

Name : Tejaswini Anil Kamble

Email: teju000kamble@gmail.com

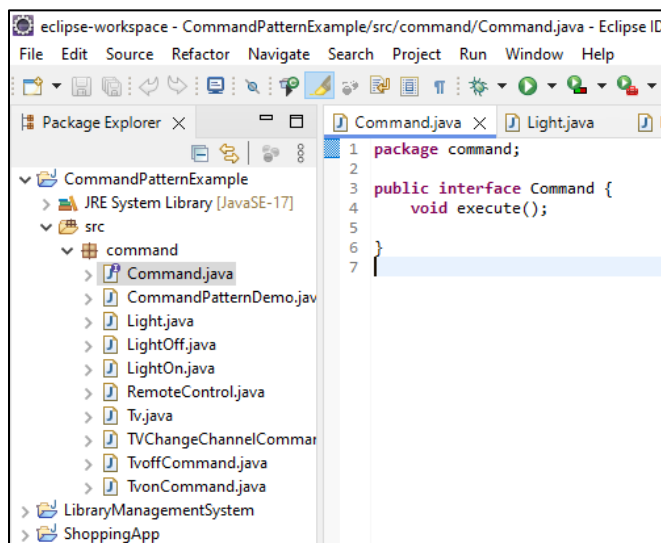
Assignment : 18/06/2022

Task

Q. Imagine a remote control with buttons(invoker) that can control various electronics devices(Receivers) such as TVs or lights. Each button press sends a command (Concreate Command) to the respective device to perform actions like turning ON / OFF or changing channels.

Use Command Design paatern and impelemt it.

Solution : Source Code



The screenshot shows the Eclipse IDE interface. The Package Explorer on the left displays the project structure for 'CommandPatternExample', including the 'src' directory and the 'command' package. The main editor window shows the 'Command.java' file, which contains the following code:

```
1 package command;
2
3 public interface Command {
4     void execute();
5 }
6
7
```

eclipse-workspace - CommandPatternExample/src/command/CommandPatternDemo.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- CommandPatternExample
 - JRE System Library [JavaSE-17]
 - src
 - command
 - Command.java
 - CommandPatternDemo.java
 - Light.java
 - LightOff.java
 - LightOn.java
 - RemoteControl.java
 - Tv.java
 - TVChangeChannelCommand.java
 - TvoffCommand.java
 - TvonCommand.java
 - LibraryManagementSystem
 - ShoppingApp

```
1 package command;
2
3 public class CommandPatternDemo {
4     public static void main(String[] args) {
5
6         Tv tv = new Tv();
7         Light light = new Light();
8
9         Command tvOn = new TvonCommand(tv);
10        Command tvOff = new TvoffCommand(tv);
11        Command tvChangeChannel = new TVChangeChannelCommand(tv, 5);
12        Command lightOn = new LightOn(light);
13        Command lightOff = new LightOff(light);
14
15        RemoteControl remote = new RemoteControl();
16
17        remote.setCommand("TV ON", tvOn);
18        remote.setCommand("TV OFF", tvOff);
19        remote.setCommand("TV CHANNEL", tvChangeChannel);
20        remote.setCommand("LIGHT ON", lightOn);
21        remote.setCommand("LIGHT OFF", lightOff);
22
23        remote.pressButton("TV ON");
24        remote.pressButton("TV CHANNEL");
25        remote.pressButton("LIGHT ON");
26        remote.pressButton("LIGHT OFF");
27        remote.pressButton("TV OFF");
28    }
29 }
```

