Adapter Design Pattern

Name: Stefan Humpelstetter

Date:15.11.2017

Content

- Issue
- Usage
- Example
- Exercise
- Sources

Issue

- reuse an already existing component which has a different „view of world“

- translate an interface into another one

Usage

- Like a bridge between two incompatible interfaces
- The card reader translates as an adapter the card for the laptop

Differenation from other Patterns

Bridge Pattern

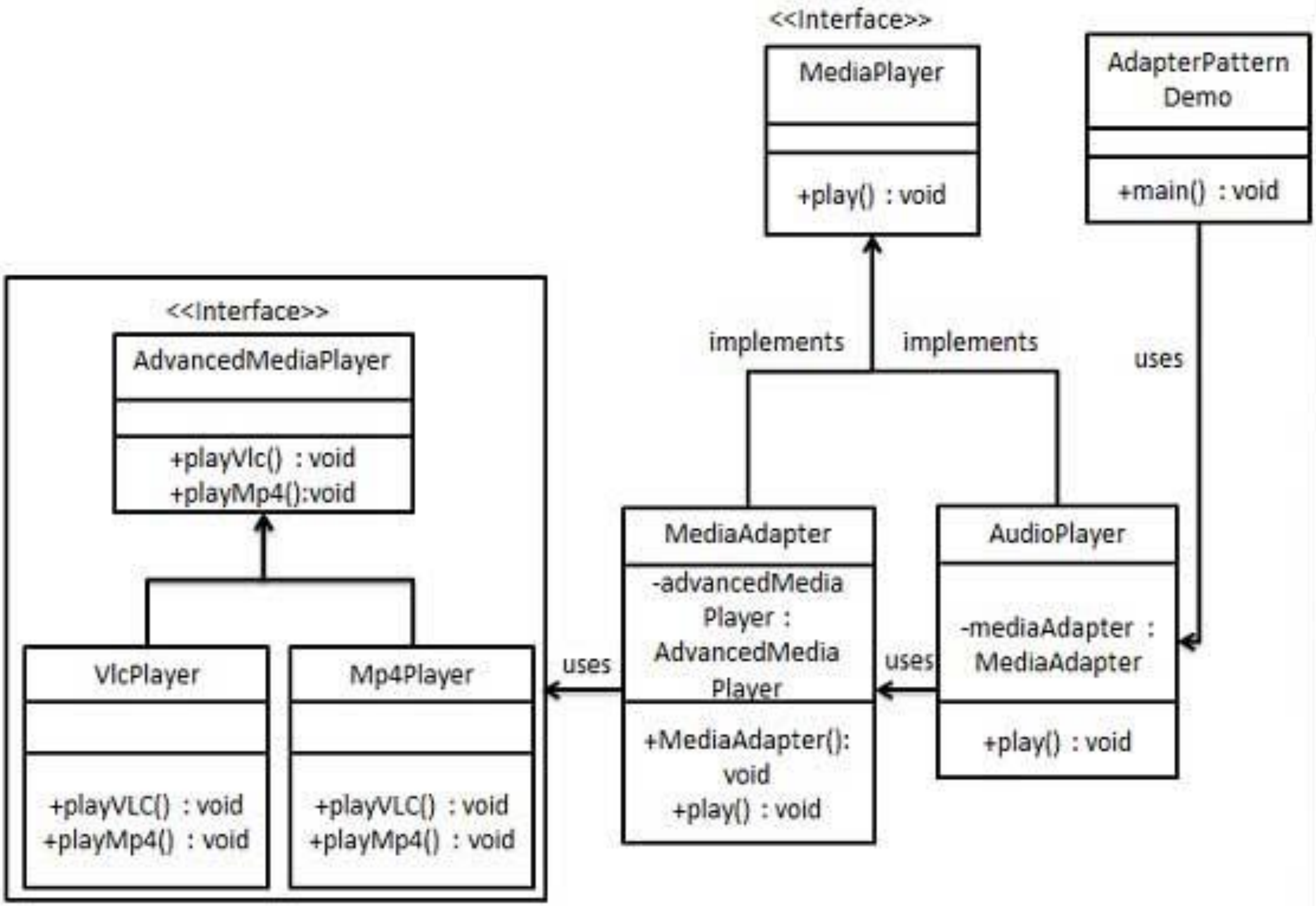
- Adapter makes things work after they're designed; Bridge makes them work before they are

