



Basi Dati

Esercitazione SQL

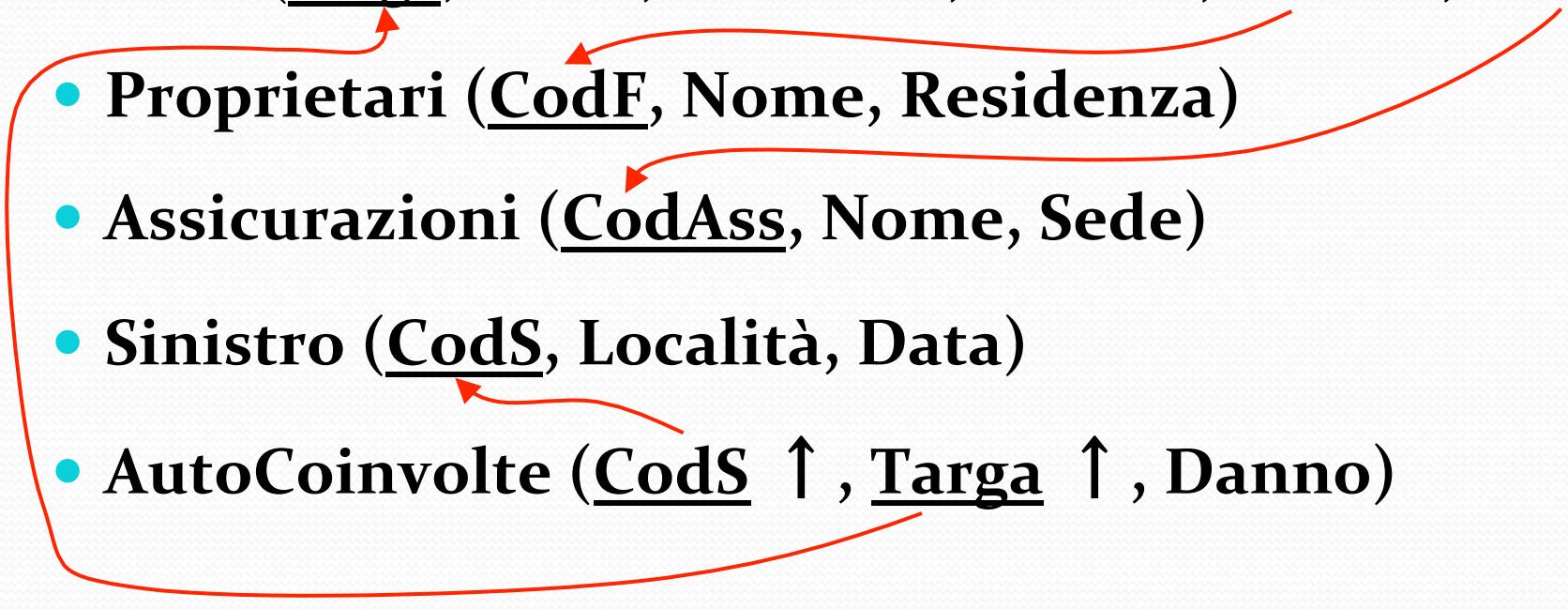
a.a 2012/2013

Prof.ssa G. Tortora

Esercizi SQL

- In ciascuno degli esercizi che seguono viene dato lo schema relazionale di una base di dati e vengono descritte delle informazioni che si vogliono estrarre dalla base.
- Si chiede di scrivere in SQL le interrogazioni che restituiscono le informazioni richieste.

Schema 1

- Auto (Targa, Marca, Cilindrata, Potenza, CodF ↑ , CodAss ↑)
 - Proprietari (CodF, Nome, Residenza)
 - Assicurazioni (CodAss, Nome, Sede)
 - Sinistro (CodS, Località, Data)
 - AutoCoinvolte (CodS ↑ , Targa ↑ , Danno)
- 

Interrogazioni

1. La targa e Marca delle Auto di cilindrata superiore a 2000 cc oppure di potenza superiore a 120 CV.
2. Targa e Nome del proprietario delle Auto di cilindrata superiore a 2000 cc oppure di potenza superiore a 120 CV.
3. Targa e Nome del proprietario delle Auto di cilindrata superiore a 2000 cc oppure di potenza superiore a 120 CV, assicurate presso la “SARA” di Pisa.
4. Targa e Nome del proprietario delle Auto assicurate presso la “SARA” di Pisa, e coinvolte in sinistri il 20/01/2010.
5. Per ciascuna Assicurazione, il nome, la sede ed il numero di auto assicurate.
6. Per ciascuna auto “Fiat”, la targa dell’ auto ed il numero di sinistri in cui è stata coinvolta.
7. Per ciascuna auto coinvolta in più di un sinistro, la targa dell’ auto ed il nome dell’ Assicurazione.

