

PROGETTO DATABASE: PALESTRA

Battelli Massimiliano

Indice

1. Requisiti
2. Glossario dei termini
3. Tabella delle operazioni
4. Tabella dei volumi
5. Valutazione dei costi
6. Schema relazionale
7. Query SQL

1. Requisiti

Database di un applicativo software per la gestione di una palestra.

La palestra offre vari servizi ai propri clienti abbonati tra cui:

- Prenotazione visite mediche specialistiche;
- Iscrizione a corsi di fitness;
- Prenotazione di sessioni di personal training coi vari istruttori di sala pesi.

Il DB registrerà i dati anagrafici di clienti, dottori, istruttori, informazioni sui corsi, visite, allenamenti.

Gli istruttori si dividono in: Esterni e Interni, gli interni sono dipendenti della palestra che offrono allenamenti e possono tenere i corsi. Gli esterni sono figure non dipendenti della palestra che possono solo tenere corsi.

I dati anagrafici di clienti, istruttori, e dottori, rimangono nel database a tempo indeterminato

I dati sui corsi attivati(lezioni settimanali, istruttore a cui appartenente) rimangono memorizzati per un anno poi automaticamente eliminati.

I dati su allenamenti e visite vengono prenotate solitamente con un anticipo massimo di 1 mese, poi automaticamente eliminate.

2. Glossario dei termini

Termine	Descrizione	Sinonimi	Collegamenti
Cliente	Persona iscritta alla palestra	-	Visita, Allenamento, Corso_Activo
Istruttore	Istruttore di fitness. Interno: dipendente della palestra Esterno: non dipendente della palestra	Interno: personal trainer, dipendente -	Interno: Allenamento, Corso_Activo Esterno: Corso_Activo
Corso_Activo	Corso fitness attivato nell'anno corrente	-	Cliente, Istruttore, Corso, Lezione
Dottore	Persona specializzata in campo medico che offre i propri servizi alla palestra.	Specialista, Medico	Visita
Allenamento	Appuntamento con un istruttore per un allenamento	Personal, Sessione	Cliente, Istruttore interno
Visita	Appuntamento con un dottore per una visita medica		Cliente, Dottore
Corso	Corso che la palestra è in grado di produrre		Corso_Activo
Lezione	Lezione settimanale di un corso che è stato attivato		Corso_Activo

3. Tabella delle Operazioni

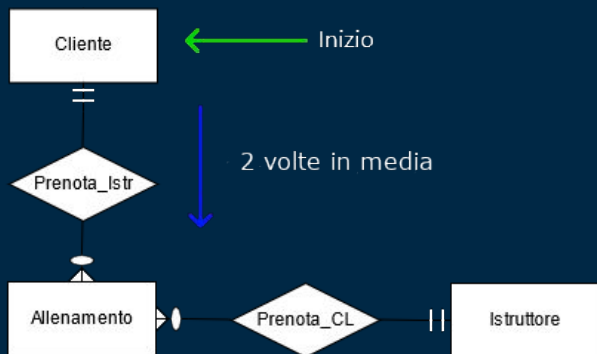
N°	Operazione	Frequenza media	Tipo
1	Inserimento cliente	3/giorno	I
2	Inserimento allenamento	13/giorno	I
3	Inserimento visita	5/giorno	I
4	Lettura dati di un cliente, e gli allenamenti prenotati	10/giorno	B
5	Lettura lezioni di un corso attivo	1/settimana	B
6	Mostra i dati dei clienti iscritti a un corso	1/mese	B
7	Mostra quanti iscritti ci sono per ciascun corso, il nome e la sua descrizione	1/settimana	B
8	Mostra tutte le visite prenotate e i dati dei corrispettivi clienti e dottori	1/settimana	B
9	Mostra le lezioni presenti in determinati giorni	1/settimana	B

4. Tabella dei Volumi

Concetto	Costrutto	Volume
Cliente	E	1000
Dottore	E	5
Istruttore	E	10
Allenamento	E	400
Visita	E	150
Corso_Attivo	E	5
Lezione	E	15
Corso	E	10

Concetto	Costrutto	Volume
Iscrizione	R	500
Responsabilità	R	5
Attivazione	R	5
Svolgimento	R	15

