

## Esercizio 1

CICLISTA (nomeciclista, nazionalità, età)

GARA (nomecorsa, anno, partenza, arrivo)

PARTECIPA (nomecorsa, anno, nomeciclista, posizione)

FK nomecorsa, anno REFERENCES GARA

FK nomeciclista REFERENCES CICLISTA

**Nota:** se un ciclista si è ritirato in una corsa abbiamo posizione = R

### Selezionare i ciclisti che si sono classificati primi in una gara partita da Milano

I dati dei ciclisti sono in CICLISTA, la partenza si trova in GARA e la posizione in PARTECIPA. Bisogna quindi eseguire un join tra GARA e PARTECIPA imponendo le condizioni, e poi un semijoin con ciclista per ottenere i ciclisti.

$$S1 = \sigma_{partenza='Milano' \text{ AND } posizione='Primo'}(GARA \bowtie PARTECIPA) \\ CICLISTA \bowtie S1$$

### Versione SQL

```
SELECT DISTINCT C.*
FROM CICLISTA C
JOIN PARTECIPA P. ON P.nomeciclista = C.nomeciclista
JOIN GARA G ON G.nomecorsa = P.nomecorsa AND G.anno = P.anno
WHERE P.posizione = 'Primo' AND G.partenza = 'Milano'
```

### Selezionare i nomi dei ciclisti che non si sono mai ritirati ad una corsa con nome 'Giro'

Bisogna prelevare i ciclisti che si sono ritirati da una gara con nome 'Giro' e sottrarli agli altri.

Prelevo i ciclisti che si sono ritirati alla corsa con nome giro

$$S1 = \sigma_{posizione='R' \text{ AND } nomecorsa='Giro'}(PARTECIPA)$$

Poi li sottraggo agli altri

$$\pi_{nomeciclista}(CICLISTA - CICLISTA \bowtie S1)$$

### Versione SQL

```
SELECT C.nomeciclista
FROM CICLISTA C
WHERE C.nomeciclista NOT IN (
    SELECT nomeciclista
    FROM PARTECIPA
    WHERE nomecorsa = 'Giro' AND posizione = 'R'
)
```

**Selezionare il nome delle gare dove per ogni edizione c'è stato almeno un ritirato**

GARA	
Nomecorsa	Anno
Giro	2010
Giro	2011
Lombardia	2010
Lombardia	2011

PARTECIPA			
Nomeciclista	Nomecorsa	Anno	Posizione
Mario	Giro	2010	R
Giuseppe	Giro	2011	R
Antonio	Lombardia	2010	1
Luigi	Lombardia	2011	R

Prendendo come esempio la situazione sopra deve essere restituito come nome solamente 'Giro', perché la gara 'Lombardia' nel 2010 non ho avuto nessun ritirato.

**Selezionare le gare che hanno avuto dei ritirati**

$$S1 = \pi_{nomecorsa, anno} \left( \sigma_{posizione='R'}(PARTECIPA) \right)$$

S1	
Nomecorsa	Anno
Giro	2010
Giro	2011
Lombardia	2011

Come si vede la gara (Lombardia, 2010) non compare, in quanto non ha avuto ritirati.

**Sottrarre S1 alle gare, ottenendo quelle per cui in almeno un'edizione non ci sono stati ritirati**

$$S2 = \pi_{nomecorsa, anno}(GARA) - S1$$

S2	
Nomecorsa	Anno
Lombardia	2010

Rimane solamente (Lombardia, 2010) che non ha avuto ritirati.

**Infine, visto che si vogliono solo i nomi delle gare per cui c'è stato un ritirato in TUTTE le edizioni, si estrae Nomecorsa da S2 e lo si sottrae da tutte le altre, in questo modo si eliminano quelle per cui esiste almeno un'edizione in cui non ci sono stati ritirati.**

$$\pi_{nomecorsa}(GARA) - \pi_{nomecorsa}(S2)$$

Ottenendo solo "Giro" come risultato, in quanto è l'unica gara che ha avuto ritirati in tutte le edizioni.

