using System;

using System.Collections.Generic;

namespace ObserverDesignPattern

{

interface IFollower

{

public void Update(string name, string message);

}

interface ICompany

{

public void Follow(IFollower followers);

public void Unfollow(IFollower followers);

public void NotifyFollower(string message);

}

class Company : ICompany

{

private string \_companyName;

private List<IFollower> \_followers;

public Company(string userName)

{

\_companyName = userName;

\_followers = new List<IFollower>();

}

public void Follow(IFollower followers)

{

\_followers.Add(followers);

}

public void Unfollow(IFollower followers)

{

\_followers.Remove(followers);

}

public void NotifyFollower(string put)

{

foreach (var follower in \_followers)

{

follower.Update(\_companyName, put);

}

}

public void Post(string post)

{

Console.WriteLine(\_companyName + " must bring " + post);

NotifyFollower(post);

}

}

class Follower : IFollower

{

private string \_followerName;

public Follower(string followerName)

{

\_followerName = followerName;

}

public void Update(string companyName, string put)

{

Console.WriteLine(\_followerName + ", you have to see the " + put + " by " + companyName);

}

}

class Program

{

static void Main(string[] args)

{

var diesel = new Company("Diesel");

var salomon = new Company("Salomon");

var leon = new Follower("Leon");

var monta = new Follower("Monta");

var joy = new Follower("Joy");

var carl = new Follower("Carl");

diesel.Follow(leon);

diesel.Follow(monta);

salomon.Follow(carl);

diesel.Post("new Jeans");

salomon.Post("new Shoes");

salomon.Unfollow(joy);

}

}

}