Mediator: Facilitates communication between components

1. Components may go in and out of a system at any time
   1. Chat room participants
   2. Players in an online game
2. It makes no sense for them to have direct references to one another
   1. Those references may go dead
3. Solution: have them all refer to some central component that facilitates communication
4. Mediator: A component that facilitates communication between other components without them necessarily being aware of each other or having direct (reference) access to each other.

Chat room

1. In chat room class all the participant references are kept. So that people don’t have references to each other.

Reactive Extensions Event Broker

1. With reactivex you can have event brokers that can publish events to subscribed objects.
2. It is a powerful library that can help you implement more than just a mediator pattern.
3. For example we have players and coaches who subscribe to an event broker and when players score they publish events to the eventbroker. (Event broker is the mediator, players and coaches don't have references to each other)

Summary

1. Create the mediator and have each object in the system refer to it
   1. E.g. in a field
2. Mediator engages in bidirectional communication with its connected components
3. Mediator has functions the components can call
4. Components have functions the mediator can call
5. Event processing (e.g., Rx) libraries make communication easier to implement.