LAB ASSIGNMENT 10

FACADE DESIGN PATTERN

Facade DP provides simplified interface to a set of interfaces in a subsystem.

It hides the complexities of the subsystem from the client.

It describes a higher-level interface that makes the sub-system easier to use.

CODE:

- Guitar.java

```java
public class Guitar {

    public void AcousticGuitar(){
        System.out.println("Playing Acoustic Guitar");
    }
    public void ElectricGuitar(){
        System.out.println("Playing Electric Guitar");
    }

}
```

- Drum.java

```java
public class Drum {

    public void AcousticDrum(){
        System.out.println("Listening Acoustic Drum");
    }
    public void ElectricDrum(){
        System.out.println("Listening Electric Drum");
    }

}
```

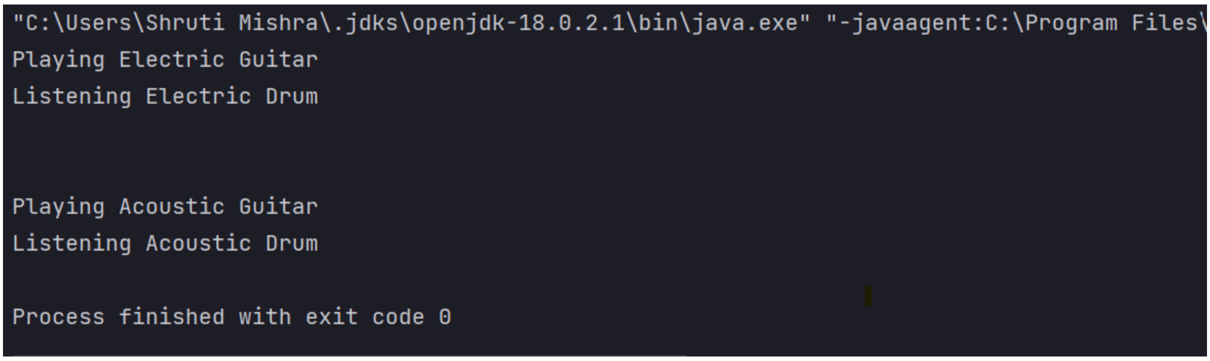
- InstrumentFacade.java //main file

```java
public class InstrumentFacade {
    Guitar guitar;
    Drum drum;

    public InstrumentFacade(Guitar guitar1 , Drum drum1){
        this.guitar = guitar1;
        this.drum = drum1;
    }

    public void acousticInstruments(){
        guitar.AcousticGuitar();
        drum.AcousticDrum();
    }

    public void electricInstrument(){
        guitar.ElectricGuitar();
        drum.ElectricDrum();
    }

}}
```

- Client.java

```
public class Client {

    public static void main(String[] args) {

        Guitar obj1 = new Guitar();
        Drum obj2 = new Drum();

        InstrumentFacade facade = new InstrumentFacade(obj1,obj2);

        facade.electricInstrument();

        System.out.println();
        System.out.println();

        facade.acousticInstruments();
    }
}
```

OUTPUT:



```
"C:\Users\Shruti Mishra\.jdks\openjdk-18.0.2.1\bin\java.exe" "-javaagent:C:\Program Files\
Playing Electric Guitar
Listening Electric Drum


Playing Acoustic Guitar
Listening Acoustic Drum

Process finished with exit code 0
```