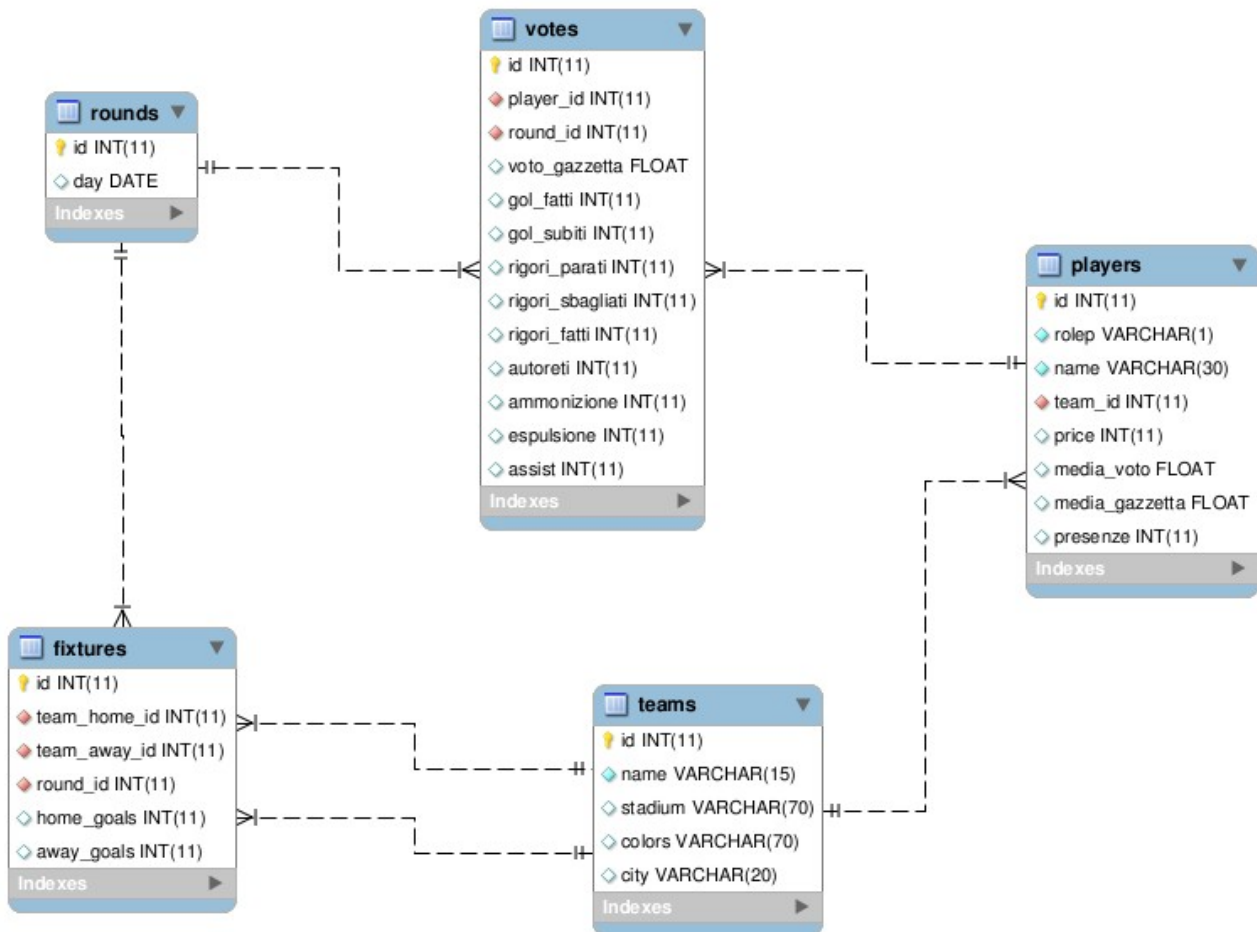


## Prova pratica di Basi di Dati – 15 febbraio 2012



### Esercizio 1

Si crei una query SQL che produce come risultato l'id e il nome dei difensori che nel mese di ottobre 2008 hanno giocato più di 2 partite, totalizzando tra 2008 e 2009 (il database contiene dati relativi solo a queste due stagioni) almeno 30 presenze. Lo studente non utilizzi viste né query annidate nel costrutto FROM. Il risultato dovrà essere del tipo: (Id del giocatore, nome del giocatore)

