

# Appunti Lezione 15-05-2023

Prof. Rosario Sorbello

SELECT  
FROM  
WHERE *CONDIZIONE BOOLEANA*

*CONDIZIONE BOOLEANA = FILTRO*

```
SELECT *  
FROM STUDENTE  
WHERE ETA >
```

```
(SELECT AVG(ETA)  
FROM CITTADINO)
```

*INTERR. + INTERNA CALCOLO ETA' DEL CITTADINO PIU' ANZIANO*

*ETA > (RISULTATO INT. + INTERNA)  
DOPO L'ESECUZIONE DELL'INT. INTERNA*

*ETA > 50*

*PRENDI TUTTI GLI STUDENTI LA CUI ETA E' MAGGIORE DELLA  
MEDIA DELL'ETA DEI CITTADINI*

```
SELECT *  
FROM STUDENTE  
WHERE ETA >
```

( INT. + INTERNA)

1. CASO SE NELL'INT. + ESTERNA USA >, <, <=, >=  
LA QUERY + INTERNA DEVE OBBLIGATORIAMENTE USARE UN  
OPERATORE AGGREGATO
2. CASO SE NELL'INT. + ESTERNA USO GLI OPERATORI INSIEMISTICI  
O I SUOI EQUIVALENTI LOGICI, LA QUERY + INTERNA PUO'  
PREVEDERE DI RESTITUIRE + TUPLE

```
SELECT *  
FROM STUDENTE  
WHERE ETA IN
```

( SELECT DISTINCT ETA  
FROM LAVORATORE)

SELECT \*  
FROM STUDENTE  
WHERE ETA **IN**

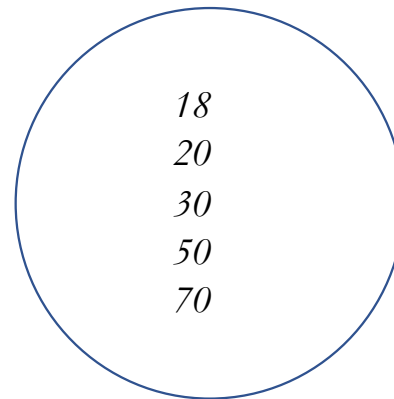
( SELECT DISTINCT ETA  
FROM LAVORATORE)

(18,20,30,50,70) QUESTO E' IL RISULTATO DELL'INTERR. + INTERNA  
OSSIA UNA TABELLA AD UNA COLONNA CON 5 TUPLE

OPERATORE INSIEMISTICO IN = INCLUSO NELL'INSIEME

32

E' INCLUSO NELL'INSIEME ?



## EQUIVALENZE DEGLI OPERATORI

- OPERATORE INSIEMISTICO **IN** = INCLUSO

IL SUO EQUIVALENTE LOGICO **=ANY** (UGUALE AD ALMENO UNO DEI VALORI RESTITUITO DALL'INTERR. + INTERNA)

SELECT \*

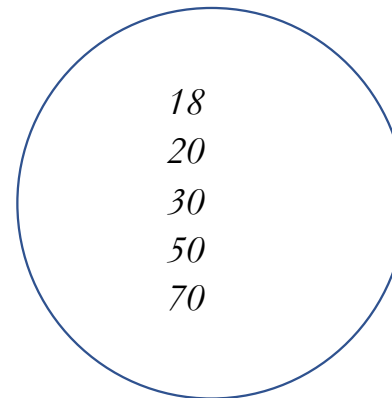
FROM STUDENTE

WHERE *ETA* **=ANY**

(*SELECT DISTINCT ETA*  
*FROM LAVORATORE*)

32

E' UGUALE AD ALMENO UN  
VALORE DELL'INSIEME?



## STUDENTE (COD\_S, NOME, COGNOME)

SELECT NOME  
FROM STUDENTE

=ANY IMPONGO UN LIMITE SINISTRO

MA IL LIMITE DESTRO NON LO  
IMPONGO

SE HO 2 ROSARIO, 2 MARCO, 2 GIUSEPPE

**CITTADINO**

MARIO  
MARIA  
MARCO

SELECT NOME  
FROM CITTADINO  
WHERE NOME = ANY ( SELECT NOME  
FROM STUDENTE)

**MARIO**

E' UGUALE AD ALMENO UN  
VALORE DELL'INSIEME?

**MARIA**

E' UGUALE AD ALMENO UN  
VALORE DELL'INSIEME?

