

OneFootball DB



Abstract

“OneFootball” consiste in un sito web/applicazione contenente dati di ambito calcistico tra cui trasferimenti dei vari giocatori e news in generale sulle varie squadre.

In particolare possiamo consultare informazioni generiche sulle squadre e i suoi piazzamenti nelle varie competizioni (per ogni anno), informazioni sulle partite giocate dalle squadre, statistiche relative al giocatore singolo (gol, assist, trasferimenti, ecc.) e una sezione dedicata agli articoli relativi alle varie squadre.

Possono essere registrate tutte le squadre del mondo con annesse competizioni nazionali e internazionali.

Analisi dei requisiti

Il progetto vuole rappresentare una base di dati che rappresenti informazioni di ambito calcistico principalmente su giocatori, squadre e partite delle varie competizioni.

Ogni **squadra** è identificata dal proprio nome univoco ed è composta da un certo numero di giocatori e da un allenatore principale.

Ogni squadra può effettuare diverse partite a seconda delle competizioni a cui partecipa, sia nel proprio stadio che in quello delle squadre avversarie, viene definita nei diversi casi come squadra in casa o squadra in trasferta.

Uno **stadio** definisce il campo di gioco di una squadra ed è identificato dal suo nome e della città in cui è situato.

Ogni stadio ha inoltre una capienza che definisce il numero di persone massimo che può contenere.

Ogni **partita** è identificata dalla data in cui viene giocata e dalla squadra che gioca in casa, è poi composta dalla squadra ospite e dal punteggio finale descritto come punti segnati dalla prima squadra e dalla rispettiva in trasferta. Viene diretta da diversi arbitri. Una partita fa parte di una competizione.

In ogni **competizione** vengono giocate svariate partite.

Una competizione può essere di tre tipi:

1. Coppa nazionale (es. Coppa Italia) identificata dalla nazione, unica per Paese;

2. Lega (es. Serie A) identificata dalla nazione e dal livello della lega, detto divisione (es. Serie A è la prima divisione, Serie B è la seconda, ecc. in Italia);
3. Torneo internazionale (es. Champions League) identificato dalla confederazione (es. UEFA, CONMEBOL, ecc.) e dal livello (es. Champions è il primo livello, Europa è il secondo, Conference il terzo in Europa).

I **professionisti** vengono identificati tramite le proprie generalità, in particolare si fa uso del Codice Fiscale (o del Tax ID fuori dall'Europa) per evitare ambiguità.

La gerarchia individuata rappresenta i professionisti essenziali nella nostra rappresentazione:

1. Le partite vengono arbitrate da una triade di **arbitri** (principale + due guardalinee) con l'aiuto del quarto uomo e gli assistenti il sala VAR, distinti dal ruolo associato che può variare di partita in partita. Ogni arbitro ha una licenza che determina quali partite può arbitrare.
2. I **calciatori** che compongono una squadra hanno un ruolo, un numero di maglia e un prezzo fissato dalla società, detto cartellino, necessario nei trasferimenti.
3. L'**allenatore** prepara la squadra in vista delle partite.

Ogni giocatore detiene delle **statistiche** come gol fatti, assist, cartellini gialli e rossi assegnati durante le partite o gol subiti nel caso il calciatore soggetto abbia il ruolo di portiere.

Il sito ha inoltre una sezione dedicata agli **articoli** che trattano le news principali delle varie squadre, ogni articolo è identificato da un ID, è scritto da un autore, pubblicato in una data e contiene un testo.