Query 1

- Targa e Marca delle Auto di cilindrata superiore a 2000 cc oppure di potenza superiore a 120 CV.

```
SELECT      Targa, Marca  
FROM        Auto  
WHERE       Cilindrata > 2000 or Potenza > 120;
```

Query 2

- Targa e Nome del proprietario delle Auto di cilindrata superiore a 2000 cc oppure di potenza superiore a 120 CV.

```
SELECT      Targa, Nome  
FROM        Auto, Proprietari  
WHERE       (Cilindrata > 2000 OR Potenza > 120)  
AND  
Auto.CodF = Proprietari.CodF;
```

Query 3

- Targa e Nome del proprietario delle Auto di cilindrata superiore a 2000 cc oppure di potenza superiore a 120 CV, assicurate presso la “SARA” di Pisa.

```
SELECT      Targa, Proprietari.Nome  
FROM        Auto, Proprietari, Assicurazioni  
WHERE       (Cilindrata > 2000 OR Potenza > 120) AND  
            Assicurazioni.Nome = 'SARA' AND  
            Assicurazioni.Sede = 'Pisa' AND  
            Auto.CodF = Proprietari.CodF AND  
            Auto.CodAss = Assicurazioni.CodAss;
```

Query 4

- Targa e Nome del proprietario delle Auto assicurate presso la “SARA” di Pisa e coinvolte in sinistri il 20/01/2010.

```
SELECT      Targa, Proprietari.Nome
FROM        Auto, Proprietari, Assicurazioni,
            AutoCoinvolte, Sinistri
WHERE       Assicurazioni.Nome= 'SARA' and
            Assicurazioni.Sede= 'Pisa' AND
            Data = '2010-01-20' AND
            Auto.CodF=Proprietari.CodF AND
            Auto.CodAss=Assicurazioni.CodAss AND
            Auto.Targa=AutoCoinvolte.Targa AND
            AutoCoinvolte.CodS=Sinistro.CodS;
```


Query 5

- Per ciascuna Assicurazione, il nome, la sede ed il numero di auto assicurate.

```
SELECT      Nome, Sede, COUNT(*) AS NumAuto  
FROM        Assicurazioni, Auto  
WHERE       Auto.CodAss=Assicurazioni.CodAss  
GROUP BY   CodAss;
```

Query 6

- Per ciascuna auto “Fiat”, la targa dell’ auto ed il numero di sinistri in cui è stata coinvolta.

```
SELECT      Targa, COUNT(*) AS NumSinistri  
FROM        Auto, AutoCoinvolte  
WHERE       Marca='Fiat' AND  
            Auto.Targa=AutoCoinvolte.Targa  
GROUP BY   Targa;
```

Query 7

- Per ciascuna auto coinvolta in più di un sinistro, la targa dell' auto ed il nome dell' Assicurazione.

```
SELECT      Targa, Nome  
FROM        AutoCoinvolte, Auto, Assicurazioni  
WHERE       Auto.Targa=AutoCoinvolte.Targa AND  
            Auto.CodAss=Assicurazioni.CodAss  
GROUP BY   Targa  
HAVING      COUNT(*) >1;
```

Schema 2

- PuntiVendita(CodPV, Località, NomeResponsabile)
- Dipendenti(CodDIP, Mansione, Cognome, Nome, Stipendio, Anzianità, CodPV ↑)
- Approvvigionamenti(CodAPP, CodPV ↑ , **Giorno, Mese, Anno**)
- Prodotti(CodPR, Descrizione, Prezzo)
- ProdottiApprovv(CodPR ↑ , CodAPP ↑ , Quantità)
- Fornitori(CodFOR, Nome, Località)
- ProdottiForn(CodFOR ↑ , CodPR ↑)

Query schema 2

1. Cognome e Nome dei dipendenti che lavorano nel PuntoVendita che ha per Responsabile “Mario Rossi”.
2. Cognome e Nome dei dipendenti con più di 10 anni di anzianità che sono “Impiegati” oppure hanno uno stipendio superiore a 1000€.
3. Il Codice dei PuntiVendita che hanno ricevuto approvvigionamenti il 7 gennaio 2010.
4. Il Codice e la Località dei PuntiVendita che hanno ricevuto approvvigionamenti il 7 di ogni mese nel 2010.
5. Il Nome dei fornitori dei prodotti presenti negli approvvigionamenti destinati ai PuntiVendita di Livorno tra il 15 gennaio ed il 10 febbraio 2010.

