

## 1. ABSTRACT

La Scuola di Musica Il Pentagramma è un'associazione culturale senza fini di lucro che dal 1982 ha lo scopo di promuovere, diffondere e sviluppare la conoscenza, l'apprendimento ed il perfezionamento della musica moderna e classica in tutte le sue tecniche strumentali e canore.

L'associazione organizza corsi, senza limiti di età, per l'apprendimento di tecniche strumentali o vocali, di registrazione, composizione e di tutto ciò che sia collegato alla musica. Inoltre, promuove la creazione di gruppi musicali e organizza eventi e concerti in proprio o in accordo con locali o enti privati e locali, in modo da diffondere la cultura musicale sia ai suoi associati sia a chiunque voglia partecipare, e quindi contribuire alla diffusione di tale cultura.

I corsi, suddivisi tra corsi individuali, musica d'insieme, teatro e corsi propedeutici, sono tenuti da docenti diplomati al conservatorio e alcuni sono svolti anche dagli stessi dirigenti della scuola. Hanno durata di 30,45 o 60 minuti a seconda della tipologia del corso. I corsi individuali offrono inoltre dei momenti di controllo, in cui gli allievi mostrano i loro progressi ad una giuria interna, e per chi ambisce ad un percorso professionale l'associazione prevede una suddivisione in livelli, accessibili tramite il superamento di un apposito esame che ne certifica le qualità acquisite nel percorso.

L'associazione vuole costruire una base di dati affinché possa organizzare al meglio le attività dell'associazione, oltre che mantenere uno storico delle attività passate e dei membri che ne hanno fatto parte nel corso del tempo.

## 2. ANALISI DEI REQUISITI

Si vuole rappresentare, attraverso una base di dati, le attività e gli eventi organizzati dall'associazione Il Pentagramma.

Ogni **persona** collegata all'associazione è caratterizzata da codice fiscale, nome, cognome, data di nascita, numero di telefono e indirizzo e-mail. Una persona può essere un associato, un docente, un dirigente oppure uno studente.

Di un **associato** è rilevante il codice della tessera a lui associata, la data d'iscrizione all'associazione e la data di scadenza dell'iscrizione.

Per un **docente** si vuole sapere la specializzazione in cui si è diplomato, la data di conseguimento del diploma, la data d'inizio dell'attività di insegnamento e, nel caso il docente non insegni più nella scuola, la data di fine. Inoltre, in caso di insegnamento in più corsi, è rilevante conoscere, assieme al corso, la data di inizio e l'eventuale fine di docenza in quel corso. Per un **dirigente**, che è anche un docente, è rilevante conoscere l'unico ruolo che copre all'interno dell'associazione, la data di inizio in quel ruolo e, se presente, la data di fine in quel ruolo (che può combaciare con la data di fine insegnamento).

Uno **studente** è caratterizzato dalla tipologia (amatoriale o professionale) di percorso che segue e, in caso di percorso professionale, il livello a cui lo studente stesso si trova. Inoltre, è d'interesse sapere la data d'inizio delle lezioni e l'eventuale data di fine. Infine, se lo studente risulta iscritto all'associazione ma frequenta dei corsi diversi rispetto ai corsi con cui ha cominciato, è rilevante conoscere la data d'inizio e di eventuale fine di frequentazione di ciascun corso frequentato.

Un **corso** è caratterizzato dal nome, dalla durata (in minuti) di una lezione e dal numero d'incontri in un mese (che è un valore compreso tra 1 e 4). Un corso può essere individuale o collettivo. Per i corsi collettivi è d'interesse avere anche una descrizione breve del corso stesso. Tale attributo non è rilevante per i corsi individuali in quanto già esplicativi col nome stesso.

Una **sede** è caratterizzata da nome, città e via nella quale si trova. Una sede può essere usata come sede per le lezioni o come sede per gli eventi. Per una **sede per eventi** è rilevante conoscere anche il numero massimo di posti disponibili.

Un **evento** è caratterizzato da nome, genere, sede dove viene svolto e numero di spettatori, essendo aperti al pubblico. Se l'evento non è direttamente organizzato dall'associazione, oppure l'associazione non è l'unica organizzatrice dell'evento, allora è d'interesse anche conoscere il nome degli organizzatori o co-organizzatori.

Inoltre, è d'interesse conoscere i brani svolti durante l'evento, oltre agli studenti che li eseguono, se partecipanti. Negli eventi sono compresi anche i **saggi**, organizzati solo dall'associazione e dove partecipano solo gli studenti della scuola.

Un **brano** è caratterizzato da nome, artista e genere. Ai fini delle attività dell'associazione, non sono d'interesse altre informazioni relative ai brani e ai relativi artisti.

Un **esame** è caratterizzato dallo studente che lo effettua, la data in cui è stato fatto, il livello per cui è valido l'esame, l'esito dell'esame (passato o non passato) e la commissione, composta da al più tre docenti. Per l'associazione non è rilevante conoscere i docenti esterni presenti nelle commissioni, sono d'interesse solo i docenti interni. Per una **audizione** sono d'interesse gli stessi campi meno il livello, essendo la stessa da effettuare anche per gli allievi che seguono un percorso amatoriale, e l'esito.

