

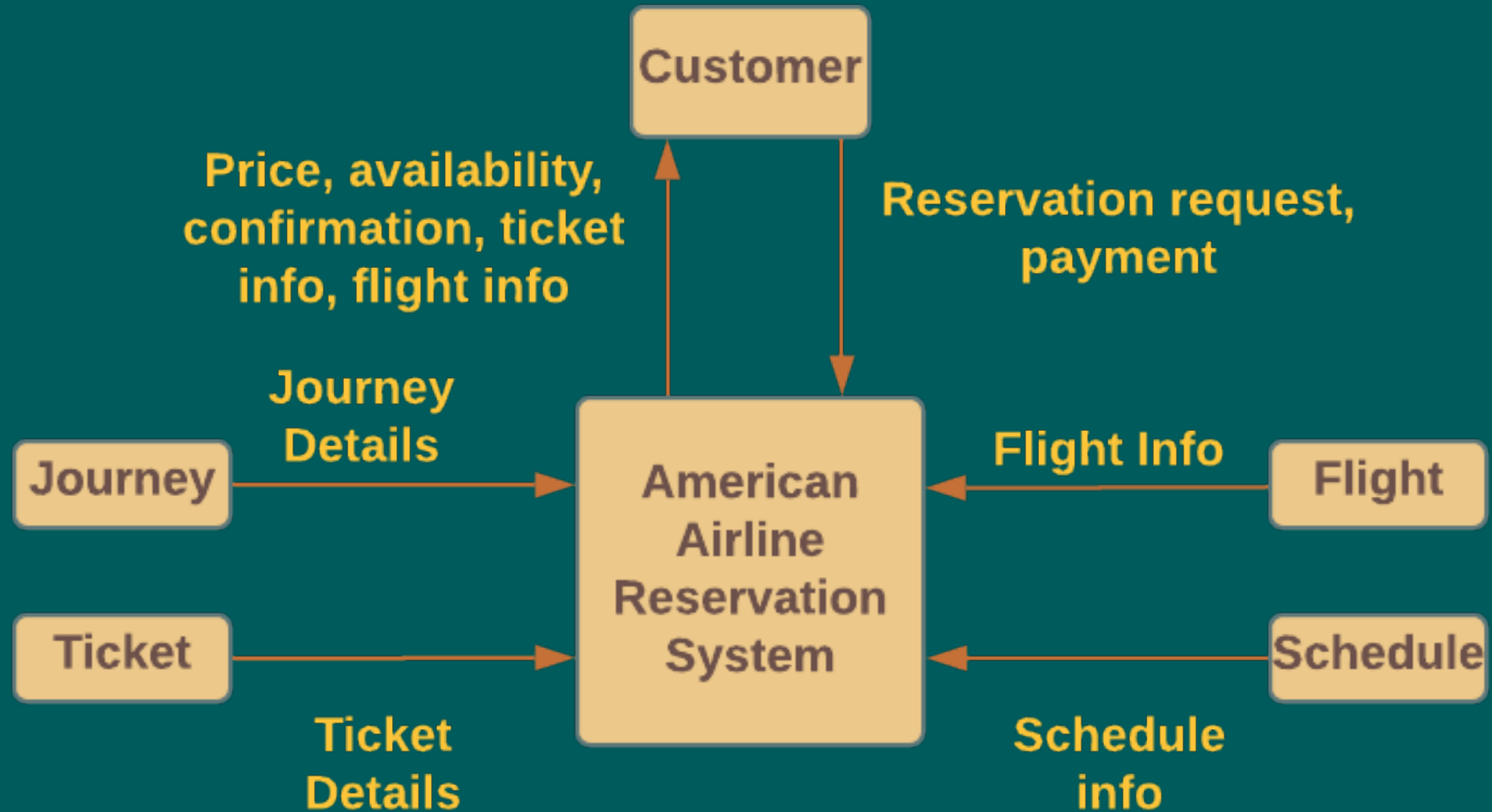
AMERICAN AIRLINES DATABASE

TEAM 1
MIS331

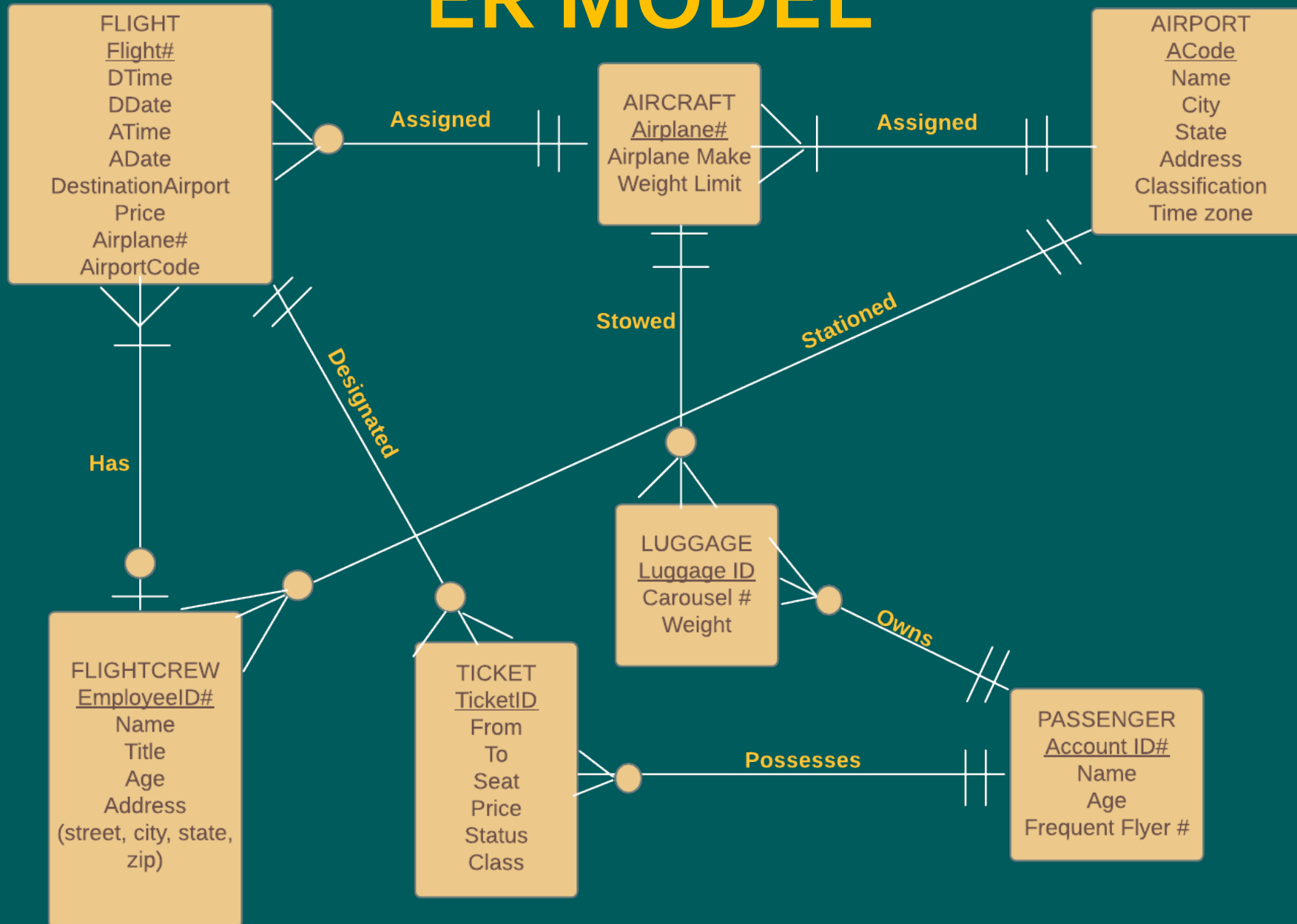
AGENDA

- Requirement Analysis
- Conceptual Model (ER Diagram)
- Relational Model
- Database Implementation
 - CRUD – Create, Read, Update, Delete

