

CAMPIONATI DI CALCIO ITALIANO

Progettazione Base di Dati

Martini Davide, Nigro Simone

Indice

1. Descrizione del progetto

- 1.1. Requisiti strutturati
- 1.2. Operazioni sulla base di dati
- 1.3. Tabella dei volumi

2. Progettazione concettuale

- 2.1. Modello concettuale: Entità-Associazione (E/R)
- 2.2. Dizionario dei dati
 - 2.2.1. Entità
 - 2.2.2. Associazioni
- 2.3. Schema concettuale, regole di vincolo

3. Progettazione logica

- 3.1. Ristrutturazione schema E/R
- 3.2. Modello logico: relazionale
- 3.3. Schema logico, regole di vincolo

4. Codice SQL

- 4.1. Struttura
- 4.2. Query

5. Note

1 Descrizione del progetto

Project manager	Martini Davide, Nigro Simone
Committente	Lega Calcio
Data inizio	1 ottobre 2016
Data fine	30 marzo 2017

Si vuole realizzare una base di dati che consenta di analizzare dettagliatamente l'andamento delle partite (formazioni, gol, sostituzioni ed eventuali provvedimenti disciplinari) e di memorizzare i risultati ed i dati delle diverse stagioni per le varie categorie del campionato italiano di calcio. È necessario poi poter tener traccia dei contratti dei giocatori e degli allenatori di ciascuna squadra.

1.1 Requisiti strutturati

<i>Categoria</i>	per ogni categoria, identificata dal nome della stessa, si vuole memorizzare il numero delle squadre che vi possono partecipare per ogni stagione, considerando che esso può variare in seguito a cambiamenti di regole stabilite dalla Lega Calcio.
<i>Stagione</i>	per ogni stagione, identificata dall'anno di riferimento della stessa (composto dall'anno di inizio e dall'anno di fine, es. 2016/2017) e dalla categoria cui appartiene, si vogliono memorizzare il calendario delle partite che si svolgeranno e le squadre partecipanti.
<i>Partita</i>	ogni partita viene identificata dalla stagione in cui viene giocata, dalla giornata del calendario e dalle squadre che vi partecipano. Per ognuna di esse si vuole memorizzare il risultato finale e le formazioni schierate dalle due squadre.
<i>Squadra</i>	ogni squadra viene identificata dal nome della società cui appartiene. Per ognuna si vogliono memorizzare l'anno di fondazione e lo stadio in cui gioca le partite di casa. Inoltre si vogliono tenere in memoria tutti i giocatori che ne fanno parte ed il loro allenatore con i relativi contratti stipulati dalla società, le partite giocate e le formazioni schierate per ognuna di esse. Infine si vuole tener traccia dei punti ottenuti (e di penalizzazione) nella stagione a cui sta partecipando la squadra.
<i>Persona</i>	per ogni persona, identificata da un codice, si vuole memorizzare il nome, il cognome, la data di nascita e la nazionalità. Sono previste due tipologie di persone: allenatore e giocatore. Per ogni allenatore vogliamo sapere quale squadra allena ed il contratto che ha stipulato con la società; mentre per il giocatore il ruolo principale, il contratto, lo stato (squalificato, infortunato, etc.) e tutti gli eventi che compie in una data partita.
<i>Contrattoallenatore</i>	per ogni contratto allenatore, identificato dalla data di inizio dell'esperienza, dalla squadra allenata e dal commissario tecnico, si vuole memorizzare la data finale del contratto, il motivo per cui è stato stipulato e la causa per la quale esso ha avuto fine.
<i>Contrattogiocatore</i>	per ogni contratto giocatore, identificato dalla data di inizio dell'esperienza, dal giocatore e dalla squadra in cui gioca, si vuole memorizzare la data finale del contratto, il motivo per cui è stato stipulato, la causa per la quale esso ha avuto fine e il numero di maglia che gli è stato assegnato.

