NIT2112

Object Oriented Programming

Semester 2 Block 1, 2023

Assignment:

Flight Management System

* Prepared By:

Diwan Malla

s8073084

Table of Contents

[1. Introduction 3](#_Toc148877619)

[a. Overview 3](#_Toc148877620)

[b. Purpose 3](#_Toc148877621)

[c. Features 3](#_Toc148877622)

[2. User Authentication 4](#_Toc148877623)

[a. Description 4](#_Toc148877624)

[b. Usage 4](#_Toc148877625)

[Sample Input/Output: 4](#_Toc148877626)

[3. Flight Class 5](#_Toc148877627)

[Attributes 5](#_Toc148877628)

[Methods 5](#_Toc148877629)

[Sample Output: 5](#_Toc148877630)

[4. Flight Creation and Sorting 6](#_Toc148877631)

[Description: 6](#_Toc148877632)

[Implementation Details: 6](#_Toc148877633)

[Sample Input/Output: 6](#_Toc148877634)

[5. GUI Implementation 7](#_Toc148877635)

[Description 7](#_Toc148877636)

[Components 7](#_Toc148877637)

[Functionality 7](#_Toc148877638)

[Screenshots 7](#_Toc148877639)

[6. BYOI (Bring Your Own Idea) 8](#_Toc148877640)

[Description 8](#_Toc148877641)

[Additional Features 8](#_Toc148877642)

[Implementation Details 8](#_Toc148877643)

# **Introduction**

## **Overview**

The Flight Management System is a Java application designed to manage airline flight information. It allows user to authenticate, create and sort flight objects, and interact with the system using a graphical user interface.

## **Purpose**

The main purpose of this project is to implement and test the Java programming object-oriented concepts, user authentication, array manipulation, and GUI concept.

## **Features**

* User authentication with username and password.
* Flight class implementation with various attributes.
* Ask user to input flight details.
* Sorting flights based on flight numbers.
* Graphical User Interface for interactive user experience.

# **User Authentication**

## **Description**

The user authentication system ensures access to the Flight Management System. Users need to provide valid credentials to login.

Login credential:

(Username: "admin", Password: "admin123")

(Username: "user", Password: "user123")

## **Usage**

* Run the application.
* Enter the username and password when prompted.
* If Authentication is successful, access is granted; otherwise, access is denied.

## **Sample Input/Output:**

* When Username and password are correct.

A close up of a number

Description automatically generated

* When Username and password are incorrect.

A close up of a sign

Description automatically generated

# **Flight Class**

## **Attributes**

* airlineName : Name of the airline.
* flightNumber: Unique identifier for the flight.
* flightOrigin: Origin airport
* destinationCities: Destination airport
* airfare: Fare of the flight
* departureTime: Time of the departure in HH:mm format
* arrivalTime: Time of arrival in HH:mm format
* distance: Distance covered by the flight in kilometers.

## **Methods**

* Getter and setter methods for all attributes.
* toString() : Returns a formatted string with flight details.