**MARCO**

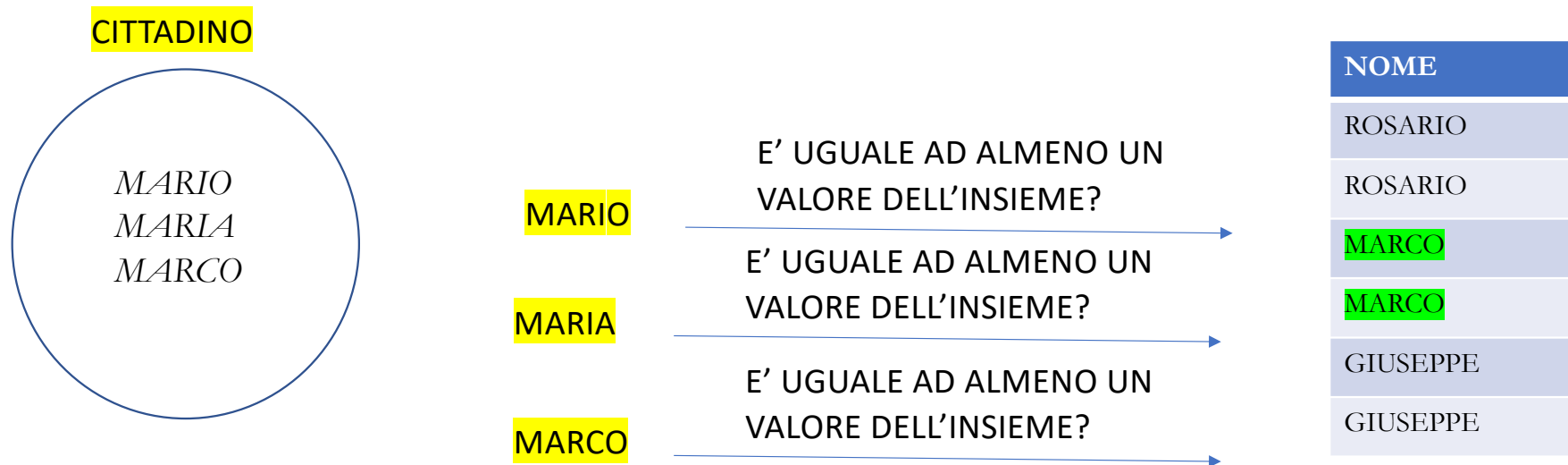
E' UGUALE AD ALMENO UN  
VALORE DELL'INSIEME?

NOME
ROSARIO
ROSARIO
MARCO
MARCO
GIUSEPPE
GIUSEPPE

## STUDENTE (COD\_S, NOME, COGNOME)

```
SELECT NOME  
FROM CITTADINO  
WHERE NOME = ANY ( SELECT NOME  
                   FROM STUDENTE)
```

INTERSEZIONE TRA L'INSIEME CITTADINO E L'INSIEME STUDENTE





```

SELECT NOME
FROM CITTADINO
WHERE NOME IN (SELECT NOME
                FROM STUDENTE)

```

=ANY e IN SONO OPERATORI EQUIVALENTI

```

SELECT NOME
FROM CITTADINO
WHERE NOME = ANY (SELECT NOME
                  FROM STUDENTE)

```

STUDENTE (COD\_S, NOME, COGNOME)  
 CITTADINO(COD\_C, NOME, COGNOME)

$\text{PROJ}_{\text{NOME}}(C) \cap \text{PROJ}_{\text{NOME}}(S)$

=ANY NON ESISTE IN ALGEBRA RELAZIONALE

INTERSEZIONE TRA L'INSIEME CITTADINO E L'INSIEME STUDENTE =  
 TROVARE UN ELEMENTO CHE SIA PRESENTE CONTEMPORANEAMENTE IN ENTRAMBI GLI INSIEMI

RISULTATO DELL'INT. + ESTERNA

NOME
MARIO
MARIA
MARCO

$\cap$

RISULTATO DELL'INT. + INTERNA

NOME
ROSARIO
ROSARIO
MARCO
MARCO
GIUSEPPE
GIUSEPPE

=

NOME
MARCO

CERCO TUTTI I CITTADINI CHE  
 SONO OMONIMI DI ALMENO UNO  
 STUDENTE

```
SELECT NOME
FROM CITTADINO
WHERE NOME NOT IN (SELECT NOME
                    FROM STUDENTE)
```

STUDENTE (COD\_S, NOME, COGNOME)  
CITTADINO(COD\_C, NOME, COGNOME)

<>ALL e NOT IN SONO OPERATORI EQUIVALENTI

```
SELECT NOME
FROM CITTADINO
WHERE NOME <>ALL (SELECT NOME
                  FROM STUDENTE)
```

$PROJ_{NOME}(C) - PROJ_{NOME}(S)$

<>ALL NON ESISTE IN ALGEBRA RELAZIONALE

NOT IN E' LA DIFFERENZA INSIEMISTICA IN ALGEBRA

INTERSEZIONE TRA L'INSIEME CITTADINO E L'INSIEME STUDENTE =  
TROVARE UN ELEMENTO CHE SIA PRESENTE CONTEMPORANEAMENTE IN ENTRAMBI GLI INSIEMI