<i>Specifica contratto</i>	è identificata da una descrizione della causa di inizio o fine di un contratto di un giocatore o di un allenatore (es. esonero, richiamo, recesso, etc.).
<i>Stato</i>	esso viene identificato dalla data in cui ha inizio, dal giocatore cui si riferisce e dal tipo di condizione in cui si trova. Vogliamo inoltre memorizzare la data di fine. Ogni giocatore potrebbe trovarsi contemporaneamente due diversi tipi di stato (es. infortunio e squalifica).
<i>Formazione</i>	ognuna viene identificata da un codice. Essa fa riferimento ai giocatori schierati da una squadra in una specifica partita e per ognuno di essi vogliamo sapere se sono titolari o a disposizione, sapendo che nel complesso ognuna deve essere composta da 11 titolari e 9 panchinari. Vogliamo inoltre memorizzare gli eventuali eventi e cambi avvenuti con i relativi minuti in cui avvengono, considerando che una squadra può effettuare da 0 a 3 cambi in una partita.
<i>Evento</i>	è identificato dalla sua tipologia, dal giocatore che ne è soggetto e dal minuto in cui avviene.
<i>Tipologia evento</i>	è identificato dalla descrizione dell'evento a cui si riferisce, intesa come insieme di possibili avvenimenti (es. ammonizione, espulsione, infortunio, goal, autogoal).
<i>Tipologia stato</i>	è identificato dalla descrizione dello stato a cui si riferisce, intesa come insieme di possibili stati (es. squalificato, infortunato, etc.).

1.2 Operazioni sulla base di dati

NOME OPERAZIONE	TIPO	FREQUENZA
Inserimento nuova stagione	Modifica	1/anno
Inserimento nuova partita	Modifica	2000/anno
Inserimento nuova squadra	Modifica	5/anno
Inserimento nuova formazione	Modifica	4000/anno
Inserimento nuovo allenatore	Modifica	10/anno
Inserimento nuovo giocatore	Modifica	100/anno
Inserimento nuovo contratto	Modifica	200/anno
Inserimento nuovo stato	Modifica	1500/anno
Inserimento nuovo evento	Modifica	20000/anno
Elenco risultati partite	Interrogazione	1/settimana
Calcolo classifica	Interrogazione	1/settimana
Elenco giocatori di una squadra	Interrogazione	1/mese
Elenco eventi di una partita	Interrogazione	10/settimana