5. Valutazione dei costi

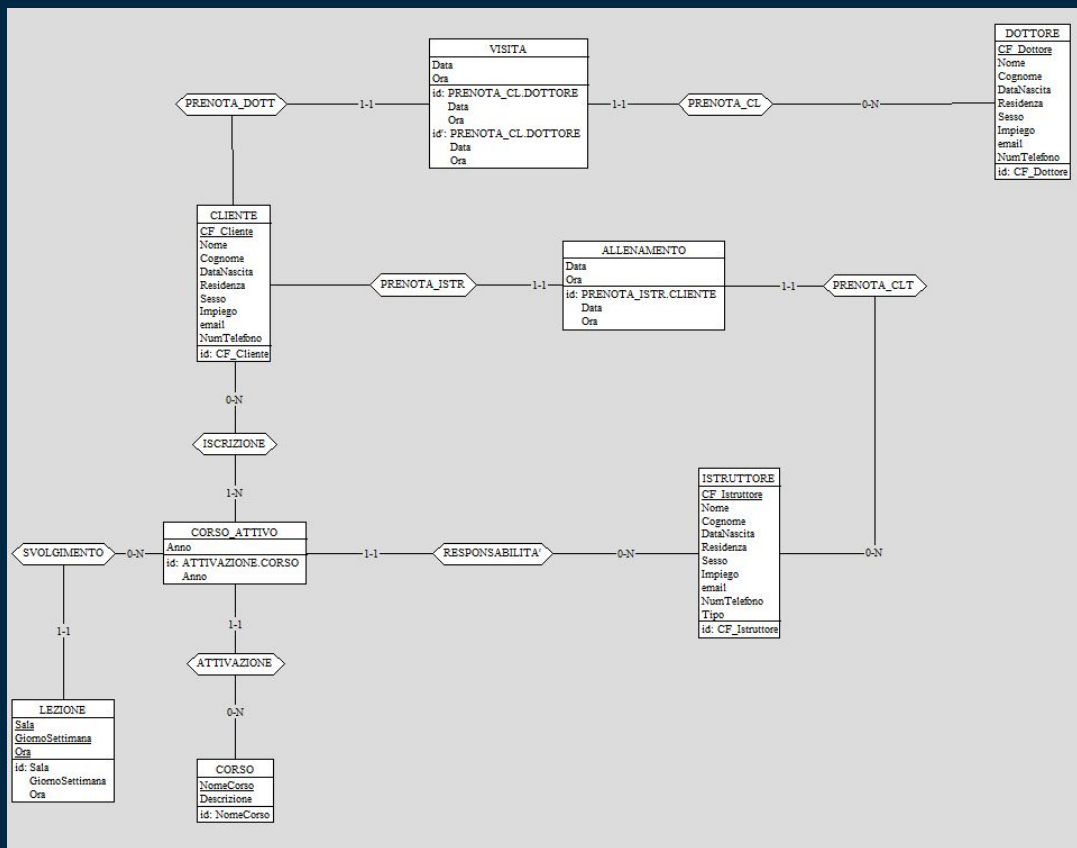


Operazione N° 4: si vuole controllare gli allenamenti prenotati da un cliente. Prima però è necessario fare qualche considerazione sulla realtà:

- Non viene contato il numero totale di clienti della palestra perchè la maggior parte di essi probabilmente si allenerà in maniera autonoma senza aiuto di alcun istruttore, perciò questa operazione non verrà quasi mai eseguita su questi.
- Di solito se si decide di essere seguiti da un istruttore si prenotano più allenamenti poi dopo si decide se proseguire, in media ne contiamo 2.

Concetto	Tipo	Accessi	Tipo
Cliente	E	1	L
Prenota_Istr	R	2	L
Allenamento	E	2	L
Prenota_CL	R	2	L
Istruttore	E	2	L

Conceptual-to-logic Model



Per tradurre lo schema E/R nel modello logico sono stati eseguiti i seguenti passaggi:

- Eliminazione gerarchia Istruttore tramite un collasso verso l'alto, e aggiungendo un attributo "Tipo" che definisce se l'istruttore è interno alla palestra o esterno.
- Per le Associazioni non è stato necessario trasformarle in entità dello schema relazionale perchè sono tutte 1-1, tranne l'associazione ISCRIZIONE per la quale è stato necessario creare un'entità in più.

6. Schema Relazionale

CLIENTE(CF_Cliente, Nome, Cognome, DataNascita, Residenza, Sesso, Impiego, Email, NumTelefono)

DOTTORE(CF_Dottore, Nome, Cognome, DataNascita, Residenza, Sesso, Impiego, Email, NumTelefono)

VISITA(Data, Ora, CF_Cliente, CF_Dottore)

FK: CF_Cliente REFERENCE CLIENTE

FK: CF_Dottore REFERENCE DOTTORE

ISTRUTTORE(CF_Istruttore, Nome, Cognome, DataNascita, Residenza, Sesso, Impiego, Email, NumTelefono, Tipo)

ALLENAMENTO(Data, Ora, CF_Cliente, CF_Istruttore)