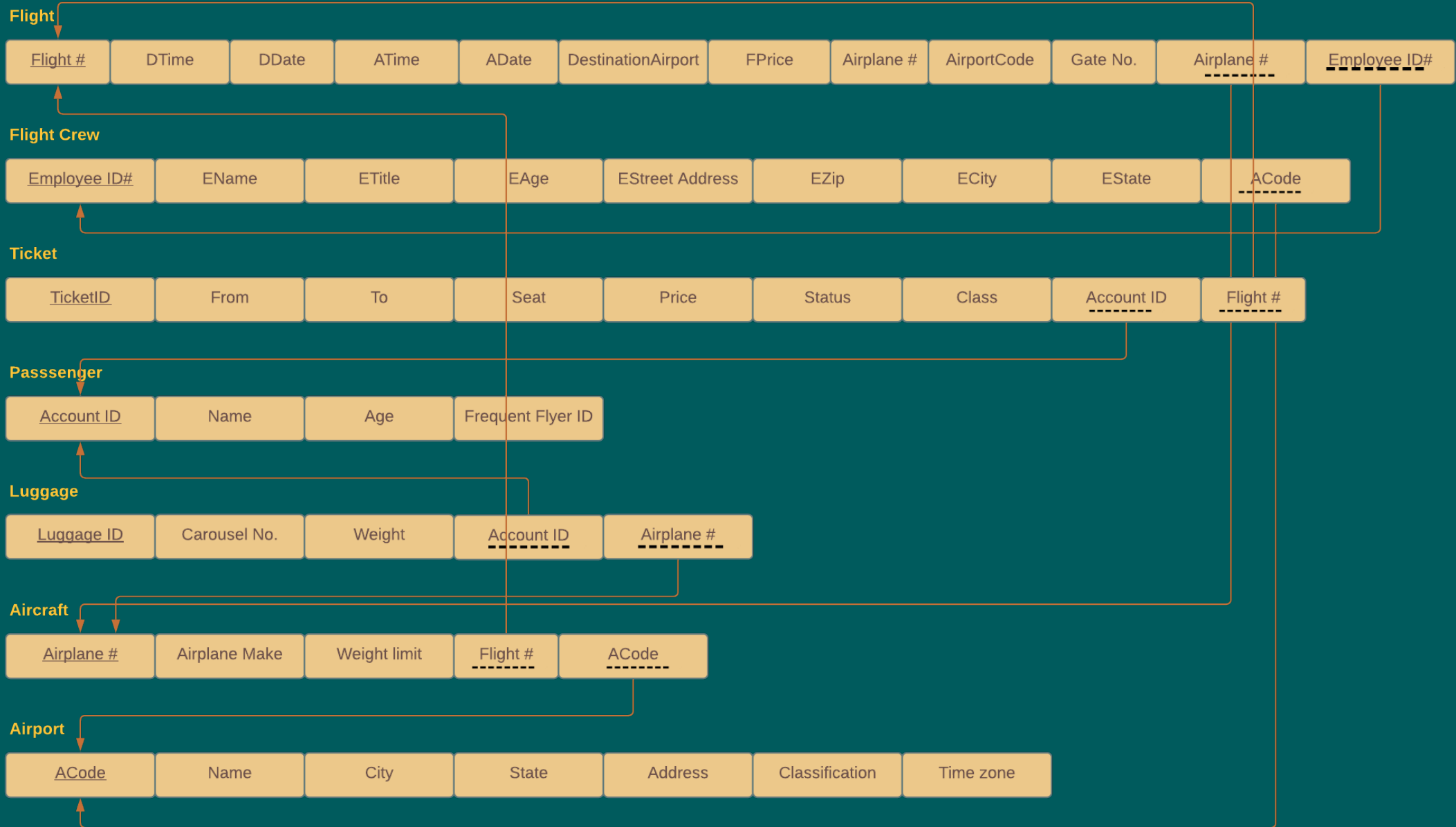
REQUIREMENT ANALYSIS



ER MODEL



RELATIONAL MODEL



CREATE & INSERT

Passenger Table

```
3  CREATE TABLE PASSENGER(  
4      AccountID INTEGER PRIMARY KEY NOT NULL,  
5      PassengerName VARCHAR(40),  
6      PassengerAge INT,  
7      FrequentflyerID INT);  
8  
9  INSERT INTO PASSENGER  
10     VALUES ("123456","Yifan Jia","20","123456");  
11  INSERT INTO PASSENGER  
12     VALUES ("136475","Mugdha Sonawane","20","136475");  
13  INSERT INTO PASSENGER  
14     VALUES ("273849","Isaiah Bae","20","273849");  
15  INSERT INTO PASSENGER  
16     VALUES ("394874","Mobeen Rana","20","394874");  
17  INSERT INTO PASSENGER  
18     VALUES ("948576","Melissa Satran","20","948576");
```

CREATE & INSERT

Airport Table

```
20 CREATE TABLE AIRPORT(  
21     AirportCode VARCHAR(8),  
22     AirportName VARCHAR(50),  
23     AirportCity VARCHAR(30),  
24     AirportState VARCHAR(30),  
25     AirportAddress VARCHAR(50),  
26     Classificarion VARCHAR(30),  
27     TimeZone VARCHAR(8),  
28     CONSTRAINT AIRPORT_PK PRIMARY KEY (AirportCode));  
29  
30 INSERT INTO AIRPORT  
31 VALUES ("ABI","Abilene Regional Airport","Abilene", "TX","6c Wireless Main Road, Airport, Tiruchirapalli","Class C international airport","GMT-6");  
32 INSERT INTO AIRPORT  
33 VALUES ("ALS","San Luis Valley Regional Airport","Alamosa", "CO","Tucson, AZ 85756","National airport","GMT-7");  
34 INSERT INTO AIRPORT  
35 VALUES ("BIL","Billings Logan International Airport","Billings", "MT","1901 Terminal Cir, Billings, Mt 59105","Regional airport","MT");  
36 INSERT INTO AIRPORT  
37 VALUES ("BGM","Greater Binghamton Airport","Binghamton", "NY","Airport Rd, Johnson City, NY 13790","Regional airport","GMT-5");  
38 INSERT INTO AIRPORT  
39 VALUES ("BOI","Boise Airport","Boise", "ID","3201 W Airport Way #1000, Boise, ID 83705","National airport","GMT-7");
```

