**Airline Cargo Management System**

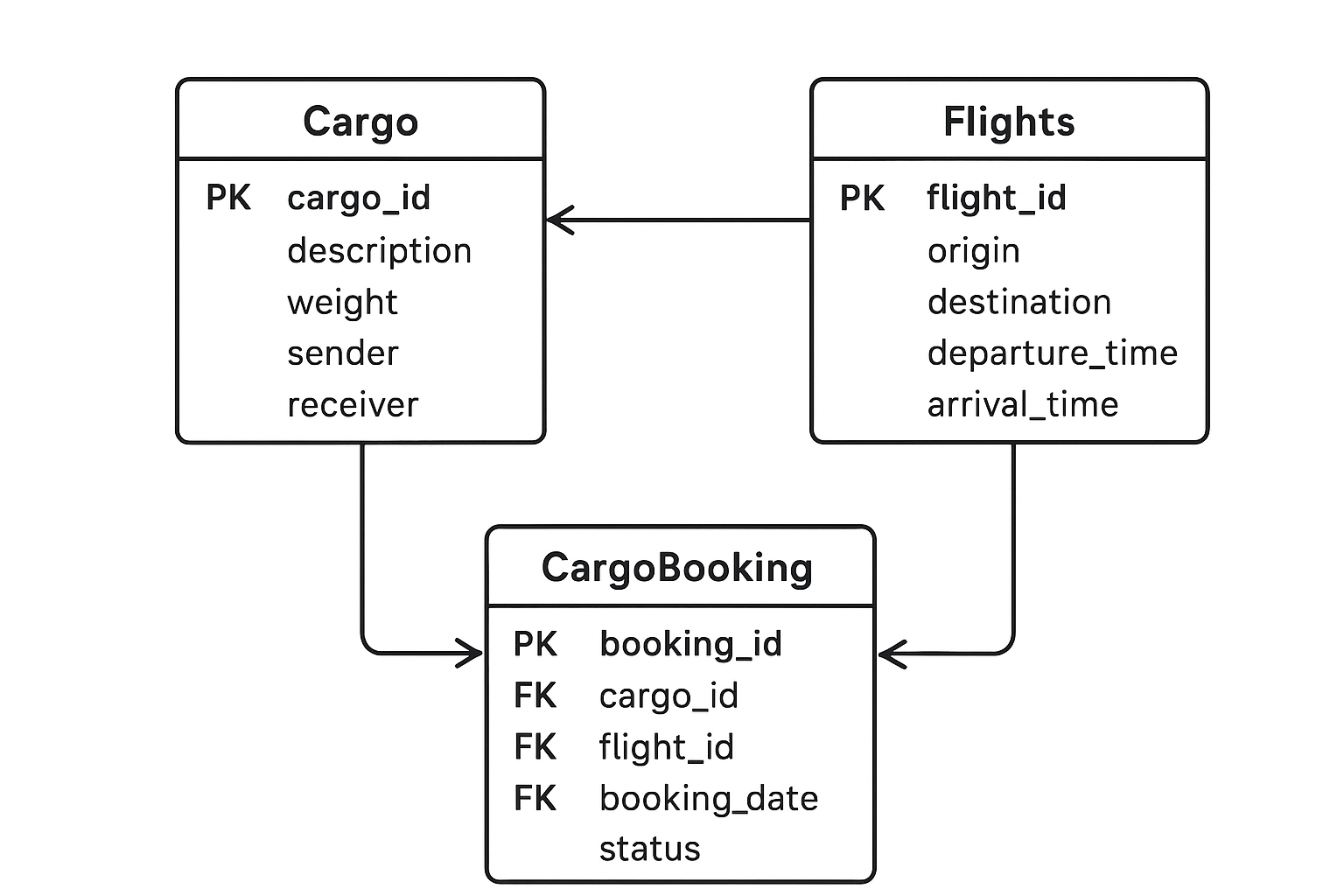
**Requirements & Use Case**

Airlines also transport **cargo** (packages, goods, freight).  
We want a system to manage:

* Cargo details (weight, type, sender, receiver)
* Flights that carry cargo
* Cargo bookings on flights
* Tracking cargo shipments

**Database Design (ER Diagram)**

Here’s the **ER Diagram** structure:



+-------------+ +--------------+ +----------------+

| Cargo | | Flights | | CargoBooking |

+-------------+ +--------------+ +----------------+

| cargo\_id PK |-------> | flight\_id PK | <-------| booking\_id PK |

| description | | origin | | cargo\_id FK |

| weight | | destination | | flight\_id FK |

| sender | | dep\_time | | booking\_date |

| receiver | | arr\_time | | status |

+-------------+ +--------------+ +----------------+

Relationship:

* A **cargo** can be booked on a **flight** (many-to-many → handled by CargoBooking).