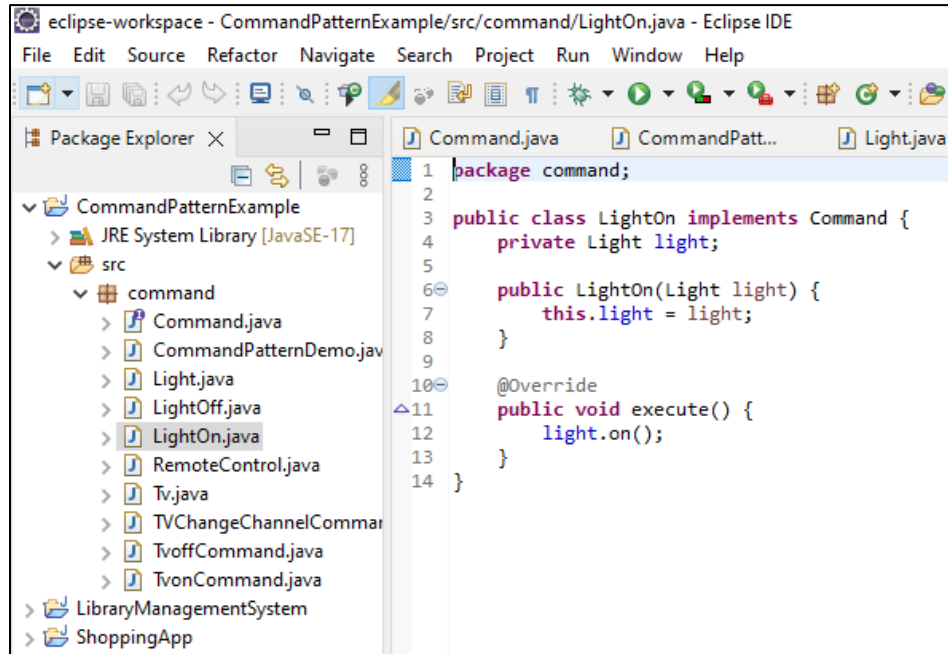
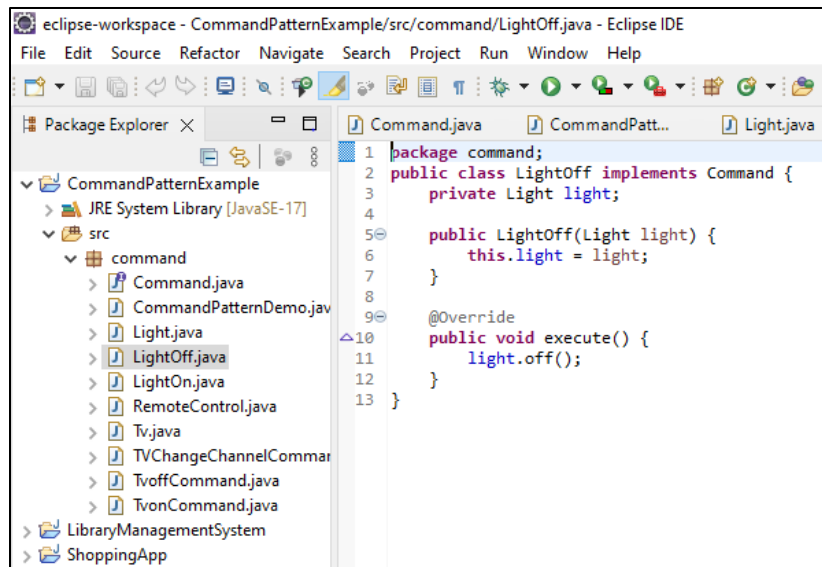
eclipse-workspace - CommandPatternExample/src/command/Light.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- CommandPatternExample
 - JRE System Library [JavaSE-17]
 - src
 - command
 - Command.java
 - CommandPatternDemo.java
 - Light.java
 - LightOff.java
 - LightOn.java
 - RemoteControl.java
 - Tv.java
 - TVChangeChannelCommand.java
 - TvoffCommand.java
 - TvonCommand.java
 - LibraryManagementSystem
 - ShoppingApp

```
1 package command;
2
3 public class Light {
4     public void on() {
5         System.out.println("Light is ON");
6     }
7
8     public void off() {
9         System.out.println("Light is OFF");
10    }
11 }
```



eclipse-workspace - CommandPatternExample/src/command/RemoteControl.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- CommandPatternExample
 - JRE System Library [JavaSE-17]
 - src
 - command
 - Command.java
 - CommandPatternDemo.java
 - Light.java
 - LightOff.java
 - LightOn.java
 - RemoteControl.java
 - Tv.java
 - TVChangeChannelCommand.java
 - TvoffCommand.java
 - TvonCommand.java
 - LibraryManagementSystem
 - ShoppingApp

```
1 package command;
2
3 import java.util.HashMap;
4
5 public class RemoteControl {
6     private Map<String, Command> buttons = new HashMap<>();
7
8     public void setCommand(String button, Command command) {
9         buttons.put(button, command);
10    }
11
12    public void pressButton(String button) {
13        if (buttons.containsKey(button)) {
14            buttons.get(button).execute();
15        } else {
16            System.out.println("No command assigned to this button");
17        }
18    }
19 }
20
21 }
```

eclipse-workspace - CommandPatternExample/src/command/Tv.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- CommandPatternExample
 - JRE System Library [JavaSE-17]
 - src
 - command
 - Command.java
 - CommandPatternDemo.java
 - Light.java
 - LightOff.java
 - LightOn.java
 - RemoteControl.java
 - Tv.java
 - TVChangeChannelCommand.java
 - TvoffCommand.java
 - TvonCommand.java
 - LibraryManagementSystem
 - ShoppingApp

```
1 package command;
2
3 public class Tv {
4     public void on() {
5         System.out.println("TV is ON");
6     }
7
8     public void off() {
9         System.out.println("TV is OFF");
10    }
11
12    public void changeChannel(int channel) {
13        System.out.println("TV channel changed to " + channel);
14    }
15 }
```