CREATE & INSERT

Aircraft Table

```
42 CREATE TABLE AIRCRAFT(  
43     AirplaneNumber VARCHAR(30) PRIMARY KEY,  
44     AirplaneMake VARCHAR(25),  
45     WeightLimit VARCHAR(30),  
46     FlightNumber VARCHAR(30),  
47     AirportCode VARCHAR(8));  
48  
49 select * from AIRCRAFT;  
50  
51 INSERT INTO AIRCRAFT  
52 VALUES ("Boeing 737-900","titanium, composites","155,000 lbs","A320","ALS");  
53 INSERT INTO AIRCRAFT  
54 VALUES ("Airbus A320-100","titanium, steel","175,000 lbs","A321","BIL");  
55 INSERT INTO AIRCRAFT  
56 VALUES ("Embraer 190","aluminum,, steel","105,000 lbs","ERJ-145","ABI");  
57 INSERT INTO AIRCRAFT  
58 VALUES ("Boeing 777-300ER","titanium, steel","180,000 lbs","RJ 900","BOI");  
59 INSERT INTO AIRCRAFT  
60 VALUES ("Gulfstream G650","composites, steel","125,000 lbs","DL 5425","BGM");  
114 ALTER TABLE AIRCRAFT  
115 ADD FOREIGN KEY (FlightNumber) REFERENCES FLIGHT(FlightNumber);  
116  
117 ALTER TABLE AIRCRAFT  
118 ADD FOREIGN KEY (AirportCode) REFERENCES AIRPORT(AirportCode);
```


CREATE & INSERT

Flight Crew Table

```
63  CREATE TABLE FLIGHTCREW(  
64      EmployeeID INT NOT NULL,  
65      EmployeeName VARCHAR(25),  
66      EmployeeTitle VARCHAR(25),  
67      EmployeeAge INT,  
68      EmployeeStreetAddress VARCHAR(50),  
69      EmployeeZipCode INT,  
70      EmployeeCity VARCHAR(30),  
71      EmployeeState VARCHAR(30),  
72      AirportCode VARCHAR(8),  
73      CONSTRAINT FLIGHTCREW_PK PRIMARY KEY (EmployeeID),  
74      CONSTRAINT FLIGHTCREW_FK FOREIGN KEY(AirportCode) REFERENCES AIRPORT(AirportCode));  
75  
76  
77  INSERT INTO FLIGHTCREW  
78  VALUES ("1859006","Virat Kohli","Passenger assistant","25","330 1st Ave","85003","Dallas","TX","ABI");  
79  INSERT INTO FLIGHTCREW  
80  VALUES ("3847905","Steve Smith","Airline food servicer","35","10400 Belle Creek Blvd","80640","Denver","CO","ALS");  
81  INSERT INTO FLIGHTCREW  
82  VALUES ("1384006","Mitchel Johnson","Airline security officer","32","2747 NY-28, Old Forge","13420","Old Forge","NY","BGM");  
83  INSERT INTO FLIGHTCREW  
84  VALUES ("9498378","James Anderson","Airline baggage handler","29","317 W Main St","59715","Bozeman","MT","BIL");  
85  INSERT INTO FLIGHTCREW  
86  VALUES ("2738904","Kevin White","Aircraft fueler","40","670 Julia Davis Dr","83702","Boise","ID","BOI");
```

CREATE & INSERT

Flight Table

```
89  CREATE TABLE FLIGHT(  
90      FlightNumber VARCHAR(30),  
91      Departure TIME,  
92      Arrival TIME,  
93      FlightPrice VARCHAR(8),  
94      AirportCode VARCHAR(8),  
95      GateNumber VARCHAR(8),  
96      AirplaneNumber VARCHAR(30),  
97      EmployeeID INT NOT NULL,  
98      CONSTRAINT FLIGHT_PK PRIMARY KEY (FlightNumber),  
99      CONSTRAINT FLIGHT_FK1 FOREIGN KEY (AirplaneNumber) REFERENCES AIRCRAFT( AirplaneNumber),  
100     CONSTRAINT FLIGHT_FK2 FOREIGN KEY (EmployeeID) REFERENCES FLIGHTCREW(EmployeeID));  
101  
102  INSERT INTO FLIGHT  
103  VALUES ("DL 5425","6:49","13:50","$820","BGM","B15","Gulfstream G650","1384006");  
104  INSERT INTO FLIGHT  
105  VALUES ("ERJ-145","16:42","19:54","$300","ABI","C11","Embraer 190","1859006");  
106  INSERT INTO FLIGHT  
107  VALUES ("RJ 900","6:45","11:59","$247","B0I","C16","Boeing 777-300ER","2738904");  
108  INSERT INTO FLIGHT  
109  VALUES ("A320","8:00","10:30","$773","ALS","A6","Boeing 737-900","3847905");  
110  INSERT INTO FLIGHT  
111  VALUES ("A321","13:37","22:26","$309","BIL","D14","Airbus A320-100","9498378");
```

CREATE & INSERT

Ticket Table

```
121 CREATE TABLE TICKET(  
122     TicketID VARCHAR(20) PRIMARY KEY,  
123     FromLocation VARCHAR(50),  
124     ToLocation VARCHAR(50),  
125     Seat CHAR(3),  
126     Price VARCHAR(5),  
127     Flightstatus VARCHAR(20),  
128     Class VARCHAR(20),  
129     AccountID INT NOT NULL,  
130     FlightNumber VARCHAR(30),  
131     CONSTRAINT TICKET_FK1 FOREIGN KEY(AccountID)  
132     REFERENCES PASSENGER(AccountID),  
133     CONSTRAINT TICKET_FK2 FOREIGN KEY(FlightNumber)  
134     REFERENCES FLIGHT(FlightNumber));  
135  
136 SELECT * FROM TICKET;  
137  
138 INSERT INTO TICKET  
139 VALUES (182774950834,"7250 S Tucson Blvd, Tucson, AZ 85756","Airport Rd, Johnson City, NY 13790","D15","$820","In flight","First class","123456","DL 5425");  
140 INSERT INTO TICKET  
141 VALUES (281049372994,"6000 N Terminal Pkwy, Atlanta, GA 30320","Tucson, AZ 85756","B07","$773","On time","Business class","136475","A320");  
142 INSERT INTO TICKET  
143 VALUES (012837649303,"2400 Aviation Dr, DFW Airport, TX 75261","1901 Terminal Cir, Billings, Mt 59105","C12","$309","On time","Business class","273849","A321");  
144 INSERT INTO TICKET  
145 VALUES (394803058740,"8500 Peña Blvd, Denver, CO 80249","6c Wireless Main Road, Airport, Tiruchirapalli","D22","$300","Delayed","Business class","394874","ERJ-145");  
146 INSERT INTO TICKET  
147 VALUES (023480394748,"10000 W O'Hare Ave, Chicago, IL 60666","3201 W Airport Way #1000, Boise, ID 83705","A5","$247","On time","Business class","948576","RJ 900");
```