## 2.1. GLOSSARIO DEI TERMINI

TERMINE	DESCRIZIONE	SINONIMI	COLLEGAMENTI
Persona	Persona che interagisce con l'associazione	Associato, Docente, Dirigente, Studente	∅
Associato	Persona iscritta all'associazione	Persona	∅
Docente	Docente di musica dell'associazione	Persona	Dirigente
Dirigente	Persona incaricata agli aspetti organizzativi dell'associazione e garante degli obiettivi programmatici	Persona, Docente	∅
Studente	Persona che beneficia delle lezioni musicali	Persona	∅
Corso	Servizio offerto dall'associazione al quale, iscrivendosi, si diventa studenti	∅	∅
Sede	Luogo per lo svolgimento dei servizi offerti	∅	∅
Evento	Spettacolo rivolto ad un pubblico. Può essere organizzato dall'associazione stessa o in collaborazione con enti esterni.	∅	∅
Saggio	Evento organizzato esclusivamente dall'associazione che permette agli studenti di mettere alla prova le conoscenze acquisite nei corsi scelti	Evento	∅
Brano	Spartito studiato ed eseguito da uno studente	∅	∅
Esame	Test che lo studente svolge per il passaggio ad un livello superiore	∅	∅
Audizione	Momento di valutazione obbligatorio per gli studenti dell'associazione	∅	Esame

## 2.2. OPERAZIONI TIPICHE

Le seguenti operazioni vengono generalmente effettuate a cadenza mensile:

1) Inserimento e lettura studenti nei corsi scelti	100 al mese
2) Controllo delle iscrizioni relative ai soci	30 al mese
3) Inserimento di eventi	10 al mese

Mentre le seguenti operazioni vengono effettuate durante l'anno:

4) Archiviazione dell'anno accademico con cadenza trimestrale	5 all'anno
5) Controllo dei docenti e dei dirigenti	60 all'anno
6) Controllo delle sedi	80 all'anno

7) Controllo dei corsi	350 all'anno
8) Controllo ed eventuale scrittura del numero di spettatori negli eventi	50 all'anno

### 3. PROGETTAZIONE CONCETTUALE

#### 3.1. LISTA DELLE ENTITA'

<b>Persona:</b> colui/colei che interagisce con l'associazione		
<u>CF</u>	char(16) primary key	Il Codice Fiscale della persona
Nome	varchar(30)	Il nome della persona
Cognome	varchar(30)	Il cognome della persona
DataNascita	date	La data di nascita della persona
Contatto		I contatti della persona, si divide in <b>Telefono</b> e <b>Email</b> .
Telefono	char(10)	Il numero di telefono della persona.
Email	varchar(50)	L'indirizzo mail della persona.

**Persona** si specializza in **Associato**, **Studente** e **Docente**.

<b>Associato:</b> un iscritto all'associazione		
<u>CodiceTessera</u>	char(10) primary key	Il codice della tessera associata all'associato
DataIscrizione	date	La data in cui l'associato risulta iscritto
DataScadenza	date	La data di scadenza dell'iscrizione

<b>Studente:</b> colui che frequenta i corsi organizzati dall'associazione		
Tipologia	char(1)	La tipologia del corso frequentato, si distingue tra amatoriale (A) o professionale (P)
Livello (0, 1)	char(2)	Il livello in cui si posiziona lo studente
PeriodoLezioni		Il periodo in cui lo studente frequenta o ha frequentato i corsi dell'associazione, si divide in <b>InizioLezioni</b> e <b>FineLezioni</b> .
InizioLezioni	date	La data d'inizio di frequenza dei corsi.
FineLezioni (0,1)	date	La data di fine di frequenza dei corsi.

<b>Docente:</b> una persona che insegna nei corsi dell'associazione		
Diploma	varchar(30)	Il diploma in possesso
DataDiploma	date	La data di acquisizione del diploma
PeriodoDocenza		Periodo in cui il docente insegna o ha insegnato per conto dell'associazione, si divide in <b>InizioDocenza</b> e <b>FineDocenza</b> .
InizioDocenza	date	Data d'inizio dell'attività d'insegnamento.
FineDocenza (0,1)	date	Data di fine dell'attività d'insegnamento.

**Docente** può specializzarsi in **Dirigente**

<b>Dirigente:</b> docente con un ruolo di rilevanza nell'organizzazione dell'associazione		
Ruolo	varchar(30)	Il ruolo ricoperto all'interno dell'associazione
PeriodoDirigenza		Periodo in cui il dirigente ha coperto un determinato ruolo all'interno dell'associazione, si divide in <b>InizioDirigenza</b> e <b>FineDirigenza</b> .
InizioDirigenza	date	Data d'inizio in cui si è ricoperto il ruolo.
FineDirigenza (0,1)	date	Data di fine del ruolo ricoperto.