Solution

**Oracle 19c Tables**

-- Flights table

CREATE TABLE Flights (

flight\_id NUMBER PRIMARY KEY,

origin VARCHAR2(50) NOT NULL,

destination VARCHAR2(50) NOT NULL,

departure\_time DATE NOT NULL,

arrival\_time DATE NOT NULL

);

-- Cargo table

CREATE TABLE Cargo (

cargo\_id NUMBER PRIMARY KEY,

description VARCHAR2(100),

weight NUMBER(6,2) CHECK (weight > 0),

sender VARCHAR2(50),

receiver VARCHAR2(50)

);

-- Cargo Booking table

CREATE TABLE CargoBooking (

booking\_id NUMBER PRIMARY KEY,

cargo\_id NUMBER REFERENCES Cargo(cargo\_id),

flight\_id NUMBER REFERENCES Flights(flight\_id),

booking\_date DATE DEFAULT SYSDATE,

status VARCHAR2(20) CHECK (status IN ('Booked','In Transit','Delivered'))

);

**Insert Sample Records**

-- Flights

INSERT INTO Flights VALUES (201, 'New York', 'London', TO\_DATE('2025-09-18 08:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO\_DATE('2025-09-18 20:00:00', 'YYYY-MM-DD HH24:MI:SS'));

INSERT INTO Flights VALUES (202, 'London', 'Dubai', TO\_DATE('2025-09-19 09:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO\_DATE('2025-09-19 15:00:00', 'YYYY-MM-DD HH24:MI:SS'));

-- Cargo

INSERT INTO Cargo VALUES (301, 'Electronics shipment', 500.75, 'ABC Corp', 'XYZ Ltd');

INSERT INTO Cargo VALUES (302, 'Pharmaceutical supplies', 120.50, 'MediCare', 'HealthPlus');

-- Cargo Booking

INSERT INTO CargoBooking VALUES (401, 301, 201, SYSDATE, 'Booked');

INSERT INTO CargoBooking VALUES (402, 302, 202, SYSDATE, 'In Transit');

**JDBC Menu-Driven Java Program**

package cargo;

import java.sql.\*;

import java.util.Scanner;

public class CargoManagement {

static final String URL = "jdbc:oracle:thin:@localhost:1521/orclpdb.localdomain";

static final String USER = "mydb"; // your username

static final String PASS = "oracle"; // your password

Connection con;

Scanner sc;

public CargoManagement() {

try {

Class.forName("oracle.jdbc.driver.OracleDriver");

con = DriverManager.getConnection(URL, USER, PASS);

sc = new Scanner(System.in);

} catch (Exception e) {

e.printStackTrace();

}

}

void menu() {

int choice = 0;

while (choice != 7) {

System.out.println("\n===== Airline Cargo Management =====");

System.out.println("1. Add Cargo");

System.out.println("2. View Cargo");

System.out.println("3. Add Flight");

System.out.println("4. View Flights");

System.out.println("5. Book Cargo on Flight");

System.out.println("6. View Cargo Bookings");

System.out.println("7. Exit");

System.out.print("Enter choice: ");

choice = sc.nextInt();

switch (choice) {

case 1: addCargo(); break;

case 2: viewCargo(); break;

case 3: addFlight(); break;

case 4: viewFlights(); break;

case 5: bookCargo(); break;

case 6: viewBookings(); break;

case 7: System.out.println("Goodbye!"); break;

default: System.out.println("Invalid choice!");

}

}

}

void addCargo() {

try {

System.out.print("Enter Cargo ID: ");

int cid = sc.nextInt();

sc.nextLine();

System.out.print("Enter Description: ");

String desc = sc.nextLine();

System.out.print("Enter Weight: ");

double weight = sc.nextDouble();

sc.nextLine();

System.out.print("Enter Sender: ");

String sender = sc.nextLine();

System.out.print("Enter Receiver: ");

String receiver = sc.nextLine();

PreparedStatement ps = con.prepareStatement("INSERT INTO Cargo VALUES (?,?,?,?,?)");

ps.setInt(1, cid);

ps.setString(2, desc);

ps.setDouble(3, weight);

ps.setString(4, sender);

ps.setString(5, receiver);

ps.executeUpdate();