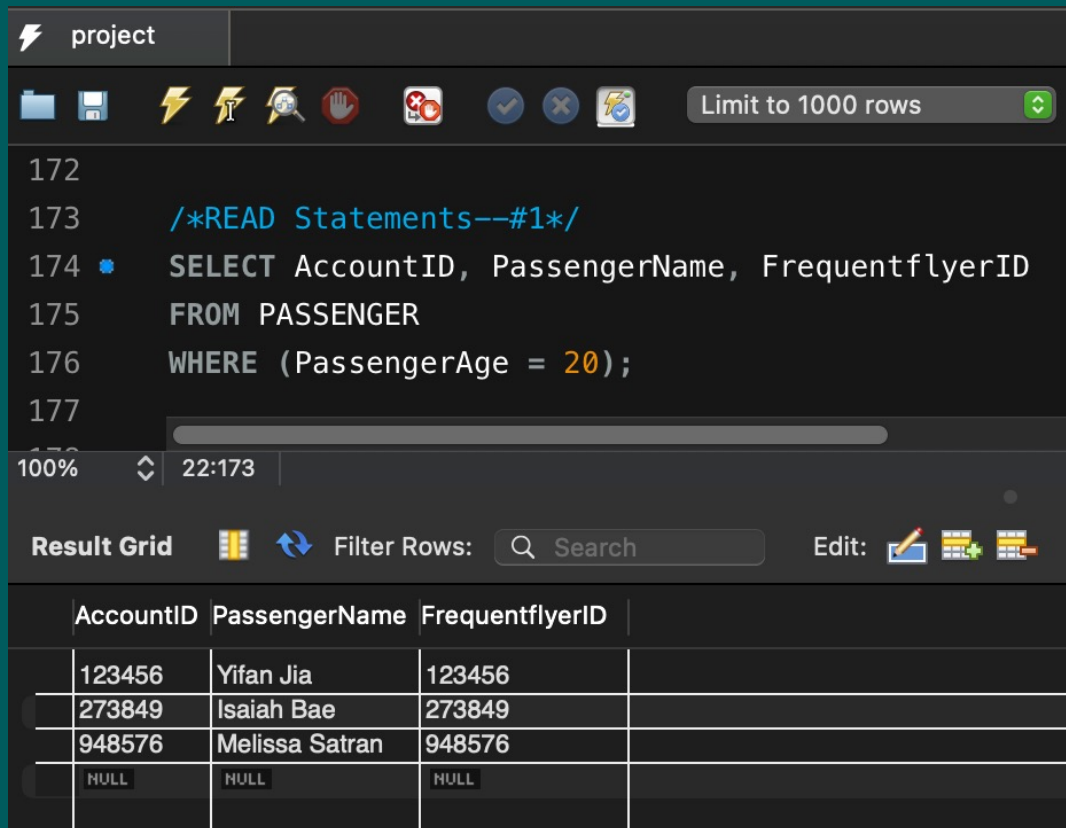
CREATE & INSERT

Luggage Table

```
149 CREATE TABLE LUGGAGE(  
150     LuggageID VARCHAR(20) PRIMARY KEY,  
151     CarouseIDNumber INT NOT NULL,  
152     Weight VARCHAR(20),  
153     AccountID INT NOT NULL,  
154     AirplaneNumber VARCHAR(30),  
155     CONSTRAINT LUGGAGE_FK1 FOREIGN KEY(AccountID) REFERENCES PASSENGER(AccountID),  
156     CONSTRAINT LUGGAGE_FK2 FOREIGN KEY(AirplaneNumber) REFERENCES AIRCRAFT(AirplaneNumber));  
157  
158 SELECT * FROM LUGGAGE;  
159 INSERT INTO LUGGAGE  
160 VALUES ("2129873","3","22 pounds","123456","Airbus A320-100");  
161 INSERT INTO LUGGAGE  
162 VALUES ("2349434","2","50 pounds","136475","Boeing 737-900");  
163 INSERT INTO LUGGAGE  
164 VALUES ("3892834","7","26.5 pounds","273849","Boeing 777-300ER");  
165 INSERT INTO LUGGAGE  
166 VALUES ("3429953","11","22 pounds","394874","Embraer 190");  
167 INSERT INTO LUGGAGE  
168 VALUES ("6050795","1","15 pounds","948576","Gulfstream G650");
```


| READ | JOIN

Passengers who were 20 years old



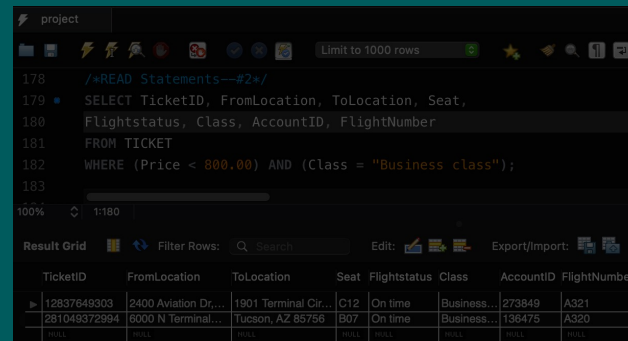
The screenshot shows a database IDE with a SQL query editor and a result grid. The query is:

```
/*READ Statements--#1*/  
SELECT AccountID, PassengerName, FrequentflyerID  
FROM PASSENGER  
WHERE (PassengerAge = 20);
```

The result grid displays the following data:

AccountID	PassengerName	FrequentflyerID
123456	Yifan Jia	123456
273849	Isaiah Bae	273849
948576	Melissa Satran	948576
NULL	NULL	NULL

TicketIDs of Passengers who were in Business Class AND had ticket cost of <\$800



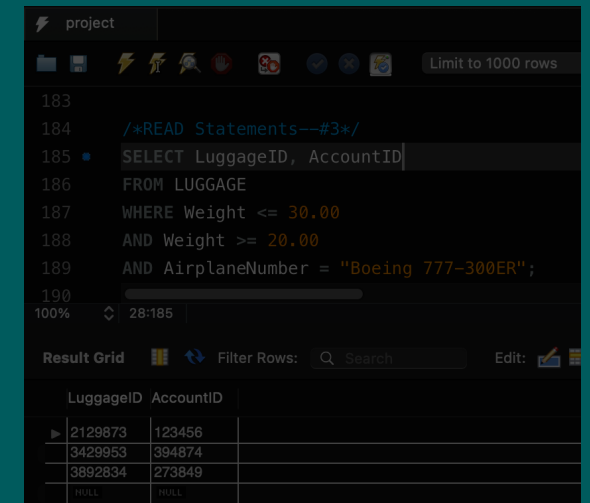
The screenshot shows a database IDE with a SQL query editor and a result grid. The query is:

```
/*READ Statements--#2*/  
SELECT TicketID, FromLocation, ToLocation, Seat,  
Flightstatus, Class, AccountID, FlightNumber  
FROM TICKET  
WHERE (Price < 800.00) AND (Class = "Business class");
```

The result grid displays the following data:

TicketID	FromLocation	ToLocation	Seat	Flightstatus	Class	AccountID	FlightNumber
12837649303	2400 Aviation Dr...	1901 Terminal Cir...	C12	On time	Business...	273849	A321
281049372994	6000 N Terminal...	Tucson, AZ 85756	B07	On time	Business...	136475	A320
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

LuggageID, AccountID of luggage whose weight was <= 30 pounds AND >= 320 pounds, and which were on the Boeing 777-300ER aircraft



The screenshot shows a database IDE with a SQL query editor and a result grid. The query is:

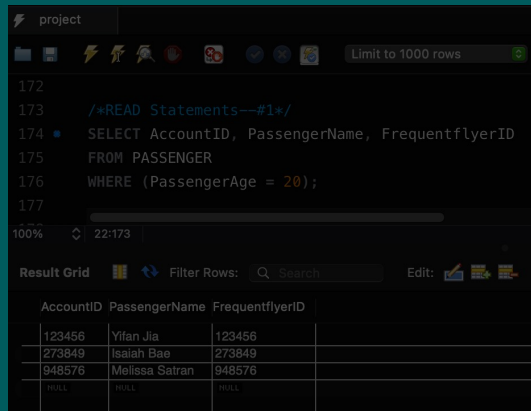
```
/*READ Statements--#3*/  
SELECT LuggageID, AccountID  
FROM LUGGAGE  
WHERE Weight <= 30.00  
AND Weight >= 20.00  
AND AirplaneNumber = "Boeing 777-300ER";
```

The result grid displays the following data:

LuggageID	AccountID
2129873	123456
3429953	394874
3892834	273849
NULL	NULL

| READ | JOIN

Passengers who were 20 years old

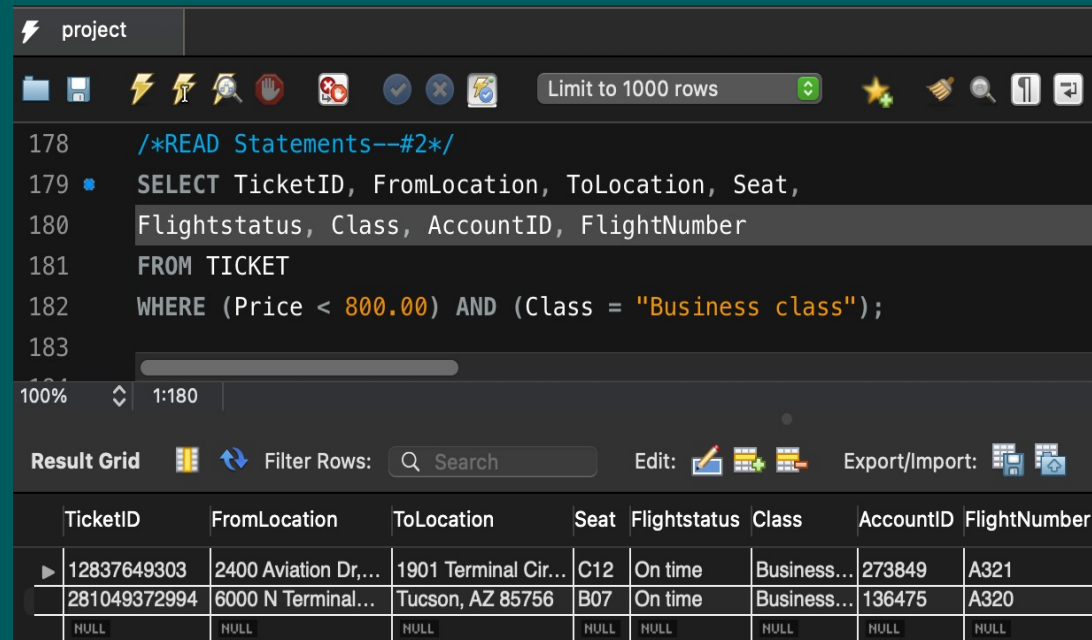


```
172
173 /*READ Statements--#1*/
174 SELECT AccountID, PassengerName, FrequentflyerID
175 FROM PASSENGER
176 WHERE (PassengerAge = 20);
177
```

Result Grid

AccountID	PassengerName	FrequentflyerID
123456	Yifan Jia	123456
273849	Isalah Bae	273849
948576	Melissa Satran	948576
NULL	NULL	NULL

TicketIDs of Passengers who
were in Business Class AND
had ticket cost of <\$800

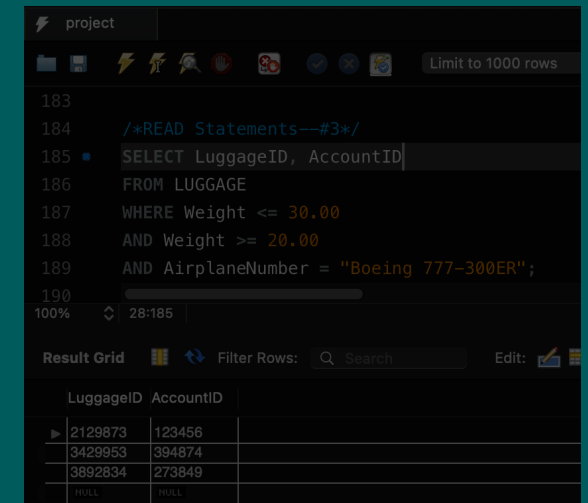


```
178 /*READ Statements--#2*/
179 SELECT TicketID, FromLocation, ToLocation, Seat,
180 Flightstatus, Class, AccountID, FlightNumber
181 FROM TICKET
182 WHERE (Price < 800.00) AND (Class = "Business class");
183
```

Result Grid

TicketID	FromLocation	ToLocation	Seat	Flightstatus	Class	AccountID	FlightNumber
12837649303	2400 Aviation Dr...	1901 Terminal Cir...	C12	On time	Business...	273849	A321
281049372994	6000 N Terminal...	Tucson, AZ 85756	B07	On time	Business...	136475	A320
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

LuggageID, AccountID of luggage
whose weight was <= 30 pounds AND
>= 320 pounds, and which were on the
Boeing 777-300ER aircraft



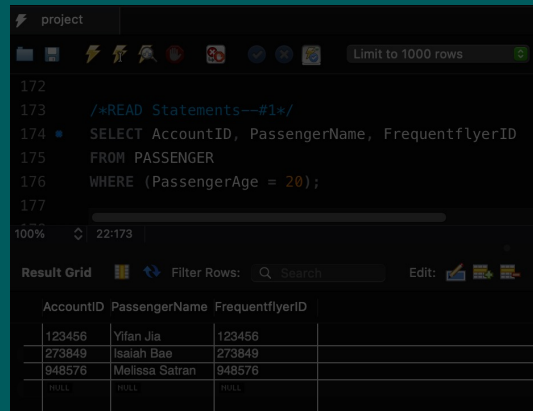
```
183
184 /*READ Statements--#3*/
185 SELECT LuggageID, AccountID
186 FROM LUGGAGE
187 WHERE Weight <= 30.00
188 AND Weight >= 20.00
189 AND AirplaneNumber = "Boeing 777-300ER";
190
```

Result Grid

LuggageID	AccountID
2129873	123456
3429953	394874
3892834	273849
NULL	NULL

| READ | JOIN

Passengers who were 20 years old

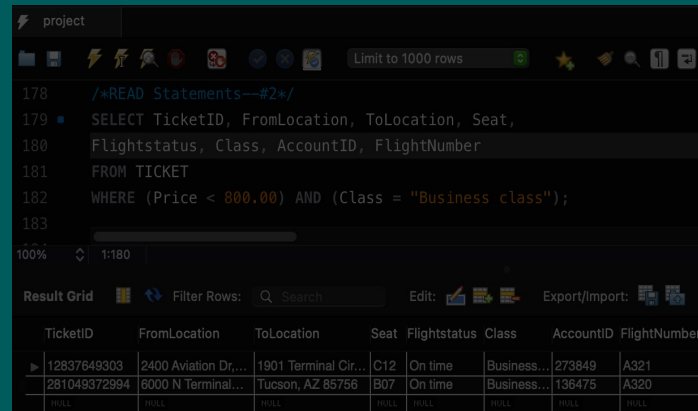


```
172
173 /*READ Statements--#1*/
174 SELECT AccountID, PassengerName, FrequentflyerID
175 FROM PASSENGER
176 WHERE (PassengerAge = 20);
177
```

