using System;

using System.Collections.Generic;

using System.Text;

namespace BehaviouralPatternDemos

{

public interface FacebookGroupMediator

{

void SendMessage(string msg, User user);

void RegisterUser(User user);

}

public class ConcreteFacebookGroupMediator : FacebookGroupMediator

{

private List<User> usersList = new List<User>();

public void RegisterUser(User user)

{

usersList.Add(user);

}

public void SendMessage(string message, User user)

{

foreach (var u in usersList)

{

// message should not be received by the user sending it

if (u != user)

{

u.Receive(message);

}

}

}

}

public abstract class User

{

protected FacebookGroupMediator mediator;

protected string name;

public User(FacebookGroupMediator mediator, string name)

{

this.mediator = mediator;

this.name = name;

}

public abstract void Send(string message);

public abstract void Receive(string message);

}

public class ConcreteUser : User

{

public ConcreteUser(FacebookGroupMediator mediator, string name) : base(mediator, name)

{

}

public override void Receive(string message)

{

Console.WriteLine(this.name + ": Received Message:" + message);

}

public override void Send(string message)

{

Console.WriteLine(this.name + ": Sending Message=" + message + "\n");

mediator.SendMessage(message, this);

}

}

class MediatorDemo2

{

public static void Main()

{

FacebookGroupMediator facebookMediator = new ConcreteFacebookGroupMediator();

User Ram = new ConcreteUser(facebookMediator, "Ram");

User Dave = new ConcreteUser(facebookMediator, "Dave");

User Smith = new ConcreteUser(facebookMediator, "Smith");

User Rajesh = new ConcreteUser(facebookMediator, "Rajesh");

User Sam = new ConcreteUser(facebookMediator, "Sam");

User Pam = new ConcreteUser(facebookMediator, "Pam");

User Anurag = new ConcreteUser(facebookMediator, "Anurag");

User John = new ConcreteUser(facebookMediator, "John");

facebookMediator.RegisterUser(Ram);

facebookMediator.RegisterUser(Dave);

facebookMediator.RegisterUser(Smith);

facebookMediator.RegisterUser(Rajesh);

facebookMediator.RegisterUser(Sam);

facebookMediator.RegisterUser(Pam);

facebookMediator.RegisterUser(Anurag);

facebookMediator.RegisterUser(John);

Dave.Send("dotnettutorials.net - this website is very good to learn Design Pattern");

Console.WriteLine();

Rajesh.Send("What is Design Patterns? Please explain ");

Console.Read();

}

}

}