

Operatori di Aggregazione Con riferimento alla Tabella Esami Calcolare il Voto più basso, Calcolare il Voto più alto, Calcolare la somma dei Voti, Contare il Numero di Voti,

Esercizio: Studenti

Operatori di Aggregazione

Con riferimento agli studenti che hanno sostenuto ESAMI Calcolare il Voto più basso

Select MIN(Voto)

From Esami, Studenti Where (Studente=Matricola);

Calcolare il Voto più alto

Select MAX(Voto)

From Esami, Studenti Where (Studente=Matricola);

Calcolare la somma dei Voti

Select SUM(Voto)

From Esami, Studenti Where (Studente=Matricola);

Contare il Numero di Voti Diversi

Select COUNT(Voto)

From Esami, Studenti Where (Studente=Matricola);

Select COUNT(distinct Voto)

From Esami, Studenti Where (Studente=Matricola);

Operatori di Aggregazione

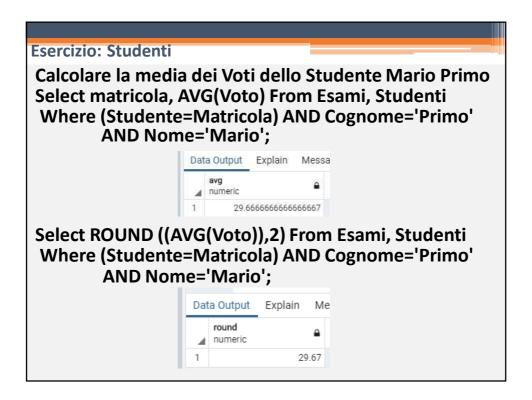
Calcolare la media dei Voti dello Studente Mario Primo

Esercizio: Studenti

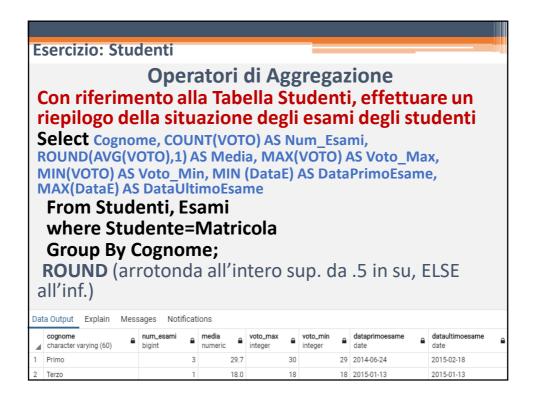
Operatori di Aggregazione

Calcolare la media dei Voti dello Studente Mario Primo

Ovviamente dobbiamo collegare le due Tabelle Studenti ed Esami con un JOIN sulla Matricola.



Operatori di Aggregazione
Con riferimento alla Tabella Studenti,
effettuare un riepilogo della
situazione degli esami degli studenti
Numero Esami;
Media;
Voto Max;
Voto Min;
Data primo esame;
Data secodo esame.



Operatori di Aggregazione

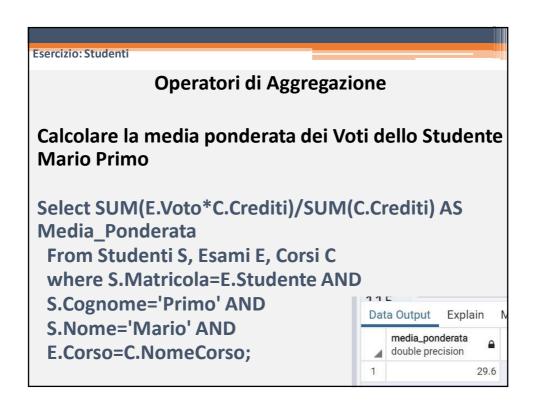
Calcolare la media ponderata dei Voti dello Studente Mario Primo

Quindi
SUM(Voto ei x Crediti ei) /
SUM(Crediti ei)

Operatori di Aggregazione

Calcolare la media ponderata dei Voti dello Studente Mario Primo

Ovviamente dobbiamo collegare le tre Tabelle Studenti, Esami sulla Matricola selezionare Mario Primo e ancora in JOIN con Corsi sul nome del corso.



ATTENZIONE

Provare il seguente comando: Select 5/2;

Come potrete osservare il risultato è: 2 e non 2,5 come ci aspettavamo. Se dividiamo due interi il risultato è ancora un intero, per avere il risultato corretto, almeno uno dei due operandi deve essere in virgola mobile.