RISULTATO DELL'INT. + ESTERNA

NOME
MARIO
MARIA
MARCO

-

RISULTATO DELL'INT. + INTERNA

NOME
ROSARIO
ROSARIO
MARCO
MARCO
GIUSEPPE
GIUSEPPE

=

NOME
MARIO
MARIA

CERCO TUTTI I CITTADINI CHE  
NON SONO OMONIMI DI NESSUN  
STUDENTE

```
SELECT NOME
FROM STUDENTE
WHERE NOME NOT IN (SELECT NOME
                    FROM CITTADINO)
```

<>ALL e NOT IN SONO OPERATORI EQUIVALENTI

```
SELECT NOME
FROM STUDENTE
WHERE NOME <>ALL (SELECT NOME
                  FROM CITTADINO)
```

STUDENTE (COD\_S, NOME, COGNOME)  
CITTADINO (COD\_C, NOME, COGNOME)

$PROJ_{NOME}(S) - PROJ_{NOME}(C)$

<>ALL NON ESISTE IN ALGEBRA RELAZIONALE

NOT IN E' LA DIFFERENZA INSIEMISTICA IN ALGEBRA

INTERSEZIONE TRA L'INSIEME CITTADINO E L'INSIEME STUDENTE =  
TROVARE UN ELEMENTO CHE SIA PRESENTE CONTEMPORANEAMENTE IN ENTRAMBI GLI INSIEMI

RISULTATO DELL'INT. + ESTERNA

NOME
ROSARIO
ROSARIO
MARCO
MARCO
GIUSEPPE
GIUSEPPE

RISULTATO DELL'INT. + INTERNA

NOME
MARIO
MARIA
MARCO

=

NOME
ROSARIO
GIUSEPPE

CERCO TUTTI GLI STUDENTI CHE NON SONO OMONIMI DI  
NESSUN CITTADINO

**S**TUDENTE (COD\_S, NOME, COGNOME, DATA\_NASCITA )  
**C**ITTADINO(COD\_C, NOME,COGNOME, INDIRIZZO)

```
SELECT C.NOME  
FROM CITTADINO C  
WHERE NOME NOT IN ( SELECT S.NOME  
                     FROM STUDENTE S)
```

SE L'INTERROGAZIONE + INTERNA VIENE ESEGUITA UNA SOLA VOLTA ALLORA NON AVRO' NESSUN CICLO FOR.

COME CAPISCO CHE L'INTERROGAZIONE + INTERNA VERRA' ESEGUITA UNA SOLA VOLTA ?

SE GLI ATTRIBUTI USATI NELL'INTERROGAZIONE PIU' INTERNA FANNO RIFERIMENTO SOLO ALLE RELAZIONI INDICATE ALL'INTERNO DEL FROM DELL'INT. + INTERNA ALLORA L'INTERROGAZIONE + INTERNA VERRA' ESEGUITA UNA SOLA VOLTA

```
SELECT *  
FROM IMPIEGATO I  
WHERE EXISTS
```

```
(SELECT *  
FROM IMPIEGATO I1  
WHERE I1.NOME = I.NOME  
AND I1.COGNOME = I.COGNOME  
AND I1.CODFISCALE <> I.CODFISCALE)
```

#### IMPIEGATO

Nome	Cognome	<u>CodFiscale</u>
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I.NOME, I.COGNOME, I.CODFISCALE FANNO RIFERIMENTO ALLA RELAZIONE I CHE E' LA RIDENOMINAZIONE DI IMPIEGATO CHE E' USATA NELL'INTERROGAZIONE + ESTERNA  
L'INT. + INTERNA NON VERRA' ESEGUITA UNA SOLA VOLTA

```
SELECT *
FROM IMPIEGATO I
WHERE EXISTS
```

```
(SELECT *
FROM IMPIEGATO I1
WHERE I1.NOME = I.NOME
AND I1.COGNOME = I.COGNOME
AND I1.CODFISCALE <> I.CODFISCALE)
```

IMPIEGATO

Nome	Cognome	<u>CodFiscale</u>
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	<u>CodFiscale</u>
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	<u>CodFiscale</u>
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

PER OGNI TUPLA DELLA RELAZIONE I ESEGUIRA' L'INTERROGAZIONE + INTERNA CON DATI DIVERSI  
 FOR T=TUPLA DI IMPIEGATO I, PASSO LA TUPLA NELL'INTERROGAZIONE PIU' INTERNA COME PARAMETRO ED ESEGUIRO'  
 L'INT. + INTERNA AVENDO SOSTITUITO TUTTI I RIFERIMENTI ALLA TABELLA IMPIEGATO I CON I PARAMETRI

```
SELECT *
FROM IMPIEGATO I
WHERE EXISTS
```

```
(SELECT *
FROM IMPIEGATO I1
WHERE I1.NOME = I.NOME
AND I1.COGNOME = I.COGNOME
AND I1.CODFISCALE <> I.CODFISCALE)
```

PASSO COME PARAMETRO

IMPIEGATO

Nome	Cognome	<u>CodFiscale</u>
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	<u>CodFiscale</u>
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	<u>CodFiscale</u>
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