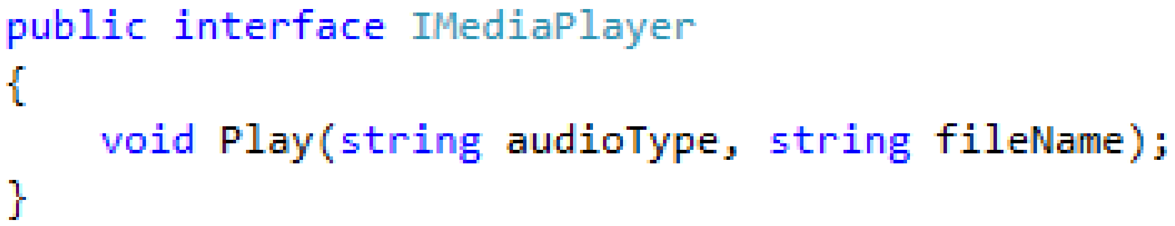
Facade Pattern

- The Facade Pattern is just to simplify a user interaction with an interface, the Adapter Pattern handles complex interactions by reconstructing incoming data for the underlying objects

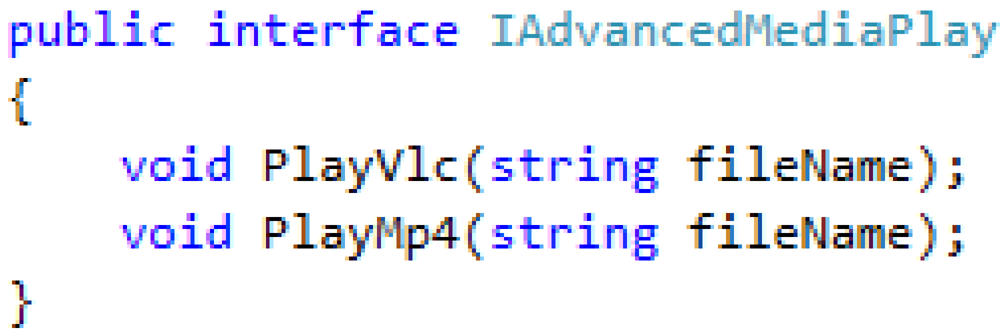
Example

- an Audio Player which can play mp3 files only wants to use an advanced audio player which can play vlc and mp4 files

