

## Exercise 2

### A Chat System Exercise

- Your task is to create a simple chat system!
  - The system is comprised of two classes: **ChatServer** and **ChatClient**.
  - The basic chatting scenario should be:
    - A chat client sends a message (through the ChatServer).
    - The chat server:
      - Prints the client's name following by the message arrived.
      - Raises a **MsgArrived** event to all registered clients.
    - When a client receives a message it prints its name, followed by the arrived message and its sender.
- The **ChatServer** class should provide the following services:
  - A **SendMessage()** method, enabling chat clients to send messages.
  - The **SendMessage** method should receive the sender object and the message.
  - To enable chat clients register themselves as listeners of arrived messages, it should define the following:
    - A **MsgArrivedEventArgs** event class, holding event arguments:
      - The **ChatClient** that sent the message.
      - The sent message
    - A **MsgArrivedEventHandler** delegate type, receiving as parameters
      - The sender **object** ,and
      - A **MsgArrivedEventArgs** object.
    - Event listeners should define such a method and provide it as a callback when registering to the event.
      - An **public event** field:
      - **MsgArrivedEventHandler MsgArrived;**
- In addition, the server should define an **OnMsgArrived** method, raising the event when the message arrives.
- The **ChatClient** class should:
  - Provide users with a **Chat()** method that receives the chat message as parameter.

- Define a method according to the server's delegate type signature. This method should perform the operation required to be done when a **MsgArrived** event is notified.
  - In its constructor, the client should get a chat server object as a parameter and register to the server's **MsgArrived** event, providing its callback method as parameter.
- Small Fix:
  - As you can see in program's output demonstration, there is still a problem with the above chatting system.
  - The chatter that sent the message also receives it...
  - Can you think of a change that can be done to the Chat Server **OnMsgArrived()** method, such that it will notify all listening chatters except of the one that sent the message?