Soluzione 1

- Cognome e Nome dei dipendenti che lavorano nel PuntoVendita che ha per Responsabile “Mario Rossi”.

```
SELECT      Cognome, Nome  
FROM        Dipendenti , PuntiVendita  
WHERE       Dipendenti.CodPV = PuntiVendita.CodPV  
AND NomeResponsabile = 'Mario Rossi' ;
```

Souzione 2

- Cognome e Nome dei dipendenti con più di 10 anni di anzianità che sono “Impiegati” oppure hanno uno stipendio superiore a 1000€.

```
SELECT      Cognome, Nome  
FROM        Dipendenti  
WHERE       Anzianità > 10 AND  
            (Mansione = 'Impiegati' OR  
            Stipendio > 1000);
```

Soluzione 3

- Il Codice dei PuntiVendita che hanno ricevuto approvvigionamenti il 7 gennaio 2010.

```
SELECT      DISTINCT CodPV  
FROM        Approvvigionamenti  
WHERE       Giorno = 7 AND Mese = 1 AND Anno = 2010;
```


Soluzione 4

- Il Codice e la Località dei PuntiVendita che hanno ricevuto approvvigionamenti il 7 di ogni mese nel 2010.

(sostituiamo gli attributi di Giorno, Mese, Anno di Approvvigionamenti con l'attributo Data di tipo Date)

```
SELECT    Approvvigionamenti.CodPV, Località  
FROM      Approvvigionamenti, PuntiVendita  
WHERE     Approvvigionamenti.CodPV =  
                PuntiVendita.CodPV AND  
                Data LIKE '2010-__-07' ;
```

Soluzione 5

- Il Nome dei fornitori dei prodotti presenti negli approvvigionamenti destinati ai PuntiVendita di 'Livorno' tra il 15 gennaio ed il 10 febbraio 2010.

```
SELECT      Nome
FROM        Fornitori , ProdottiFor, ProdottiAppr ,
            Approvvigionamenti , PuntiVendita
WHERE       Fornitori.CodFor = ProdottiFor.CodFor AND
            ProdottiFor.CodPr = ProdottiAppr.CodPr AND
            ProdottiAppr.CodApp = Approvvigionamenti.CodApp AND
            Approvvigionamenti.CodPV = PuntiVendita. CodPV AND
            PuntiVendita.Località = 'Livorno' AND
            Date >= '2010-01-15' AND Date <= '2010-02-10' ;
```

Schema 3

- Persone(CodF, Cognome, Nome, Età, Città, Via)
 - Autisti(CodAut ↑ , NumPat)
 - HaGuidato(CodViaggio ↑ , CodAut ↑)
 - Viaggi(CodViaggio , Targa ↑ , Carico, CittàPartenza, DataPart, OraPart, CittàDest, DataArr*, OraArr*)
 - Automezzi(Targa, Km, Portata, Peso)
-
- ```
graph TD; Autisti["Autisti(CodAut ↑, NumPat)"] --> HaGuidato["HaGuidato(CodViaggio ↑, CodAut ↑)"]; HaGuidato --> Viaggi["Viaggi(CodViaggio, Targa ↑, Carico, CittàPartenza, DataPart, OraPart, CittàDest, DataArr*, OraArr*)"]; Automezzi["Automezzi(Targa, Km, Portata, Peso)"] --> Viaggi;
```

## Query schema 3

1. Il Numero di Patente degli autisti che hanno compiuto viaggi con partenza da Firenze oppure con destinazione 'Genova' .
2. Il Numero di Patente e l' età degli autisti che hanno compiuto viaggi con partenza da 'Firenze' oppure con destinazione 'Genova' .
3. Il Numero di Patente ed il Cognome degli autisti di età compresa tra 30 e 40 anni che hanno compiuto viaggi con partenza da 'Firenze' oppure con destinazione 'Genova' Guidando automezzi con peso superiore a 100.
4. La città di partenza e la città di destinazione dei viaggi in corso (*cioè iniziati ma non terminati*).
5. Nome e Cognome degli autisti in viaggio (*cioè impegnati in viaggi iniziati ma non terminati*).