Result Grid

AccountID	PassengerName	FrequentflyerID
123456	Yifan Jia	123456
273849	Isalah Bae	273849
948576	Melissa Satran	948576
NULL	NULL	NULL

TicketIDs of Passengers who were
in Business Class AND had ticket
cost of <\$800

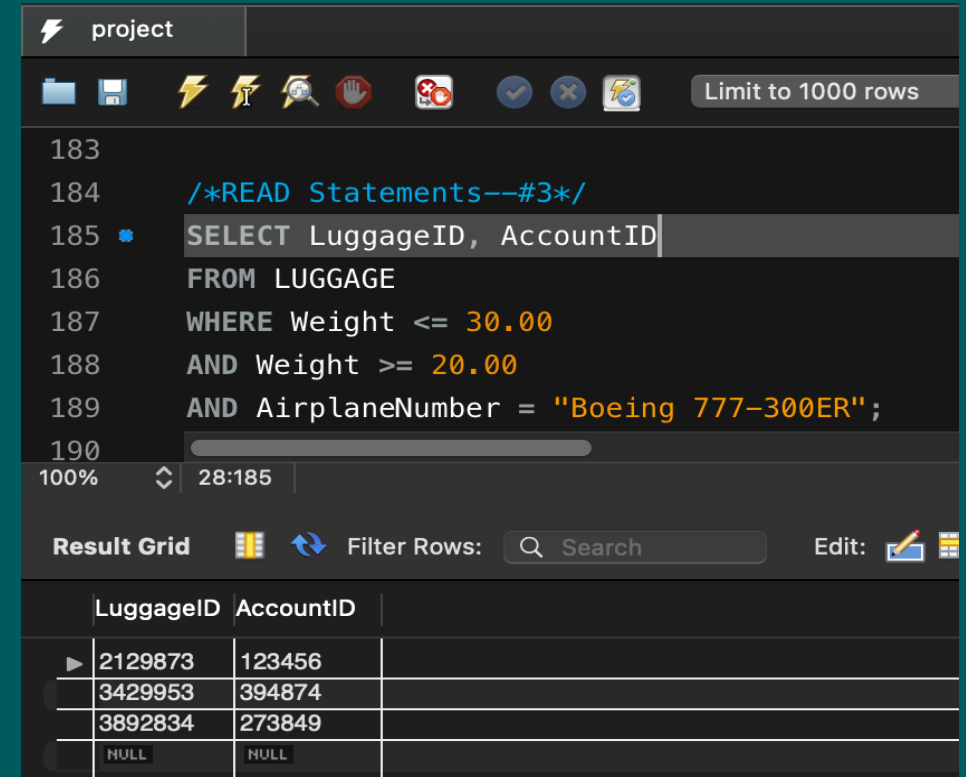


```
178 /*READ Statements--#2*/
179 SELECT TicketID, FromLocation, ToLocation, Seat,
180 Flightstatus, Class, AccountID, FlightNumber
181 FROM TICKET
182 WHERE (Price < 800.00) AND (Class = "Business class");
183
```

Result Grid

TicketID	FromLocation	ToLocation	Seat	Flightstatus	Class	AccountID	FlightNumber
12837649303	2400 Aviation Dr...	1901 Terminal Cir...	C12	On time	Business...	273849	A321
261049372994	6000 N Terminal...	Tucson, AZ 85756	B07	On time	Business...	136475	A320
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

LuggageID, AccountID of luggage
whose weight was <= 30 pounds
AND >= 20 pounds, and which were
on the Boeing 777-300ER aircraft



```
183
184 /*READ Statements--#3*/
185 SELECT LuggageID, AccountID
186 FROM LUGGAGE
187 WHERE Weight <= 30.00
188 AND Weight >= 20.00
189 AND AirplaneNumber = "Boeing 777-300ER";
190
```

Result Grid

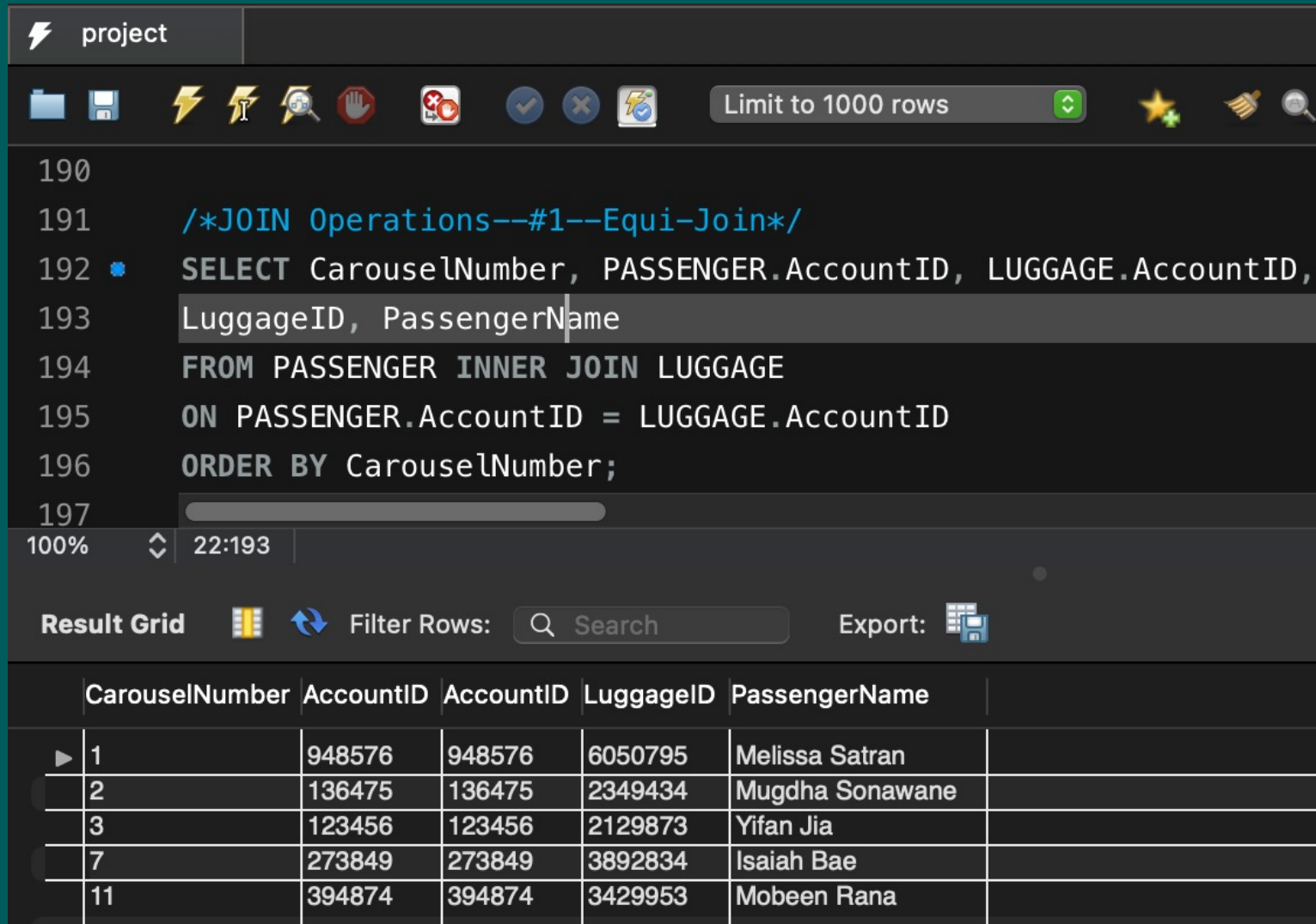
LuggageID	AccountID
2129873	123456
3429953	394874
3892834	273849
NULL	NULL

