ESERCITAZIONE [27/03/23]

1. Trovare Nome, Marca e Modello dei prodotti acquistabili con meno di 2000 euro.
2. Trovare i nomi dei fornitori che distribuiscono prodotti IBM.
3. Trovare i codici e i nomi di tutti i prodotti che sono forniti da almeno due fornitori.
4. Trovare i nomi dei fornitori che distribuiscono tutti i prodotti.

RELAZIONI

**FORNITORI**(COD\_F, NOME, INDIRIZZO, CITTA)

**PRODOTTI**(COD\_P, NOME\_P, MARCA, MODELLO)

**CATALOGO**(REF\_F, REF\_P, COSTO)

RISOLUZIONE

1:

Per trovare i dettagli dei prodotti acquistabili con meno di 2000 euro mi interessano la relazione PRODOTTI per i dettagli e CATALOGO per il costo

**SEL**CA.COSTO < 2000 (CA **JOIN**CA.REF\_P = PR.COD\_P PR)

Proietto gli attributi delle tuple selezionate:

**PROJ**PR.NOME, PR.MARCA, PR.MODELLO (**SEL**CA.COSTO < 2000 (CA **JOIN**CA.REF\_P = PR.COD\_P PR)))

2:

Devo unire PRODOTTI con FORNITORI, passando attraverso la relazione slave CATALOGO

(PR **JOIN**PR.COD\_P = CA.REF\_P CA) **JOIN**CA.REF\_F = FO.COD\_P FO

Mi interessano solamente le tuple che contengono ‘IBM’ come MARCA, allora seleziono:

**SEL**CA.MARCA = ‘IBM’ ((PR **JOIN**PR.COD\_P = CA.REF\_P CA) **JOIN**CA.REF\_F = FO.COD\_P FO)

Proietto solamente l’attributo NOME dei fornitori della relazione sovrastante:

**PROJ**FO.NOME (**SEL**CA.MARCA = ‘IBM’ ((PR **JOIN**PR.COD\_P = CA.REF\_P CA) **JOIN**CA.REF\_F = FO.COD\_P FO))

3:

Eseguendo un JOIN fra il ridenominato CATALOGO1 e sé stesso ridenominato CATALOGO2

(CA1 JOINCA1.COD\_P1 = CA2.COD\_P2 CA2)

In questo modo ottengo una relazione contenente identità e valori allo specchio, devo eliminare le identità e le informazioni ripetute 2 volte sfruttando l’ordine alfabetico delle stringhe, prendo le tuple dove la stringa contenuta in REF\_F1 è in ordine alfabetico PRIMA della stringa contenuta in REF\_F2

SELCA1.REF\_F1 > CA2.REF\_F2 (PR JOINPR.COD\_P = CA1.REF\_P1 (CA1 JOINCA1.COD\_P1 = CA2.COD\_P2 CA2))

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| REF\_F1 | REF\_P1 | COSTO1 | REF\_F2 | REF\_P2 | COSTO2 |
| F1 | P1 | 10 | F1 | P1 | 10 |
| F1 | P1 | 10 | F2 | P1 | 12 |
| F1 | P1 | 10 | F3 | P1 | 20 |
| F2 | P1 | 12 | F2 | P1 | 12 |
| F2 | P1 | 12 | F1 | P1 | 10 |
| F2 | P1 | 12 | F3 | P1 | 20 |
| F2 | P4 | 16 | F2 | P4 | 16 |
| F1 | P3 | 14 | F1 | P3 | 14 |
| F1 | P3 | 14 | F2 | P3 | 18 |
| F2 | P3 | 18 | F2 | P3 | 18 |
| F2 | P3 | 18 | F1 | P3 | 14 |

Vengono selezionate le tuple che non sono identità né ripetute in modo speculare, si nota che il prodotto P1 è fornito da 3 FORNITORI, e P3 da 2 FORNITORI

**PROJ**PR.COD\_P, PR.NOME\_P (**SEL**CA1.REF\_F1 > CA2.REF\_F2 (PR **JOIN**PR.COD\_P = CA1.REF\_P1 (CA1 **JOIN**CA1.COD\_P1 = CA2.COD\_P2 CA2)))

4:

Divisione non esplosa:

Per trovare i fornitori che distribuiscono tutti i prodotti nel catalogo basta dividere:

**PROJ**CA.COD\_F, PR.COD\_P (CA) **/** **PROJ**PR.COD\_P (PR)

Ottengo tutti i fornitori che forniscono tutti i prodotti, basta JOINARE la relazione FORNITORI e proiettare le informazioni

**PROJ**FO.COD\_F, FO.NOME (F **JOIN**FO.COD\_F = CA.REF\_F (**PROJ**CA.COD\_F, PR.COD\_P (CA) **/** **PROJ**PR.COD\_P (PR)))