FK: CF_Cliente REFERENCE CLIENTE

FK: CF_Istruttore REFERENCE ISTRUTTORE

ISCRIZIONE(CF_Cliente, Anno, NomeCorso)

FK: CF_Cliente REFERENCE CLIENTE

FK: NomeCorso REFERENCE CORSO

CORSO_ATTIVO(Anno, NomeCorso, CF_Istruttore)

FK: NomeCorso REFERENCE CORSO

FK: CF_Istruttore REFERENCE ISTRUTTORE

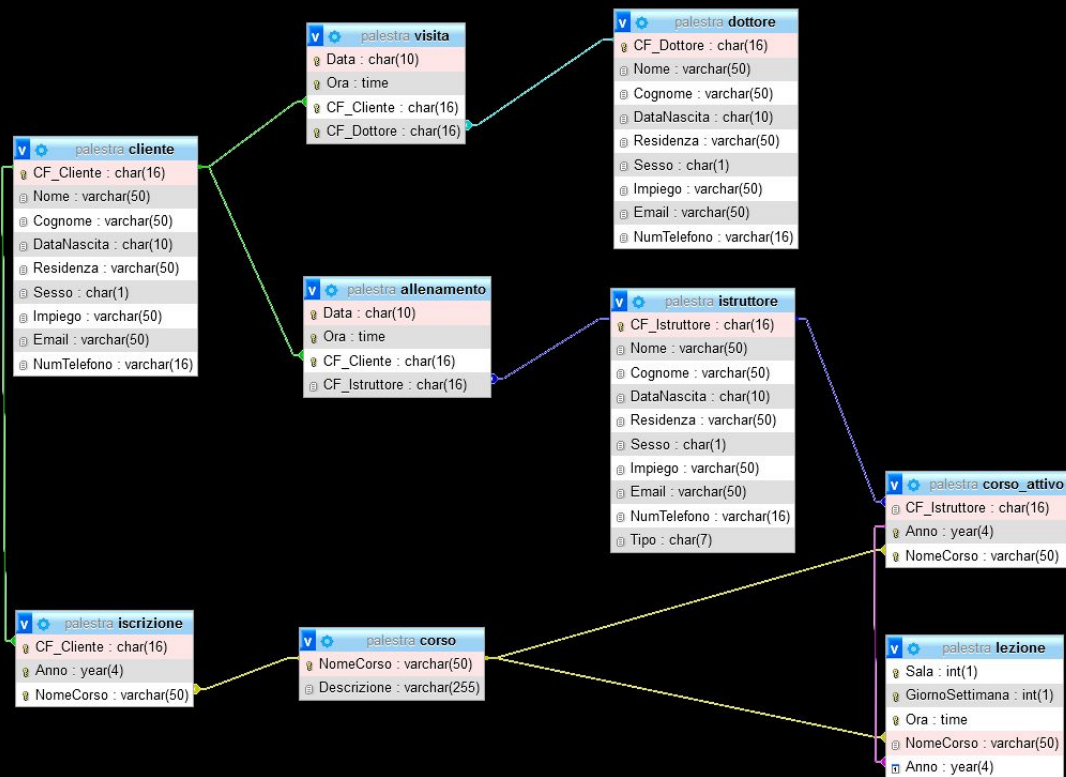
CORSO(NomeCorso, Descrizione)

LEZIONE(Sala, GiornoSettimana, Ora, NomeCorso, Anno)

FK: NomeCorso REFERENCE CORSO

FK: Anno REFERENCE CORSO_ATTIVO

6b.Screenshot modello relazionale



Query SQL

Op.6: Mostra i dati dei clienti iscritti a un corso (es. Corso "Sem")

```
SELECT *  
FROM Iscrizione I NATURAL JOIN Cliente C  
WHERE Nomecorso = "sem"
```

Op.7: Mostra quanti iscritti ci sono per ciascun corso, il nome e la sua descrizione

```
SELECT COUNT(CF_Cliente) AS NumIsritti, Nomecorso, Descrizione  
FROM Iscrizione I NATURAL JOIN Corso C  
GROUP by Nomecorso  
ORDER by NumIsritti
```

Op.8: Mostra tutte le visite prenotate e i dati dei corrispettivi clienti e dottori

```
SELECT *  
FROM Cliente c NATURAL JOIN Visita v JOIN Dottore d ON(v.CF_Dottore = d.CF_Dottore)
```

Op. 9: Mostra le lezioni presenti il mercoledì e il venerdì

```
SELECT *  
FROM lezione  
WHERE giornosettimana = "3" OR giornosettimana ="5"  
ORDER by giornosettimana, ora
```