## **Sample Output:**



## **Flight Creation and Sorting**

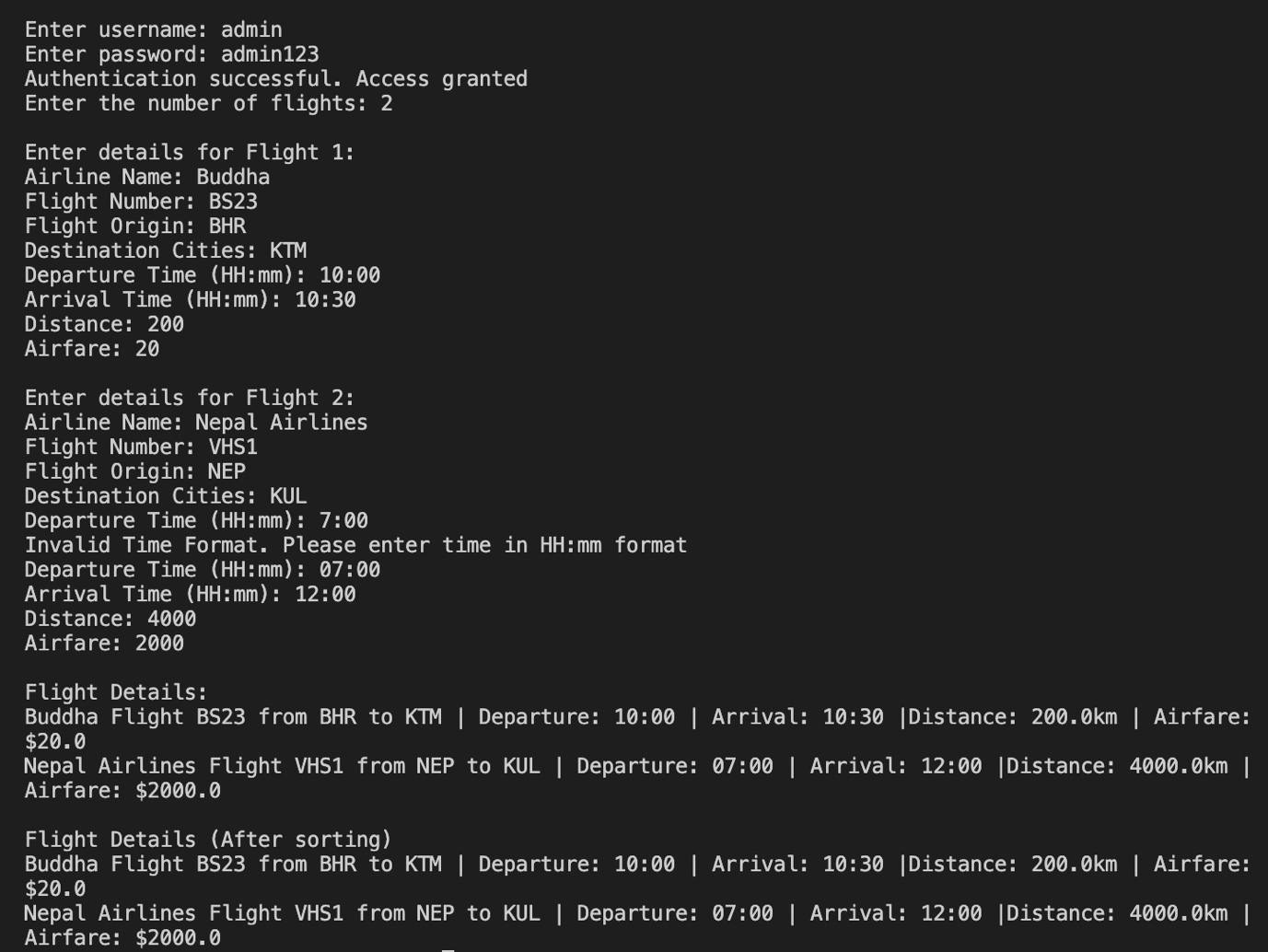
## **Description:**

Flight objects are created based on user input. The flights can be sorted based on their flight numbers.

## **Implementation Details:**

* User inputs the number of flights and their details.
* Flights detail are stored in an array of Flight objects.
* Flights are sorted using the ‘Arrays.sort()’ method and a custom comparator lambda expression.

## **Sample Input/Output:**



# **GUI Implementation**

## **Description**

The graphical user interface (GUI) allows users to interact with the Flight Management System visually. It displays flight information in table and text Area too and provides option for sorting and searching for flights.

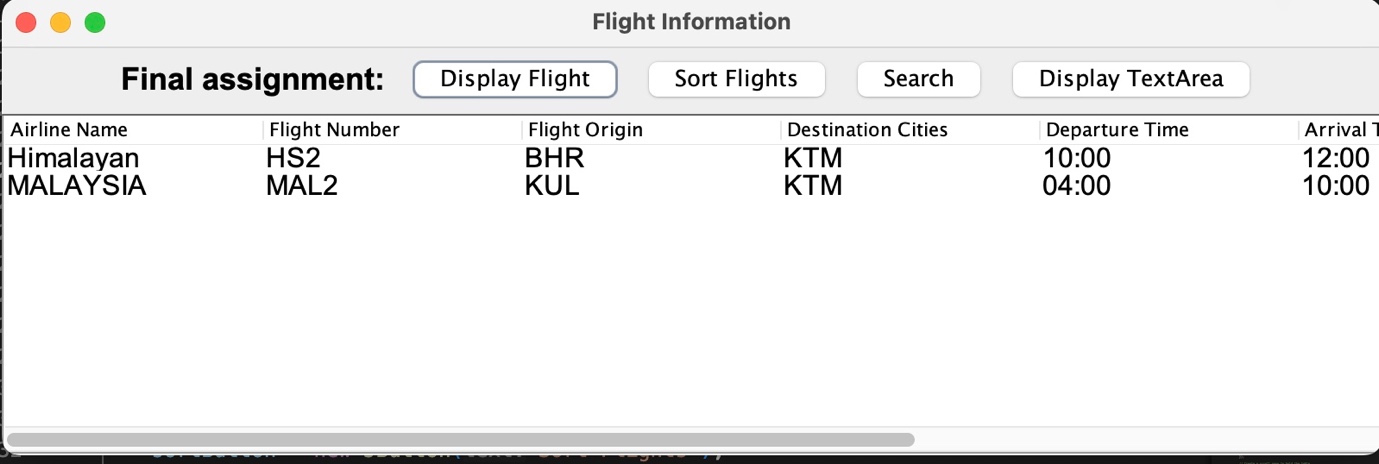
## **Components**

* JTable: Displays flight information in a tabular form
* JTextArea: Displays flight information in a text area.
* JLabel: Provide a label.
* JButtons:
* Display Flight: Populates the table with flight data.
* Sort Flight: Sorts flights based on flight numbers.
* Search: Allows users to search for flights by flight number.
* Display TextArea: Displays flight details in a text area.

## **Functionality**

* User clicks “Display Flight” to populate the table with flight data.
* User clicks “Sort Flights” to sort flights based on flight numbers.
* User clicks “Search” to search for a flight-by-flight number.
* User clicks “Display TextArea” to view flight details in a text area.

## **Screenshots**



# **BYOI (Bring Your Own Idea)**

## **Description**

This encourages students to enhance the project creativity. In this project, additional features beyond the initial requirements were implemented to enrich the user experience.

## **Additional Features**

* Search Functionality: Users can search for flights by flight number.
* Table display: Flight details are displayed in a table format.
* Error handling: There are error handling implemented when user input invalid data in program.
* User Authentication: User required to have access login credential to run this program.
* Button enable and disable: Sort button and search button are implemented in such way that they are enabled and disabled in certain steps.

## **Implementation Details**

* Search Functionality: Users input a flight number, and the system searches for the flight. If found, flight details are displayed.
* Table display: Clicking the “Display Flight” button populates the data in table and displayed it.
* Error handling: When user enter invalid time and data type then it shows error message.
* User authentication: When program is run, user requires to enter login credential. If authentication success, then it continues the program.
* Button enable and disable: When Display TextArea button is clicked, sortbutton and searchbutton gets disabled and when Display Flight button is clicked, it appears again.