Basi di Dati  
  
Prova di SQL del 21-02-2018 GRUPPO A

# Viene valutato quanto memorizzato su file (denominare il file COGNOME.SQL- suffisso SQL OBBLIGATORIO)

Facendo riferimento allo schema del database PROVA, scrivere espressioni SQL per le seguenti interrogazioni:

1) Contare il numero dei “colleghi”: due dipendenti sono “colleghi” se hanno lo stesso superiore.

select Count(distinct D1.matricola) as Colleghi,D1.sup as Superiore

from dipendenti D1 join dipendenti D2 on D1.sup=D2.sup

where D1.matricola <> D2.matricola

group By D1.SUP

2) Si considerino le città dei fornitori, prodotti e parti presenti in fornitura. Contare il numero di volte in cui una città  
compare nelle forniture, indipendentemente dal fatto che sia città di fornitore, prodotto o parte.

select CITTA, sum(C) occorrenze

from

((select CITTA, count(\*) C, "fornitori" as provenienza

from FORNITURE join FORNITORI on FCOD=FORNITORI.COD

where CITTA is not null

group by CITTA) union

(select CITTA, count(\*) C, "prodotti"

from FORNITURE join PRODOTTI on PRCOD=PRODOTTI.COD

where CITTA is not null

group by CITTA) union

(select CITTA, count(\*) C, "parti"

from FORNITURE join PARTI on PCOD=PARTI.COD

where CITTA is not null

group by CITTA)) as a

group by CITTA;

Basi di Dati  
  
Prova di SQL del 21-02-2018 GRUPPO B

# Viene valutato quanto memorizzato su file (denominare il file COGNOME.SQL- suffisso SQL OBBLIGATORIO)

Facendo riferimento allo schema del database PROVA, scrivere espressioni SQL per le seguenti interrogazioni:

1) Stabilire, per ogni mansione, quante volte si verifica che un dipendente abbia la stessa mansione del suo superiore diretto. Elencare la mansione e il numero di tali volte.

SELECT MANSIONE, COUNT(MANSIONI) AS VOLTE

FROM (SELECT MANS1 AS MANSIONI, MANS2

FROM (select MATRICOLA AS MAT1, MANSIONE AS MANS1, SUP AS SUP1

from DIPENDENTI) AS D1 JOIN

(select MATRICOLA AS MAT2, MANSIONE AS MANS2, SUP AS SUP2

from DIPENDENTI) AS D2 ON SUP1=MAT2

WHERE MANS1=MANS2) AS UGUALI

RIGHT outer join

(SELECT DISTINCT MANSIONE

FROM DIPENDENTI) AS MANSIONI

ON MANSIONE=MANSIONI

GROUP BY MANSIONE;

2) Produrre una tabella che elenchi la distribuzione delle città nelle varie forniture: una riga della tabella deve riportare il numero di forniture nelle quali le tre città di parte, prodotto e fornitore sono uguali, una seconda riga il numero di forniture nelle quali le tre città sono diverse, e un’ultima riga il numero di forniture nelle quali ci sono solo due città uguali. La tabella DEVE essere la seguente (ovviamente, XX, YY e ZZ sono il risultato, qui sono solo simboli)

|  |  |
| --- | --- |
| Casi | Numerosità |
| Triplette di città uguali | XX |
| Triplette di città distinte | YY |
| Triplette di città uguali a coppie | ZZ |

select 'Triplette di città uguali' as "CASI",

(select count(\*) as "NUMEROSITA'"

from ((FORNITURE as FO join PARTI as PA on FO.PCOD=PA.COD) join

PRODOTTI as PR on FO.PRCOD=PR.COD) join FORNITORI as F on FO.FCOD=F.COD

where PA.CITTA=PR.CITTA and PA.CITTA=F.CITTA) as "NUMEROSITA'"

union

select 'Triplette di città uguali a coppie' as "CASI",

(select count(\*)

from ((FORNITURE as FO join PARTI as PA on FO.PCOD=PA.COD) join

PRODOTTI as PR on FO.PRCOD=PR.COD) join FORNITORI as F on FO.FCOD=F.COD

where (PA.CITTA=PR.CITTA and not( PA.CITTA=F.CITTA))

or (PA.CITTA=F.CITTA and not( PA.CITTA=PR.CITTA))

or (PR.CITTA=F.CITTA and not( PA.CITTA=PR.CITTA))) as "NUMEROSITA'"

union

select 'Triplette di città distinte' as "CASI",

(select count(\*)

from ((FORNITURE as FO join PARTI as PA on FO.PCOD=PA.COD) join

PRODOTTI as PR on FO.PRCOD=PR.COD) join FORNITORI as F on FO.FCOD=F.COD

where PA.CITTA<>F.CITTA and PA.CITTA<>PR.CITTA and F.CITTA<>PR.CITTA)