La Conversione di un'espressione verso un altro dominio prende il nome di CAST e si ottiene mediante la seguente funzione

CAST (Espressione AS Dominio)

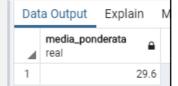
Select CAST(SUM(E.Voto*C.Crediti) AS Real)/SUM(C.Crediti) AS Media_Ponderata

From Studenti S, Esami E, Corsi C where S.Matricola=E.Studente AND

S.Cognome='Primo' AND

S.Nome='Mario' AND

E.Corso=C.NomeCorso;

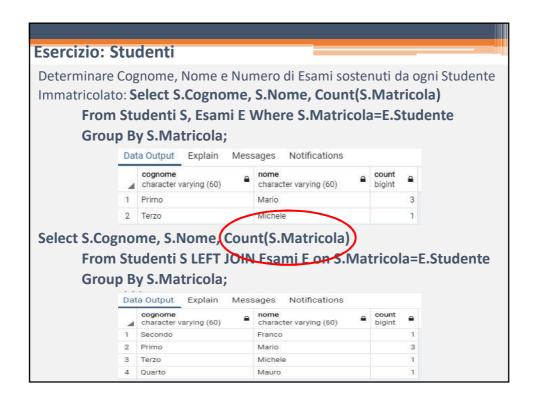


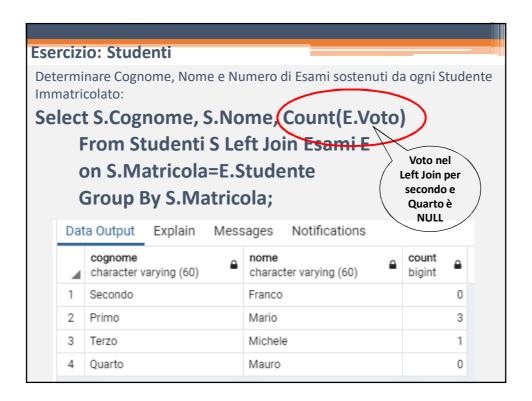
Esercizio: Studenti

Determinare Cognome, Nome e Numero di Esami sostenuti da ogni Studente Immatricolato

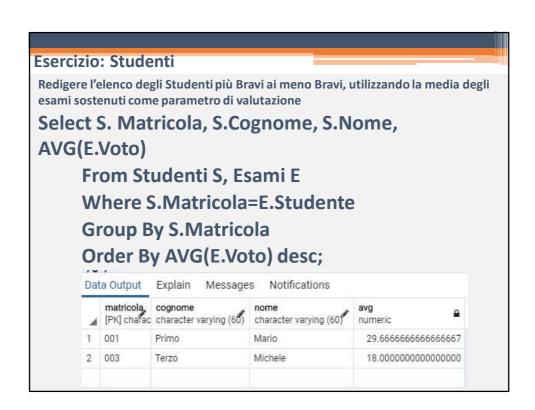
Basta un semplice JOIN Tra studenti ed Esami, fare le partizioni degli esami per singolo studente, ovvero raggruppare, contare ed è fatta,

SI, <u>ma attenzione a come si conta, anzi a</u> cosa si conta.





Redigere l'elenco degli Studenti più Bravi ai meno Bravi, utilizzando la media degli esami sostenuti come parametro di valutazione



Redigere l'elenco degli Studenti più Bravi ovvero con Voto di Media > 26

Esercizio: Studenti

Redigere l'elenco degli Studenti più Bravi ovvero con Voto di Media > 26

Spunto: dobbiamo prima realizzare le partizioni degli esami degli Studenti e poi sulla media di ogni singola partizione effettuare il controllo.

Redigere l'elenco degli Studenti più Bravi ovvero con Voto di Media > 26

Select S. Matricola, S.Cognome, S.Nome, AVG(E.Voto)

From Studenti S, Esami E
Where S.Matricola=E.Studente
Group By S.Matricola
HAVING AVG(E.Voto) >26;

Esercizio: Studenti

Determinare gli esami di Basi di Dati svolti nel gennaio 2015

Determinare gli esami di Basi di Dati svolti nel gennaio 2015 Select E.Corso, E.DataE from Esami E where E.Corso='Basi di Dati' and E.DataE between '2015-01-01' and '2015-01-30';

Esercizio: Studenti

Elencare i Corsi, stampando in caratteri MAIUSCOLI i corsi da 12 crediti, in caratteri MINUSCOLI i corsi da 6 crediti, lasciando inalterati gli altri.

