

**Docente:**

Dilian Anabel HURTADO PONCE

# **Tarea 8:** PATRONES

ESTRUCTURALES: INTRODUCCIÓN A  
PATRONES ESTRUCTURALES PATRÓN  
ADAPTER Y PATRÓN FACADE.

**Semana: 9**

**GRUPO 5**

**Integrantes:**

Roberto Agustín Mejía Collazos

Miguel Ángel Velásquez Ysuiza

Manuel Ángel Pecho Santos

Daniel Wilfredo Sotomayor Beteta

Cristina Paola Burgos Salazar

10/10/24

```
interface FabricaAbstracta {
    Televisor crearTelevisor();
}

class FabricaAltaGama implements FabricaAbstracta
{
    public Televisor crearTelevisor() {
        return new TelevisorAltaGama();
    }
}

class FabricaBajaGama implements FabricaAbstracta
{
    public Televisor crearTelevisor() {
        return new TelevisorBajaGama();
    }
}

interface Televisor {
    void mostrar();
}

class TelevisorAltaGama implements Televisor {
    public void mostrar() {
        System.out.println("Televisor de Alta
Definición");
    }
}

class TelevisorBajaGama implements Televisor {
    public void mostrar() {
        System.out.println("Televisor de
Definición Estándar");
    }
}
```

# Codigo GitHub

```
Bienvenido  VideoConversionFacade.java M  Demo.java M X
src > main > java > pattern > facade > Demo.java > ...
victorHugoAguilar, hace 3 años | 1 author (victorHugoAguilar)
1 package pattern.facade;
2 import java.io.File;
victorHugoAguilar, hace 3 años | 1 author (victorHugoAguilar)
3 public class Demo {
  Run | Debug
4     public static void main(String[] args) {
5         VideoConversionFacade converter = new VideoConversionFacade();
6         @SuppressWarnings("unused")
7         File mp4Video = converter.convertVideo(fileName:"youtubevideo.ogg", format:"mp4");
8     }
9 }
```

```
Bienvenido  VideoConversionFacade.java X  Demo.java
> main > java > pattern > facade > VideoConversionFacade.java > ...
1 package pattern.facade;
2 import java.io.File;
3 import pattern.facade.some_complex_media_library.*;
4 public class VideoConversionFacade {
5     public File convertVideo(String fileName, String format) {
6         System.out.println(x:"VideoConversionFacade: conversion started.");
7         VideoFile file = new VideoFile(fileName);
8         Codec sourceCodec = CodecFactory.extract(file);
9         Codec destinationCodec;
10        if (format.equals(anObject:"mp4")) {
11            destinationCodec = new OggCompressionCodec();
12        } else {
13            destinationCodec = new MPEG4CompressionCodec();
14        }
15        VideoFile buffer = BitrateReader.read(file, sourceCodec);
16        VideoFile intermediateResult = BitrateReader.convert(buffer, destinationCodec);
17        File result = (new AudioMixer()).fix(intermediateResult);
18        System.out.println(x:"VideoConversionFacade: conversion completed.");
19        return result;
20    }
21 }
```

```
Bienvenido  VideoConversionFacade.java M  Demo.java M  FacadeMain.java M X
src > main > java > facade > FacadeMain.java > ...
1 package facade;
2 import facade.impl.IPaymentFacade;
3 import facade.impl.OnlineFacadeImpl;
4 import facade.impl.PaymentRequest;
5 public class FacadeMain {
6     public static void main(String[] args) {
7         PaymentRequest request = new PaymentRequest();
8         request.setAmount(amount:1000);
9         request.setCardExtDate(cardExtDate:"10/2015");
10        request.setCardName(cardName:"Victor Aguilar");
11        request.setCardNumber(cardNumber:"1234567890123456");
12        request.setCardSecureNum(cardSecureNum:"345");
13        request.setCustomerId(customerId:1L);
14        try {
15            IPaymentFacade paymentFacade = new OnlineFacadeImpl();
16            paymentFacade.pay(request);
17        } catch (Exception e) {
18            System.out.println(e.getMessage());
19        }
20    }
21 }
```

## DIAGRAMA UML