Glossario dei termini

Termine	Descrizione
<u>Squadra</u>	Insieme di calciatori preparati da un allenatore in una società.
<u>Allenatore</u>	Figura sportiva incaricata dalla società alla preparazione della squadra. Unico per ogni squadra.
<u>Arbitro</u>	Ufficiale di gara incaricato alla direzione di una partita. Può assumere diversi ruoli oltre al principale, tra cui "guardalinee", assistenti come "quarto ufficiale" e addetti alla sala VAR.
<u>Stadio</u>	Struttura sportiva in cui vengono svolte le partite delle varie competizioni. Lo stadio determina la squadra che gioca come "in casa" e la squadra ospitata che gioca come "in trasferta", utile nell'assegnazione dei punteggi. Solitamente lo stadio determina anche il luogo della squadra.
<u>Divisione</u>	Livello che identifica il grado professionistico di una lega nazionale o di un torneo internazionale.
<u>Lega</u>	Ciclo di partite per l'assegnazione del titolo di campione alla squadra che si classifica al primo posto.
<u>Coppa nazionale</u>	Torneo ad eliminazione diretta svolto in una nazione al quale possono partecipare le squadre appartenenti alle leghe di divisione più alta (solitamente le prime tre).
<u>Confederazione</u>	Organizzazione sportiva tra diversi stati in una determinata zona geografica che si occupa di istituire tornei di livello internazionale.

<u>Torneo internazionale</u>	Torneo che viene gestito da una confederazione a cui prendono parte le squadre di una nazione in base alla posizione raggiunta nel proprio campionato.
------------------------------	--

Operazioni tipiche

Operazione	Tipo	Frequenza
Aggiornamento punteggi delle partite	S	10000 al mese
Visualizzazione delle partite	L	15000 al mese
Inserimento articoli	S	300 al mese
Visualizzazione della classifica di un campionato	L	1000 al mese
Assegnazione degli arbitri alle partite	S	30000 al mese
Aggiornamento delle statistiche dei giocatori	S	500 al giorno
Aggiornamento dei trasferimenti dei giocatori	S	200 al mese
Visualizzazione di informazioni sugli stadi	L	50 al mese
Aggiornamento di professionisti	S	1000 al mese
Visualizzazione dei vincitori delle coppe nazionali e non	L	100 al mese

Progettazione concettuale

Lista entità

Squadra		
<u>Nome</u>	varchar(30) Primary Key	
Fondazione (0,1)	date	
Città	varchar(40)	

Stadio		
<u>Nome</u>	varchar(50) Primary Key	
<u>Città</u>	varchar(40) Primary Key	Città in cui è situato lo stadio
Capienza	int	Numero di posti massimo

Articolo		
<u>ID</u>	int Primary Key	Identificativo articolo
Data	date	Data pubblicazione articolo
Testo	text	Contenuto articolo
Autore	varchar(40)	

Professionista		
<u>CF</u>	char(16) Primary Key	Codice fiscale
Nome	varchar(20)	
Cognome	varchar(20)	
DataNascita	date	
Nazionalità	varchar(30)	

Professionista si divide in **Arbitro**, **Allenatore** e **Calciatore** (gerarchia parziale):

Arbitro		
Livello licenza	smallint	Licenza che descrive le partite che può dirigere l'arbitro
Allenatore		
Calciatore		
Ruolo	ruolo AS ENUM(POR, DIF, CEN, ATT)	Portiere, difensore, centrocampista o attaccante
Numero	smallint	Numero di maglia
Cartellino (0,1)	smallint	Prezzo fissato dalla società per scambiare il giocatore (in milioni)

Statistica			
Periodo (attr. composto)	<u>DataInizio</u>	date Primary Key	Data inizio di raccolta dati
	DataFine	date	Data fine di raccolta dati
Gol	smallint		
Assist	smallint		
Gialli	smallint		Cartellini gialli assegnati
Rossi	smallint		Cartellini rossi assegnati
GolSubiti (0,1)	smallint		Gol subiti

Partita		
<u>Data</u>	date Primary Key	Data in cui viene svolta la partita
PunteggioCasa	smallint	Gol fatti dalla squadra di casa
PunteggioOspite	smallint	Gol fatti dalla squadra ospite

Competizione			
<u>Zona</u>		varchar(30) Primary Key	Nazione in cui si svolge la competizione
<u>Divisione</u>		smallint Primary key	Livello della competizione
Stagione (attr. composto)	Inizio	date	Mese e giorno di inizio stagione regolamentare
	Fine	date	Mese e giorno di fine stagione regolamentare
Nome		varchar(30)	Denominazione della competizione