### Soluzione

```
SELECT p.id, name
FROM players p JOIN votes v
ON p.id = v.player_id
JOIN rounds r ON v.round_id = r.id
WHERE r.day_r BETWEEN '2008-10-01' AND '2008-10-31'
      AND p.rolep = 'D'
      AND p.presenze >= 30
GROUP BY name, p.id
HAVING COUNT(*) > 2;
```

### Esercizio 2

Si crei una query SQL che produce come risultato l'elenco delle squadre, con almeno 5 attaccanti, che hanno

vinto almeno una partita in casa con più di due goal di scarto e che non hanno mai vinto almeno due partite con più di due goal di scarto fuori casa. Il risultato dovrà essere del tipo: (NomeSquadra)

## Soluzione

```
SELECT t.name
FROM teams t JOIN players p ON t.id=p.team_id
WHERE t.id NOT IN (
    SELECT f.team_away_id
    FROM fixtures f
    WHERE f.team_away_id = t.id AND (away_goals - home_goals)>2
    GROUP BY f.team_away_id
    HAVING COUNT(*)>=2
)
AND t.id IN
(
    SELECT f.team_home_id
    FROM fixtures f
    WHERE f.team_home_id = t.id AND (home_goals - away_goals)>2
)
AND p.rolep='A'
GROUP BY t.name
HAVING COUNT(*)>=5;
```

## Esercizio 3

Si supponga di voler stilare una classifica fair play, assegnando un punto per ogni partita giocata senza ricevere un'ammonizione o espulsione. Si crei una query SQL che produce come risultato il nome dei giocatori che nel corso dell'anno 2009 hanno totalizzato più di 14 punti giocando almeno 17 partite. Il risultato dovrà essere del tipo (Nomegiocatore, NomeSquadra, punteggio)

## Soluzione

```
CREATE OR REPLACE VIEW partite_giocate AS
(
    SELECT p.id, COUNT(*) AS num_partite
    FROM players p
    JOIN votes v ON v.player_id = p.id
    JOIN rounds r ON v.round_id = r.id
    WHERE r.day_r BETWEEN '2009-01-01' AND '2009-12-31'
    group by p.id
);

SELECT p.name, t.name, COUNT(*) AS Punteggio
FROM players p JOIN teams t
ON p.team_id = t.id
JOIN votes v ON v.player_id = p.id
JOIN rounds r ON v.round_id = r.id
JOIN partite_giocate pg ON p.id=pg.id
WHERE ammonizione=0 AND espulsione=0 AND r.day_r BETWEEN '2009-01-01' AND '2009-12-31' AND pg.num_partite>=17
GROUP BY p.name, t.name
HAVING Punteggio >14;
```

## Esercizio 4

Si crei una query SQL che produca come risultato, per ogni squadra del campionato, il numero di giocatori che hanno realizzato almeno due goal su rigore. Il risultato dovrà essere del tipo (NomeSquadra, NumeroRigoristi). Si possono omettere le tuple con conteggio nullo.

## Soluzione

```
CREATE OR REPLACE VIEW Rigoristi (tid , pid , rigori) AS (  
SELECT team_id , p.id , sum(rigori_fatti)  
FROM players p , votes v  
WHERE p.id=v.player_id  
GROUP BY(p.id));
```

```
SELECT t.name , count(*) as NumeroRigoristi  
FROM teams t , Rigoristi r  
WHERE t.id=r.tid  
AND r.rigori >= 2  
GROUP BY t.name,t.id;
```

## Esercizio 5

Si scriva una query che permetta di settare il campo new\_column della tabella players (crearlo del tipo appropriato se necessario). Dopo l'esecuzione della query il campo deve contenere la somma di ammonizioni ed espulsioni ricevute dal giocatore nel campionato. Si aggiunga la query “SELECT new\_column FROM players” di seguito alla soluzione dell'esercizio.

## Soluzione

```
UPDATE players p  
SET new_column = (  
SELECT COUNT(*)  
FROM votes  
WHERE player_id=p.id AND (espulsione=1 or ammonizione=1)  
GROUP BY player_id  
);
```