

*Corso di Laurea in Ingegneria Gestionale  
Sapienza Università di Roma*

# ***Corso di Basi di Dati***

## ***A.A. 2015/2016***

**E2 – SQL : Structured Query  
Language – Soluzione Esercizi**

**Tiziana Catarci, Andrea Marrella**

Ultimo aggiornamento : 19/03/2016

# Esercizi di riepilogo

## Persone

Nome	Reddito	Età	Sesso
Mario	15	80	M
Carlo	25	24	M
Giuseppe	30	45	M
Maria	76	43	F
Gianni	60	50	M
Francesca	18	26	F
Paola	45	60	F
Marco	80	35	M

## Genitori

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**RISOLVERE UTILIZZANDO ESCLUSIVAMENTE  
I COSTRUTTI FINO AD ORA ANALIZZATI**

# Interrogazioni

---

1. Estrarre i nonni di ogni persona.
2. Trovare la relazione che mostra le coppie di fratelli.
3. Trovare l'elenco ordinato dei genitori in cui almeno un figlio guadagna più di 20 milioni.
4. Trovare la relazione che mostra per ciascun figlio i rispettivi genitori (padre e madre), solo se li ha entrambe.
5. Trovare le persone che sono genitori di almeno due figli.
6. Trovare l'elenco ordinato dei genitori in cui tutti i figli guadagnano più di 20 milioni.

# Esercizio 1 – 1\7

## Persone

Nome	Reddito	Età	Sesso
Mario	15	80	M
Carlo	25	24	M
Giuseppe	30	45	M
Maria	76	43	F
Gianni	60	50	M
Francesca	18	26	F
Paola	45	60	F
Marco	80	35	M

## Genitori

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

### **ESERCIZIO** : Estrarre i nonni di ogni persona

```
SELECT P.Figlio AS Nipote, N.Genitore AS Nonno  
FROM Genitori P, Genitori N  
WHERE P.Genitore = N.Figlio
```



Nipote	Nonno
Marco	Mario

# Esercizio 1 – 2\7

## ESERCIZIO : Estrarre i nonni di ogni persona

```
SELECT P.Figlio AS Nipote, N.Genitore AS Nonno  
FROM Genitori P, Genitori N  
WHERE P.Genitore = N.Figlio
```

Vengono definite *due variabili di range per la tabella **Genitore***. In pratica è come se si avessero a disposizione due tabelle **Genitore** identiche (**P** ed **N**).

**P**

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**N**

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

# Esercizio 1 – 3\7

## ESERCIZIO : Estrarre i nonni di ogni persona

```
SELECT P.Figlio AS Nipote, N.Genitore AS Nonno  
FROM Genitori P, Genitori N  
WHERE P.Genitore = N.Figlio
```

Si effettua un *equi-join* tra **P** e **N**, utilizzando come condizione di uguaglianza gli attributi *P.Genitore* e *N.Figlio*.

Il primo valore di *P.Genitore* (=“Mario”) non soddisfa l'*equi-join* con nessuna tupla di **N** (in particolare con nessun valore di *N.Figlio*).

**P**

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**N**

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

# Esercizio 1 – 4\7

## ESERCIZIO : Estrarre i nonni di ogni persona

```
SELECT P.Figlio AS Nipote, N.Genitore AS Nonno
FROM Genitori P, Genitori N
WHERE P.Genitore = N.Figlio
```

Il secondo valore di *P.Genitore* (=“Paola”) soddisfa l'*equi-join solo* con la prima tupla di N.

La tupla (“Marco”, “Paola”, “Paola”, “Mario”) farà parte della tabella ottenuta come *equi-join* tra P ed N).

P

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

N

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

# Esercizio 1 – 5\7

**ESERCIZIO** :Estrarre i nonni di ogni persona

```
SELECT P.Figlio AS Nipote, N.Genitore AS Nonno
FROM Genitori P, Genitori N
WHERE P.Genitore = N.Figlio
```

Il terzo valore di *P.Genitore* (=“Gianni”) non soddisfa l'*equi-join* con nessuna tupla di *N*.

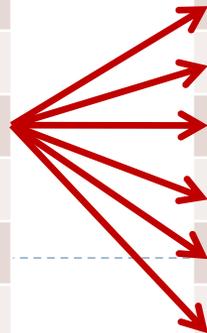
Questo stesso processo di comparazione di valori deve essere effettuato per tutti gli altri valori di *P.Genitore*.

**P**

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**N**

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe



# Esercizio 1 – 6\7

**ESERCIZIO** :Estrarre i nonni di ogni persona

```
SELECT P.Figlio AS Nipote, N.Genitore AS Nonno  
FROM Genitori P, Genitori N  
WHERE P.Genitore = N.Figlio
```

L'*equi-join* tra **P** e **N**, utilizzando come condizione di uguaglianza gli attributi *P.Genitore* e *N.Figlio*, produce una tabella con una sola tupla.