Competizione si divide in **Coppa**, **Lega** e **TorneoInt** (gerarchia totale):

Coppa		
<u>Nazione</u>	varchar(30) Primary Key	Nazione in cui si svolge la competizione
Lega		
<u>Nazione</u>	varchar(30) Primary Key	
<u>Divisione</u>	smallint Primary Key	Livello della competizione
TorneoInt (Torneo Internazionale)		
<u>Confederazione</u>	varchar(10) Primary Key	Organizzazione che istituisce il torneo
<u>Divisione</u>	smallint Primary Key	Livello della competizione

Lista relazioni

Relazione	Entità coinvolte	Descrizione	Attributi
Dove	Squadra (1,1) Stadio (1,N)	Una squadra gioca le sue partite di casa in uno stadio. Uno stadio ha una o più squadre di casa.	
Notizia	Squadra (0,N) Articolo (1,1)	Una squadra può avere degli articoli dedicati. Un articolo è dedicato ad una squadra.	
Casa	Squadra (1,N) Partita (1,1)	Una squadra di casa gioca diverse partite. In una partita c'è solamente una squadra casalinga.	
Ospite	Squadra (1,N) Partita (1,1)	Una squadra ospite gioca diverse partite. In una partita c'è solamente una squadra in trasferta.	
Svolgimento	Competizione (1,N) Partita (1,1)	Una partita si svolge in una determinata competizione. Una competizione è formata da molteplici partite.	
Contratto	Calciatore (0,1) Squadra (1,N)	Un calciatore può avere un contratto con una squadra o essere svincolato. Una squadra ha molti calciatori.	DataInizio DataFine
Trasferimento	Calciatore (0,N) Squadra (0,N)	Una squadra può essere interessata a dei giocatori. Un giocatore può essere accostato a diverse squadre.	Veridicità Offerta
Detiene	Statistica (1,1) Calciatore (1,N)	Un calciatore detiene diverse statistiche nel corso degli anni. Una statistica annuale è detenuta da un calciatore.	
Allena	Allenatore(0,1) Squadra (0,1)	Un allenatore prepara una squadra o può essere svincolato. Una squadra ha un solo allenatore.	
Arbitraggio	Arbitro (0,N) Partita (1,N)	Un arbitro può aver diretto diverse partite. Una partita è diretta da uno o più arbitri.	Ruolo

Vincoli non rappresentabili tramite schema E-R

- In ogni entità o relazione dove sono presenti "DataInizio" e "DataFine" avremo che "DataInizio" < "DataFine".
- Ad ogni partita è necessario che le due squadre siano distinte.
- Data la possibilità per una partita di essere arbitrata da più ufficiali, un singolo arbitro non può arbitrare la stessa partita più di una volta.

Il diagramma ER illustra la struttura di un database per il calcio. Gli entità e le loro attributi sono:

- STATISTICA**: Gol Assist Gialli, Rossi, GolSubiti, Periodo.
- CALCIATORE**: Ruolo, Numero, DataInizio, DataFine, Periodo, DataFine, DataInizio.
- ALLENATORE**: CF, Nome, Cognome, Data, Nascita, Nazionalità.
- PROFESSIONISTA**: Data, Nascita, Nazionalità.
- ARBITRO**: Livello, Licenza.
- SQUADRA**: Nome, Fondazione, Nome, Città, Capienza.
- STADIO**: Nome, Città, Capienza.
- ARTICOLO**: ID, Data, Testo, Autore.
- COMPETIZIONE**: Nome, Inizio, Fine, Zona, Divisione.
- PARTITA**: Data, Casa, Ospite, Punteggio Casa, Punteggio Ospite.
- COPPA**: Nazione.
- TORNEOINT**: Confederazione, Divisione.
- LEGA**: Nazione, Divisione.