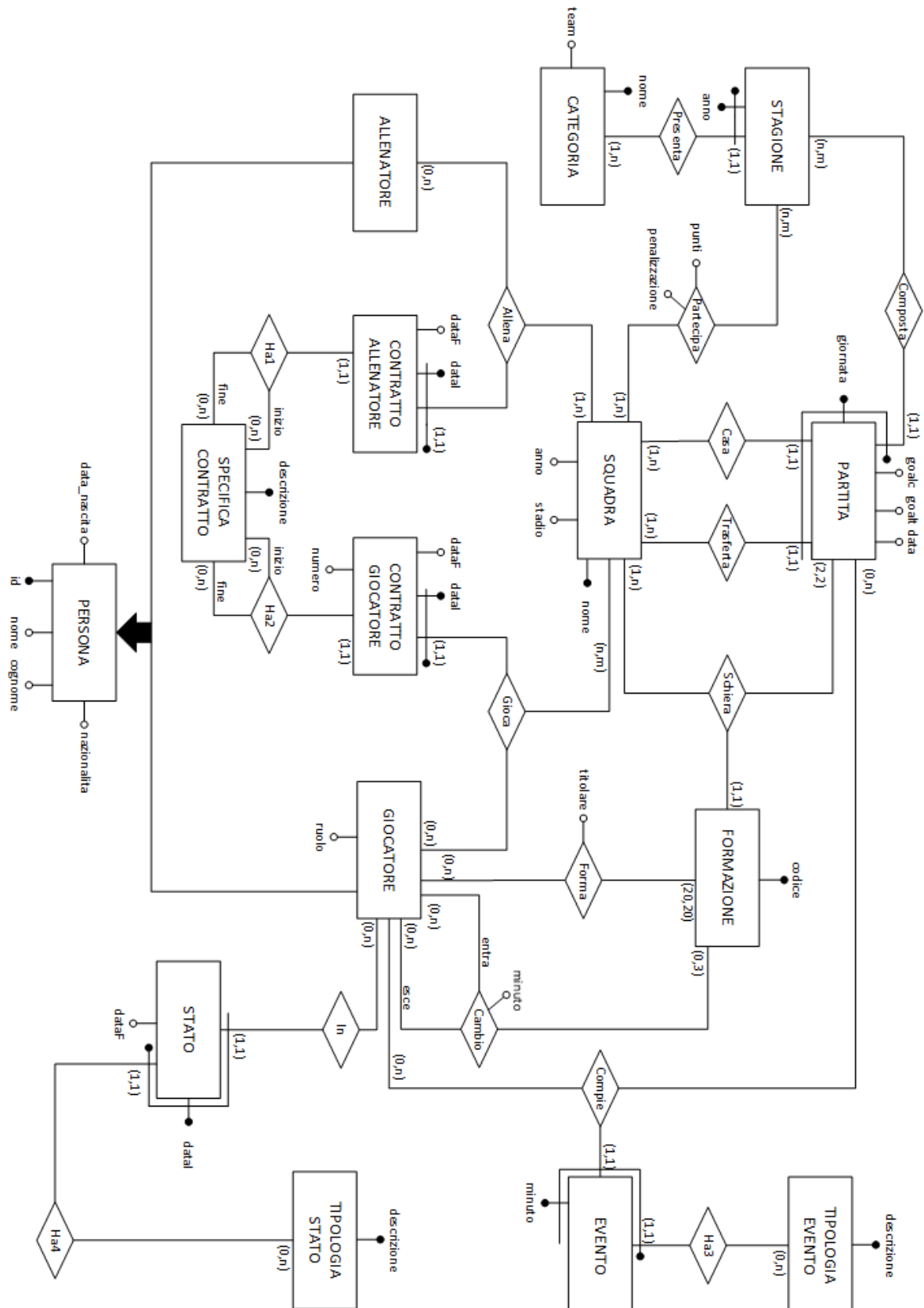
1.3 Tabella dei volumi

CONCETTO	TIPO	VOLUME
Categoria	Entità	2
Stagione	Entità	1
Partita	Entità	842
Squadra	Entità	42
Formazione	Entità	1684
Allenatore	Entità	60
Contrattoallenatore	Entità	80
Giocatore	Entità	1300
Contrattogiocatore	Entità	1500
Specificacontratto	Entità	10
Stato	Entità	160
Evento	Entità	3900
Tipologiaevento	Entità	6
Tipologistato	Entità	3
presenta	Associazione	2
composta	Associazione	850
partecipa	Associazione	42
casa	Associazione	850
trasferta	Associazione	850
schiera	Associazione	1700
allena	Associazione	80
gioca	Associazione	1300
ha1	Associazione	80
ha2	Associazione	1500
forma	Associazione	33680
cambio	Associazione	4300
in	Associazione	150
compie	Associazione	3900
ha3	Associazione	3900
ha4	Associazione	160

2 Progetto concettuale

2.1 Modello concettuale: Entity/Relationship (E/R)

Schema concettuale modellato sulla base delle specifiche richieste:



2.2 Dizionario dei dati

2.2.1 Entità

ENTITA'	DESCRIZIONE	ATTRIBUTI	IDENTIFICATORE
Categoria	Livello competizione	nome, team	nome
Stagione	Anno competizione	anno	anno, categoria
Partita	Incontro tra due squadre	giornata, goalc, goalt, data	giornata, stagione, squadrac, squadrat
Squadra	Organico composto da più giocatori e da un allenatore	nome, anno, stadio	nome
Formazione	Giocatori schierati in una partita	codice	codice
Allenatore	Commissario tecnico addetto all'organizzazione della squadra	id, nome, cognome, nazionalità, datanascita	id
Contrattoallenatore	Accordo tra società e allenatore	datai, dataf	datai, squadra, allenatore
Giocatore	Atleta appartenente ad una squadra	id, nome, cognome, nazionalità, datanascita, ruolo	id
Contrattogiocatore	Accordo tra società e giocatore	datai, dataf, numero	datai, squadra, giocatore
Specificacontratto	Tipologia di tesseramento	descrizione	descrizione
Stato	Condizione atletica o disciplinare di un giocatore	datai, dataf	tipologiastato, datai, giocatore
Evento	Avvenimento sorto durante una partita che coinvolge un giocatore	minuto	minuto, giocatore, partita, tipologiaevento
Tipologiaevento	Genere di evento	descrizione	descrizione
Tipologiastato	Genere di stato	descrizione	descrizione

2.2.2 Associazioni

ASSOCIAZIONE	ATTRIBUTI	ENTITA' COLLEGATE
presenta		categoria(1,n), stagione(1,1)
composta		stagione(n,m), partita(1,1)
partecipa	punti, penalizzazione	stagione(n,m), squadra(1,n)
casa		partita(1,1), squadra(1,n)
trasferta		partita(1,1), squadra(1,n)
schiera		squadra(1,n), partita(2,2), formazione(1,1)
allena		allenatore(0,n), contrattoallenatore(1,1), squadra(1,n)
gioca		giocatore(0,n), contrattogiocatore(1,1), squadra(n,m)
ha1		contrattoallenatore(1,1), specificacontratto(0,n), specificacontratto(0,n)
ha2		contratto giocatore(1,1), specificacontratto(0,n), specificacontratto(0,n)
forma	titolare	formazione(20,20), giocatore(0,n)
cambio	minuto	formazione(0,3), giocatore(0,n), giocatore(0,n)
in		giocatore(0,n), stato(1,1)
compie		evento(1,1), formazione(0,n), giocatore(0,n)
ha3		tipologiaevento(0,n), evento(1,1)
ha4		tipologistato(0,n), stato(1,1)