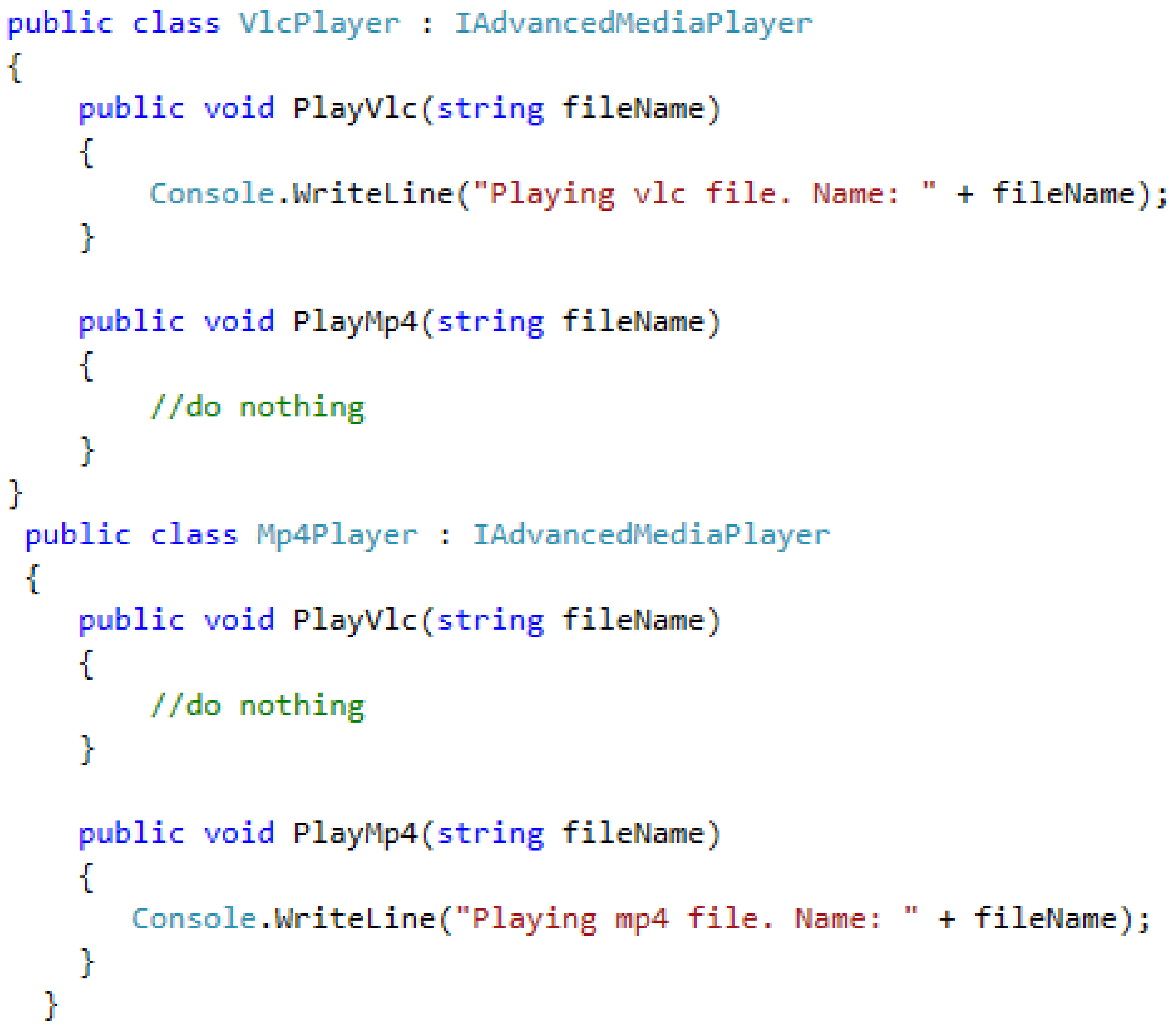

<<Interface>>
MediaPlayer
+play() : void
AdapterPattern Demo
+main() : void
<<Interface>>
AdvancedMediaPlayer
+playVlc() : void
+playMp4():void
implements
implements
uses
MediaAdapter
-advancedMedia Player : AdvancedMedia Player
+MediaAdapter(): void
+play() : void
AudioPlayer
-mediaAdapter : MediaAdapter
+play() : void
uses
uses
VlcPlayer
+playVLC() : void
+playMp4() : void
Mp4Player
+playVLC() : void
+playMp4() : void