Le relazioni e le loro cardinalità sono:

- Detiene** (CALCIATORE a STATISTICA): 1, N.
- Contratto** (CALCIATORE a SQUADRA): 0, 1 a 1, N.
- Trasferimento** (SQUADRA a CALCIATORE): 1, 1 a 0, N.
- Allena** (ALLENATORE a SQUADRA): 0, 1 a 1, N.
- Dove** (SQUADRA a STADIO): 1, 1 a 1, N.
- Notizia** (SQUADRA a ARTICOLO): 1, 1 a 0, N.
- Casa** (SQUADRA a PARTITA): 1, 1 a 1, 1.
- Ospite** (SQUADRA a PARTITA): 1, 1 a 1, 1.
- Arbitraggio** (ARBITRO a PARTITA): 0, N a 1, N.
- Svolgimento** (COMPETIZIONE a PARTITA): 1, N a 1, 1.

Altre informazioni: La relazione **Detiene** ha un vincolo di totalità (cerchio pieno). La relazione **Svolgimento** ha un vincolo di totalità (cerchio pieno). La relazione **Arbitraggio** ha un vincolo di totalità (cerchio pieno). La relazione **Casa** ha un vincolo di totalità (cerchio pieno). La relazione **Ospite** ha un vincolo di totalità (cerchio pieno). La relazione **Detiene** ha un vincolo di totalità (cerchio pieno).

Analisi delle ridondanze

Eliminazione delle generalizzazioni

La tipologia Coppa, essendo singola per Nazione, occuperà una sola divisione (identificata dallo 0).

