Basi di Dati
Prova di SQL del 02-02-2020

Facendo riferimento allo schema del database PROVA, contenente le tabelle DIPART (dipartimenti), DIPENDENTI, FORNITORI, PARTI, PRODOTTI, FORNITURE, del quale è fornita un’istanza, scrivere la query SQL corrispondenti alle seguenti interrogazioni:

1. Si identifichi per ogni MANSIONE dei DIPENDENTI la prima data di assunzione di un dipendente con quella mansione. Elencare poi tutti i dipendenti, la cui data di assunzione è successiva alla data di prima assunzione nella sua mansione.

Opzionale. Stabilire inoltre se il dipartimento dei dipendenti così identificati è lo stesso di quello nel quale c’è il dipendente che è stato assunto per primo in quella mansione, indicandolo nella “target list” con la stringa “speciale” se vale la condizione, altrimenti non indicare nulla.

select \*

from DIPENDENTI D1

where DATA\_ASS > (select min(DATA\_ASS)

 from DIPENDENTI D2

 where D1.MANSIONE=D2.MANSIONE)

VERSIONE OPZIONALE

select \*, case

when D1.num\_dipart IN (select distinct(num\_dipart)

 from DIPENDENTI D3

 where D3.DATA\_ASS = (select min(DATA\_ASS)

 from DIPENDENTI D4

 where D1.MANSIONE=D4.MANSIONE))

then "speciale"

else "null"

 end

from DIPENDENTI D1

where DATA\_ASS > (select min(DATA\_ASS)

 from DIPENDENTI D2

 where D1.MANSIONE=D2.MANSIONE)

1. Si consideri l’elenco di tutte le città (sia Fornitori, sia Parti, sia Prodotti). Produrre un output che riporti, per ogni città nell’elenco, il numero di Fornitori, di Parti e di Prodotti per quella città. Se una città non è associata ad una o più delle tabelle, indicare 0 (nel l’esempio qui sotto, con dati di fantasia, Pavia non è associata a nessun prodotto, Torino a nessun fornitore né parte).

|  |  |  |  |
| --- | --- | --- | --- |
| CITTA | N\_fornitori | N\_parti | N\_Prodotti |
| Pavia | 3 | 4 | 0 |
| Milano | 8 | 2 | 8 |
| Torino | 0 | 0 | 1 |
| Venezia | 1 | 1 | 1 |

select ElencoCitta.CITTA, coalesce(N\_FORN,0) as N\_FORN, coalesce(N\_PAR,0) AS N\_PAR, coalesce(N\_PROD,0) AS N\_PROD

from (select distinct(citta)

 from fornitori

 where citta is not null

 union

 select distinct(citta)

 from parti

 where citta is not null

 union

 select distinct(citta)

 from prodotti

 where citta is not null

) as ElencoCitta

 left join

 (select citta, count(cod) as N\_FORN

 from fornitori

 group by citta) as F on ElencoCitta.citta = F.citta

 left join

 (select citta, count(cod) as N\_PAR

 from parti

 group by citta) as P on ElencoCitta.citta = P.citta

 left join

 (select citta, count(cod) as N\_PROD

 from prodotti

 group by citta) as PR on ElencoCitta.citta = PR.citta;