1. List the name, age and EID of employees who are certified to fly \*all\* aircraft (AID) in the database that have a cruising range greater than 3000 miles.

AircraftOver3000 <- PI\_AID (SIGMA\_CruisingRange > 3000)

CertifiedEmp <- PI\_EID(CERTIFIED\_TO\_FLY DIVIDE AircraftOver3000)

Result <- PI\_EName, Age, EID (EMPLOYEES JOIN CertifiedEmp)

2. List the names and age of all pilots who have \*never\* had a maintenance issue on a flight where they were the pilot (415/615)

```
FD <- FLIGHTDETAILS

MI <- MAINTENANCE_ISSUE

IssueEID <- RHO(EID) PI_FD.PilotEID(FD JOIN MI)

GoodEID <- PI_EID (EMPLOYEES) - IssueEID

Result <- PI_EName, Age (EMPLOYEES JOIN GoodEID)
```

List the names and age of all pilots who have \*ever\* had a maintenance issue on a flight where they were the pilot (315)

```
PilotsWithMaintenance \leftarrow \pi PilotEID (FLIGHTDETAILS \bowtie MAINTENANCE_ISSUE) PilotsWithMaintenance \leftarrow \rho PilotEID \rightarrow EID (PilotsWithMaintenance) Result \leftarrow \pi EName, Age (PilotsWithMaintenance \bowtie EMPLOYEES)
```

# 3 (315):

JARED\_CERTIFIED <- PI\_(AID) SELECT\_(EName = "Jared Eisner") (EMPLOYEES NATURAL\_JOIN CERTIFIED\_TO\_FLY)

AIRCRAFT\_NOTCERT <- PI\_(AID) AIRCRAFT – JARED\_CERTIFIED

RESULT <- PI\_(EName, AName, AID) AIRCRAFT\_NOTCERT NATURAL\_JOIN AIRCRAFT NATURAL\_JOIN EMPLOYEES

# 3 (415):

```
ALL_PAIRS <- PI (EID, AID) (EMPLOYEES X AIRCRAFT)

CANT_FLY <- ALL_PAIRS - CERTIFIED_TO_FLY

RESULT <- PI (EName, AID, AName) (CANT_FLY NATURAL_JOIN EMPLOYEES NATURAL_JOIN AIRCRAFT)
```

# 4 (Both):

```
{ t | EXISTS a IN AIRCRAFT ( a[AName] = "Boeing787" ^

EXISTS fd IN FLIGHTDETAILS ( fd[PlaneAID] = a[AID] ^

EXISTS f1 IN FLIGHTS ( f1[FLNO] = fd[FNO] ^

EXISTS f2 IN FLIGHTS (f2[Dist] < 2000 ^

f1[FromCode] = f2[FromCode] ^

f1[ToCode] = f2[ToCode]))) ^

t[FromCode] = f1[FromCode] ^ t[ToCode] - t[Distance] = f2[Distance]}
```

### 5a: 315)

SELECT SUM(FLIGHTS.Distance)

FROM FLIGHTS, FLIGHTDETAILS, EMPLOYEES

WHERE FLIGHTS.FLNO = FLIGHTDETAILS.FNO

AND FLIGHTDETAILS.PilotEID = EMPLOYEES.EID

AND EMPLOYEES.EName = "Jason Eisner";

## 5a: 415/615)

SELECT SUM(FLIGHTS.DISTANCE)

FROM FLIGHTDETAILS, EMPLOYEES, AIRCRAFT, FLIGHTS

WHERE EMPLOYEES.EID = FLIGHTDETAILS.PilotEID

AND EMPLOYEES.EName = 'Jason Eisner'

AND AIRCRAFT.AID = FLIGHTDETAILS.PlaneAID

AND AIRCRAFT.AName = 'Embraer 135'

AND FLIGHTS.FLNO = FLIGHTDETAILS.FNO;

## 5b: 315)

SELECT AIRCRAFT.AName, COUNT(AIRCRAFT.AName)

FROM AIRCRAFT, MAINTENANCE\_ISSUE

WHERE AIRCRAFT.AID = MAINTENANCE\_ISSUE.PlaneAID

**GROUP BY AIRCRAFT.AName** 

HAVING COUNT(AName) = ( SELECT MAX(numlssues)

FROM (SELECT COUNT(AName) as numIssues

FROM AIRCRAFT, MAINTENANCE\_ISSUE

WHERE AIRCRAFT.AID = MAINTENANCE\_ISSUE.PlaneAID

GROUP BY AIRCRAFT.AName ) as R )