7

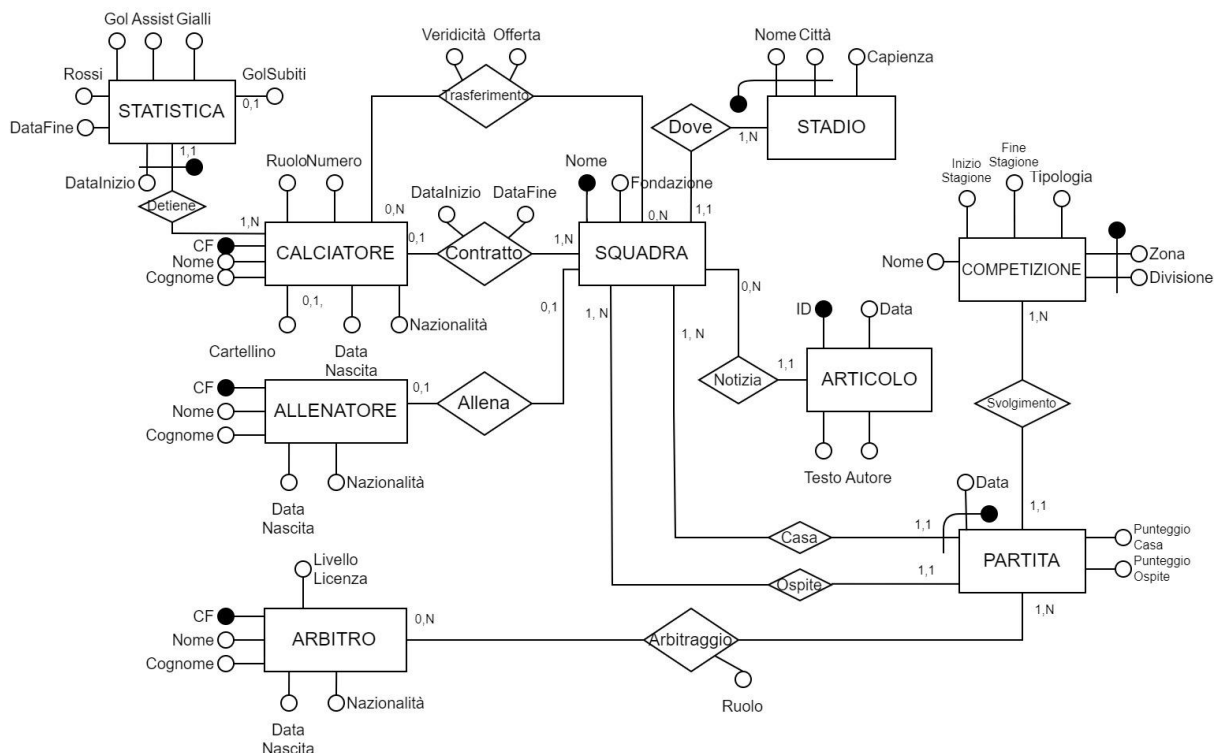
Scelta degli identificatori principali

Per l'entità Statistica era stato scelto di utilizzare il periodo, un attributo composto formato da una data di inizio ed una di fine, per identificare (insieme alla relazione "detiene") l'elenco di statistiche di un giocatore in un determinato lasso di tempo, in realtà la data finale risulta superflua in quanto il prossimo periodo registrerebbe le statistiche a partire da un giorno successivo rispetto la fine del periodo scorso. Si è scelto quindi di scomporre l'attributo composto Periodo e di implementare nella chiave primaria solamente la data di inizio e la relazione "detiene" con Calciatore.

Per l'entità Competizione si è scelto di utilizzare la zona o l'organizzazione che istituisce la singola competizione in modo da rappresentare anche tutte le varie divisioni calcistiche offerte dal paese o dalla confederazione (ad esempio "Serie B" è la seconda divisione Italiana), tenendo anche conto che per motivi legati alla sponsorizzazione il nome della lega potrebbe variare di stagione in stagione (ad esempio "Serie Bwin", "Serie BKT", ecc.).

Inoltre, considerando che la coppa nazionale è unica per Paese, si utilizza la stessa notazione della Lega (Nazione, Divisione) identificandola con il campo Divisione posto a 0.

Diagramma E-R ristrutturato / Implementazione dello schema logico



Schema relazionale e vincoli di integrità referenziale

Gli attributi con l'asterisco possono assumere valori nulli.

Dopo ogni entità vengono riportati i vincoli di integrità referenziale.

Squadra(Nome, Fondazione*, Stadio, Città, Allenatore*)

- Squadra.Stadio, Squadra.Città → Stadio.Nome, Stadio.Città
- Squadra.Allenatore → Allenatore.CF

Stadio(Nome, Città, Capienza)

Articolo(ID, Data, Testo, Autore, Squadra)

- Articolo.Squadra → Squadra.Nome

Calciatore(CF, Nome, Cognome, DataNascita, Nazionalità, Numero, Ruolo, Cartellino, Squadra*, DataInizio*, DataFine*)

- Calciatore.Squadra → Squadra.Nome

Statistica(DataInizio, Calciatore, DataFine, Gol, Assist, Gialli, Rossi, GolSubiti*)

- Statistica.Calciatore → Calciatore.CF

Trasferimento(Calciatore, Squadra, Veridicità, Offerta)

- Trasferimento.Calciatore → Calciatore.CF
- Trasferimento.Squadra → Squadra.Nome

Allenatore(CF, Nome, Cognome, DataNascita, Nazionalità)

Partita(Data, SquadraCasa, SquadraOspite, PunteggioCasa, PunteggioOspite, Zona, Divisione)

- Partita.SquadraCasa → Squadra.Nome
- Partita.SquadraOspite → Squadra.Nome
- Partita.Zona, Partita.Divisione → Competizione.Zona, Competizione.Divisione

Arbitro(CF, Nome, Cognome, DataNascita, Nazionalità, LivelloLicenza)

Arbitraggio(Arbitro, DataPartita, SquadraCasa, Ruolo)

- Arbitraggio.Arbitro → Arbitro.CF
- Arbitraggio.DataPartita, Arbitraggio.SquadraCasa → Partita.Data, Partita.SquadraCasa

Competizione(Zona, Divisione, Tipologia, InizioStagione, FineStagione)

Definizione delle Query e degli indici associati

Per mostrare il funzionamento delle query sono fornite un numero di tuple strettamente necessarie per l'esecuzione di esse, tenendo conto che è possibile aggiungere dati in ogni momento.