1° VOLTA SELEZIONO LA PRIMA TUPLA DELLA RELAZIONE I E LA PASSO COME PARAMETRO ALL' INT. + INTERNA

```
SELECT *
FROM IMPIEGATO I
WHERE EXISTS
```

```
(SELECT *
FROM IMPIEGATO I1
WHERE I1.NOME = "MARIO"
AND I1.COGNOME = "ROSSI"
AND I1.CODFISCALE <> "A012")
```

DOPO IL PASS. DI PARAMETRI, ESEGUO UNA PRIMA VOLTA L'INT. + INTERNA

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

CERCO TUTTI GLI IMPIEGATI CHE SI CHIAMANO MARIO ROSSI MA CHE HANNO CODICE FISCALE DIVERSO DA A012, STO CERCANDO TUTTI GLI OMONIMI DI MARIO ROSSI A012



```
SELECT *
FROM IMPIEGATO I
WHERE EXISTS
```

```
(SELECT *
FROM IMPIEGATO I1
WHERE I1.NOME = "MARIO"
AND I1.COGNOME = "ROSSI"
AND I1.CODFISCALE <> "A012")
```

DOPO IL PASS. DI PARAMETRI, ESEGUO UNA PRIMA VOLTA L'INT. + INTERNA

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
UNA TABELLA VUOTA		

INTERR. + INTERNA CON I PARAMETRI MARIO,ROSSI, A012 MI RESTITUISCE LA TABELLA VUOTA

**SELECT \***  
**FROM IMPIEGATO I**  
**WHERE**

TUPLA **MARIO,ROSSI,A012** VIENE SCARTATA DAL  
 RISULTATO FINALE

**EXISTS**

FALSO

RESTITUISCE IL VALORE  
 BOOLEANO VERO SOLO  
 SE L'INTERR. +  
 INTERNA RESTITUISCE  
 ALMENO UNA TUPLA,  
 SE CIO' NON ACCADE  
 MI RESTITUISCE FALSO

(**SELECT \***  
**FROM IMPIEGATO I1**  
**WHERE I1.NOME** = **"MARIO"**  
**AND I1.COGNOME** = **"ROSSI"**  
**AND I1.CODFISCALE** <> **"A012"**)

**IMPIEGATO**

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

**I**

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

**I1**

Nome	Cognome	CodFiscale
UNA TABELLA VUOTA		

INTERR. + INTERNA CON I PARAMETRI MARIO,ROSSI, A012 MI  
 RESTITUISCE LA TABELLA VUOTA

SELECT \*  
FROM IMPIEGATO I  
WHERE EXISTS

(SELECT \*  
FROM IMPIEGATO I1  
WHERE I1.NOME = I.NOME  
AND I1.COGNOME = I.COGNOME  
AND I1.CODFISCALE <> I.CODFISCALE)

PASSO COME PARAMETRO

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

2° VOLTA SELEZIONO LA SECONDA TUPLA DELLA RELAZIONE I E LA PASSO COME PARAMETRO ALL' INT. + INTERNA

SELECT \*  
FROM IMPIEGATO I  
WHERE EXISTS

(SELECT \*  
FROM IMPIEGATO I1  
WHERE I1.NOME = 'CARLO'  
AND I1.COGNOME = 'BIANCHI'  
AND I1.CODFISCALE <> 'B013')

DOPO IL PASS. DI PARAMETRI, ESEGUO UNA PRIMA VOLTA L'INT. + INTERNA

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

CERCO TUTTI GLI IMPIEGATI CHE SI CHIAMANO CARLO BIANCHI MA CHE HANNO CODICE FISCALE DIVERSO DA B013, STO CERCANDO TUTTI GLI OMONIMI DI MARIO ROSSI A012

```
SELECT *
FROM IMPIEGATO I
WHERE EXISTS
```

```
(SELECT *
FROM IMPIEGATO I1
WHERE I1.NOME = 'CARLO'
AND I1.COGNOME = 'BIANCHI'
AND I1.CODFISCALE <> 'B013')
```

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

DOPO IL PASS. DI PARAMETRI, ESEGUO UNA PRIMA VOLTA L'INT. + INTERNA

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Carlo	Bianchi	C014

INTERR. + INTERNA CON I PARAMETRI CARLO, BIANCHI, B013 MI  
RESTITUISCE UNICO OMONIMO DI CARLO BIANCHI OSSIA  
CARLO, BIANCHI, C014

SELECT \*  
FROM IMPIEGATO I  
WHERE EXISTS

LA TUPLA (CARLO, BIANCHI, B013) VIENE  
SELEZIONATO PER FAR PARTE DEL RISULTATO

EXISTS → VERO

(SELECT \*  
FROM IMPIEGATO I1  
WHERE I1.NOME = 'CARLO'  
AND I1.COGNOME = 'BIANCHI'  
AND I1.CODFISCALE <> 'B013')

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

DOPO IL PASS. DI PARAMETRI, ESEGUO UNA PRIMA VOLTA L'INT. + INTERNA

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Carlo	Bianchi	C014

INTERR. + INTERNA CON I PARAMETRI CARLO, BIANCHI, B013 MI  
RESTITUISCE UNICO OMONIMO DI CARLO BIANCHI OSSIA  
CARLO, BIANCHI, C014