| READ | JOIN |

INNER JOIN

Example #1

Ordering Luggage by Carousel Number

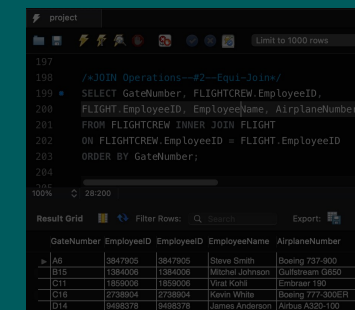


The screenshot shows a SQL IDE interface with a query editor and a result grid. The query is an INNER JOIN between PASSENGER and LUGGAGE tables, ordered by CarouselNumber. The result grid displays 5 rows of data.

```
190
191 /*JOIN Operations--#1--Equi-Join*/
192 SELECT CarouselNumber, PASSENGER.AccountID, LUGGAGE.AccountID,
193        LuggageID, PassengerName
194 FROM PASSENGER INNER JOIN LUGGAGE
195 ON PASSENGER.AccountID = LUGGAGE.AccountID
196 ORDER BY CarouselNumber;
```

	CarouselNumber	AccountID	AccountID	LuggageID	PassengerName
▶	1	948576	948576	6050795	Melissa Satran
	2	136475	136475	2349434	Mugdha Sonawane
	3	123456	123456	2129873	Yifan Jia
	7	273849	273849	3892834	Isaiah Bae
	11	394874	394874	3429953	Mobeen Rana

EQUI JOIN

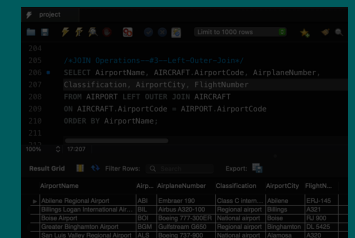


The screenshot shows a SQL IDE interface with a query editor and a result grid. The query is an EQUI JOIN between FLIGHTCREW and FLIGHT tables, ordered by GateNumber. The result grid displays 5 rows of data.

```
197
198 /*JOIN Operations--#2--Equi-Join*/
199 SELECT GateNumber, FLIGHTCREW.EmployeeID,
200        FLIGHT.EmployeeID, EmployeeName, AirplaneNumber
201 FROM FLIGHTCREW INNER JOIN FLIGHT
202 ON FLIGHTCREW.EmployeeID = FLIGHT.EmployeeID
203 ORDER BY GateNumber;
```

GateNumber	EmployeeID	EmployeeID	EmployeeName	AirplaneNumber
A6	3847905	3847905	Slove Smith	Boeing 737-900
B15	1384006	1384006	Michael Johnson	Gulfstream G650
D11	1869006	1869006	Vital Kori	Embraer 175
D18	2738904	2738904	Karen White	Boeing 777-300ER
D14	9488378	9488378	James Anderson	Airbus A320-100

LEFT OUTER JOIN



The screenshot shows a SQL IDE interface with a query editor and a result grid. The query is a LEFT OUTER JOIN between AIRPORT and AIRCRAFT tables, ordered by AirportCode. The result grid displays 5 rows of data.

```
204
205 /*JOIN Operations--#3--Left Outer-Join*/
206 SELECT AirportName, AIRCRAFT.AirportCode, AirplaneNumber,
207        Classification, AircraftType, FlightNumber
208 FROM AIRPORT LEFT OUTER JOIN AIRCRAFT
209 ON AIRCRAFT.AirportCode = AIRPORT.AirportCode
210 ORDER BY AirportName;
```

AirportName	Airp...	AirplaneNumber	Classification	AircraftType	Flight...
Airport Regional Report	AP	Boeing 787	Wide Body	Boeing	Boeing 787-9
Boeing 737-900	737	Boeing 737-900	Narrow body	Boeing	Boeing 737-900
Boeing 777-300ER	777	Boeing 777-300ER	Narrow body	Boeing	Boeing 777-300ER
Boeing 787-9	787	Boeing 787-9	Narrow body	Boeing	Boeing 787-9