1. **Visualizzare la data, e i risultati di tutte le partite di una determinata stagione (ad esempio della Serie A).**

```
SELECT p.Data, p.SquadraCasa AS Casa, p.PunteggioCasa, p.PunteggioOspite,
p.SquadraOspite AS Ospite
FROM Competizione AS c JOIN Partita AS p
    on (c.Zona, c.Divisione) = (p.Zona, p.Divisione)
WHERE (EXTRACT(YEAR FROM p.Data) = 2021
    and TO_CHAR(p.Data, 'MM DD') >= TO_CHAR(c.InizioStagione, 'MM DD')
    or EXTRACT(YEAR FROM p.Data) = 2022
    and TO_CHAR(p.Data, 'MM DD') <= TO_CHAR(c.FineStagione, 'MM DD'))
    and c.Zona = 'Italia' and c.Divisione = 1
ORDER BY p.Data
```

	data date	casa character varying (30)	punteggiocasa smallint	punteggioospite smallint	ospite character varying (30)
1	2021-08-21	Inter	4	0	Genoa
2	2021-08-21	Verona	2	3	Sassuolo
3	2021-08-21	Empoli	1	3	Lazio
4	2021-08-21	Torino	1	2	Atalanta
5	2021-08-22	Bologna	3	2	Salernitana
6	2021-08-22	Udinese	2	2	Juventus
7	2021-08-22	Napoli	2	0	Venezia

2. Visualizzare la classifica attuale di una determinata competizione (ad esempio la Serie A).

-- Ottiene tutte le partite di una determinata stagione e lega mostrandone la data e i risultati.

```
DROP VIEW IF EXISTS Stagione2122 CASCADE;
```

```
CREATE VIEW Stagione2122 AS
```

```
SELECT p.Data, p.SquadraCasa, p.PunteggioCasa, p.PunteggioOspite,  
p.SquadraOspite
```

```
FROM Competizione AS c join Partita AS p ON (c.Zona, c.Divisione) = (p.Zona,  
p.Divisione)
```

```
WHERE (EXTRACT(YEAR FROM p.Data) = 2021 and TO_CHAR(p.Data, 'MM DD') >=
```

```
TO_CHAR(c.InizioStagione, 'MM DD'))
```

```
or EXTRACT(YEAR FROM p.Data) = 2022 and TO_CHAR(p.Data, 'MM DD') <=
```

```
TO_CHAR(c.FineStagione, 'MM DD'))
```

```
and c.Zona = 'Italia' and c.Divisione = 1
```

```
ORDER BY p.Data
```

```
;
```

-- Calcola le vittorie in casa e le somma alle vittorie in trasferta.

```
CREATE VIEW Vittorie AS
```

```
SELECT vc.Squadra,
```

```
(COALESCE(VittorieCasa, 0)+COALESCE(VittorieOspite, 0)) AS Vittorie
```

```
FROM (
```

```
SELECT SquadraCasa AS Squadra, COUNT(*) AS VittorieCasa
```

```
FROM Stagione2122
```

```
WHERE PunteggioCasa > PunteggioOspite
```

```
GROUP BY SquadraCasa
```

```
) AS vc FULL JOIN (
```

```
SELECT SquadraOspite AS Squadra, COUNT(*) AS VittorieOspite
```

```
FROM Stagione2122
```

```
WHERE PunteggioCasa < PunteggioOspite
```

```
GROUP BY SquadraOspite
```

```
) AS vo ON vc.Squadra = vo.Squadra
```

```
ORDER BY Vittorie DESC
```

```
;
```

```
-- Calcola i pareggi in casa e li somma ai pareggi in trasferta.
CREATE VIEW Pareggi AS
SELECT pc.Squadra,
       (COALESCE(PareggiCasa, 0)+COALESCE(PareggiOspite, 0)) AS Pareggi
FROM (
  SELECT SquadraCasa AS Squadra, COUNT(*) AS PareggiCasa
  FROM Stagione2122
  WHERE PunteggioCasa = PunteggioOspite
  GROUP BY SquadraCasa
) AS pc FULL JOIN (
  SELECT SquadraOspite AS Squadra, COUNT(*) AS PareggiOspite
  FROM Stagione2122
  WHERE PunteggioCasa = PunteggioOspite
  GROUP BY SquadraOspite
) AS po ON pc.Squadra=po.Squadra
ORDER BY Pareggi DESC
;

SELECT v.Squadra,
       (COALESCE(Vittorie, 0)*3 + COALESCE(Pareggi, 0)) AS Punti,
       COALESCE(Vittorie, 0) AS Vittorie,
       COALESCE(Pareggi, 0) AS Pareggi,
       ((SELECT COUNT(*) FROM Stagione2122)/
        (SELECT COUNT(DISTINCT(SquadraCasa)) FROM Stagione2122)*2
        -COALESCE(Vittorie, 0)-COALESCE(Pareggi, 0)) AS Sconfitte
FROM Vittorie AS v FULL JOIN Pareggi AS p
  ON v.Squadra = p.Squadra
ORDER BY Punti DESC
```