# Soluzione 1

- Il Numero di Patente degli autisti che hanno compiuto viaggi con partenza da 'Firenze' oppure con destinazione 'Genova' .

```
SELECT NumPatente
FROM Autisti, HaGuidato, Viaggi
WHERE (CittàPartenza = 'Firenze' OR
 CittàDest = 'Genova') AND
 Autisti.CodAut = HaGuidato.CodAut AND
 HaGuidato.CodViaggio =
 Viaggi.CodViaggio;
```

## Soluzione 2

- Il Numero di Patente e l'età degli autisti che hanno compiuto viaggi con partenza da 'Firenze' oppure da 'Genova'.

```
SELECT NumPatente, Età
FROM Autisti, HaGuidato, Viaggi, Persone
WHERE (CittàPartenza = 'Firenze' OR
 CittàPartenza = 'Genova') AND
 Autisti.CodAut = HaGuidato.CodAut AND
 HaGuidato.CodViaggio = Viaggi.CodViaggio AND
 Autisti.CodAut = Persone.CodF;
```

## Soluzione 3

- Il Numero di Patente ed il Cognome degli autisti di età compresa tra 30 e 40 anni che hanno compiuto viaggi con partenza da Firenze oppure con destinazione Genova guidando automezzi con peso superiore a 100.

```
SELECT NumPatente, Cognome
FROM Autisti, HaGuidato, Viaggi, Persone, Automezzi
WHERE (CittàPartenza = "Firenze" OR
 CittàDest = "Genova") AND
 Età > 30 AND Età <= 40 AND
 Autisti.CodAut = HaGuidato.CodAut AND
 HaGuidato.CodViaggio = Viaggi.CodViaggio AND
 Autisti.CodAut = Persone.CodF AND
 Viaggi.Targa = Automezzi.Targa AND
 Peso > 100 AND
 Autisti.CodAut = Persone.CodF;
```

## Soluzione 4

- La città di partenza e la città di destinazione dei viaggi in corso (*cioè iniziati ma non terminati*).

```
SELECT CittàPartenza, CittàDest
FROM Viaggi
WHERE DataArrivo IS NULL;
```



## Soluzione 5

- Nome e Cognome degli autisti in viaggio (cioè *impegnati in viaggi iniziati ma non terminati*).

```
SELECT Nome, Cognome
FROM Persone, Autisti, HaGuidato, Viaggi
WHERE DataArrivo IS NULL AND
 Persone.CodF = Autisti.CodAut AND
 Autisti.CodAut = HaGuidato.CodAut AND
 HaGuidato.CodViaggio = Viaggi.CodViaggio;
```

# Esercizi SQL – Funzioni di aggregazione

## Schema 3

- PuntiVendita(CodPV, Località, NomeResponsabile)
- Dipendenti(CodDIP, Mansione, Cognome, Nome, Stipendio, Anzianità, CodPV ↑)
- Approvvigionamenti(CodAPP, CodPV ↑, Data)
- Prodotti(CodPR, Descrizione, Prezzo)
- ProdottiApprovv(CodPR ↑, CodAPP ↑, Quantità)
- Fornitori(CodFOR, Nome, Località)
- ProdottiForn(CodFOR ↑, CodPR ↑)

# Query 1 e Soluzione

- La descrizione e la spesa totale di ciascun prodotto approvvigionato il 7 gennaio 2010.

```
SELECT Descrizione, SUM(Prezzo) AS PrezzoTotale
FROM Approvvigionamenti, ProdottiAppr, Prodotti
WHERE Approvvigionamenti.CodApp =
 ProdottiAppr.CodApp AND
 ProdottiAppr.CodPr = Prodotti.CodPr AND
 Data = '2010-07-01'
GROUP BY Prodotti.CodPr;
```

## Query 2 e Soluzione

- La quantità di 'Spaghetti De C' ricevuta nel mese di Febbraio 2010 dai Punti Vendita con sede a 'Livorno'.

```
SELECT SUM(Quantità) AS QuantitàTotale
FROM Prodotti , ProdottiAppr , Approvvigionamenti,
 PuntiVendita
WHERE Prodotti.CodPr = ProdottiAppr.CodPr AND
 ProdottiAppr.CodApp = Approvvigionamenti.CodApp AND
 Approvvigionamenti.CodPV = PuntiVendita.CodPV
AND
 Località = 'Livorno' AND
 Descrizione = 'Spaghetti De C' AND
 Data LIKE '2010-02-__';
```