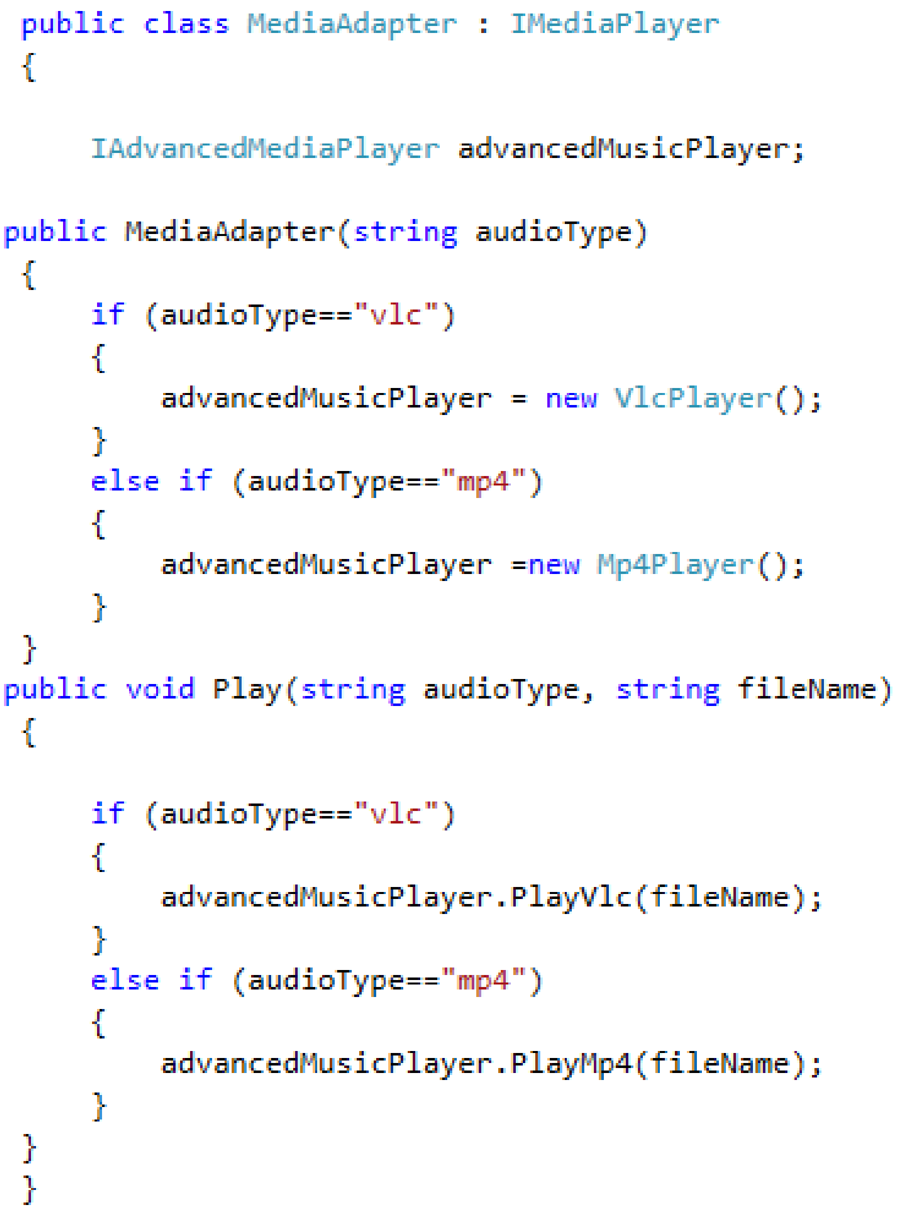
```
public interface IMediaPlayer
{
    void Play(string audioType, string fileName);
}
```