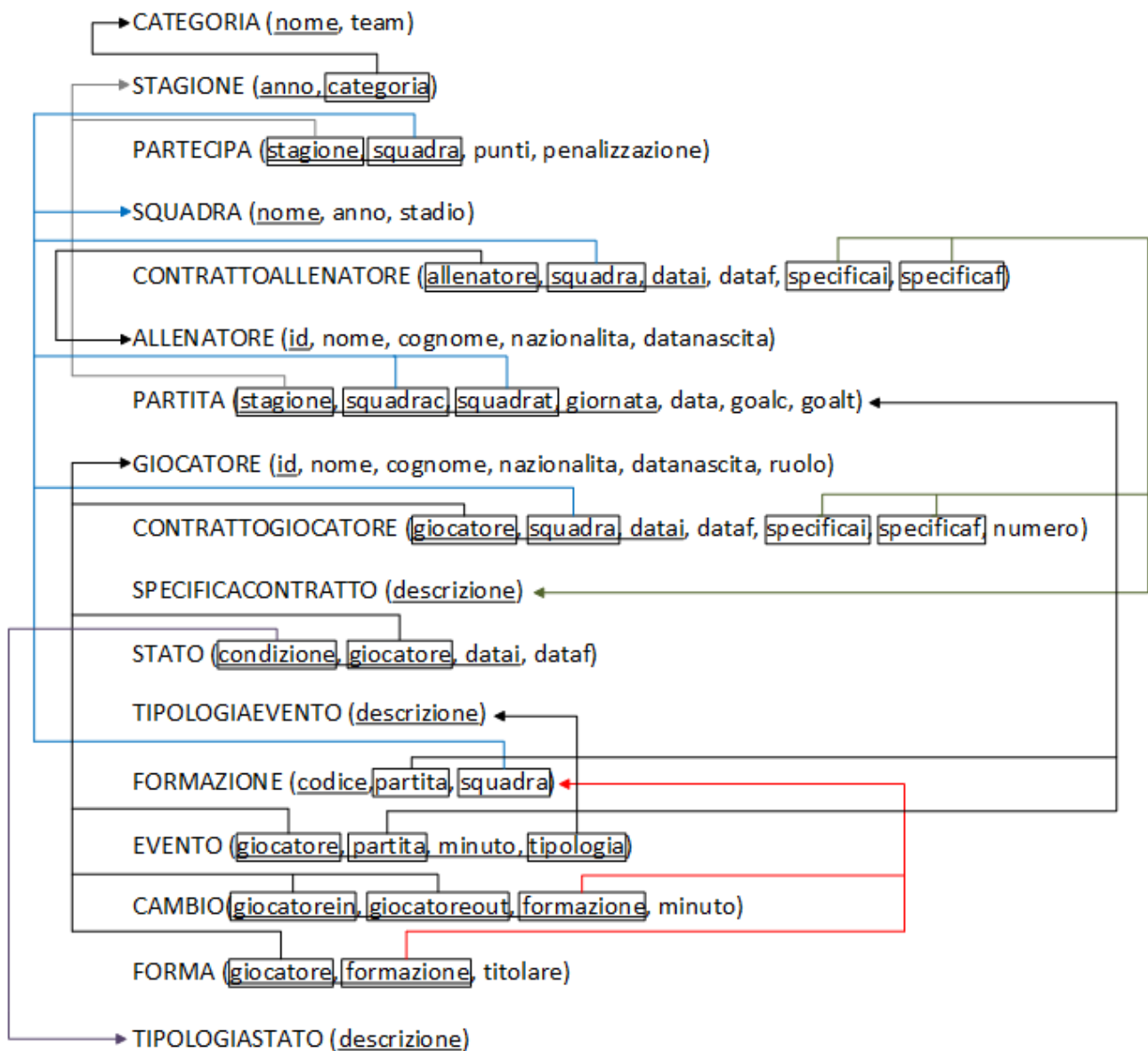
2.3 Schema concettuale, regole di vincolo

Per rispettare la rappresentazione fedele della realtà di interesse descritta è necessario definire delle regole di vincolo per concetti altrimenti non esprimibili utilizzando il modello E.R..

