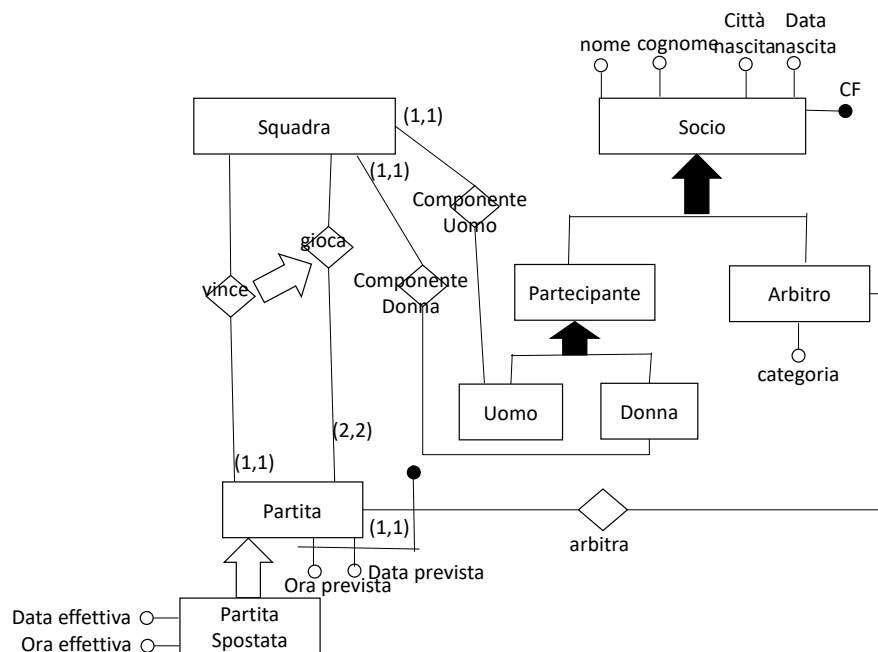


Appello di Basi di Dati – 12/04/2018

Prof. Tiziana Catarci

A.A. 2017-2018

DOMANDA 1 (10 punti)



Una stessa squadra non può giocare due partite alla stessa ora

DOMANDA 2

La ristrutturazione dello schema consiste nell'eliminazione e traduzione in opportuni vincoli delle generalizzazioni complete, della ISA tra entità e dell'ISA tra relazioni (quest'ultima sostituita da opportuno vincolo). Occorre inoltre introdurre una chiave per Squadra (ad esempio un codice). La ristrutturazione è lasciata allo studente. Di seguito la progettazione logica:

SOCIO (CF, NOME, COGNOME, CITTANASCITA, DATANASCITA)

PARTECIPANTE (CF)

FK: PARTECIPANTE[CF] $\subseteq$ SOCIO[CF]

ARBITRO (CF, CATEGORIA)

FK: ARBITRO[CF] $\subseteq$ SOCIO[CF]

UOMO (CF)

FK: UOMO[CF] $\subseteq$ PARTECIPANTE[CF]

DONNA(CF)

FK: DONNA[CF] $\subseteq$ PARTECIPANTE[CF]

SQUADRA(CODICE, COMPONENTE_UOMO, COMPONENTE_DONNA)

FK: SQUADRA[COMPONENTE_UOMO] $\subseteq$ UOMO[CF]

FK: SQUADRA[COMPONENTE_DONNA] $\subseteq$ DONNA[CF]

PARTITA(DATA_PREVISTA, ORA_PREVISTA, ARBITRO, SQUADRA_VINCENTE)

FK: PARTITA[ARBITRO] $\subseteq$ ARBITRO[CF]

FK: PARTITA[SQUADRA_VINCENTE] $\subseteq$ SQUADRA[CODICE]

INC: PARTITA[DATA_PREVISTA, ORA_PREVISTA, ARBITRO] $\subseteq$
GIOCA[DATA_PREVISTA, ORA_PREVISTA, ARBITRO]

PARTITA_SPOSTATA(DATA_PREVISTA, ORA_PREVISTA, ARBITRO,
DATA_EFFETTIVA, ORA_EFFETTIVA)