<b>Attività:</b> attività didattica organizzata dall'associazione		
InizioAttività	date	La data d'inizio dell'attività
FineAttività (0,1)	date	La data di fine dell'attività

<b>Corso:</b> percorso didattico offerto dall'associazione		
<u>NomeCorso</u>	varchar(30) primary key	Il nome del corso
DurataLezione	int	La durata della singola lezione, in minuti

LezioneMese	int	Il numero di lezioni mensili che il corso prevede
-------------	-----	---

**Corso** si specializza in **Individuale** e **Collecttivo**.

**Individuale:** corso che offre lezioni individuali

<b>Collettivo:</b> corso con lezioni in cui più studenti partecipano contemporaneamente		
Descrizione	varchar(50)	Breve descrizione sul corso

<b>Sede:</b> luogo dove le attività e gli eventi dell'associazione sono tenuti		
<u>NomeSede</u>	varchar(50) primary key	Il nome della sede
Locazione		Le indicazioni dove si trova una sede, si divide in <b>Citta</b> e <b>Indirizzo</b> .
Citta	varchar(30)	La città dove si trova la sede
Indirizzo	varchar(50)	L'indirizzo della sede

**Sede** si specializza in **SedeLezione** e **Sedevento**.

**SedeLezione:** luogo indicato per usufruire dei corsi dell'associazione

<b>SedeEvento:</b> luogo indicato per gli eventi in cui l'associazione è coinvolta		
MaxPosti	int	La capienza relativa alla sede

<b>Evento:</b> manifestazione in cui è coinvolta l'associazione		
<u>NomeEvento</u>	varchar(50)	Il nome dell'evento
NumSpettatori	int	Gli spettatori che hanno partecipato
GenereEvento	varchar(20)	Il genere dell'evento
Organizzazione		Chi ha organizzato l'evento, si divide in <b>Organizzatore</b> e <b>Co-Organizzatore</b> .
Organizzatore (0, 1)	varchar(50)	Elenco di enti che hanno organizzato l'evento di cui l'associazione fa da promoter
Co-Organizzatore (0, 1)	varchar(50)	Elenco di enti che hanno organizzato l'evento assieme all'associazione

**Evento** può specializzarsi in **Saggio**

**Saggio:** manifestazione organizzata dall'associazione dove si esibiscono gli allievi della stessa

**Scaletta:** elenco di brani eseguiti da vari allievi in determinati eventi

<b>Brano:</b> canzone suonata negli eventi dagli allievi		
<u>NomeBrano</u>	varchar(50) primary key	Il nome del brano
<u>Artista</u>	varchar(50) primary key	Il nome dell'artista
Genere	varchar(50)	Il genere del brano

<b>Esame:</b> verifica per il passaggio di livello di un allievo		
<u>LivelloAbilitazione</u>	char(2) primary key	Il livello a cui l'esame è indirizzato
<u>DataEsame</u>	date primary key	La data in cui l'esame è stato sostenuto

<b>Audizione:</b> verifica del progresso di un allievo		
<u>DataAudizione</u>	date primary key	La data in cui l'audizione è stata sostenuta

## 3.2. LISTA DELLE RELAZIONI

Studente-Audizione: **Accertamento**

- Uno studente può aver già partecipato ad almeno una audizione (0, N)
- Un'audizione deve essere svolta da almeno uno studente (1, N)

Docente-Audizione: **Auditore**

- Un docente deve essere presente ad almeno una audizione (1, N)
- Un'audizione deve essere presente almeno un docente (1, N)

Studente-Esame: **Esaminato**

- Uno studente può partecipare a uno o più esami (0, N)
- Ad ogni esame partecipa almeno uno studente (1, N)

Attributi di <b>Esaminato</b>		
EsitoEsame	bool	L'esito dell'esame

Docente-Esame: **Esaminatore**

- Un docente può prendere parte a uno o più esami (0, N)
- Ogni esame ha almeno un docente (1, N)

Studente-Scaletta: **Performance**

- Uno studente può essere inserito in una o più scalette (0, N)
- Una scaletta è composta da almeno uno studente (1, N)

Studente-Attività: **Apprendimento**

- Uno studente è coinvolto in almeno un'attività (1, N)
- Ogni attività ha almeno uno studente (1, N)

Docente-Attività: **Insegnamento**

- Ogni attività è tenuta da un docente (1, 1)
- Ogni docente prende parte ad almeno un'attività (1, N)

Brano-Scaletta: **BranoEseguito**

- Ogni scaletta ha almeno un brano (1, N)
- Ogni brano è presente in almeno una scaletta (1, N)

Attività-SedeLezione: **LuogoLezione**

- Un'attività viene svolta in una sede per le lezioni (1,1)
- Ogni sede per le lezioni ospita almeno un'attività (1, N)

Attività-Corso: **Erogazione**

- Un'attività è legata ad un solo corso (1, 1)
- Un corso è legato ad almeno un'attività (1, N)