P.Figlio	P.Genitore	N.Figlio	N.Genitore
Marco	Paola	Paola	Mario

# Esercizio 1 – 7\7

**ESERCIZIO** :Estrarre i nonni di ogni persona

```
SELECT P.Figlio AS Nipote, N.Genitore AS Nonno  
FROM Genitori P, Genitori N  
WHERE P.Genitore = N.Figlio
```

P.Figlio	P.Genitore	N.Figlio	N.Genitore
Marco	Paola	Paola	Mario

P.Figlio	N.Genitore
Marco	Mario

Nipote	Nonno
Marco	Mario

L'ultimo passaggio consiste in una proiezione rispetto agli attributi *P.Figlio* e *N.Genitore*, e in una loro successiva ridenominazione.

# Esercizio 2 – 1\6

## Persone

Nome	Reddito	Età	Sesso
Mario	15	80	M
Carlo	25	24	M
Giuseppe	30	45	M
Maria	76	43	F
Gianni	60	50	M
Francesca	18	26	F
Paola	45	60	F
Marco	80	35	M

## Genitori

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**ESERCIZIO** : Trovare la relazione che mostra le coppie di fratelli

```
SELECT DISTINCT G1.Figlio AS Nome1,  
G2.Figlio AS Nome2  
FROM Genitori G1, Genitori G2  
WHERE G1.Genitore = G2.Genitore AND  
G1.Figlio <> G2.Figlio
```



Nome1	Nome2
Marco	Francesca
Francesca	Marco

# Esercizio 2 – 2\6

**ESERCIZIO** : Trovare la relazione che mostra le coppie di fratelli

```
SELECT DISTINCT G1.Figlio AS Nome1, G2.Figlio AS Nome2
FROM Genitori G1, Genitori G2
WHERE G1.Genitore = G2.Genitore AND G1.Figlio <> G2.Figlio
```

**G1**

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**G2**

Genitore	Figlio
Mario	Paola
Paola	Marco
Gianni	Carlo
Maria	Carlo
Giuseppe	Francesca
Giuseppe	Marco

Vengono definite *due variabili di range per la tabella **Genitore***. In pratica è come se si avessero a disposizione due tabelle Genitore identiche.



# Esercizio 2 – 3\6

**ESERCIZIO** : Trovare la relazione che mostra le coppie di fratelli

```
SELECT DISTINCT G1.Figlio AS Nome1, G2.Figlio AS Nome2
FROM Genitori G1, Genitori G2
WHERE G1.Genitore = G2.Genitore AND G1.Figlio <> G2.Figlio
```

**G1**

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**G2**

Genitore	Figlio
Mario	Paola
Paola	Marco
Gianni	Carlo
Maria	Carlo
Giuseppe	Francesca
Giuseppe	Marco

Si effettua un *equi-join* tra **G1** e **G2**, utilizzando come condizione di uguaglianza gli attributi **G1.Genitore** e **G2.Genitore**.

Il primo valore di **G1.Genitore** (=“Mario”) soddisfa l'*equi-join* **solo** con la prima tupla di **G2** (in particolare con il primo valore di **G2.Genitore**)



# Esercizio 2 – 4\6

**ESERCIZIO** : Trovare la relazione che mostra le coppie di fratelli

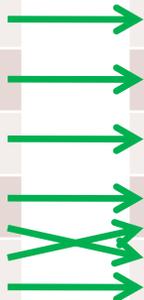
```
SELECT DISTINCT G1.Figlio AS Nome1, G2.Figlio AS Nome2
FROM Genitori G1, Genitori G2
WHERE G1.Genitore = G2.Genitore AND G1.Figlio <> G2.Figlio
```

**G1**

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**G2**

Genitore	Figlio
Mario	Paola
Paola	Marco
Gianni	Carlo
Maria	Carlo
Giuseppe	Francesca
Giuseppe	Marco



Questo stesso processo di comparazione di valori deve essere effettuato per tutti gli altri campi di *G1.Genitore*.  
Le tuple che soddisfano la condizione di uguaglianza sono segnalate da **freccie verdi**.



# Esercizio 2 – 5\6

**ESERCIZIO** :Trovare la relazione che mostra le coppie di fratelli

```
SELECT DISTINCT G1.Figlio AS Nome1, G2.Figlio AS Nome2
FROM Genitori G1, Genitori G2
WHERE G1.Genitore = G2.Genitore AND G1.Figlio <> G2.Figlio
```

G1.Figlio	G1.Genitore	G2.Genitore	G2.Figlio
Paola	Mario	Mario	Paola
Marco	Paola	Paola	Marco
Carlo	Gianni	Gianni	Carlo
Carlo	Maria	Maria	Carlo
Francesca	Giuseppe	Giuseppe	Francesca
Francesca	Giuseppe	Giuseppe	Marco
Marco	Giuseppe	Giuseppe	Francesca
Marco	Giuseppe	Giuseppe	Marco

L'equi-join tra **G1** e **G2**, utilizzando come condizione di uguaglianza gli attributi **G1.Genitore** e **G2.Genitore**, produce la tabella in figura.

Il secondo passo del *join* consiste nella verifica che **G1.Figlio** sia diverso da **G2.Figlio**. Le tuple che non soddisfano tale condizione vengono eliminate (quelle sottolineate in **rosso** in figura).