LINGUA TEDESCA

Esercizio: Studenti Elencare i Corsi, stampando in caratteri MAIUSCOLI i corsi da 12 crediti, in caratteri MINUSCOLI i corsi da 6 crediti, lasciando inalterati gli altri Select C.Cognomedocente, CASE C.Crediti when 12 then UPPER(C.NomeCorso) when 6 then LOWER(C.NomeCorso) ELSE C.NomeCorso Data Output Explain Messages Notifications cognomedocente character varying (60) a nomecorso character varying **END** TECNOLOGIE PER IL W... From Corsi C 3 Gaeta sistemi informativi LINGUA INGLESE Order by C.Cognomedocente;

4 Jones
5 Muller

Esercizio: Ancora su CASE

Obiettivo

Elencare Matricola, Cognome e Media di tutti gli Studenti, ma con il seguente vincolo

IF AVG(E.Voto)>28 then 'Eccellente'

IF AVG(E.Voto)>26 then 'Ottimo'

IF AVG(E.Voto)>24 then 'Buono'

IF AVG(E.Voto)>22 then 'Dicreto'

IF AVG(E.Voto) IS NULL then 'Nessun Esame Svolto'

Esercizio: Ancora su CASE Select S.Matricola, S.Cognome, CASE when AVG(E.Voto)>28 then 'Eccellente' when AVG(E.Voto)>26 then 'Ottimo' when AVG(E.Voto)>24 then 'Buono' when AVG(E.Voto)>22 then 'Dicreto' when AVG(E.Voto) IS NULL then 'Nessun Esame Svolto' **ELSE 'Sufficiente' FND** from Studenti S LEFT JOIN Esami E ON S.Matricola=E.Studente Group by S.Matricola order by S.Matricola; Data Output Explain Messages Notifications matricola cognome case [PK] character varying (16) character varying (60) text Eccellente 001 002 Secondo Nessun Esame Svolto 4 004 Ouarto Nessun Esame Svolto

Ancora su CASE

- L'espressione CASE ha due formati supportano un argomento facoltativo ELSE :
 - L'espressione CASE semplice confronta un'espressione con un set di espressioni semplici per determinare il risultato.
 - L'espressione CASE avanzata valuta un set di espressioni booleane per determinare il risultato.
- L'espressione CASE può essere utilizzata in qualsiasi istruzione o clausola che consenta un'espressione valida. È possibile, ad esempio, utilizzare CASE in istruzioni quali SELECT, UPDATE, DELETE e SET e in clausole quali select_list, IN, WHE RE, ORDER BY e HAVING
- Simple CASE expression:

```
CASE input expression WHEN
     when expression THEN result expression [...n]
     [ ELSE else_result_expression ]
END
```

Searched CASE expression:

END

```
CASE
      WHEN Boolean_expression THEN result_expression [ ...n ]
      [ ELSE else_result_expression ]
```

```
Esercizio: Ancora su CASE
/* Creare una tabella DIPARTIMENTI */
CREATE TABLE DIPARTIMENTI (
       CODICE CHAR(4) NOT NULL,
       NOME VARCHAR(20) NOT NULL UNIQUE,
       DIP_IND varchar (50),
       DIP_CITTA varchar(20),
       CONSTRAINT PK_DIPARTIMENTI PRIMARY KEY (CODICE) );
/* Creare una tabella IMPIEGATI */
CREATE TABLE IMPIEGATI (
       matricola char(6) primary key,
       nome varchar(20),
       cognome varchar (20),
       dipart varchar (20) references DIPARTIMENTI(nome),
       ufficio numeric (3),
       stipendio numeric (9) default 0,
       citta character varying(50),
       unique (cognome, nome));
```

4	matricola [PK] character (6)	nome character varying (20)	cognome character varying (20)	dipart character varying (20)	ufficio numeric (3)	stipendio numeric (9)	citta character varying (50
1	000001	Mario	Rossi	Amministrazione	10		Milano
2	000002	Carlo	Bianchi	Produzione	20	36	Torino
3	000003	Giovanni	Verdi	Amministrazione	20	40	Roma
4	000004	Franco	Neri	Distribuzione	16	45	Napoli
5	000005	Carlo	Rossi	Direzione	14	80	Milano
6	000006	Lorenzo	Gialli	Direzione	7	73	Genova
7	000007	Paola	Rosati	Amministrazione	75	40	Venezia
8	000008	Marco	Franco	Produzione	20	46	Roma

Ancora su CASE ...