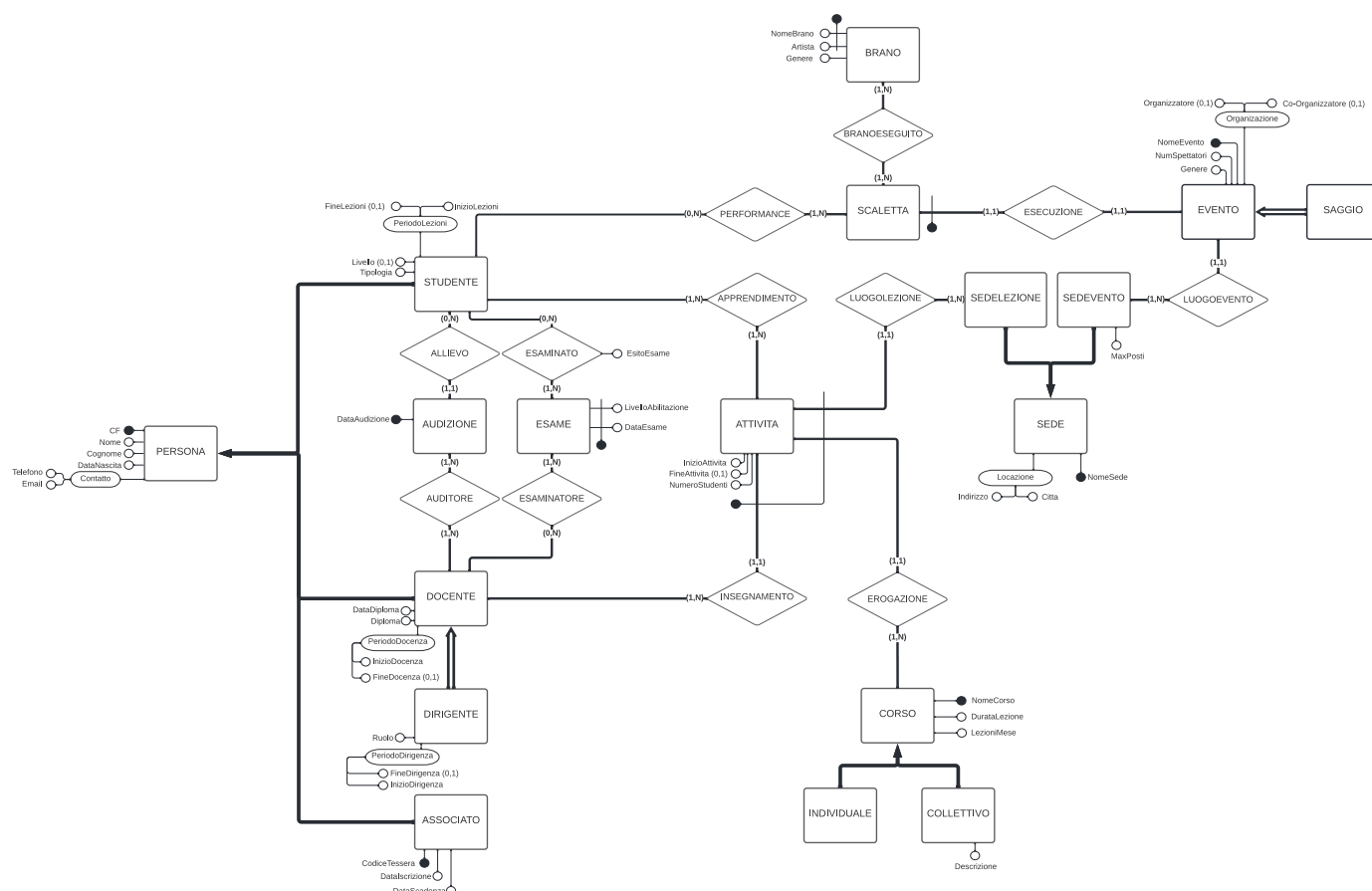
Scaletta-Evento: **Esecuzione**

- Ogni scaletta si riferisce ad un evento (1, 1)
- Ogni evento ha una scaletta (1, 1)

Evento-Sedeevento: **LuogoEvento**

- Ogni evento è tenuto in una sede per eventi (1, 1)
- Ogni sede per eventi ospita almeno un evento (1, N)