## Query 3 e Soluzione

- Per ciascun Punto Vendita con più di 20 dipendenti, lo stipendio medio dei dipendenti.

```
SELECT CodPV, AVG(Stipendio) AS StipendioMedio
FROM PuntiVendita, Dipendenti
WHERE PuntiVendita.CodPV = Dipendenti.CodPV
GROUP BY CodPV
HAVING COUNT(*) > 20;
```

## Query 4 e Soluzione

- Per ciascuna Località, lo stipendio minimo e massimo dei dipendenti con più di 10 anni di anzianità che lavorano nei Punti Vendita della Località.

```
SELECT Località, MIN(Stipendio) AS StipendioMin,
 MAX(Stipendio) AS StipendioMax
FROM PuntiVendita , Dipendenti
WHERE PuntiVendita.CodPV = Dipendenti.CodPV AND
 Anzianità > 10
GROUP BY Località;
```

## Query 5 e Soluzione

- Per le mansioni con più di 5 addetti nei PuntiVendita con sede a 'Livorno', l'anzianità media dei dipendenti per ciascuna mansione.

```
SELECT Mansione, AVG(Anzianità) AS AnzianitàMedia
FROM PuntiVendita , Dipendenti
WHERE PuntiVendita.CodPV = Dipendenti.CodPV AND
 Località = 'Livorno'
GROUP BY Mansione
HAVING COUNT(*) > 5
```



# Esercizi SQL – Query complesse

## Schema 4

- Suppliers(sid, sname, address)
- Parts(pid, pname, color)
- Catalog(sid ↑ , pid ↑ , cost)

# Query 1

- Trova i sid dei supplier che forniscono qualche part Red oppure sono al “221 Parcker Street”.

```
SELECT S.sid
FROM Supplier AS S
WHERE S.address= '221 Parker Street'
 OR S.sid IN (SELECT C.sid
 FROM Parts AS P, Catalog AS C
 WHERE P.color= 'red' AND P.pid=C.pid);
```



## Query 3

- Trova i sid dei supplier che forniscono ogni part  
(*suggerimento: ovvero non esiste una parte che non è fornita da questi sid*).

```
SELECT C.sid
FROM Catalog AS C
WHERE (NOT EXISTS (SELECT P.pid
 FROM Parts AS P
 WHERE NOT EXISTS (
 SELECT C1.sid
 FROM Catalog AS C1
 WHERE C1.sid=C.sid AND
 C1.pid=P.pid)));
```

## Query 4

- Trova i sid dei supplier che forniscono ogni part Red.

```
SELECT C.sid
FROM Catalog AS C
WHERE (NOT EXISTS (
 SELECT P.pid
 FROM Parts AS P
 WHERE P.color= 'red' AND
 NOT EXISTS (SELECT C1.sid
 FROM Catalog AS C1
 WHERE C1.sid=C.sid AND
 C1.pid=P.pid));
```



# Query 6

- Trova i sid dei supplier che forniscono ogni part Red oppure ogni Part Green.

```
SELECT C.sid
FROM Catalog AS C
WHERE (NOT EXISTS (SELECT P.pid
 FROM Parts AS P
 WHERE P.color= 'red' AND
 NOT EXISTS (SELECT C1.sid
 FROM Catalog AS C1
 WHERE C1.sid=C.sid AND C1.pid=P.pid)))
OR NOT EXISTS (SELECT P1.pid
 FROM Parts AS P1
 WHERE P1.color= 'green' AND
 NOT EXISTS (SELECT C2.sid
 FROM Catalog AS C2
 WHERE C2.sid=C.sid AND C2.pid=P1.pid)));
```



## Query 7

- Trova i pid delle più costose Parts fornite dal fornitore “Tizio Caio”.

```
SELECT C.pid
FROM Catalog AS C, Supplier AS S
WHERE S.sname= 'Tizio Caio' AND C.sid=S.sid
 AND C.cost >= ALL (
 SELECT C2.cost
 FROM Catalog AS C2, Supplier AS S2
 WHERE S2.sname= 'Tizio Caio'
 AND C2.sid=S2.sid);
```

## Schema 5

- **Flights(no, from, to, distance, departs, arrives, price)**
- **Aircraft(aid, aname, cruisingrange)**
- **Employees(eid, ename, salary)**
- **Certified(eid ↑ , aid ↑ )**