- **Regola di vincolo 1:** il giocatore coinvolto nell'evento in una determinata partita deve essere tra i giocatori presenti in campo nel determinato minuto in cui accade.
- **Regola di vincolo 2:** un giocatore schierato in una formazione deve appartenere alla squadra che partecipa alla partita di riferimento e il suo stato non può essere indisponibile (squalificato, infortunato, etc.).
- **Regola di vincolo 3:** non si tiene conto dei minuti di recupero e gli eventi avvenuti in tali minuti vengono inglobati nel minuto 45 e/o 90.
- **Regola di vincolo 4:** i punti di penalizzazione di una squadra vengono detratti dai punti totali nel momento in cui vengono inseriti.
- **Regola di vincolo 5:** la seconda ammonizione allo stesso giocatore nella stessa partita viene considerata come espulsione e provoca una squalifica per la giornata successiva.
- **Regola di vincolo 6:** un giocatore che contrae una serie di 5 ammonizioni in partite differenti provoca una squalifica per la giornata successiva a quella in cui riceve la quinta ammonizione.
- **Regola di vincolo 7:** in caso di esonero di un allenatore il suo contratto viene interrotto con causale finale "Esonero". In caso di richiamo si ipotizza che venga stipulato un nuovo contratto con opportuna causale di inizio.
- **Regola di vincolo 8:** un giocatore coinvolto in un cambio non deve essere stato espulso in un minuto antecedente nella stessa partita. Inoltre un giocatore che è stato sostituito non può rientrare in campo nella medesima partita.

3.2 Modello Logico: Relazionale

Lo schema logico prodotto per la rappresentazione della realtà di interesse è il seguente:



3.3 Schema Logico, Regole di vincolo

- RV1: In una stagione devono partecipare almeno n squadre (con $n > 1$);
- RV2: Una squadra può effettuare da 0 a 3 cambi in una partita;
- RV3: Una formazione è composta da 20 giocatori per squadra, di cui 11 titolari e 9 a disposizione in panchina;
- RV4: Nessun attributo nello schema relazionale è opzionale (cioè non sono ammessi valori nulli);
- RV5: Ogni partita deve avere obbligatoriamente due formazioni di cui una per la squadra di casa e una per la squadra che gioca in trasferta.

