

com.booking

Class FlightController

java.lang.Object
com.booking.FlightController

```
public class FlightController
extends java.lang.Object
```

This class provides methods pertaining to any required functionality and processing that is related to flights.

Field Summary

Fields

Modifier and Type	Field and Description
private FlightEntity	fe AirportController requires the FlightEntity class to write/read data to the database.
private FleetController	ftc FlightController requires the FleetController class to query plane data.
private java.util.Scanner	in Scanner allows class to use the basic input from the console.
private RouteController	rc FlightController requires the RouteController class to query route data.

Constructor Summary

Constructors

Constructor and Description
FlightController() Default constructor.

Method Summary

Methods

Modifier and Type	Method and Description
private void	addFlight() Method called by the manageFlightsOption() method when the Flight Manager chooses to add a flight.
private void	deleteFlight() Method called by the manageFlightsOption() method when the Flight Manager chooses to delete a flight.
private boolean	doesFlightExist(java.lang.String flight_id) Checks whether the flight exists.
private void	editFlight() Method called by the manageFlightsOption() method when the Flight Manager chooses to edit a flight.
private java.lang.String	enterArrivalTime() A UI method that allows the Flight Manager to enter a arrival time.
private java.lang.String	enterDepartureTime() A UI method that allows the Flight Manager to enter a departure time.
java.lang.String	enterFlightId(boolean check_exists) A UI method that allows the Flight Manager to enter a flight ID.
private Plane	enterPlane() A UI method that allows the Flight Manager to enter a Plane.
private double	enterPrice(int class_type) A UI method that allows the Flight Manager to enter the flight price.
Flight	getFlight(java.lang.String flight_id) Gets the flight based on the given flight ID.
java.util.List<Flight>	getFlights(java.lang.String origin_airport, java.lang.String destination_airport) Gets a List of Flight objects based on the given origin and destination.
java.util.List<Flight>	getFlightsForMonth(java.lang.String month, int year) Allows user to get flights for a particular month.
java.util.Map.Entry<java.lang.String, java.lang.String>	getRouteCities(int routeNumber) Gets the route cities for a given route number.
java.util.AbstractMap.SimpleImmutableEntry<java.lang.String, java.lang.String>	getRouteLocations(int routeNumber)

6/1/2015	FlightController
double[]	getSeatPrices (java.lang.String flight_id) Gets the seat prices for a certain flight.
boolean	isInternationalFlight (java.lang.String origin_airport, java.lang.String destination_airport) Checks if the flight with the origin airport and destination airport.
void	manageFlightsOption () Provides an interface for the Flight Manager to choose whether to add, edit or delete flights.
void	setSeatPrices (java.lang.String flight_id) Sets the seat prices for the given flight.
void	updateAvailableSeats (java.lang.String flight_id, int[] available_seats) Updates the available seats of a particular flight.
Methods inherited from class java.lang.Object	
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	

Field Detail

rc
<pre>private RouteController rc</pre> <p>FlightController requires the RouteController class to query route data.</p>
ftc
<pre>private FleetController ftc</pre> <p>FlightController requires the FleetController class to query plane data.</p>
fe
<pre>private FlightEntity fe</pre> <p>AirportController requires the FlightEntity class to write/read data to the database.</p>
in
<pre>private java.util.Scanner in</pre> <p>Scanner allows class to use the basic input from the console.</p>

Constructor Detail

FlightController
<pre>public FlightController()</pre> <p>Default constructor.</p>

Method Detail

manageFlightsOption
<pre>public void manageFlightsOption()</pre> <p>Provides an interface for the Flight Manager to choose whether to add, edit or delete flights.</p>
addFlight
<pre>private void addFlight()</pre> <p>Method called by the manageFlightsOption() method when the Flight Manager chooses to add a flight.</p>
editFlight
<pre>private void editFlight()</pre> <p>Method called by the manageFlightsOption() method when the Flight Manager chooses to edit a flight.</p>