

Gebze Teknik Üniversitesi
Bilgisayar Mühendisliği
Object Oriented Analysis and Design
CSE 443
#Hw2

Jwan hussein

151044078

Q1)

1. Any attempt to clone a Singleton object using the clone () method inherited from the Object class will result in throwing an exception at run time.

according to java documentation:

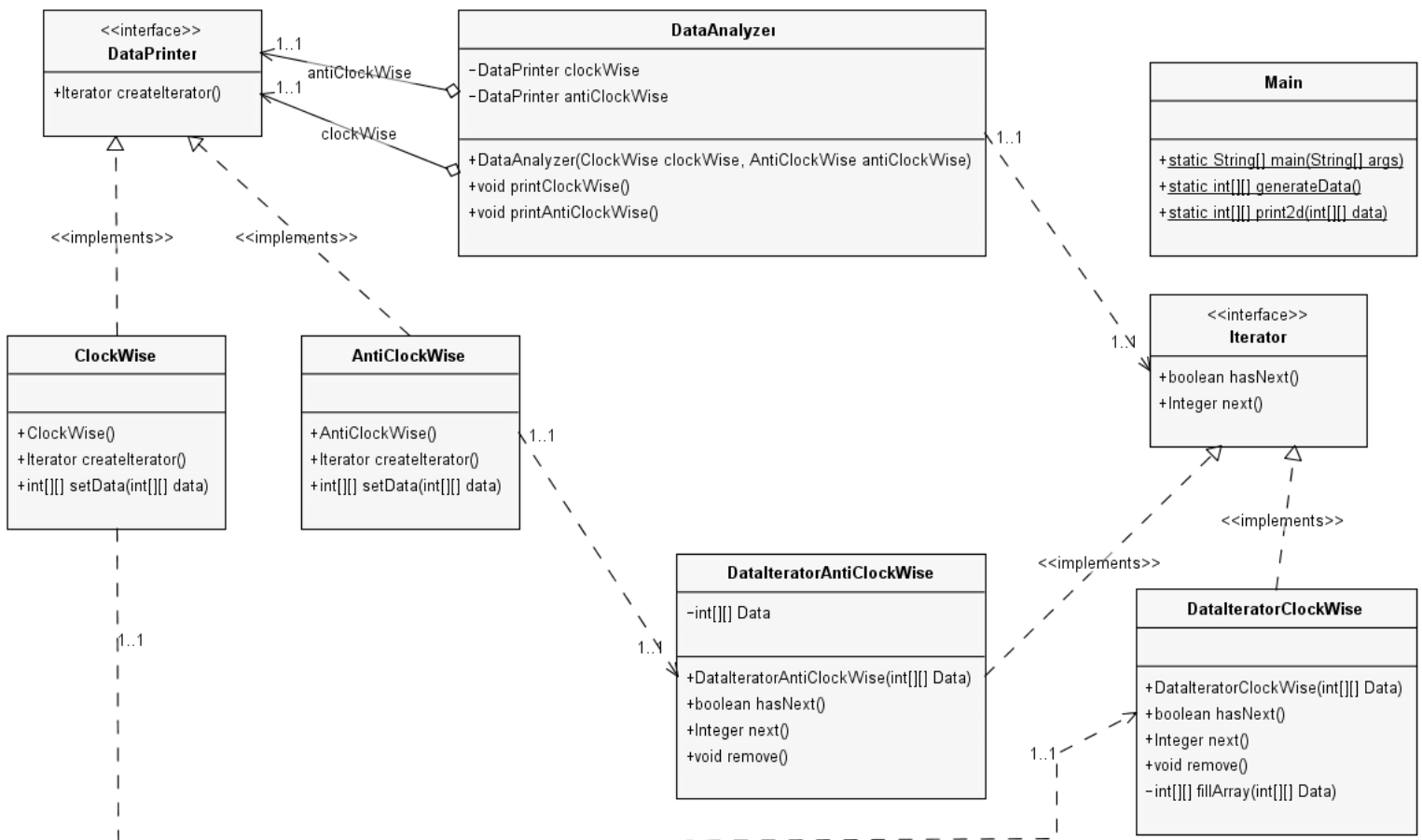
The class `Object` does not itself implement the interface `Cloneable`, so calling the `clone` method on an object whose class is `Object` will result in throwing an exception at run time.

