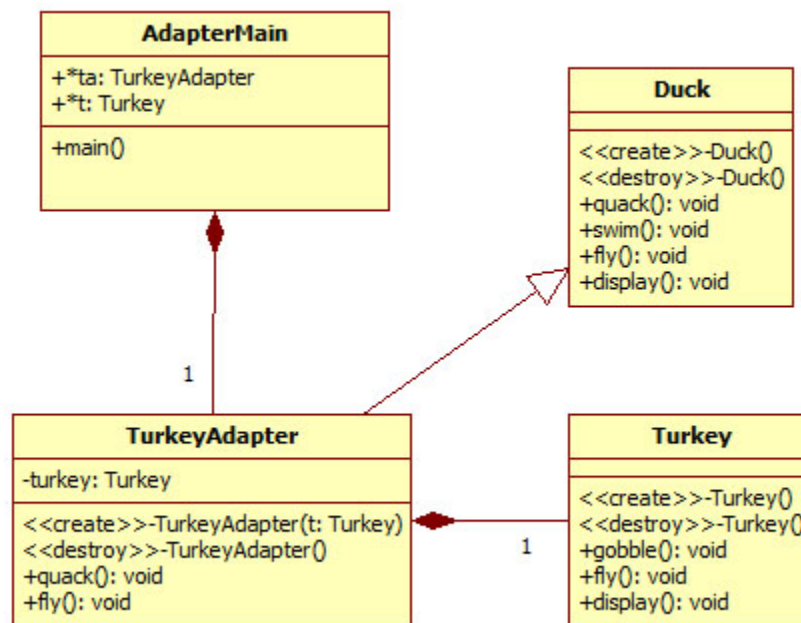
---

**GoF Statement:** Converts the interface of a class into another interface the clients expect. Adapter lets classes work together that couldn't otherwise because of incompatible interfaces.

**Category:** Structural

**UML Diagram:**

---



### Description of the Demonstration:

This demonstration shows how to create an Adapter class which handles calls to one type of class by translating them into appropriate calls to another type of class. In the demonstration of the Strategy Design Pattern a Duck object was created in which calls were made to functions to handle the actions of "quack" and "fly". This demonstration shows how an adapter, **TurkeyAdapter**, as a sub-class of **Duck** can take calls to a **Duck** object and translate them into the appropriate "gobble" and "fly" functions for a **Turkey** object.