# Esercizio 2 – 6\6

**ESERCIZIO** : Trovare la relazione che mostra le coppie di fratelli

```
SELECT DISTINCT G1.Figlio AS Nome1, G2.Figlio AS Nome2
FROM Genitori G1, Genitori G2
WHERE G1.Genitore = G2.Genitore AND G1.Figlio <> G2.Figlio
```

G1.Figlio	G1.Genitore	G2.Genitore	G2.Figlio
Francesca	Giuseppe	Giuseppe	Marco
Marco	Giuseppe	Giuseppe	Francesca

L'ultimo passaggio consiste in una proiezione rispetto agli attributi *G1.Figlio* e *G2.Figlio*, e in una loro successiva ridenominazione.

G1.Figlio	G2.Figlio
Francesca	Marco
Marco	Francesca

Nome1	Nome2
Francesca	Marco
Marco	Francesca

# Esercizio 3

## Persone

Nome	Reddito	Età	Sesso
Mario	15	80	M
Carlo	25	24	M
Giuseppe	30	45	M
Maria	76	43	F
Gianni	60	50	M
Francesca	18	26	F
Paola	45	60	F
Marco	80	35	M

## Genitori

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**ESERCIZIO** : Trovare l'elenco ordinato dei genitori in cui almeno un figlio guadagna più di 20 milioni

```
SELECT DISTINCT Genitore  
FROM Genitori, Persone  
WHERE Nome = Figlio AND Reddito >20  
ORDER BY Genitore
```



### Genitore

Gianni

Giuseppe

Maria

Mario

Paola



# Esercizio 4 – 1\2

## Persone

Nome	Reddito	Età	Sesso
Mario	15	80	M
Carlo	25	24	M
Giuseppe	30	45	M
Maria	76	43	F
Gianni	60	50	M
Francesca	18	26	F
Paola	45	60	F
Marco	80	35	M

## Genitori

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**ESERCIZIO** : Trovare la relazione che mostra per ciascun figlio i rispettivi genitori (padre e madre), solo se li ha entrambi

```
SELECT GP.Genitore AS Padre, GM.Genitore AS Madre, GP.Figlio
FROM Genitori GP, Genitori GM, Persone PP, Persone PM
WHERE GP.Figlio=GM.Figlio AND GP.Genitore=PP.Nome AND PP.Sesso='M'
AND GM.Genitore=PM.Nome AND PM.Sesso='F'
```



# Esercizio 4 – 2\2

**ESERCIZIO** :Trovare la relazione che mostra per ciascun figlio i rispettivi genitori (padre, madre, figlio), solo se li ha entrambi

```
SELECT GP.Genitore AS Padre, GM.Genitore AS Madre, GP.Figlio
FROM Genitori GP, Genitori GM, Persone PP, Persone PM
WHERE GP.Figlio=GM.Figlio AND GP.Genitore=PP.Nome AND PP.Sesso='M'
AND GM.Genitore=PM.Nome AND PM.Sesso='F'
```

**PP**

Nome	Reddito	Età	Sesso
------	---------	-----	-------

**GP**

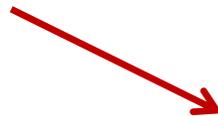
Figlio	Genitore
--------	----------

**PM**

Nome	Reddito	Età	Sesso
------	---------	-----	-------

**GM**

Figlio	Genitore
--------	----------



Padre	Madre	Figlio
Giuseppe	Paola	Marco
Gianni	Maria	Carlo

# Esercizio 5

## Persone

Nome	Reddito	Età	Sesso
Mario	15	80	M
Carlo	25	24	M
Giuseppe	30	45	M
Maria	76	43	F
Gianni	60	50	M
Francesca	18	26	F
Paola	45	60	F
Marco	80	35	M

## Genitori

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**ESERCIZIO** : Trovare le persone che sono genitori di almeno due figli

```
SELECT DISTINCT G1.Genitore  
FROM Genitori G1, Genitori G2  
WHERE G1.Genitore = G2.Genitore AND  
G1.Figlio <> G2.Figlio
```



Genitore

Giuseppe

# Esercizio 6

## Persone

Nome	Reddito	Età	Sesso
Mario	15	80	M
Carlo	25	24	M
Giuseppe	30	45	M
Maria	76	43	F
Gianni	60	50	M
Francesca	18	26	F
Paola	45	60	F
Marco	80	35	M

## Genitori

Figlio	Genitore
Paola	Mario
Marco	Paola
Carlo	Gianni
Carlo	Maria
Francesca	Giuseppe
Marco	Giuseppe

**ESERCIZIO** : Trovare l'elenco ordinato dei genitori in cui tutti i figli guadagnano più di 20 milioni

```
SELECT DISTINCT Genitore
FROM Genitori, Persone
WHERE Nome = Figlio
ORDER BY Genitore
EXCEPT
SELECT DISTINCT Genitore
FROM Genitori, Persone
WHERE Nome = Figlio AND Reddito <=20
```



Genitore
Gianni
Maria
Mario
Paola