4 Codice SQL

4.1 Struttura

```
CREATE TABLE allenatore (  
    id integer NOT NULL,  
    nome character varying(20) NOT NULL,  
    cognome character varying(20) NOT NULL,  
    nazionalita character varying(3) NOT NULL,  
    datanascita date NOT NULL  
);  
  
CREATE TABLE cambio (  
    giocatorein integer NOT NULL,  
    giocatoreout integer NOT NULL,  
    formazione integer NOT NULL,  
    minuto smallint NOT NULL,  
    CONSTRAINT cambio_minuto_check CHECK ((minuto > 0) AND (minuto < 91))  
);  
  
CREATE TABLE categoria (  
    nome character varying(20) NOT NULL,  
    team smallint NOT NULL,  
    CONSTRAINT categoria_team_check CHECK ((team < 50))  
);  
  
CREATE TABLE contrattogiocatore (  
    giocatore integer NOT NULL,  
    squadra character varying(20) NOT NULL,  
    datai date NOT NULL,  
    dataf date NOT NULL,  
    specifica_i character varying(20) NOT NULL,  
    specifica_f character varying(20) NOT NULL,  
    numero smallint NOT NULL,  
    CONSTRAINT contrattogiocatore_numero_check CHECK (((numero > 0) AND (numero < 100)))  
);  
  
CREATE TABLE evento (  
    giocatore integer NOT NULL,  
    anno character varying(9) NOT NULL,  
    categoria character varying(20) NOT NULL,  
    squadrac character varying(20) NOT NULL,  
    squadrat character varying(20) NOT NULL,
```

```

giornata smallint NOT NULL,
minuto smallint NOT NULL,
tipologia character varying(20) NOT NULL,
CONSTRAINT evento_giornata_check CHECK ((giornata < 100)),
CONSTRAINT evento_minuto_check CHECK ((minuto > 0) AND (minuto < 91)))
);

CREATE TABLE giocatore (
    id integer NOT NULL,
    nome character varying(20) NOT NULL,
    cognome character varying(20) NOT NULL,
    nazionalita character varying(3) NOT NULL,
    datanascita date NOT NULL,
    ruolo character varying(14) NOT NULL
);

CREATE TABLE partecipa (
    anno character varying(9) NOT NULL,
    categoria character varying(20) NOT NULL,
    squadra character varying(20) NOT NULL,
    punti smallint NOT NULL,
    penalizzazione smallint DEFAULT 0 NOT NULL
);

CREATE TABLE contrattoallenatore (
    allenatore integer NOT NULL,
    squadra character varying(20) NOT NULL,
    datai date NOT NULL,
    dataf date NOT NULL,
    specifica character varying(20) NOT NULL,
    specifica character varying(20) NOT NULL
);

CREATE TABLE forma (
    giocatore integer NOT NULL,
    formazione integer NOT NULL,
    titolare boolean NOT NULL
);

CREATE TABLE formazione (
    codice integer NOT NULL,
    anno character varying(9) NOT NULL,
    categoria character varying(20) NOT NULL,
    squadrac character varying(20) NOT NULL,
    squadrat character varying(20) NOT NULL,
    giornata smallint NOT NULL,
    squadra character varying(20) NOT NULL,
    CONSTRAINT formazione_giornata_check CHECK (((giornata > 0) AND (giornata < 100)))
);

CREATE TABLE partita (
    anno character varying(9) NOT NULL,
    categoria character varying(20) NOT NULL,
    squadrac character varying(20) NOT NULL,
    squadrat character varying(20) NOT NULL,
    giornata smallint NOT NULL,
    data timestamp without time zone NOT NULL,
    goalc smallint NOT NULL,
    goalt smallint NOT NULL,
    CONSTRAINT partita_giornata_check CHECK (((giornata > 0) AND (giornata < 100)))
);

CREATE TABLE stato (
    condizione character varying(20) NOT NULL,
    giocatore integer NOT NULL,
    datai date NOT NULL,
    dataf date NOT NULL
);

```

```

CREATE TABLE specificacontratto (
    descrizione character varying(20) NOT NULL
);

CREATE TABLE squadra (
    nome character varying(20) NOT NULL,
    anno smallint NOT NULL,
    stadio character varying(20) NOT NULL,
    CONSTRAINT squadra_anno_check CHECK ((anno < 2050))
);

CREATE TABLE stagione (
    anno character varying(9) NOT NULL,
    categoria character varying(20) NOT NULL
);

CREATE TABLE tipologiaevento (
    descrizione character varying(20) NOT NULL
);

CREATE TABLE tipologiastato (
    descrizione character varying(20) NOT NULL
);

ALTER TABLE ONLY allenatore
    ADD CONSTRAINT allenatore_pkey PRIMARY KEY (id);

ALTER TABLE ONLY cambio
    ADD CONSTRAINT cambio_pkey PRIMARY KEY (giocatorein, giocatoreout, formazione);

ALTER TABLE ONLY categoria
    ADD CONSTRAINT categoria_pkey PRIMARY KEY (nome);

ALTER TABLE ONLY contrattoallenatore
    ADD CONSTRAINT contrattoallenatore_pkey PRIMARY KEY (allenatore, squadra, datai);

ALTER TABLE ONLY contrattogiocatore
    ADD CONSTRAINT contrattogiocatore_pkey PRIMARY KEY (giocatore, squadra, datai);

ALTER TABLE ONLY evento
    ADD CONSTRAINT evento_pkey PRIMARY KEY (giocatore, anno, categoria, squadrac,
squadrat, giornata, minuto, tipologia);

ALTER TABLE ONLY forma
    ADD CONSTRAINT forma_pkey PRIMARY KEY (giocatore, formazione);

ALTER TABLE ONLY formazione
    ADD CONSTRAINT formazione_pkey PRIMARY KEY (codice);

ALTER TABLE ONLY giocatore
    ADD CONSTRAINT giocatore_pkey PRIMARY KEY (id);

ALTER TABLE ONLY partecipa
    ADD CONSTRAINT partecipa_pkey PRIMARY KEY (anno, categoria, squadra);

ALTER TABLE ONLY partita
    ADD CONSTRAINT partita_pkey PRIMARY KEY (anno, categoria, squadrac, squadrat,
giornata);

ALTER TABLE ONLY specificacontratto
    ADD CONSTRAINT specificacontratto_pkey PRIMARY KEY (descrizione);

ALTER TABLE ONLY squadra
    ADD CONSTRAINT squadra_pkey PRIMARY KEY (nome);

ALTER TABLE ONLY stagione
    ADD CONSTRAINT stagione_pkey PRIMARY KEY (anno, categoria);

ALTER TABLE ONLY stato
    ADD CONSTRAINT stato_pkey PRIMARY KEY (condizione, giocatore, datai);