SELECT \*  
FROM IMPIEGATO I  
WHERE EXISTS

(SELECT \*  
FROM IMPIEGATO I1  
WHERE I1.NOME = I.NOME  
AND I1.COGNOME = I.COGNOME  
AND I1.CODFISCALE <> I.CODFISCALE)

PASSO COME PARAMETRO

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

3° VOLTA SELEZIONO LA SECONDA TUPLA DELLA RELAZIONE I E LA PASSO COME PARAMETRO ALL' INT. + INTERNA

```
SELECT *
FROM IMPIEGATO I
WHERE EXISTS
```

```
(SELECT *
FROM IMPIEGATO I1
WHERE I1.NOME = 'CARLO'
AND I1.COGNOME = 'BIANCHI'
AND I1.CODFISCALE <> 'C014')
```

DOPO IL PASS. DI PARAMETRI, ESEGUO UNA PRIMA VOLTA L'INT. INTERNA

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

CERCO TUTTI GLI IMPIEGATI CHE SI CHIAMANO CARLO BIANCHI MA CHE HANNO CODICE FISCALE DIVERSO DA C014, STO CERCANDO TUTTI GLI OMONIMI DI CARLO BIANCHI C014



```
SELECT *
FROM IMPIEGATO I
WHERE EXISTS
```

```
(SELECT *
FROM IMPIEGATO I1
WHERE I1.NOME = 'CARLO'
AND I1.COGNOME = 'BIANCHI'
AND I1.CODFISCALE <> 'C014')
```

DOPO IL PASS. DI PARAMETRI, ESEGUO UNA PRIMA VOLTA L'INT. + INTERNA

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Carlo	Bianchi	B013

INTERR. + INTERNA CON I PARAMETRI CARLO, BIANCHI, C014 MI  
RESTITUISCE UNICO OMONIMO DI CARLO BIANCHI OSSIA  
CARLO, BIANCHI, B013

SELECT \*  
FROM IMPIEGATO I  
WHERE EXISTS

LA TUPLA (CARLO, BIANCHI, C014) VIENE  
SELEZIONATO PER FAR PARTE DEL RISULTATO

EXISTS → VERO

(SELECT \*  
FROM IMPIEGATO I1  
WHERE I1.NOME = 'CARLO'  
AND I1.COGNOME = 'BIANCHI'  
AND I1.CODFISCALE <> 'C014')

DOPO IL PASS. DI PARAMETRI, ESEGUO UNA PRIMA VOLTA L'INT. + INTERNA

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Carlo	Bianchi	B013

INTERR. + INTERNA CON I PARAMETRI CARLO, BIANCHI, B013 MI  
RESTITUISCE UNICO OMONIMO DI CARLO BIANCHI OSSIA  
CARLO, BIANCHI, C014

SELECT \*  
FROM IMPIEGATO I  
WHERE EXISTS

TUTTI I DATI DEGLI IMPIEGATI CHE  
HANNO ALMENO UN OMONIMO SIA NEL  
NOME CHE NEL COGNOME

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

(SELECT \*  
FROM IMPIEGATO I1  
WHERE I1.NOME = I.NOME  
AND I1.COGNOME = I.COGNOME  
AND I1.CODFISCALE <> I.CODFISCALE)

DOPO IL PASS. DI PARAMETRI, ESEGUO UNA PRIMA VOLTA L'INT. + INTERNA

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

RISULTATO FINALE DELL'INT.  
NIDIFICATA CON PASS. DI  
PARAMETRI

Nome	Cognome	CodFiscale
Carlo	Bianchi	B013
Carlo	Bianchi	C014

HO ESEGUITO LA QUERY + INTERNA CON  
VALORI SEMPRE DIVERSI E TANTE VOLTE  
QUANTE SONO LE TUPLE DELLA RELAZIONE  
I, HO ESEGUITO 3 VOLTE L'INT. + INTERNA E  
HO RACCOLTO I RISULTATI PER OTTENERE

**SELECT \***  
**FROM IMPIEGATO I**  
**WHERE NOT EXISTS**

NOT EXISTS RESTITUISCE VERO  
 SE L'INTERROGAZIONE PIU'  
 INTERNA RESTITUISCE UNA  
 TABELLA VUOTA

(**SELECT \***  
**FROM IMPIEGATO I1**  
**WHERE I1.NOME = I.NOME**  
**AND I1.COGNOME = I.COGNOME**  
**AND I1.CODFISCALE <> I.CODFISCALE**)

PASSO COME PARAMETRO

**IMPIEGATO**

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

**I**

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

TUTTI GLI IMPIEGATI CHE  
 NON HANNO OMONIMI SIA  
 DI NOME CHE DI COGNOME



**I1**

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

Nome	Cognome	CodFiscale
Mario	Rossi	A012

```
SELECT *
FROM IMPIEGATO I
WHERE NOT EXISTS
```

```
(SELECT *
FROM IMPIEGATO I1
WHERE I1.NOME = I.NOME
AND I1.COGNOME = I.COGNOME
AND I1.CODFISCALE <> I.CODFISCALE)
```