System.out.println("Cargo added successfully!");

} catch (SQLException e) {

System.out.println("Error: " + e.getMessage());

}

}

void viewCargo() {

try {

Statement stmt = con.createStatement();

ResultSet rs = stmt.executeQuery("SELECT \* FROM Cargo");

while (rs.next()) {

System.out.println(rs.getInt(1) + " | " + rs.getString(2) +

" | " + rs.getDouble(3) + "kg | " +

rs.getString(4) + " -> " + rs.getString(5));

}

} catch (SQLException e) {

System.out.println("Error: " + e.getMessage());

}

}

void addFlight() {

try {

System.out.print("Enter Flight ID: ");

int fid = sc.nextInt();

sc.nextLine();

System.out.print("Enter Origin: ");

String origin = sc.nextLine();

System.out.print("Enter Destination: ");

String dest = sc.nextLine();

System.out.print("Enter Departure (YYYY-MM-DD HH24:MI:SS): ");

String dep = sc.nextLine();

System.out.print("Enter Arrival (YYYY-MM-DD HH24:MI:SS): ");

String arr = sc.nextLine();

PreparedStatement ps = con.prepareStatement("INSERT INTO Flights VALUES (?, ?, ?, TO\_DATE(?, 'YYYY-MM-DD HH24:MI:SS'), TO\_DATE(?, 'YYYY-MM-DD HH24:MI:SS'))");

ps.setInt(1, fid);

ps.setString(2, origin);

ps.setString(3, dest);

ps.setString(4, dep);

ps.setString(5, arr);

ps.executeUpdate();

System.out.println("Flight added successfully!");

} catch (SQLException e) {

System.out.println("Error: " + e.getMessage());

}

}

void viewFlights() {

try {

Statement stmt = con.createStatement();

ResultSet rs = stmt.executeQuery("SELECT \* FROM Flights");

while (rs.next()) {

System.out.println(rs.getInt(1) + " | " + rs.getString(2) +

" -> " + rs.getString(3) + " | " +

rs.getTimestamp(4) + " to " + rs.getTimestamp(5));

}

} catch (SQLException e) {

System.out.println("Error: " + e.getMessage());

}

}

void bookCargo() {

try {

System.out.print("Enter Booking ID: ");

int bid = sc.nextInt();

System.out.print("Enter Cargo ID: ");

int cid = sc.nextInt();

System.out.print("Enter Flight ID: ");

int fid = sc.nextInt();

sc.nextLine();

System.out.print("Enter Status (Booked/In Transit/Delivered): ");

String status = sc.nextLine();

PreparedStatement ps = con.prepareStatement("INSERT INTO CargoBooking VALUES (?, ?, ?, SYSDATE, ?)");

ps.setInt(1, bid);

ps.setInt(2, cid);

ps.setInt(3, fid);

ps.setString(4, status);

ps.executeUpdate();

System.out.println("Cargo booking successful!");

} catch (SQLException e) {

System.out.println("Error: " + e.getMessage());

}

}

void viewBookings() {

try {

Statement stmt = con.createStatement();

ResultSet rs = stmt.executeQuery(

"SELECT b.booking\_id, c.description, f.origin, f.destination, b.booking\_date, b.status " +

"FROM CargoBooking b JOIN Cargo c ON b.cargo\_id=c.cargo\_id " +

"JOIN Flights f ON b.flight\_id=f.flight\_id");

while (rs.next()) {

System.out.println("Booking ID: " + rs.getInt(1) +

" | Cargo: " + rs.getString(2) +

" | Flight: " + rs.getString(3) + " -> " + rs.getString(4) +

" | Date: " + rs.getTimestamp(5) +

" | Status: " + rs.getString(6));

}

} catch (SQLException e) {

System.out.println("Error: " + e.getMessage());

}

}

public static void main(String[] args) {

CargoManagement app = new CargoManagement();

app.menu();

}

}

**Explanation of Steps**

1. **Create tables** (Flights, Cargo, CargoBooking).
2. **Insert records** (sample flights, cargos, bookings).
3. **JDBC Program**:
   * Load driver: Class.forName("oracle.jdbc.driver.OracleDriver");
   * Connect: DriverManager.getConnection(...)
   * Menu-driven options for CRUD.
   * PreparedStatement for inserts.
   * JOIN query in **View Bookings** to show complete cargo shipment details.

Now you have a **Cargo Management System** with:

* ER Diagram
* Oracle 19c schema + records
* JDBC Java menu-driven project