### 3.3. SCHEMA ER



### 3.4. VINCOLI NON ESPRIMIBILI NELLO SCHEMA ER

Entità o Relazioni	Vincolo
Auditore	Per ogni studente, tra gli uditori dev'essere presente il docente con cui frequenta il corso.
Docente	Per ogni docente, se ha valore non nullo, $FineDocenza \geq InizioDocenza$
Studente	Per ogni studente, se ha valore non nullo, $FineLezioni \geq InizioLezioni$
Dirigente	Per ogni dirigente, se ha valore non nullo, $FineDirigenza \geq InizioDirigenza$ Per ogni D1, D2 dirigenti distinti con ruolo in comune, $D2.InizioDirigenza \geq D1.FineDirigenza$ Per ogni dirigente, se $FineDocenza$ e $FineDirigenza$ hanno valori non nulli, $FineDirigenza \leq FineDocenza$
Associato	$DataScadenza \geq DataIscrizione$
Esaminato	Tipologia di studente: Professionale
Esaminatore	$1 \leq \text{numero di esaminatori} \leq 3$
Attività	Per ogni attività, se ha valore non nullo, $FineAttività \geq InizioAttività$
Corso	$durataLezione \leq 60$ $1 \leq lezioniMese \leq 4$
Evento	Organizzatore ha valore non nullo $\Leftrightarrow$ Co-Organizzatore ha valore nullo e Organizzatore ha valore nullo $\Leftrightarrow$ Co-Organizzatore ha valore non nullo
Saggio	Organizzatore e Co-Organizzatore hanno entrambi valori nulli

## 4. PROGETTAZIONE LOGICA

### 4.1. ANALISI DELLE RIDONDANZE

L'attributo "NumeroStudenti" nell'entità "Attività" è ridondante in quanto tale informazione può essere ricavata tramite la relazione "Apprendimento".

Tale ridondanza affligge l'operazione n°1, ovvero l'inserimento e lettura degli studenti e dei relativi corsi scelti dagli studenti stessi.

Si hanno le seguenti tabelle degli accessi, una con ridondanza assente ed una con ridondanza presente:

Tavola degli accessi in presenza di ridondanze			
Concetto	Costrutto	Accessi	Operazione
Attività	E	1	L
Attività	E	1	S
Apprendimento	R	1	S
Studente	E	1	S

Tavola degli accessi in assenza di ridondanze			
Concetto	Costrutto	Accessi	Operazione
Attività	E	1	L
Apprendimento	R	1	L
Apprendimento	R	1	S
Studente	E	1	S

Tali accessi devono essere moltiplicati per ogni studente inserito, che equivale a 100 studenti al mese.

Per gli accessi in scrittura, tale ridondanza non porta alcun beneficio evidente, dovendo aggiornare anche l'attributo in attività (per quanto sia rilevante il peso di aggiornamento dello stesso).

Si noti però come tale ridondanza sia ottimale nella fase di lettura, poiché il controllo del numero di iscritti ad una attività senza la ridondanza prevede un'unione tra Attività ed Apprendimento e, specialmente per determinati parametri, tale operazione può risultare onerosa in termini di tempo (si ricorda che la base di dati assume anche una funzione di archivio), per cui si è deciso di mantenere la ridondanza.

### 4.2. ELIMINAZIONE DELLE GENERALIZZAZIONI

Per la generalizzazione **Persona-Studente, Docente, Associato** si è scelto di sostituire la generalizzazione con più relazioni, poiché accorpare gli attributi delle entità figlie nell'entità Persona porterebbe a tuple nell'entità molto grandi e con una quantità di valori nulli non indifferente, mentre l'operazione inversa, ovvero accorpare gli attributi dell'entità Persona nelle varie entità figlie, porterebbe ad una ridondanza di tale attributi, quindi a possibili inconsistenze non previste dalla base di dati (per esempio, non si possono avere studenti che sono contemporaneamente docenti e/o associati).

Tale soluzione permette di evitare pericolose inconsistenze, senza avere un peso consistente durante la lettura dei dati, poiché le entità figlie ottengono come attributo identificativo la chiave di Persona.

Per le altre generalizzazioni si è optato per accorpare gli attributi dell'entità figlia nell'entità genitore. In particolare:

- Per la generalizzazione parziale **Docente-Dirigente**, un docente svolge anche una funzione da dirigente se e solo se almeno la coppia di attributi **RuoloDirigente-InizioDirigenza** è e non nulla, ovvero entrambi gli attributi hanno valori non nulli.
- Per la generalizzazione totale **Corso-Individuale, Collettivo**, un corso individuale si distingue da un corso collettivo tramite l'attributo **Tipologia-Corso**, che contiene la distinzione tra individuale (I) e collettivo (C). Inoltre, l'attributo **Descrizione** avrà un valore nullo se e solo se **TipologiaCorso='I'**.
- Per la generalizzazione totale **Sede-SedeLezione, SedEvento**, una sede adibita per le lezioni si distingue da una sede adibita per gli eventi tramite l'attributo **Utilizzo**, che può assumere due valori: 'L' oppure 'E'. L'attributo **MaxPosti** assume valore non nullo se e solo se **utilizzo='E'**.
- Per la generalizzazione parziale **Evento-Saggio** viene semplicemente aggiunto l'attributo **TipoEvento** per distinguere un saggio (S) da un evento normale (N), dato che l'entità Saggio non contribuiva con informazioni aggiuntive.  
Gli attributi **Organizzatore** e **Co-Organizzatore** devono avere valore nullo se e solo se **TipoEvento='S'**.