Select Matricola, Nome, Cognome,

CASE

when dipart='Amministrazione' then 'Impiegato di Concetto'
when dipart='Produzione' then 'Operaio'
when dipart='Ricerca' then 'Ricercatore'
when dipart='Distribuzione' then 'Autista'

ELSE 'Direzione'

END as Ruolo

from Impiegati;

4	matricola [PK] character (6)	nome character varying (20)	cognome character varying (20)	ruolo text
1	000001	Mario	Rossi	Impiegato di Concetto
2	000002	Carlo	Bianchi	Operaio
3	000003	Giovanni	Verdi	Impiegato di Concetto
4	000004	Franco	Neri	Autista
5	000005	Carlo	Rossi	Direzione
6	000006	Lorenzo	Gialli	Direzione
7	000007	Paola	Rosati	Impiegato di Concetto
8	000008	Marco	Franco	Operaio

Ancora su CASE ... Le insiemistiche

Select matricola, nome, cognome, 'Impiegato di Concetto' as Ruolo

from Impiegati where Dipart='Amministrazione'

UNION

Select matricola, nome, cognome, Operaio' as Ruolo

from Impiegati where Dipart='Produzione'

UNION

Select matricola, nome, cognome, 'Ricercatore' as Ruolo

from Impiegati where Dipart='Ricerca'

UNION

Select matricola, nome, cognome, 'Autista' as Ruolo

from Impiegati where Dipart='Distribuzione'

UNION

Select matricola, nome, cognome, 'Direzione' as Ruolo

from Impiegati where (Dipart<>'Amministrazione') AND (Dipart<>'Produzione')

AND (Dipart<>'Ricerca') AND (Dipart<>'Distribuzione');

4	matricola character (6)	nome character varying (20)	cognome character varying (20) □	ruolo text
1	000004	Franco	Neri	Autista
2	000001	Mario	Rossi	Impiegato di Concetto
3	000002	Carlo	Bianchi	Operaio
4	000008	Marco	Franco	Operaio
5	000005	Carlo	Rossi	Direzione
6	000003	Giovanni	Verdi	Impiegato di Concetto
7	000006	Lorenzo	Gialli	Direzione
8	000007	Paola	Rosati	Impiegato di Concetto

BASI DI DATI

Materiale utilizzato e bibliografia

- > Le slide utilizzate dai docenti per le attività frontali sono in gran parte riconducibili e riprese dalle slide originali (con alcuni spunti parziali ripresi dai libri indicati) realizzate da:
- ✓ autori del libro Basi di Dati (Atzeni e altri) testo di riferimento del corso Basi di Dati e sono reperibili su internet su molteplici link oltre che laddove indicato dagli stessi autori del libro;
- ✓ Prof.ssa Tiziana Catarci e dal dott. Ing. Francesco Leotta corso di Basi di Dati dell'Università degli Studi La Sapienza di Roma al seguente link ed altri: http://www.dis.uniroma1.it/~catarci/basidatGEST.html (molto Interessanti anche le lezioni su YouTube).
- ✓ Proff. Luca Allulli e Umberto Nanni, Libro Fondamenti di basi di dati, editore HOEPLI (testo di facile lettura ed efficace).
- Diverse slide su specifici argomenti utilizzate dai docenti per le attività frontali sono anche in parte riconducibili e riprese dalle slide originali – facilmente reperibili e accessibili su internet - realizzate da:

Prof.ssa Roberta Aiello – corso Basi di Dati dell'Università di Salerno

Prof. Dario Maio - corso Basi di Dati dell'Università di Bologna al seguente link ed altri: http://bias.csr.unibo.it/maio
Prof. Marco Di Felice - corso Basi di Dati dell'Università di Bologna al seguente link ed altri: http://www.cs.unibo.it/difelice/dbsi/

Prof Marco Maggini e prof Franco Scarselli - corso Basi di Dati dell'Università di Siena ai seguenti link ed altri: http://staff.icar.cnr.it/pontieri/didattica/LabS//lezioni/_preliminari-DB1%20(Maggini).pdf

Prof.ssa Raffaella Gentilini - corso Basi di Dati dell'Università di Perugia al seguente link ed altri: http://www.dmi.unipg.it/raffaella.gentilini/BD.htm

Prof. Enrico Giunchiglia - corso Basi di Dati dell'Università di Genova al seguente link ed altri: http://www.star.dist.unige.it/~enrico/BasiDiDati/

Prof. Maurizio Lenzerini - corso Basi di Dati dell'Università degli Studi La Sapienza di Roma al seguente link ed altri http://didatticainfo.altervista.org/Quinta/Database2.pdf

- ${\blacktriangleright} \ \ {\it The PostgreSQL Global Development Group PostgreSQL nn.xx} \ \ {\it Documentation}$
- > PostgreSQL (appendice scaricabile dal sito del libro (area studenti) e www.postgresql.org