- Any class that doesn't implement the `Cloneable` interface isn't able to be cloned.

assume the class Singleton is a subclass of class Parent, that fully implements the Cloneable interface.

1. In this case a new instance will be returned hence the class is no more singleton.
2. The Singleton class should override the clone() method .

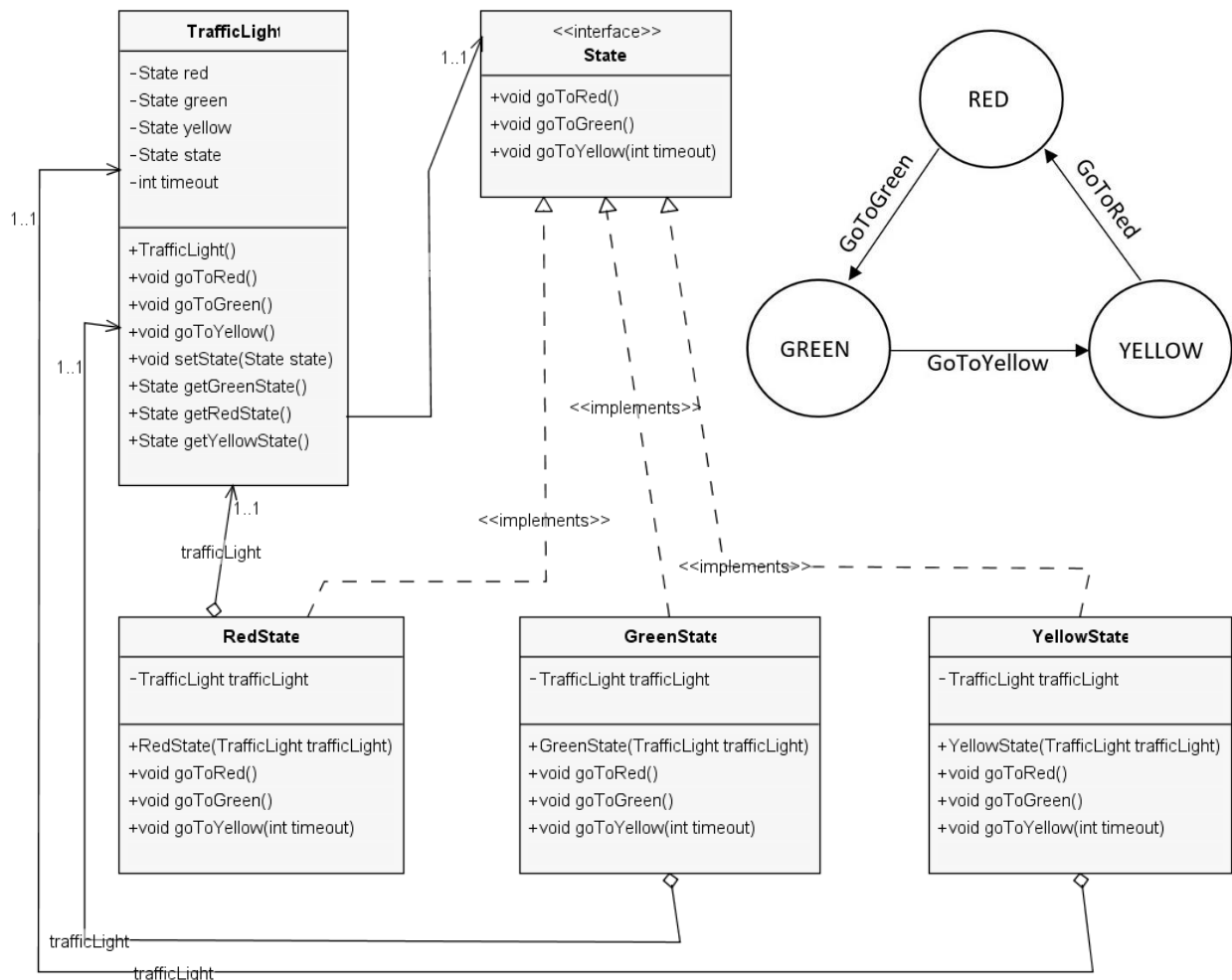
Q2)



