Ej 1

1. Se puede usar el patrón observer para hacer con que se notifique a mas estudiantes
2. Public interface IObserver{

Public void notify()

}

Public class Exam {

… // lo que tenia antes

Public void subscribe (Student s){

Students.add(s)

}

Public void NotifyStudents(){

Foreach(Student s in students){

s.notify(this)

}

}

}

Public class Student: IObserver{

… // lo que tenia antes

Public void notify(Exam e){

Console.writeLine(“Hay un nuevo examen de” + e.subject)

}

}

Ej 2

1. Se puede usar el patrón memento para restaurar al estado inicial.
2. Public class GameCharacterMemento {

Public string name {get; set;}

Public int health {get; set;}

Public int mana {get; set;}

Public GameCharacterMemento(String name, int health, int mana) {

This.mana=mana;

This.health=health;

This.name=name;

}

}

Public class GameCharacter {

… // lo que estaba antes

Public GameCharacterMemento save(){

Return new GameCharacterMemento(name, health, mana);

}

Public void restore(GameCharacterMemento memento){

Name=memento.name

Health=memento.health

Mana=memento.mana

}

}

Public class program{

Public void main(){

Var gameCharacter= new GameCharacter(“name”, 100, 50)

gameCharacter.displayStatus()

GameCharacterMemento memento= gameCharacter.save()

gameCharacter.health=80

gameCharacter.mana=40

gameCharacter.displayStatus()

gameCharacter.restore(memento)

gameCharacter.displayStatus()

}

}

Ej 3

1. Capaz que usar mediator?

Ej 4

1. Se puede usar el state???

Ej 5

1. Command?

Ej 6

1. State

Ej 7

1. Strategy o chan of responsability

Ej 8

1. Strategy

Ej 9

1. Chain of responsability

Ej 10

1. Chain of responsability