PASSO COME PARAMETRO

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

1° VOLTA SELEZIONO LA PRIMA TUPLA DELLA RELAZIONE I E LA PASSO COME PARAMETRO ALL' INT. + INTERNA

```
SELECT *
FROM IMPIEGATO I
WHERE NOT EXISTS
```

```
(SELECT *
FROM IMPIEGATO I1
WHERE I1.NOME = "MARIO"
AND I1.COGNOME = "ROSSI"
AND I1.CODFISCALE <> "A012")
```

DOPO IL PASS. DI PARAMETRI, ESEGUO UNA PRIMA VOLTA L'INT. + INTERNA

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

CERCO TUTTI GLI IMPIEGATI CHE SI CHIAMANO MARIO ROSSI MA CHE HANNO CODICE FISCALE DIVERSO DA A012, STO CERCANDO TUTTI GLI OMONIMI DI MARIO ROSSI A012

```
SELECT *
FROM IMPIEGATO I
WHERE NOT EXISTS
```

```
(SELECT *
FROM IMPIEGATO I1
WHERE I1.NOME = "MARIO"
AND I1.COGNOME = "ROSSI"
AND I1.CODFISCALE <> "A012")
```

DOPO IL PASS. DI PARAMETRI, ESEGUO UNA PRIMA VOLTA L'INT. + INTERNA

IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
UNA TABELLA VUOTA		

INTERR. + INTERNA CON I PARAMETRI MARIO,ROSSI, A012 MI RESTITUISCE LA TABELLA VUOTA

**SELECT \*** TUPLA **MARIO,ROSSI,A012** VIENE INCLUSA NEL RISULTATO FINALE  
**FROM IMPIEGATO I**  
**WHERE NOT EXISTS** → VERO

RESTITUISCE IL VALORE  
 BOOLEANO FALSO  
 SOLO SE L'INTERR. +  
 INTERNA RESTITUISCE  
 ALMENO UNA TUPLA,  
 SE CIO' NON ACCADE  
 MI RESTITUISCE VERO

(**SELECT \***  
**FROM IMPIEGATO I1**  
**WHERE I1.NOME** = "MARIO"  
**AND I1.COGNOME** = "ROSSI"  
**AND I1.CODFISCALE** <> "A012")

## IMPIEGATO

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I

Nome	Cognome	CodFiscale
Mario	Rossi	A012
Carlo	Bianchi	B013
Carlo	Bianchi	C014

I1

Nome	Cognome	CodFiscale
UNA TABELLA VUOTA		

INTERR. + INTERNA CON I PARAMETRI MARIO,ROSSI, A012 MI  
 RESTITUISCE LA TABELLA VUOTA



# Doppia negazione = affermazione

Trovare tutti gli acquirenti che su amazon hanno acquistato **tutti** i cellulari presenti  
Per dimostrare che un acquirente ha acquistato tutti i cellulari, devo verificare che **non esiste** un cellulare presente su amazon che quello specifico acquirente **non abbia** acquistato, ho dimostrato in logica che quell'acquirente ha acquistato tutti i cellulari.

2° soluzione: conto quanti siano i cellulari diversi presenti su amazon (es. 1000), poi conto quanti sono i cellulari diversi acquistati da uno specifico acquirente su amazon e controllo che il numero di cellulari acquistati sia uguale al numero di cellulari presenti su amazon (*dovrò usare una interrogazione nidificata con la presenza di group by in entrambe e con passaggio di parametri*)

ESAME(**REF\_S**, **REF\_C**, DATA, VOTO)  
 STUDENTE(**COD\_S**, NOME, COGNOME)  
 CORSO(**COD\_C**, NOME\_C)

DIVISIONE TRA I CODICI DELLO SLAVE E LA CHIAVE PRIMARIA  
 DEL MASTER RIDENOMINATA IN MODO DA ESSERE UGUALE  
 AL CODICE DELLO SLAVE

2° MASTER

SLAVE

1° MASTER

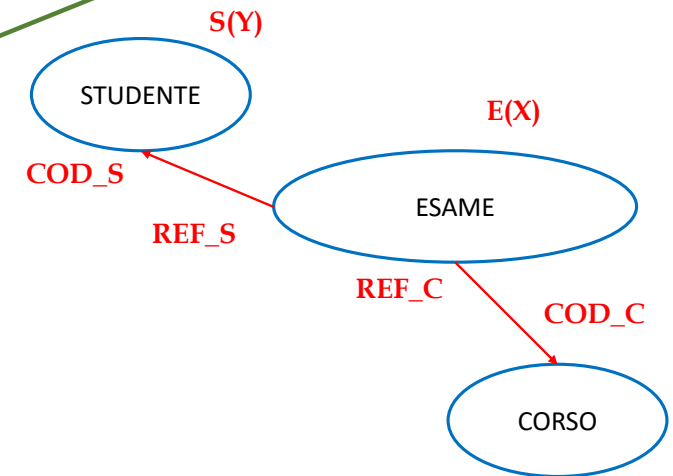
PROJ<sub>CODS, NOME, COGNOME</sub> (S JOIN<sub>S.COD\_S=REF\_S</sub> (PROJ<sub>REF\_S, REF\_C</sub> E) / REN<sub>REF\_C<-COD\_C</sub> (PROJ<sub>COD\_C</sub> (SEL<sub>NOME\_C='BASI' OR NOME\_C='CALC'</sub> C)))