#### 5b: 415/615)

SELECT TypeIssues.AName, TypeIssues.Total

FROM ( SELECT MAX(Total) AS Total

FROM ( SELECT AIRCRAFT.AName, COUNT(MAINTENANCE\_ISSUE.PlaneAID) AS Total

FROM AIRCRAFT, MAINTENANCE\_ISSUE

WHERE AIRCRAFT.aid = MAINTENANCE\_ISSUE.PlaneAID

GROUP BY AIRCRAFT. AName ) AS Totalissues ) AS Maxissues,

( SELECT AIRCRAFT.AName, COUNT(MAINTENANCE\_ISSUE.PlaneAID) AS Total

FROM AIRCRAFT, MAINTENANCE\_ISSUE

WHERE AIRCRAFT.aid = MAINTENANCE ISSUE.PlaneAID

GROUP BY AIRCRAFT. AName ) AS Typelssues

WHERE MaxIssues. Total = TypeIssues. Total

### 5c: 315)

SELECT EMPLOYEES.EName, EMPLOYEES.Age, C.c.

FROM EMPLOYEES, (SELECT MAX(c) AS m

FROM (SELECT SUM(FLIGHTS.Distance) AS c

FROM FLIGHTS, CODE\_NAMES, FLIGHTDETAILS,

WHERE FLIGHTDETAILS.FNO = FLIGHTS.FLNO

AND FLIGHTS.FromCode = CODE NAMES.Code

AND CODE\_NAMES.StateName = "Maryland"

GROUP BY FLIGHTDETAILS.PilotEID) as B),

(SELECT D.PilotEID, SUM(F.Distance) AS c

FROM FLIGHTS, CODE\_NAMES, FLIGHTDETAILS,

WHERE FLIGHTDETAILS.FNO = FLIGHTS.FLNO

AND FLIGHTS.FromCode = CODE NAMES.Code

AND CODE\_NAMES.StateName = "Maryland"

GROUP BY FLIGHTDETAILS.PilotEID) as C

WHERE C.c = B.m

AND EMPLOYEES.EID = C.PilotEID;

5c: 415/615)

SELECT T.EName, T.Age, T.sum\_distance

FROM (SELECT MAX(MarylandInfo.MarylandDistance) AS MaxDistance

FROM (SELECT SUM(FLIGHTS.Distance) AS MarylandDistance, EMPLOYEES.EID

FROM FLIGHTS, FLIGHTDETAILS, EMPLOYEES, CODE\_NAMES

WHERE CODE NAMES.StateName = "Maryland"

AND (CODE\_NAMES.Code = FLIGHTS.FromCode

OR CODE NAMES.Code = FLIGHTS.ToCode)

AND FLIGHTS.FLNO = FLIGHTDETAILS.FNO

AND FLIGHTDETAILS.PIIotEID = EMPLOYEES.EID

GROUP BY EMPLOYEES.EID ) AS MarylandInfo) AS MAX,

(SELECT SUM(FLIGHTS.Distance) AS MarylandDistance, EMPLOYEES.EID, EMPLOYEES.EName, EMPLOYEES.Age

FROM FLIGHTS, FLIGHTDETAILS, EMPLOYEES, CODE NAMES

WHERE CODE\_NAMES.StateName = "Maryland"

AND (CODE NAMES.Code = FLIGHTS.FromCode

OR CODE\_NAMES.Code = FLIGHTS.ToCode)

AND FLIGHTS.FLNO = FLIGHTDETAILS.FNO

AND FLIGHTDETAILS.PilotEID = EMPLOYEES.EID

GROUP BY EMPLOYEES.EID ) AS MarylandInfo

WHERE MarylandInfo.MarylandDistance = MAX.MarylandDistance

5d: 315)

SELECT FromCode, ToCode, min(Price) as minDirectPrice

FROM FLIGHTS

GROUP BY FromCode, ToCode

5d: 415)

SELECT MIN(F.Price)
FROM FLIGHTS F
WHERE F.Distance > 5000
GROUP BY F.FromCode, F.ToCode

6a: (Both)

FLIGHTS	FLNO	FromCode	ToCode	DepTime	ArrTime	Price	Distance	
	_f1 _f2	'BWI' _c	_c 'SFO'	_t2	_t1		_d1 _d2	
CODE_NAMES		Code	Code		CityName		StateName	
	_c		_cityname					