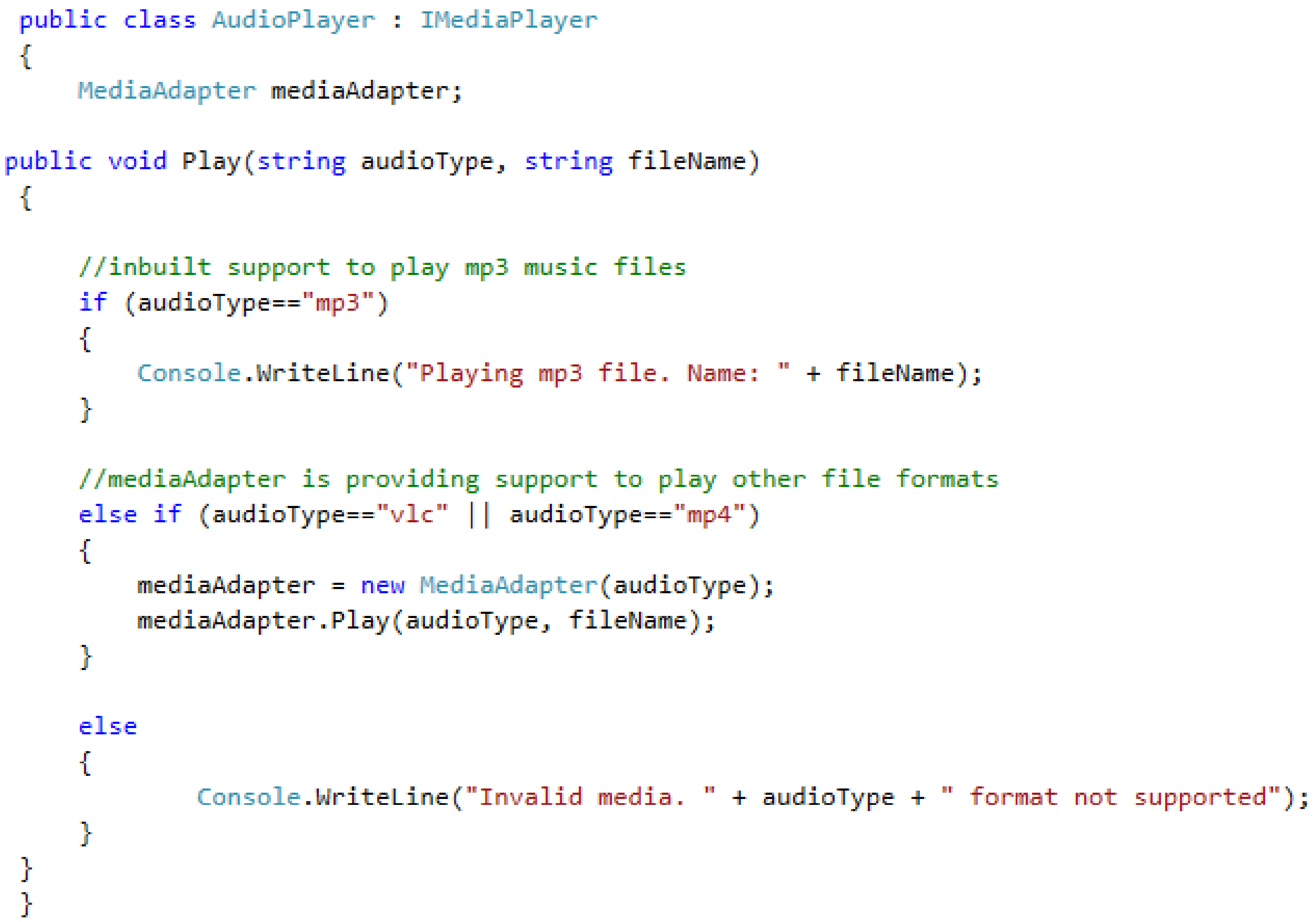
```
public interface IAdvancedMediaPlayer
{
    void PlayVlc(string fileName);
    void PlayMp4(string fileName);
}
```



```
public class VlcPlayer : IAdvancedMediaPlayer
{
    public void PlayVlc(string fileName)
    {
        Console.WriteLine("Playing vlc file. Name: " + fileName);
    }

    public void PlayMp4(string fileName)
    {
        //do nothing
    }
}
 public class Mp4Player : IAdvancedMediaPlayer
 {
    public void PlayVlc(string fileName)
    {
        //do nothing
    }

    public void PlayMp4(string fileName)
    {
       Console.WriteLine("Playing mp4 file. Name: " + fileName);
    }
  }
```