IL NOME E COGNOME DEGLI STUDENTI CHE HANNO SOSTENUTO **TUTTI** GLI ESAMI DI BASI DI DATI E CALCOLATORI

SELECT S.CODS, S.NOME, S.COGNOME  
 FROM STUDENTE S  
 WHERE NOT EXISTS (

SELECT \*  
 FROM CORSO C  
 WHERE C.NOME\_C='BASI' OR C.NOME\_C='CALC'  
 AND NOT EXISTS

(SELECT \*  
 FROM ESAME E  
 WHERE E.REF\_S=S.COD\_S AND E.REF\_C=C.REF\_C))



ESAME(**REF\_S**, **REF\_C**, DATA, VOTO)  
 STUDENTE(**COD\_S**, NOME, COGNOME)  
 CORSO(**COD\_C**, NOME\_C)

DIVISIONE TRA I CODICI DELLO SLAVE E LA CHIAVE PRIMARIA DEL MASTER RIDENOMINATA  
 IN MODO DA ESSERE UGUALE AL CODICE DELLO SLAVE ED OTTENGONO LA DIFFERENZA OSSIA  
 $(REF\_S, REF\_C) / (REF\_C) = \text{LOGICO } REF\_S$

2° MASTER

SLAVE

$PROJ_{REF\_S, REF\_C} \text{E} / REF\_C \text{C} = REF\_S$  1° MASTER

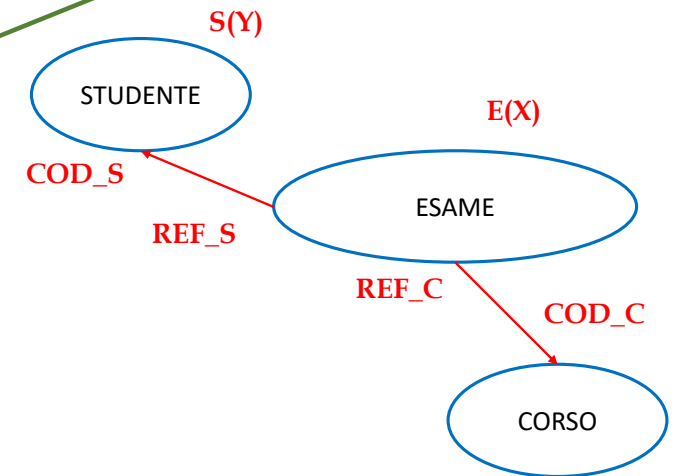
$PROJ_{CODS, NOME, COGNOME} (S JOIN_{S.COD\_S=REF\_S} (PROJ_{REF\_S, REF\_C} E) / REN_{REF\_C < -COD\_C} (PROJ_{COD\_C} (SEL_{NOME\_C='BASI' OR NOME\_C='CALC'} C)))$

IL NOME E COGNOME DEGLI STUDENTI CHE HANNO SOSTENUTO **TUTTI** GLI ESAMI DI BASI DI DATI E CALCOLATORI

SELECT S.CODS, S.NOME, S.COGNOME  
 FROM STUDENTE S  
 WHERE NOT EXISTS (

SELECT \*  
 FROM CORSO C  
 WHERE C.NOME\_C='BASI' OR C.NOME\_C='CALC'  
 AND NOT EXISTS

(SELECT \*  
 FROM ESAME E  
 WHERE E.REF\_S=**S.COD\_S** AND E.REF\_C=**C.REF\_C**))



ESAME(**REF\_S**, **REF\_C**, DATA, VOTO)  
 STUDENTE(**COD\_S**, NOME, COGNOME)  
 CORSO(**COD\_C**, NOME\_C)

