Add Flight using JDBC

Zaro Flight System wants to automate the process in their organization. As a start up, they need to automate the flight management system. Help them to develop this application.

You are provided with a public class Flight with following private attribute :

int flightId

String source

String destination

int noOfSeats

double flightFare

Appropriate setter and getter are written.

A public 5 argument constructor with arguments – flightId, source, destination, noOfSeats and flightFare is also provided.

Create a class FlightManagementSystem which has the following method. Use Database for manipulation.

public boolean addFlight(Flight flightObj) - This method should accept a flight object and add that flight details into the database. If flight details are added successfully, return true. Else, return false.

To connect to the database you are provided with database.properties file and DB.java file.

The flight table is already created at the backend. The structure of flight table is:

Column Name	Datatype
flightId	int
source	varchar2(30)
destination	varchar2(30)
noofseats	int
flightfare	number(8,2)

Create a class Main which has main method to perform the above operation.

In main method,

When **addFlight** method is invoked and if added successfully, print “Flight details added successfully” else print “Addition not done”.

To execute on your machine, you can make the necessary changes to the values of connection url, username and password in the database.properties file.

Flight.java

```java

public class Flight {

        private int flightId;
        private String source;
        private String destination;
        private int noOfSeats;
        private double flightFare;
        public int getFlightId() {
                return flightId;
        }
        public void setFlightId(int flightId) {
                this.flightId = flightId;
        }
        public String getSource() {
                return source;
        }
        public void setSource(String source) {
                this.source = source;
        }
        public String getDestination() {
                return destination;
        }
        public void setDestination(String destination) {
                this.destination = destination;
        }
        public int getNoOfSeats() {
                return noOfSeats;
        }
        public void setNoOfSeats(int noOfSeats) {
                this.noOfSeats = noOfSeats;
        }
        public double getFlightFare() {
                return flightFare;
        }
        public void setFlightFare(double flightFare) {
                this.flightFare = flightFare;
        }
        public Flight(int flightId, String source, String destination,
                        int noOfSeats, double flightFare) {
                super();
                this.flightId = flightId;
                this.source = source;
                this.destination = destination;
                this.noOfSeats = noOfSeats;
                this.flightFare = flightFare;
        }



}

```

FlightManagementSystem.java

```java
import java.sql.*;
public class FlightManagementSystem
{
    public boolean addFlight(Flight flightObj)
    {
        try {

            DB db=new DB();
            Connection con=db.getConnection();
            Statement st=con.createStatement();
            int fid=flightObj.getFlightId();
            String s=flightObj.getSource();
            String d=flightObj.getDestination();
            int nos=flightObj.getNoOfSeats();
            double amnt=flightObj.getFlightFare();
            PreparedStatement ps=con.prepareStatement("insert into flight values(?,?,?,?,?)");
            ps.setInt(1,fid);
            ps.setString(2,s);
            ps.setString(3,d);
            ps.setInt(4,nos);
            ps.setDouble(5,amnt);
            int rs1=ps.executeUpdate();

        } catch(Exception e) {
            e.printStackTrace();
            return false;
        }
        return true;
    }
}
```

Main.java

```java
import java.util.*;
public class Main{
    public static void main(String[] args){
        Scanner sc=new Scanner(System.in);
        Flight f=new Flight(1234,"Mumbai","Delhi",50,15000);
        FlightManagementSystem fms=new FlightManagementSystem();
        boolean res=fms.addFlight(f);
        if(res==true)
        System.out.println("Flight details added successfully");
         else
         System.out.println("Addition not done");
    }
}
```

DB.java

```java


import java.io.FileInputStream;
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.util.Properties;

public class DB {

          private static Connection con = null;
          private static Properties props = new Properties();


    //ENSURE YOU DON'T CHANGE THE BELOW CODE WHEN YOU SUBMIT
          public static Connection getConnection() throws ClassNotFoundException, SQLException {
              try{

                                                  FileInputStream fis = null;
                                                  fis = new FileInputStream("database.properties");
                                                  props.load(fis);

                                                  // load the Driver Class
```

```
                        Class.forName(props.getProperty("DB_DRIVER_CLASS"));

                        // create the connection now
         con = DriverManager.getConnection(props.getProperty("DB_URL"),props.getProperty("DB_USERNAME"),props.getProperty("DB_PASSWORD"));
          }
          catch(IOException e){
            e.printStackTrace();
          }
                 return con;
        }
}

```

database.properties

```
#IF NEEDED, YOU CAN MODIFY THIS PROPERTY FILE
#ENSURE YOU ARE NOT CHANGING THE NAME OF THE PROPERTY
#YOU CAN CHANGE THE VALUE OF THE PROPERTY
#LOAD THE DETAILS OF DRIVER CLASS, URL, USERNAME AND PASSWORD IN DB.java using this properties file only.
#Do not hard code the values in DB.java.

DB_DRIVER_CLASS=com.mysql.jdbc.Driver
DB_URL=jdbc:mysql://localhost:3306/${sys:DB_USERNAME}
DB_USERNAME=${sys:DB_USERNAME}
DB_PASSWORD=${sys:DB_USERNAME}

```