	squadra character varying (30)	punti bigint	vittorie bigint	pareggi bigint	sconfitte bigint
1	Milan	86	26	8	4
2	Inter	84	25	9	4
3	Napoli	79	24	7	7
4	Juventus	70	20	10	8
5	Lazio	64	18	10	10
6	Roma	63	18	9	11
7	Fiorentina	62	19	5	14

3. Visualizzare i calciatori che hanno almeno N trasferimenti “Possibili” (in questo caso 3).

```
SELECT t.Calciatore, c.Nome, c.Cognome, COUNT(*) AS Numero
FROM Trasferimento AS t JOIN Calciatore AS c ON t.Calciatore = c.CF
WHERE t.Veridicita = 'POSSIBILE'
GROUP BY t.Calciatore, c.Nome, c.Cognome
HAVING COUNT(*) >= 3
```

	calciatore character (16)	nome character varying (20)	cognome character varying (20)	numero bigint
1	GARRID19810901ME	Gyarik	Riddlesden	4
2	NIGSUD19980719FR	Nigel	Suddick	3
3	PENASK19850131PO	Pen	Askin	4

4. Visualizzare l'elenco delle squadre vincitrici di una determinata coppa ordinate per anno (ad esempio della Coppa Italia).

```
SELECT (case
    when p.PunteggioCasa > p.punteggioospite then p.SquadraCasa
    when p.PunteggioCasa < p.punteggioospite then p.SquadraOspite
    end) AS Vincitore,
    (TO_CHAR(p.Data, 'YYYY')) AS Anno
FROM Competizione AS c JOIN Partita AS p
    ON (c.Zona, c.Divisione) = (p.Zona, p.Divisione)
WHERE c.Zona = 'Italia' and c.Divisione = 0
ORDER BY (TO_CHAR(p.Data, 'MM')) DESC,
    (TO_CHAR(p.Data, 'YYYY')) DESC,
    (TO_CHAR(p.Data, 'DD')) DESC
LIMIT (
SELECT COUNT(DISTINCT((TO_CHAR(p.Data, 'YYYY'))))
FROM Competizione AS c JOIN Partita AS p
    ON (c.Zona, c.Divisione) = (p.Zona, p.Divisione)
WHERE c.Zona = 'Italia' and c.Divisione = 0
)
```

	vincitore character varying	anno text
1	Inter	2022
2	Juventus	2021
3	Napoli	2020

5. Visualizzare Nome, Cognome e squadra allenata degli allenatori che allenano una squadra che ha almeno un giocatore con N gol in tutta la carriera (in questo caso 50).

```
SELECT DISTINCT a.Nome, a.Cognome, s.Nome as Allena
FROM (Allenatore AS a JOIN Squadra AS s ON a.CF = s.Allenatore)
    JOIN Calciatore AS c ON s.Nome = c.Squadra
WHERE c.CF in (SELECT c.CF
    FROM Statistica AS s JOIN Calciatore AS c ON s.Calciatore = c.CF
    GROUP BY c.CF
    HAVING SUM(s.Gol) > 50)
```

	nome character varying	cognome character	allena character varying (30)
1	Simone	Inzaghi	Inter
2	Massimiliano	Allegri	Juventus

