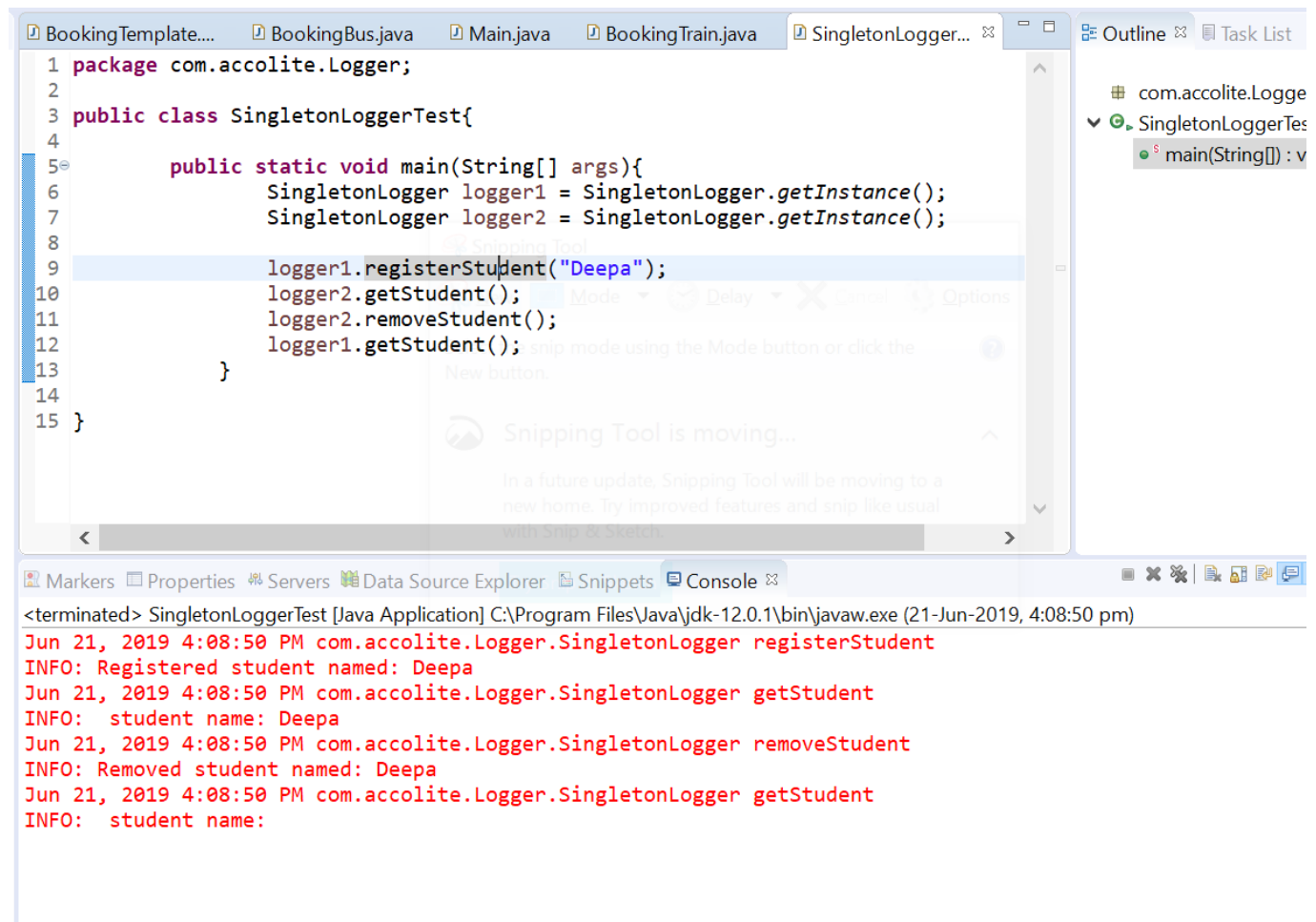


# Outputs of Design Patterns

## Output of Singleton logger



The screenshot displays an IDE with a Java project. The main editor shows the `SingletonLoggerTest` class. The code defines a package `com.accolite.Logger` and a public class `SingletonLoggerTest` with a `main` method. The `main` method creates two instances of `SingletonLogger` using `getInstance()`, registers a student named "Deepa" with `logger1`, and then uses `logger2` to get and remove the student. The console output at the bottom shows the execution results, confirming that both loggers interact with the same singleton instance.

```
1 package com.accolite.Logger;
2
3 public class SingletonLoggerTest{
4
5     public static void main(String[] args){
6         SingletonLogger logger1 = SingletonLogger.getInstance();
7         SingletonLogger logger2 = SingletonLogger.getInstance();
8
9         logger1.registerStudent("Deepa");
10        logger2.getStudent();
11        logger2.removeStudent();
12        logger1.getStudent();
13    }
14 }
15 }
```

Console Output:

```
<terminated> SingletonLoggerTest [Java Application] C:\Program Files\Java\jdk-12.0.1\bin\javaw.exe (21-Jun-2019, 4:08:50 pm)
Jun 21, 2019 4:08:50 PM com.accolite.Logger.SingletonLogger registerStudent
INFO: Registered student named: Deepa
Jun 21, 2019 4:08:50 PM com.accolite.Logger.SingletonLogger getStudent
INFO: student name: Deepa
Jun 21, 2019 4:08:50 PM com.accolite.Logger.SingletonLogger removeStudent
INFO: Removed student named: Deepa
Jun 21, 2019 4:08:50 PM com.accolite.Logger.SingletonLogger getStudent
INFO: student name:
```

## Output of Travel Tciket

The screenshot shows an IDE with the following components:

- Editor:** Displays `Main.java` with the following code:

```
2
3 import com.accolite.bookingBus.BookingBus;
4 import com.accolite.bookingTrain.BookingTrain;
5
6 public class Main {
7
8     public static void main(String... args) {
9
10         // Creating objects
11         BookingBus b = new BookingBus();
12         BookingTrain t = new BookingTrain();
13
14         // Load home page and login
15         System.out.println("Load home page and login");
16         b.loadHomePage();
17         b.login();
18         System.out.println();
19
20         // getting and setting customer details
```
- Outline:** Shows the package structure: `com.acc` > `Main` > `main`.
- Console:** Shows the execution output:

```
<terminated> Main [Java Application] C:\Program Files\Java\jdk-12.0.1\bin\javaw.exe (21-Jun-2019, 4:06:54 pm)
Load home page and login
Hello in Home Page
Logged in

getting and setting customer details
customer name: Deepa
customer contact: 92837560792

getting and setting transport details
Mode of Transport: Bus
Transport mode : Adriathi
Transporting Date: 12-2-2019
Transporting time :9:30 AM
Seat number : 23

customer name: Deepa
```
- Bottom Bar:** Includes tabs for `Markers`, `Properties`, `Servers`, `Data Source Explorer`, `Snippets`, and `Console`. At the bottom, there are status indicators for `Writable`, `Smart Insert`, and a clock showing `12 : 21`.