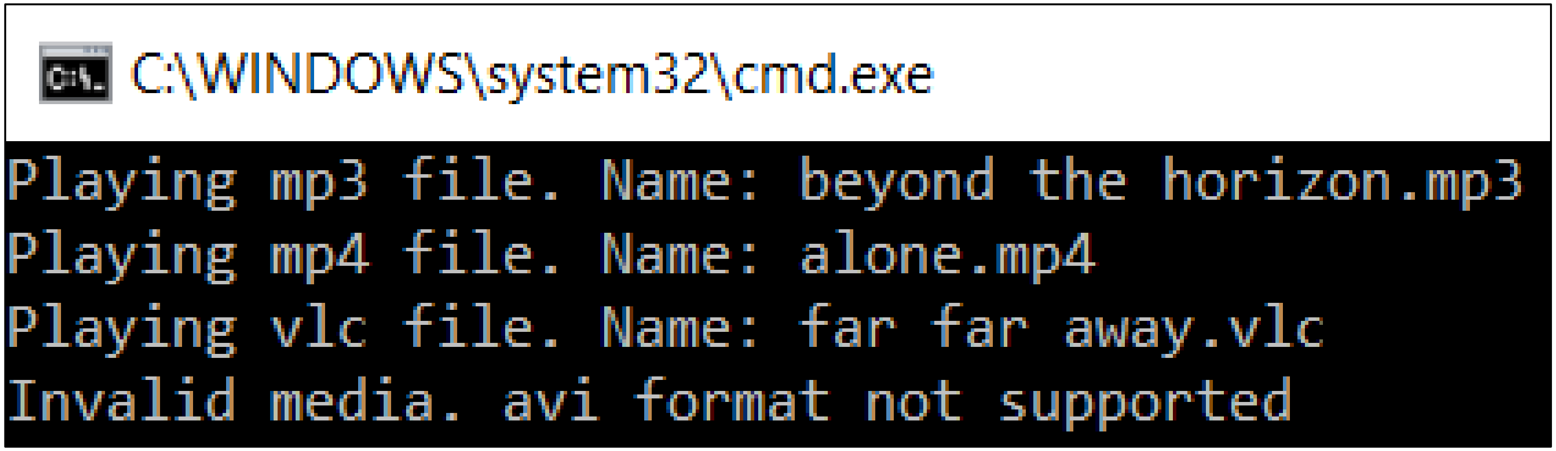
```
 public class MediaAdapter : IMediaPlayer
 {

     IAdvancedMediaPlayer advancedMusicPlayer;

public MediaAdapter(string audioType)
 {
     if (audioType=="vlc")
     {
         advancedMusicPlayer = new VlcPlayer();
     }
     else if (audioType=="mp4")
     {
         advancedMusicPlayer =new Mp4Player();
     }
 }
public void Play(string audioType, string fileName)
 {

     if (audioType=="vlc")
     {
         advancedMusicPlayer.PlayVlc(fileName);
     }
     else if (audioType=="mp4")
     {
         advancedMusicPlayer.PlayMp4(fileName);
     }
 }
 }
```



```
 public class AudioPlayer : IMediaPlayer
 {
     MediaAdapter mediaAdapter;

public void Play(string audioType, string fileName)
 {

     //inbuilt support to play mp3 music files
     if (audioType=="mp3")
     {
         Console.WriteLine("Playing mp3 file. Name: " + fileName);
     }

     //mediaAdapter is providing support to play other file formats
     else if (audioType=="vlc" || audioType=="mp4")
     {
         mediaAdapter = new MediaAdapter(audioType);
         mediaAdapter.Play(audioType, fileName);
     }

     else
     {
             Console.WriteLine("Invalid media. " + audioType + " format not supported");
     }
 }
 }
```

Test

```
class Program
{
    static void Main(string[] args)
    {
        AudioPlayer audioPlayer = new AudioPlayer();

        audioPlayer.Play("mp3", "beyond the horizon.mp3");
        audioPlayer.Play("mp4", "alone.mp4");
        audioPlayer.Play("vlc", "far far away.vlc");
        audioPlayer.Play("avi", "mind me.avi");
    }
}
```



C:\WINDOWS\system32\cmd.exe

```
Playing mp3 file. Name: beyond the horizon.mp3
Playing mp4 file. Name: alone.mp4
Playing vlc file. Name: far far away.vlc
Invalid media. avi format not supported
```

Excercise

- Create an even more advanced media player which can play mp5 and vlx format files
- Use a new Adapter called AdvancedMediaAdapter
- Let the already existing audio player play mp5 and vlx format files

Sources

https://sourcemaking.com/design_patterns/adapter

https://www.tutorialspoint.com/design_pattern/adapter_pattern.htm

https://de.wikipedia.org/wiki/Adapter_(Entwurfsmuster)