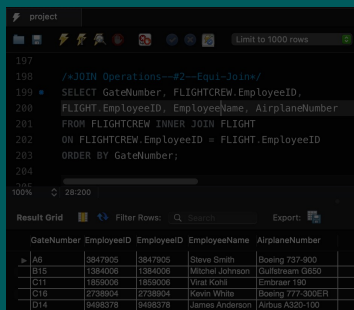
| READ | JOIN |

LEFT OUTER JOIN

Ordering Aircrafts by Airports

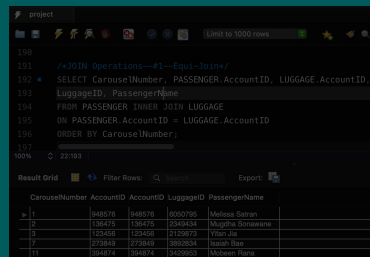
EQUI JOIN

EQUI JOIN



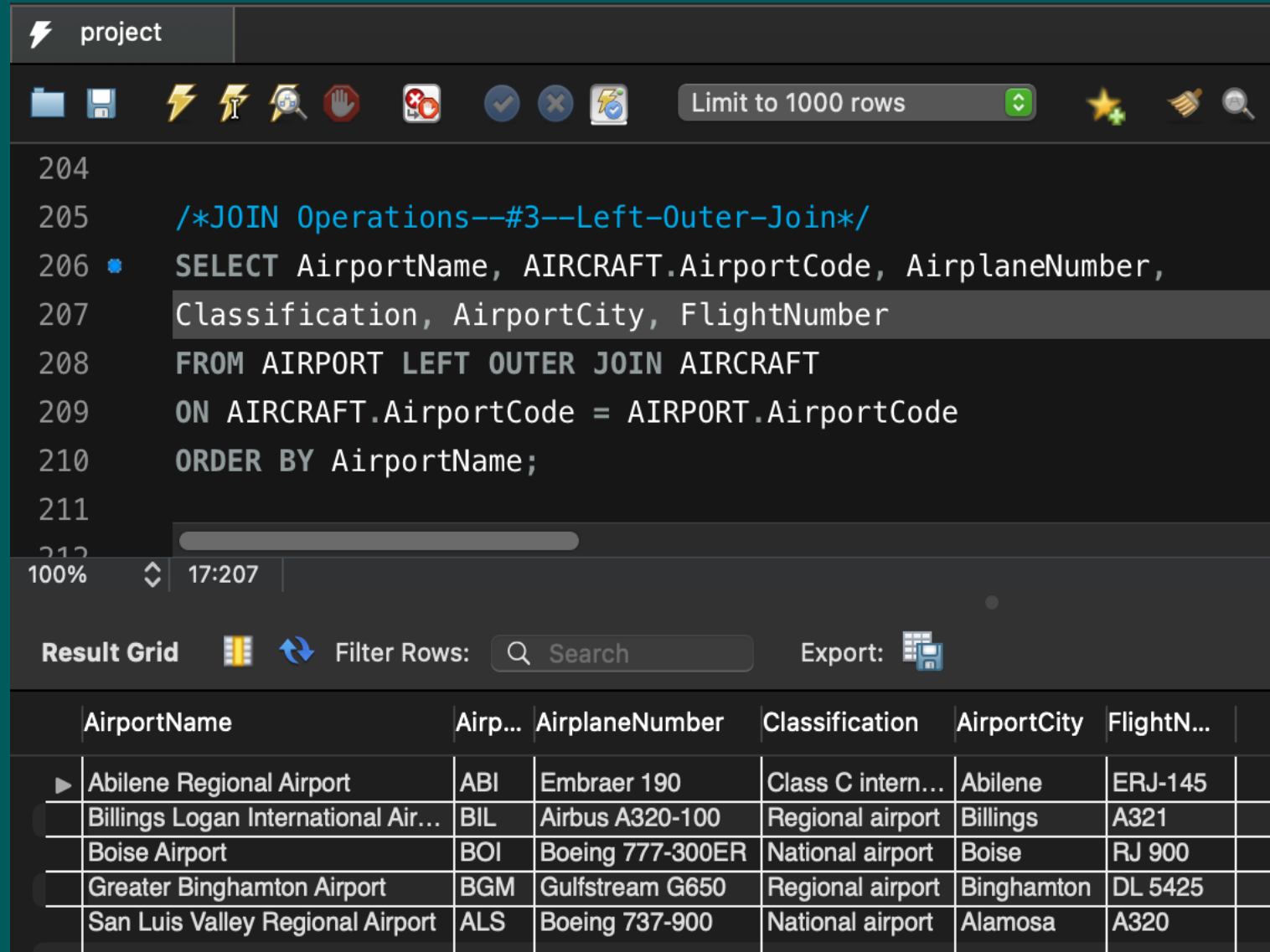
```
197 /*JOIN Operations--#2--EQUI-Join*/
198 SELECT GateNumber, FLIGHTCREW.EmployeeID,
199 FLIGHT.EmployeeID, EmployeeName, AirplaneNumber
200 FROM FLIGHTCREW INNER JOIN FLIGHT
201 ON FLIGHTCREW.EmployeeID = FLIGHT.EmployeeID
202 ORDER BY GateNumber;
```

GateNumber	EmployeeID	EmployeeID	EmployeeName	AirplaneNumber
AB	1347005	1347005	Steve Smith	Boeing 737-900
BIS	1280006	1280006	Michael Johnson	Boeing 737-900
CTT	1850006	1850006	Vinod Kulkarni	Boeing 737-900
CTB	2738004	2738004	Karen White	Boeing 737-900
CTA	1488378	1488378	James Anderson	Airbus A320-100



```
199 /*JOIN Operations--#1--EQUI-Join*/
200 SELECT CarouseNumber, PASSENGER.AccountID, LUGGAGE.AccountID,
201 LuggageID, PassengerName
202 FROM PASSENGER INNER JOIN LUGGAGE
203 ON PASSENGER.AccountID = LUGGAGE.AccountID
204 ORDER BY CarouseNumber;
```

CarouseNumber	AccountID	AccountID	LuggageID	PassengerName
1	1488378	1488378	1488378	Michael Johnson
2	1280006	1280006	1280006	Michael Johnson
3	1280006	1280006	1280006	Michael Johnson
4	1280006	1280006	1280006	Michael Johnson
5	1280006	1280006	1280006	Michael Johnson
6	1280006	1280006	1280006	Michael Johnson
7	1280006	1280006	1280006	Michael Johnson
8	1280006	1280006	1280006	Michael Johnson
9	1280006	1280006	1280006	Michael Johnson
10	1280006	1280006	1280006	Michael Johnson



```
204
205 /*JOIN Operations--#3--Left-Outer-Join*/
206 SELECT AirportName, AIRCRAFT.AirportCode, AirplaneNumber,
207 Classification, AirportCity, FlightNumber
208 FROM AIRPORT LEFT OUTER JOIN AIRCRAFT
209 ON AIRCRAFT.AirportCode = AIRPORT.AirportCode
210 ORDER BY AirportName;
```

AirportName	Airp...	AirplaneNumber	Classification	AirportCity	FlightN...
Abilene Regional Airport	ABI	Embraer 190	Class C intern...	Abilene	ERJ-145
Billings Logan International Air...	BIL	Airbus A320-100	Regional airport	Billings	A321
Boise Airport	BOI	Boeing 777-300ER	National airport	Boise	RJ 900
Greater Binghamton Airport	BGM	Gulfstream G650	Regional airport	Binghamton	DL 5425
San Luis Valley Regional Airport	ALS	Boeing 737-900	National airport	Alamosa	A320

UPDATE

```
202 * UPDATE LUGGAGE
203     SET WEIGHT = 25
204     WHERE LuggageID = 2129873;
205
206 * UPDATE TICKET
207     SET Class = "First Class"
208     WHERE TicketID = 281049372994;
209
210 * UPDATE FLIGHT
211     SET Departure = "9:00"
212     WHERE FlightNumber = "A320";
213
214 * UPDATE FLIGHTCREW
215     SET EmployeeAge = 36
216     WHERE EmployeeID = "3847905";
217
218 * UPDATE PASSENGER
219     SET PassengerAge = "21"
220     WHERE PassengerName = "Isaiah Bae";
```

The weight of the luggage with the ID
"2129873" is updated to 25 units

The class for the ticket with the ID
"281049372994" is upgraded to first class

The departure time of flight number
A320 is updated to 9:00

The age of the employee with the ID
"3847905" is updated to 36

The age of passenger "Isaiah Bae"
is updated to 21

DELETE

```
222  *   DELETE From LUGGAGE
```

```
223      WHERE WEIGHT > 50 ;
```

```
224
```

```
225  *   DELETE From FLIGHTCREW
```

```
226      WHERE EmployeeID = '1859006';
```

```
227
```

```
228  *   DELETE From FLIGHT
```

```
229      WHERE EmployeeID = '1859006';
```

```
230
```

```
231  *   DELETE From Ticket
```

```
232      WHERE Flightstatus = 'arrived';
```

```
233
```

```
234  *   DELETE From PASSENGER
```

```
235      WHERE FrequentflyerID = '948576';
```

Records are deleted from the luggage table for all luggage that is heavier than 50 units

Records are deleted for the flight crew table for employee 1859006

The employee with the ID "1859006" is removed from their previously assigned flight

The ticket records are deleted from the ticket table for tickets where the flight status is arrived

Records are deleted from the passenger table for the passenger with the frequent flyer ID "948576"

Thank you!