FK: PARTITA_SPOSTATA(DATA_PREVISTA, ORA_PREVISTA, ARBITRO) $\subseteq$
PARTITA[DATA_PREVISTA, ORA_PREVISTA, ARBITRO]

GIOCA(DATA, ORA, ARBITRO, SQUADRA)

FK: GIOCA[DATA, ORA, ARBITRO] $\subseteq$ PARTITA[DATA_PREVISTA,
ORA_PREVISTA, ARBITRO]

FK: GIOCA[SQUADRA] $\subseteq$ SQUADRA[CODICE]

Vincolo di generalizzazione SOCIO

PARTECIPANTE[CF] $\cap$ ARBITRO[CF] = $\emptyset$

PARTECIPANTE[CF] $\cup$ ARBITRO[CF] $\subseteq$ SOCIO[CF]

Vincolo di generalizzazione PARTECIPANTE

UOMO[CF] $\cap$ DONNA[CF] = $\emptyset$

UOMO[CF] $\cup$ DONNA[CF] $\subseteq$ PARTECIPANTE[CF]

Vincolo ISA tra relazioni

PARTITA[DATA_PREVISTA, ORA_PREVISTA, ARBITRO, SQUADRA_VINCENTE] $\subseteq$
GIOCA[DATA, ORA, ARBITRO, SQUADRA]

Vincolo di cardinalità su relazione gioca

$\forall (d, o, a) \in \text{PARTITA}[\text{DATA_PREVISTA}, \text{ORA_PREVISTA}, \text{ARBITRO}] :$

$|\text{GIOCA}[\text{DATA}, \text{ORA}, \text{ARBITRO}, \text{SQUADRA}] : \text{DATA}=d \wedge \text{ORA}=o \wedge \text{ARBITRO} = a|$
 $= 2$

Traduzione vincolo esterno

$\nexists c \in \text{SQUADRA}[\text{CODICE}] : \exists d, o, a_1, a_2 \text{ tale che } (d, o, a_1, c) \in$

$\text{GIOCA}[\text{DATA}, \text{ORA}, \text{ARBITRO}, \text{SQUADRA}] \wedge (d, o, a_2, c) \in$

$\text{GIOCA}[\text{DATA}, \text{ORA}, \text{ARBITRO}, \text{SQUADRA}]$

DOMANDA 3

Dato il seguente schema relazionale:

STUDENTI (Matricola, Cognome)

CORSI (Cod, aa, CognomeDocente)

FREQUENZA (Cod, aa, Matricola)

che contiene informazioni sui corsi seguiti dagli studenti di una Facoltà, si esprimano in SQL le seguenti interrogazioni:

- 1) La matricola degli studenti che seguono corsi in cui il docente ha il loro stesso cognome. **(3 punti)**

```
SELECT distinct s.Matricola from STUDENTI s, FREQUENZA f, CORSI c
WHERE s.Matricola = f.Matricola AND f.Cod = c.Cod AND f.aa = c.aa
AND c.CognomeDocente = s.Cognome
```

- 2) Il cognome degli studenti che non seguono più di un corso. **(4 punti)**

```
SELECT s.Cognome FROM STUDENTI s WHERE 1 >= (SELECT count(*) FROM
FREQUENZA f where f.Matricola = s.Matricola)
```

- 3) I docenti che insegnano nei corsi che sono i più frequentati dagli studenti per almeno due anni accademici. **(4 punti)**

```
CREATE VIEW StudentiPerCorso AS
SELECT Cod, aa, count(*) AS NUM_STUDENTI FROM FREQUENZA
GROUP BY Cod, aa
```

```
CREATE VIEW PiuFrequentatiPerAnno AS
SELECT Cod, aa FROM StudentiPerCorso s1 WHERE NUM_STUDENTI =
(SELECT MAX(NUM_STUDENTI) FROM StudentiPerCorso s2 WHERE s2.aa =
s1.aa)
```

```
SELECT c.CognomeDocente
FROM PiuFrequentatiPerAnno p, CORSI c
WHERE p.Cod = c.Cod and p.aa = c.aa and 2 <= (SELECT count(*)
FROM PiuFrequentatiPerAnno p2 where p.Cod = p2.Cod)
```