#### Versione SQL

```

SELECT DISTINCT G.nomecorsa
FROM GARA G
WHERE G.nomecorsa NOT IN (
    SELECT G1.nomecorsa
    FROM GARA G1
    WHERE NOT EXISTS (
        SELECT *
        FROM PARTECIPA P
        WHERE P.nomecorsa = G1.nomecorsa AND P.anno = G1.anno AND posizione = 'R'
    )
)

```

## Esercizio 2

USER(USERID, NAME, CITY)

TAG(TAGID, TAGNAME)

AK(TAGNAME)

POST(POSTID, USERID, TAGID, TITLE, BODY)

FK USERID REFERENCES USER NOT NULL

FK TAGID REFERENCES TAG (**NOTA TAGID PUO' ESSERE NULL**)

COMMENT(COMMENTID, POSTID, USERID, COMMENT)

FK POSTID REFERENCES POST NOT NULL

FK USERID REFERENCES USER NOT NULL

**Id distinti dei post che sono stati commentati da almeno un utente di Modena**

$$\pi_{POSTID}(COMMENT \bowtie \sigma_{CITY='Modena'}(USER))$$

```
SELECT DISTINCT POSTID
FROM COMMENT C
JOIN USER U ON U.USERID = C.USERID
WHERE U.CITY = 'Modena'
```

**I nomi degli utenti residenti a Bologna che hanno scritto o commentato un post**

Seleziono gli utenti che risiedono a Bologna

$$S1 = \sigma_{CITY='Bologna'}(USER)$$

Seleziono il nome degli utenti residenti a Bologna che hanno scritto un post

$$S2 = \pi_{NAME}(S1 \bowtie POST)$$

Seleziono il nome degli utenti residenti a Bologna che hanno commentato un post

$$S3 = \pi_{NAME}(S1 \bowtie COMMENT)$$

Visto che voglio quelli che hanno scritto o commentato unisco i risultati

$$S2 \cup S3$$

### Versione SQL

```
SELECT DISTINCT U.NAME
FROM USER U
JOIN POST P ON P.USERID = U.USERID
WHERE U.CITY = 'Bologna'
```

UNION

```
SELECT DISTINCT U.NAME
FROM USER U
JOIN COMMENT C ON C.USERID = U.USERID
WHERE U.CITY = 'Bologna'
```

### Nome degli utenti che hanno commentato solo post con tag 'Sport'

La richiesta può essere riscritta come: "Nome degli utenti che hanno commentato post con tag sport ma non hanno commentato post con tag diverso da sport."

C'è da notare che TAGID in un post può essere nullo, quindi devo verificare effettivamente che un post abbia il tag sport.

Prelevo i post taggati sport

$$S1 = POST \bowtie \sigma_{TAGNAME='SPORT'}(TAG)$$

Prelevo i post con tag diverso da sport

$$S2 = POST \bowtie \sigma_{TAGNAME \neq 'SPORT'}(TAG)$$

Ora da COMMENT estraggo gli id degli utenti che hanno commentato i post di S1 e S2

Utenti che hanno commentato post con tag SPORT

$$S3 = \pi_{USERID}(COMMENT \bowtie S1)$$

Utenti che hanno commentato post con tag diverso da SPORT

$$S4 = \pi_{USERID}(COMMENT \bowtie S2)$$

Notare che in questo caso il semijoin è importante, se si fosse usato il natural join bisognava specificare da quale relazione prelevare USERID visto che è presente anche in POST, ma con significato diverso (autore del post).

Infine sottraggo gli utenti che hanno commentato post con tag diverso da SPORT da quelli che hanno commentato post taggati SPORT e faccio il join con USER per estrarre i nomi.

$$\pi_{NAME}(USER \bowtie (S3 - S4))$$

### Versione SQL

```
SELECT U.NAME
FROM USER U
JOIN COMMENT C ON C.USERID = U.USERID
JOIN POST P ON P.POSTID = C.POSTID
JOIN TAG T ON T.TAGID = P.TAGID
WHERE T.TAGNAME = 'SPORT' AND U.USERID NOT IN (
    SELECT C1.USERID
    FROM COMMENT C1
    JOIN POST P1 ON P1.POSTID = C1.POSTID
    JOIN TAG T1 ON T1.TAGID = P1.TAGID
    WHERE T1.TAGNAME <> 'SPORT'
)
```