### 4.3. ATTRIBUTI MULTIPLI

Gli attributi multipli presenti tra le varie entità vengono scomposti in più attributi, non potendo rappresentare gli attributi multipli nella base di dati finale.

Entità	Attributo multiplo	Scomposizione
Persona	Contatto	Telefono, Email
Docente	PeriodoDocenza PeriodoDirigenza	InizioDocenza, FineDocenza (0,1) InizioDirigenza, FineDirigenza (0, 1)
Studente	PeriodoLezioni	InizioLezioni, FineLezioni (0,1)
Evento	Organizzazione	Organizzatore (0,1), Co-Organizzatore (0, 1)
Sede	Locazione	Indirizzo, Citta

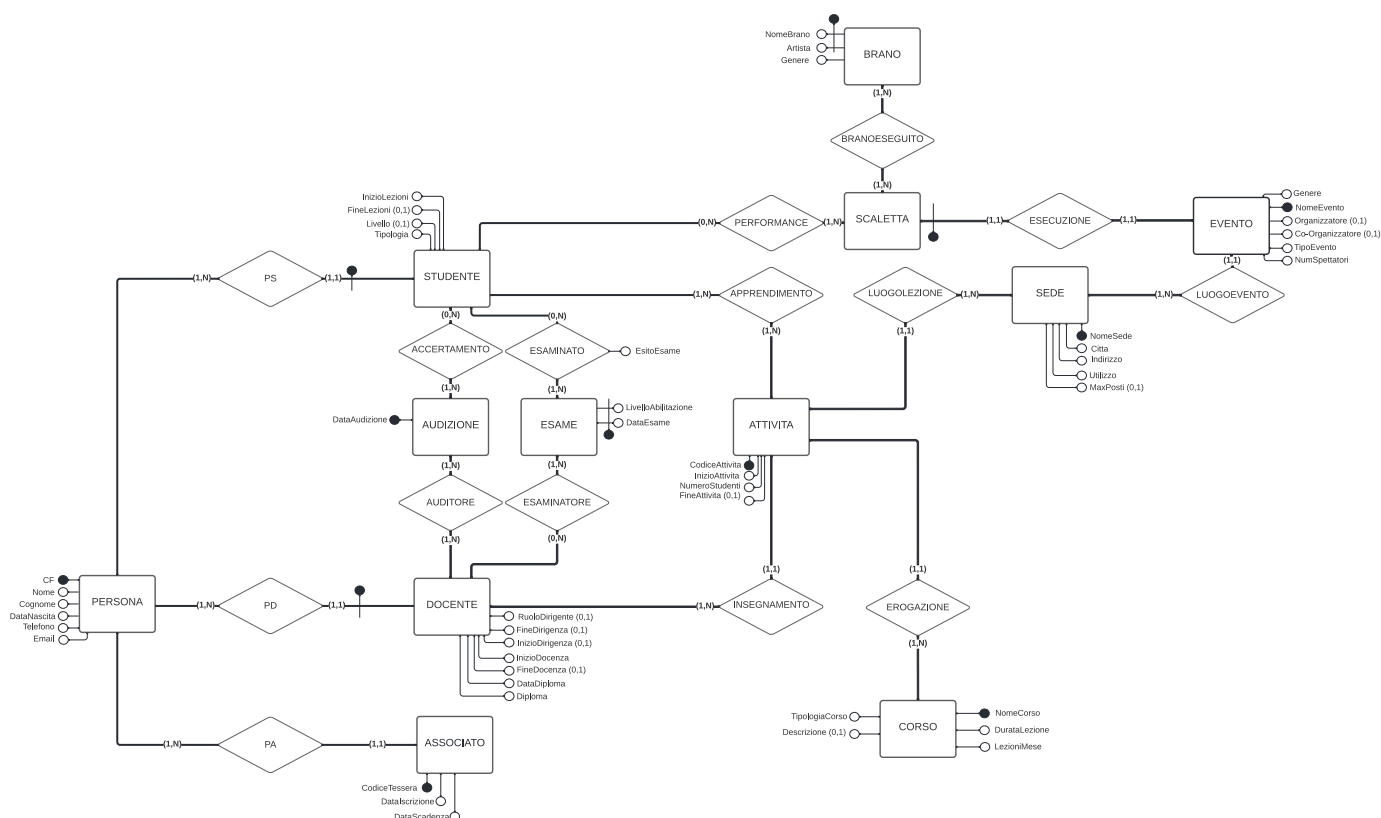
#### 4.4. PARTIZIONAMENTO / ACCORPAMENTO DI ENTITA' E ASSOCIAZIONI

!!!

#### 4.5. SCELTA DEGLI IDENTIFICATORI PRINCIPALI

Lo schema concettuale presenta nell'entità "ATTIVITA" la chiave primaria ha tre attributi esterni (NomeSede, NomeCorso, CF). Questa chiave a tre attributi non è efficiente quindi si è preferito specificare un nuovo attributo specifico per attività con lo scopo di identificare ogni attività dell'associazione univocamente attraverso un codice a otto caratteri.