The ClockWise and AntiClockWise classes implement an interface, `DataPrinter`. `DataAnalyier` can refer to each `DataPrinter` object using the interface.
The `DataPrinter` interface has one method, `createIterator()`, that is implemented by `ClockWise` and `AntiClockWise` classes.
Each concrete `DataPrinter` is responsible for creating the appropriate concrete `Iterator` class.

Q3)

1.



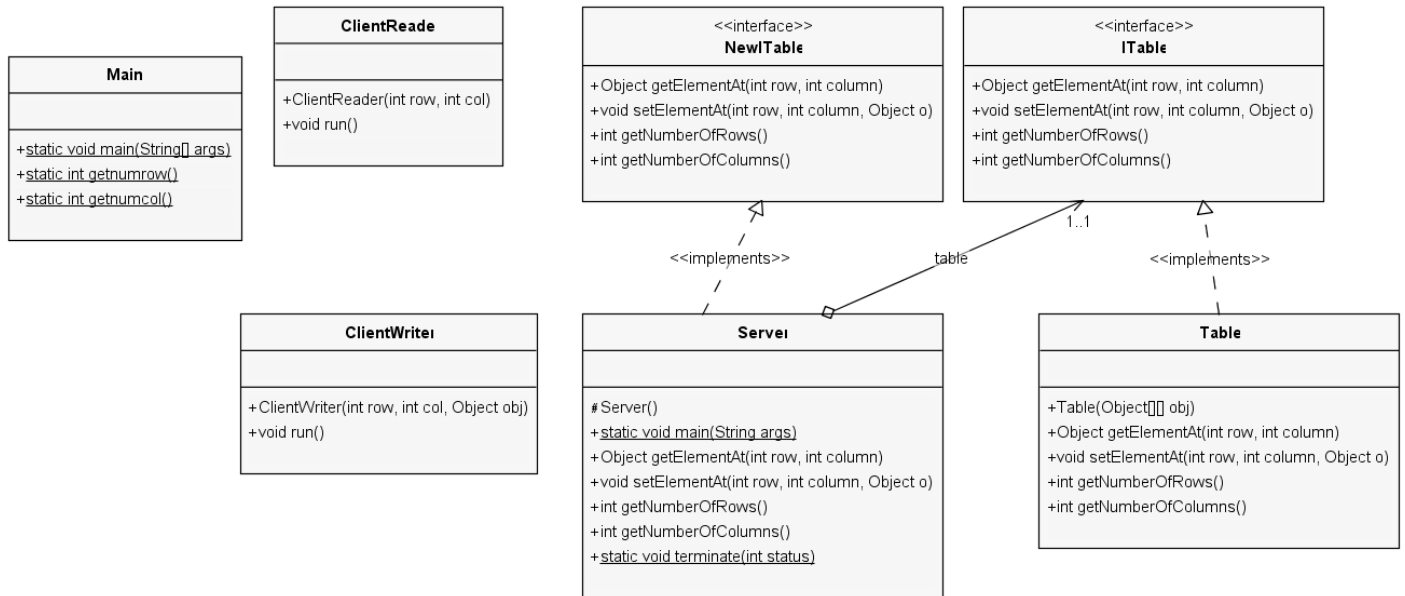
The `State` interface defines a common interface for all concrete states;
The states all implement the same interface, so they are interchangeable.

Concrete States handle requests from the `TrafficLight` class. Each Concrete State provides its own implementation for a request. In this way, when the `TrafficLight` changes state, its behavior will change as well.

The `TrafficLight` is the class that can have a number of internal states.

Whenever a request is made to change the state of the TrafficLight it is delegated to the stateto handle.

Q4)



the given interface ITable is not compatible with the proxy pattern so a new interface was written and the server class is now working as an adaptor too .