# Interrogazioni

1. Trova i nomi degli aircraft tale che tutti i piloti certificate per usarli hanno un salario maggiore di \$80,000.
2. Trova i nomi dei piloti il cui salario è minore del prezzo del volo più costoso da Los Angeles to Honolulu.
3. Per tutti gli aircraft con *cruisingrange* più di 1000 miles, e il nome dell'aircraft e il salario medio dei piloti certificati per questo aircraft.
4. Trova i nomi di tutti i piloti certificate per qualche Boeing aircraft.
5. Trova gli *aids* di tutti gli aircraft che possono essere usati su voli da Los Angeles a Chicago.
6. Identifica i viaggi che posso essere pilotati da ogni pilota che guadagna più di \$100,000.
7. Stampa gli *enames* dei piloti che possono operare piani con *cruisingrange* più grandi di 3000 miles ma non sono certificati su ogni Boeing aircraft.
8. Calcola la differenza tra il salario medio di un pilota e del salario medio di tutti gli employees (incluso i piloti).
9. Stampa il nome e il salario di ogni non pilota il cui salario è più grande del salario medio dei piloti.

# Query 1

- Trova i nomi degli aircraft tale che tutti i piloti certificati per usarli hanno un salario maggiore di \$80,000.

[illegible]

## Query 2

- Trova i nomi dei piloti il cui salario è minore del prezzo del volo più costoso da Los Angeles to Honolulu.

```
SELECT DISTINCT E.ename
FROM Employees E
WHERE E.salary < (SELECT MIN (F.price)
 FROM Flights F
 WHERE F.from = 'Los Angeles' AND
 F.to = 'Honolulu');
```

## Query 3

- Per tutti gli aircraft con *cruisingrange* più di 1000 miles, e il nome dell'aircraft e il salario medio dei piloti certificati per questo aircraft.

```
SELECT Temp.name, Temp.AvgSalary
FROM (SELECT A.aid, A.aname AS name,
 AVG (E.salary) AS AvgSalary
 FROM Aircraft A, Certi.ed C, Employees E
 WHERE A.aid = C.aid AND
 C.eid = E.eid AND A.cruisingrange > 1000
 GROUP BY A.aid, A.aname) AS Temp;
```

## Query 4

- Trova i nomi di tutti i piloti certificate per qualche Boeing aircraft.

```
SELECT DISTINCT E.ename
FROM Employees E, Certi.ed C, Aircraft A
WHERE E.eid = C.eid AND
 C.aid = A.aid AND
 A.aname LIKE 'Boeing%';
```

## Query 5

- Trova gli *aids* di tutti gli aircraft che possono essere usati su voli da Los Angeles a Chicago.

```
SELECT A.aid
FROM Aircraft A
WHERE A.cruisingrange > (SELECT MIN (F.distance)
 FROM Flights F
 WHERE F.from = 'Los Angeles'
 AND F.to = 'Chicago');
```



## Query 6

- Identifica i viaggi che posso essere pilotati da ogni pilota che guadagna più di \$100,000.

[illegible]

## Query 7

- Stampa gli *enames* dei piloti che possono operare piani con *cruisingrange* più grandi di 3000 miles ma non sono certificati su ogni Boeing aircraft.

```
SELECT DISTINCT E.ename
FROM Employees E
WHERE E.eid IN ((SELECT C.eid
 FROM Certi.ed C
 WHERE EXISTS (SELECT A.aid
 FROM Aircraft A
 WHERE A.aid = C.aid
 AND A.cruisingrange > 3000)
)
AND
NOT EXISTS (SELECT A1.aid
 FROM Aircraft A1
 WHERE A1.aid = C.aid
 AND A1.aname LIKE 'Boeing%'));
```

## Query 8

- Calcola la differenza tra il salario medio di un pilota e del salario medio di tutti gli employees (*incluso i piloti*).

```
SELECT Temp1.avg - Temp2.avg
FROM (SELECT AVG (E.salary) AS avg
 FROM Employees E
 WHERE E.eid IN (SELECT DISTINCT C.eid
 FROM Certi.ed C)) AS Temp1,
(SELECT AVG (E1.salary) AS avg
 FROM Employees E1) AS Temp2;
```

## Query 9

- Stampa il nome e il salario di ogni non pilota il cui salario è più grande del salario medio dei piloti.

```
SELECT E.ename, E.salary
FROM Employees E
WHERE E.eid NOT IN (SELECT DISTINCT C.eid
 FROM Certi.ed C)
AND E.salary > (SELECT AVG (E1.salary)
 FROM Employees E1
 WHERE E1.eid IN
 (SELECT DISTINCT C1.eid
 FROM Certi.ed C1));
```