EMPLOYEES	EID	EName	Age	Salary

AIRCRAFT	AID	AName	CruisingRange

CERTIFIED_TO_FLY	EID	AID

RESULT	FlightNum1	FlightNum2	Connection	Distance	
P.	_f1	_f2	_cityname	_d1+_d2	

	Conditions	
_t2 > _t1 AND _d1 + _d2 < 3200		

**6b: 315)** (6 points) A customer wants to fly from BWI to SFO on exactly 3 connecting flights (e.g. a flight from BWI to ORD, another from ORD to DEN and another flight from DEN to SFO). List all 3-flight options from BWI to SFO including their departure time from BWI, their arrival time in SFO and both the airport codes and name of the first city where the flights connect. ANSWER IN QBE

FLIGHTS	<u>FLNO</u>	FromCode	ToCode	DepTime	ArrTime	Price	Distance
	_f1 _f2 _f3	BWI _t1 _t2	_t1 _t2 SFO	_d1 _d2 _d3	_a1 _a2 _a3		

CODE_NAMES	Code	CityName	StateName
	_t1	_name	

EMPLOYEES	EID	EName	Age	Salary

AIRCRAFT	AID	AName	CruisingRange

RESULT	FLNOs	Code	City	DepTime	ArrTime
P.	(_f1, _f2, _f3)	_t1	_name	_d1	_a3

Conditions	
D1 < a1 < d2 < a2 < d3 < a3	

**6b: 415/615)** (6 points) A customer wants to fly from BWI to SFO on 2 or 3 connecting flights (e.g. a flight from BWI to ORD, another from ORD to DEN and another flight from DEN to SFO). List all 2 or 3-flight options from BWI to SFO including their departure time from BWI, their arrival time in SFO and both the airport codes and name of the first city where the flight connects. ANSWER IN QBE

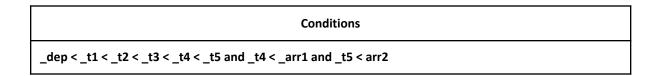
FLIGHTS	FLNO	FromCode	ToCode	DepTime	ArrTime	Price	Distance
	_f1 _f2 _f3 _f4	BWI _a _b _a	_a _b SFO SFO	_dep _t2 _t4 _t5	_t1 _t3 _arr1 _arr2		

CODE_NAMES	Code	CityName	StateName	
	_a	_name		

EMPLOYEES	EID	EName	Age	Salary

AIRCRAFT	AID	AName	CruisingRange

RESULT	Flights	Code	CityName	Departure	Arrival
P.	(_f1, _f2, _f3) (_f1, _f4)	_a	_name	_dep	_arr1 _arr2

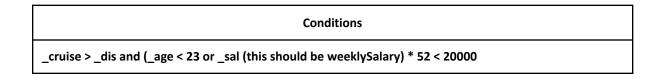


**6C: 315)** . (6 points) List the city, state and airport code of cities in California which are possible to reach from BWI with 2 flights (one connection) but for which HopAir offers no nonstop flights.

FLIGHTS	FLNO	FromCod	e T	oCode	DepTime	Arı	Time	Price	e Distance		
NOT		BWI _a BWI	_	a b b							
CODE_NAM	IES	Code			CityName			Stat	eName		
P.		_b			_name			California			
EMPLOYEES	EMPLOYEES EID			EName		Age			Salary		
AIRCRAFT AID		AID			AName			CruisingRange			
CERTIFIED_	TO_FLY		EID				AID				
RESULT											
				Cone	ditions						

**6C:** 415/615) (6 points) List the flight number of all nonstop flights from a city in Maryland to a city in California which could possibly be piloted by an certified employee who is less than 23 years old or makes less than 20,000 a year.

FLIGHTS	FLNO	FromCode	То	oCode	DepTime	Arı	rTime	Price	•	Distance
	Pfn	_a		b						_dis
	l .		!			!				I
CODE_NAM	ES	Code			CityName		StateName			ne
								Maryland		
		_b						California		
EMPLOYEES		EID		EName		Age	Age Salary			nry
		_e			_age			_sal		
		-		Ī						
AIRCRAFT		AID			AName	CruisingRange			inge	
_a						_cruise				
CERTIFIED_TO_FLY			EID		A		AID	AID		
			е				а			



# 7a (Both 315 and 415/615)

	Derivable from FD's above (type YES/NO) You <b>don't</b> need to give a justification or derivation.
B -> C ?	YES
A -> C?	NO
AB -> A?	YES
ABD -> E?	YES
ABC -> D?	YES
AB -> E?	YES
AC -> E?	NO
A -> E?	NO
B -> E?	YES
C -> E?	NO

# 7b

AB -> C B -> D

AB -> CD -> E

A cannot derive B and B cannot derive A

Thus the candidate key is AB