SELECT S.CODS, S.NOME, S.COGNOME  
 FROM STUDENTE S  
 WHERE NOT EXISTS (

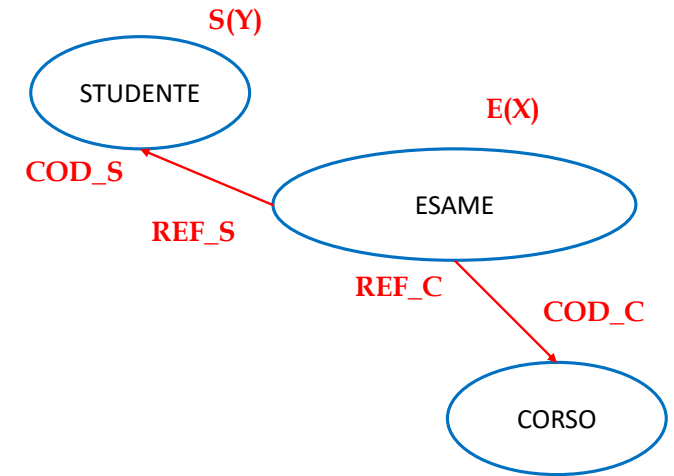
→ (0291435, ROSARIO, SORBELLO)

SELECT \*  
 FROM CORSO C  
 WHERE C.NOME\_C='BASI' OR C.NOME\_C='CALC'  
 AND NOT EXISTS

→ (BD, C1)

2 CORSI DI BASI DI DATI (BD, C1), (BD, C2)  
 2 CORSI DI CALCOLATORI (CC, C3), (CC, C4)

(SELECT \*  
 FROM ESAME E  
 WHERE E.REF\_S=**S.COD\_S** AND E.REF\_C=**C.REF\_C**))



DOPPIO CICLO FOR:

PRENDO LA PRIMA TUPLA DI STUDENTE = (0291435, ROSARIO, SORBELLO)

POI PRENDO IL PRIMO RISULTATO DELL'INTERROGAZIONE INTERNA CON CORSO E SUPPONIAMO

ESAME(**REF\_S**, **REF\_C**, DATA, VOTO)  
 STUDENTE(**COD\_S**, NOME, COGNOME)  
 CORSO(**COD\_C**, NOME\_C)

SELECT S.CODS, S.NOME, S.COGNOME  
 FROM STUDENTE S  
 WHERE NOT EXISTS (

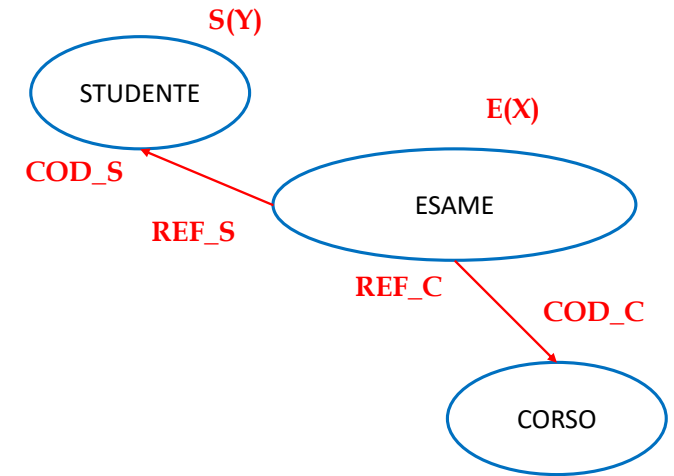
→ (0291435, ROSARIO, SORBELLO)

SELECT \*  
 FROM CORSO C  
 WHERE C.NOME\_C='BASI' OR C.NOME\_C='CALC'  
 AND NOT EXISTS

2 CORSI DI BASI DI DATI (BD, C1), (BD, C2)  
 2 CORSI DI CALCOLATORI (CC, C3), (CC, C4)

(SELECT \*  
 FROM ESAME E  
 WHERE E.REF\_S='0291435' AND E.REF\_C='C1'))

→ ( C1, C2, C3, C4)



ESEGUO L'INTERROGAZIONE + INTERNA DOPO IL PASSAGGIO DEI DUE PARAMETRI E VERIFICO SE ESISTA UN ESAME DELLO STUDENTE CON CODICE 0291435 RELATIVAMENTE ALLA MATERIA CON CODICE C1, POI PER LA MATERIA C2...ETC SEMPRE LASCIANDO 0291435.

ESAME(**REF\_S**, **REF\_C**, DATA, VOTO)  
 STUDENTE(**COD\_S**, NOME, COGNOME)  
 CORSO(**COD\_C**, NOME\_C)

SELECT S.CODS, S.NOME, S.COGNOME  
 FROM STUDENTE S  
 WHERE NOT EXISTS (

→ (0291435, ROSARIO, SORBELLO)

SELECT \*  
 FROM CORSO C  
 WHERE C.NOME\_C='BASI' OR C.NOME\_C='CALC'

AND NOT EXISTS

2 CORSI DI BASI DI DATI (BD, C1), (BD, C2)  
 2 CORSI DI CALCOLATORI (CC, C3), (CC, C4)

(SELECT \*  
 FROM ESAME E  
 WHERE E.REF\_S='S.COD\_S' AND E.REF\_C='C.REF\_C'))

SE TROVO 8 ESAMI DI ROSARIO SORBELLO PER LE 8  
 MATERIE OTTENUTE NELL'INTERROGAZIONE  
 INTERMEDIA ALLORA ROSARIO SORBELLO VERRA'  
 SCELTO PER IL RISULTATO FINALE