#### 4.6. DIAGRAMMA ER RISTRUTTURATO



#### SCHEMA RELAZIONALE E VINCOLI DI INTEGRITA' REGERENZIALE

Persona(CF, Nome, Cognome, DataNascita, Telefono, Email)

Audizione(DataAudizione)

Esame(LivelloAbitazione, DataEsame)

Brano(NomeBrano, Artista, Genere)

Evento(NomeEvento, NumSpettatori, Genere, Organizzatore\*, Co-Organizzatore\*, TipoEvento)

Sede(NomeSede, Utilizzo, MaxPosti\*, Indirizzo, Citta)

Attivita(CodiceAttività, InizioAttività, FineAttività\*, NumStudenti)



Corso(NomeCorso, DurataLezione, LezioniMese, TipologiaCorso, Descrizione\*)

Associato(CodiceTessera, CF, DataIscrizione, DataScadenza)

- Associato.CF→Persona.CF

Studente(CF, Tipologia, Livello\*, InizioLezioni, FineLezioni\*)

- Studente.CF→Persona.CF

Docente(CF, DataDiploma, Diploma, RuoloDirigente\*, InizioDocenza, FineDocenza\*, InizioDirigenza\*, FineDirigenza\*)

- Docente.CF→Persona.CF

Accertamento(CF, DataAudizione)

- Accertamento.CF→Studente.CF
- Accertamento.DataAudizione→Audizione.DataAudizione

Auditore(CF, DataAudizione)

- Auditore.CF→Docente.CF
- Auditore.DataAudizione→Audizione.DataAudizione

Esaminatore(LivelloAbilitazione, DataEsame, CF)

- Esaminatore.LivelloAbilitazione,Esaminatore.DataEsame→Esame.LivelloAbilitazione,Esame.DataEsame
- Esaminatore.CF→Docente.CF

Esaminato(LivelloAbilitazione, DataEsame, CF, EsitoEsame)

- Esaminato.LivelloAbilitazione,Esaminato.DataEsame→Esame.LivelloAbilitazione,Esame.DataEsame
- Esaminato.CF→Studente.CF

Performance(CF, NomeEvento)

- Performance.CF→Studente.CF
- Performance.NomeEvento→Scaletta.NomeEvento

Scaletta(NomeEvento)

- Scaletta.NomeEvento→Evento.NomeEvento

BranoEseguito(NomeEvento, NomeBrano, Artista)

- BranoEseguito.NomeEvento→Scaletta.NomeEvento
- BranoEseguito.NomeBrano,BranoEseguito.NomeArtista→Brano.NomeBrano,Brano.NomeArtista

Apprendimento(CF, CodiceAttivita)

- Apprendimento.CF→Studente.CF
- Apprendimento.CodiceAttivita→Attivita.CodiceAttivita

## 5. QUERY

- 1) Trovare i nominativi degli studenti che attualmente frequentano il corso di Pianoforte.

```
SELECT "Nome", "Cognome"
FROM "Persona"
JOIN "Studente"
ON "Persona"."CF"="Studente"."CF"
JOIN "Apprendimento"
ON "Studente"."CF"="Apprendimento"."CF"
JOIN "Attivita"
ON "Apprendimento"."CodiceAttivita"="Attivita"."CodiceAttivita"
WHERE "FineAttivita" IS NULL
AND "Corso"='Pianoforte'
```

	Nome character varying (30)	Cognome character varying (30)
1	Mona	Tyler

- 2) Fornire le generalità (nome, cognome e numero di telefono) degli associati iscritti dal 2018.

```
SELECT "Nome", "Cognome", "Telefono"
FROM "Persona", "Associato"
WHERE "Persona"."CF"="Associato"."CF"
AND "Associato"."DataIscrizione">='2019-01-01'
```

	Nome character varying (30)	Cognome character varying (30)	Telefono character (10)
1	James	Osborne	6578814929
2	Colby	Dickerson	7230818818
3	Emmanuel	Baird	1039152146
4	Rhona	Walters	0751869251
5	Anthony	Riggs	7718564225
6	Stacey	Prince	8761405273
7	Ray	Rogers	3216725091
8	Elmo	Powell	6680437665
9	Reagan	Dennis	4064291628
10	Acton	Meyer	6622738898
11	Mark	Hogan	1555573862
12	Zephania	Stark	1428922188
13	Vincent	Parsons	3885154261
14	Alyssa	Bass	8122682067

- 3) Selezionare i docenti (basta il codice fiscale) che sono stati esaminatori almeno una volta.

```
SELECT "Nome", "Cognome"
FROM "Docente", "Persona"
WHERE "Docente"."CF"="Persona"."CF"
AND "Docente"."CF"
IN
(
SELECT "CF"
FROM "Esaminatore"
GROUP BY "CF"
)
```