6. Visualizzare gli allenatori delle squadre che hanno una media gol casalinga superiore al doppio (o un altro moltiplicatore) della media gol complessiva casalinga.

```
SELECT A.Nome, A.Cognome, S.Nome
FROM Allenatore AS a JOIN Squadra AS s ON a.CF = s.Allenatore
WHERE s.Nome IN (SELECT p.SquadraCasa
                  FROM Partita AS p
                  WHERE p.PunteggioCasa > p.PunteggioOspite
                  GROUP BY p.SquadraCasa
                  HAVING AVG(p.PunteggioCasa) > 2*(SELECT AVG(p.PunteggioCasa)
                                                    FROM Partita AS p))
```

	nome character varying (20)	cognome character varying (20)	nome character varying (30)
1	Gian Piero	Gasperini	Atalanta
2	Aurelio	Andreazzoli	Empoli
3	Gabriele	Cioffi	Udinese
4	Fabio	Caserta	Benevento
5	Roberto	Venturato	Spal

Indici

Le partite hanno dei dati che vengono utilizzati molto spesso in lettura perché vengono visualizzati molto frequentemente (per per la creazione delle classifiche o comunque per conoscere l'esito della partita (vittoria, pareggio, sconfitta)).

Quindi si decide, ipotizzando uno sviluppo su larga scala dei punteggi, di indicizzare gli attributi PunteggioCasa e PunteggioOspite nell'indice "Punteggi".

```
CREATE INDEX punteggi ON Partita(PunteggioCasa, PunteggioOspite)
```

Applicazione Software

L'applicazione software sviluppata consiste di un file C++ che fa uso della libreria *libpq* per mostrare il funzionamento e il risultato delle query sopra descritte.

Per la compilazione, il file `main.cpp` necessita di librerie e header forniti da PostgreSQL. Questi sono già presenti nella cartella `dependencies` e sono essenziali per interfacciarsi al database.

Prima di compilare è però necessario modificare i parametri di accesso al database definiti all'inizio del codice. Una volta configurati, la compilazione potrà avvenire mediante il seguente comando (da eseguire nella cartella del sorgente)

```
g++ main.cpp -L dependencies\lib -lpq -o program
```

Per l'esecuzione invece basterà avviare l'eseguibile `program.exe` generato.

Il programma consiste in un semplice menù che dà possibilità all'utente di scegliere quale delle query eseguire selezionando il numero desiderato. E' possibile inoltre, per alcune query, inserire un parametro direttamente attraverso il programma per adattare l'interrogazione al database in base alle esigenze.

Funzioni

```
PGconn* connect(const char* host, const char* user, const char* db, const char* psw, const int port)
```

Consente al programma di interfacciarsi al database utilizzando i parametri definiti all'inizio del codice.

```
void initQueries(string* q) {
```

Inizializza l'elenco di query in un array di stringhe.

```
void printMenu()
```

Stampa il menù iniziale con le descrizioni delle query.

```
void checkResults(PGresult* res, const PGconn* conn)
```

Controlla lo stato del risultato della query restituito, se ci sono errori ferma il programma e mostra un messaggio d'errore.

```
PGresult* params(PGconn* conn, string queries[], int nq)
```

Consente all'utente di inserire parametri a scelta per adattare l'interrogazione al database.

```
void print(PGresult* res)
```

Stampa il risultato della query sotto forma di tabella scorrendo tutti i campi restituiti da PQexecPrepared.

```
int main()
```

Il main() gestisce gli input dell'utente e attraverso uno switch decide quale query stampare a schermo.