```

```

ALTER TABLE ONLY tipologiaevento
    ADD CONSTRAINT tipologiaevento_pkey PRIMARY KEY (descrizione);

ALTER TABLE ONLY tipologiastato
    ADD CONSTRAINT tipologiastato_pkey PRIMARY KEY (descrizione);

ALTER TABLE ONLY cambio
    ADD CONSTRAINT cambio_formazione_fkey FOREIGN KEY (formazione) REFERENCES
    formazione(codice) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY cambio
    ADD CONSTRAINT cambio_giocatorein_fkey FOREIGN KEY (giocatorein) REFERENCES
    giocatore(id) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY cambio
    ADD CONSTRAINT cambio_giocatoreout_fkey FOREIGN KEY (giocatoreout) REFERENCES
    giocatore(id) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY contrattoallenatore
    ADD CONSTRAINT contrattoallenatore_allenatore_fkey FOREIGN KEY (allenatore)
    REFERENCES allenatore(id) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY contrattoallenatore
    ADD CONSTRAINT contrattoallenatore_specificaf_fkey FOREIGN KEY (specificaf)
    REFERENCES specificacontratto(descrizione) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY contrattoallenatore
    ADD CONSTRAINT contrattoallenatore_specificai_fkey FOREIGN KEY (specificai)
    REFERENCES specificacontratto(descrizione) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY contrattoallenatore
    ADD CONSTRAINT contrattoallenatore_squadra_fkey FOREIGN KEY (squadra) REFERENCES
    squadra(nome) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY contrattogiocatore
    ADD CONSTRAINT contrattogiocatore_giocatore_fkey FOREIGN KEY (giocatore) REFERENCES
    giocatore(id) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY contrattogiocatore
    ADD CONSTRAINT contrattogiocatore_specificaf_fkey FOREIGN KEY (specificaf) REFERENCES
    specificacontratto(descrizione) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY contrattogiocatore
    ADD CONSTRAINT contrattogiocatore_specificai_fkey FOREIGN KEY (specificai) REFERENCES
    specificacontratto(descrizione) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY contrattogiocatore
    ADD CONSTRAINT contrattogiocatore_squadra_fkey FOREIGN KEY (squadra) REFERENCES
    squadra(nome) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY evento
    ADD CONSTRAINT evento_anno_fkey FOREIGN KEY (anno, categoria, squadrac, squadrat,
    giornata) REFERENCES partita(anno, categoria, squadrac, squadrat, giornata) ON UPDATE
    CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY evento
    ADD CONSTRAINT evento_giocatore_fkey FOREIGN KEY (giocatore) REFERENCES giocatore(id)
    ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY evento
    ADD CONSTRAINT evento_tipologia_fkey FOREIGN KEY (tipologia) REFERENCES
    tipologiaevento(descrizione) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY forma
    ADD CONSTRAINT forma_formazione_fkey FOREIGN KEY (formazione) REFERENCES
    formazione(codice) ON UPDATE CASCADE ON DELETE RESTRICT;

```

```

ALTER TABLE ONLY forma
    ADD CONSTRAINT forma_giocatore_fkey FOREIGN KEY (giocatore) REFERENCES giocatore(id)
ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY formazione
    ADD CONSTRAINT formazione_anno_fkey FOREIGN KEY (anno, categoria, squadrac, squadrat,
giornata) REFERENCES partita(anno, categoria, squadrac, squadrat, giornata) ON UPDATE
CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY formazione
    ADD CONSTRAINT formazione_squadra_fkey FOREIGN KEY (squadra) REFERENCES squadra(nome)
ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY partecipa
    ADD CONSTRAINT partecipa_anno_fkey FOREIGN KEY (anno, categoria) REFERENCES
stagione(anno, categoria) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY partecipa
    ADD CONSTRAINT partecipa_squadra_fkey FOREIGN KEY (squadra) REFERENCES squadra(nome)
ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY partita
    ADD CONSTRAINT partita_anno_fkey FOREIGN KEY (anno, categoria) REFERENCES
stagione(anno, categoria) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY partita
    ADD CONSTRAINT partita_squadrac_fkey FOREIGN KEY (squadrac) REFERENCES squadra(nome)
ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY partita
    ADD CONSTRAINT partita_squadrat_fkey FOREIGN KEY (squadrat) REFERENCES squadra(nome)
ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY stagione
    ADD CONSTRAINT stagione_categoria_fkey FOREIGN KEY (categoria) REFERENCES
categoria(nome) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY stato
    ADD CONSTRAINT stato_condizione_fkey FOREIGN KEY (condizione) REFERENCES
tipologiastato(descrizione) ON UPDATE CASCADE ON DELETE RESTRICT;

ALTER TABLE ONLY stato
    ADD CONSTRAINT stato_giocatore_fkey FOREIGN KEY (giocatore) REFERENCES giocatore(id)
ON UPDATE CASCADE ON DELETE RESTRICT;