	Nome character varying (30)	Cognome character varying (30)
1	Gregory	Mann
2	Adrian	Solis
3	Brenna	Saunders
4	Kareem	Ford
5	Phelan	Kelly
6	Vance	Benjamin
7	Duncan	Hatfield
8	Lenore	Jensen
9	Summer	Workman
10	Keegan	Bailey
11	Raven	Noble
12	Chadwick	Pugh
13	Caldwell	Olson
14	Wing	Henson

- 4) Elencare i corsi che siano stati tenuti più di una volta

```
SELECT "Corso"
FROM "Docente", "Attivita"
WHERE "Docente"."CF"="Attivita"."Docente"
GROUP BY "Corso"
HAVING COUNT(*)>=2
```

	Corso character varying (30)
1	Batteria
2	Pianoforte
3	Violino
4	Chitarra Elettrica
5	Basso
6	Cornamusa

- 5) Mostrare il numero di scalette in cui sia stato eseguito almeno una volta un brano dal genere "Metal".

```
SELECT "GenereBrano", COUNT(*)
FROM "Scaletta"
JOIN "BranoEseguito"
ON "Scaletta"."NomeEvento"="BranoEseguito"."NomeEvento"
JOIN "Brano"
ON "BranoEseguito"."NomeBrano"="Brano"."NomeBrano"
AND "BranoEseguito"."Artista"="Brano"."Artista"
WHERE "GenereBrano"='Metal'
GROUP BY "GenereBrano"
```

	GenereBrano character varying (20)	count bigint
1	Metal	20

- 6) Mostrare Indirizzo e Citta della sede con il maggior numero di spettatori in un singolo evento.

```
SELECT "Indirizzo", "Citta", "NumSpettatori"
FROM "Evento"
JOIN "Sede"
ON "Evento"."LuogoEvento"="Sede"."NomeSede"
WHERE "NumSpettatori"=
(
SELECT MAX("NumSpettatori")
FROM "Evento"
)
```

	Indirizzo character varying (50)	Citta character varying (30)	NumSpettatori integer
1	Riviera S. Trentin, 3	Mira	1000

- 7) Selezionare i docenti (è sufficiente CF) che insegnano dal 2010.

```
SELECT "CF"
FROM "Docente"
WHERE "InizioDocenza">>='2010-01-01'
AND "FineDocenza" IS NULL
```

	CF [PK] character (16)
1	OIBPBN38G17K415C
2	UHEPKR56Y58E429C
3	FCJFSE25C20V588J
4	FLQMSL06X75D245E
5	FUJYS002X60D552V
6	VPPMBC83L68W079L
7	JQDBHY47L85N710L
8	CNWWYL75H80E135T

- 8) Mostrare il nome di tutti gli eventi che NON sono saggi.

```
SELECT "NomeEvento"
FROM "Evento"
WHERE "TipoEvento"='N'
```

	NomeEvento [PK] character varying (50)
1	Note Future 2020
2	Note Future 2015
3	Pentagramma Rock 2018
4	Pentagramma Metal Fest 2022
5	Pop Fest 2022

- 9) Contare tutte le occorrenze in cui la sede sia stata usata per evento (LuogoEvento).

```
SELECT "LuogoEvento", COUNT(*)  
FROM "Evento"  
GROUP BY "LuogoEvento"
```

	LuogoEvento character varying (50)	count bigint
1	Teatro Dario Fo Camponogara	4
2	Teatro Villa dei Leoni	2
3	Villa Valier	2

- 10) Trovare per ogni brano (eseguito da qualunque artista) il numero di esecuzioni.

```
SELECT "NomeBrano", COUNT(*)  
AS "NumEsecuzioni"  
FROM "BranoEseguito"  
GROUP BY "NomeBrano"  
ORDER BY "NomeBrano"
```

	NomeBrano character varying (50)	NumEsecuzioni bigint
1	Back in Black	3
2	Brividi	2
3	Creeping Death	1
4	Disposable Heroes	2
5	Enter Sandman	2
6	Faithless	2
7	Fur Elise	3
8	Gold Digger	2
9	Holy Diver	3
10	I wanna be your slave	1
11	Inno alla Gioia	2
12	L'estate addosso	1
13	La danza della fata confetto	3
14	Paranoid	4
15	Primavera	3
16	Rumor has it	1
17	Someone Like You	2
18	T.N.T.	2
19	Technical Difficulties	3
20	Toccata e Fuga	4
21	Trust	1

## 6. INDICE

Essendo questo database anche a scopo di archiviazione è necessario prendere in considerazione l'inserimento di indici. In particolare, la relazione "Attivita" contiene dati che vengono inseriti con una frequenza elevata vista l'operazione tipica mensile n. 1 (Inserimento e lettura studenti nei corsi scelti), per la funzionalità di storico la ricerca di determinate attività potrebbe risultare onerosa.

Da queste premesse ricaviamo l'indicizzazione della tabella "Attivita" sull'attributo "CodiceAttivita":

```
CREATE INDEX idx_Activita ON "Attivita"("CodiceAttivita");
```

## 7. CODICE C++