```

4.2 Query

1) Classifica ammonizioni Serie A 2016/2017:

```

SELECT evento.giocatore, giocatore.nome, giocatore.cognome,
contrattogiocatore.squadra, count(*) AS numeroammonizioni FROM ((evento
NATURAL JOIN contrattogiocatore) JOIN giocatore ON ((evento.giocatore =
giocatore.id))) WHERE ((evento.tipologia = 'Ammonizione') AND
(evento.categoria) = 'Serie A')) GROUP BY evento.giocatore, giocatore.nome,
giocatore.cognome, contrattogiocatore.squadra ORDER BY count(*) DESC;

```

2) Classifica marcatori Serie A 2016/2017:

```
SELECT evento.giocatore, giocatore.nome, giocatore.cognome,
contrattogiocatore.squadra, count(*) AS numerosgoal FROM ((evento NATURAL
JOIN contrattogiocatore) JOIN giocatore ON ((evento.giocatore =
giocatore.id))) WHERE ((evento.tipologia = 'Goal' or evento.tipologia=
'Rigore') AND (evento.categoria = 'Serie A')) GROUP BY evento.giocatore,
giocatore.nome, giocatore.cognome, contrattogiocatore.squadra ORDER BY
count(*) DESC;
```

3) Classifica sostituiti Serie A 2016/2017:

```
SELECT giocatore.id AS giocatore, giocatore.nome, giocatore.cognome,
contrattogiocatore.squadra, count(*) AS numerosostituzioni FROM (((cambio
JOIN giocatore ON ((cambio.giocatoreout = giocatore.id))) JOIN
contrattogiocatore ON ((giocatore.id = contrattogiocatore.giocatore)))
NATURAL JOIN partecipa) WHERE ((partecipa.categoria = 'Serie A')) GROUP BY
giocatore.id, giocatore.nome, giocatore.cognome,
contrattogiocatore.squadra ORDER BY count(*) DESC;
```

4) Classifica subentri Serie A 2016/2017:

```
SELECT giocatore.id AS giocatore, giocatore.nome, giocatore.cognome,
contrattogiocatore.squadra, count(*) AS numerosostituzioni FROM (((cambio
JOIN giocatore ON ((cambio.giocatoreout = giocatore.id))) JOIN
contrattogiocatore ON ((giocatore.id = contrattogiocatore.giocatore)))
NATURAL JOIN partecipa) WHERE ((partecipa.categoria = 'Serie A')) GROUP BY
giocatore.id, giocatore.nome, giocatore.cognome,
contrattogiocatore.squadra ORDER BY count(*) DESC;
```

5) Controllo che un giocatore indisponibile non giochi, il risultato di questa query mi restituisce una lista di giocatori presenti in partite in cui non possono esser presenti per problemi di stato:

```
CREATE VIEW giocgiorn AS
SELECT forma.giocatore, formazione.giornata, partita.data FROM
((formazione JOIN forma ON ((formazione.codice = forma.formazione)))
NATURAL JOIN partita);

SELECT stato.condizione, stato.giocatore, stato.datai, stato.dataf,
giocgiorn.giornata, giocgiorn.data AS indisponibilita FROM (stato JOIN
giocgiorn ON ((stato.giocatore = giocgiorn.giocatore))) WHERE
((giocgiorn.data > stato.datai) AND (giocgiorn.data < stato.dataf));
```

5 Note

Strumenti utilizzati: Microsoft Word2016, Microsoft Visio, PostgreSQL, Firefox Web Browser, Komodo Edit.