

**Corso di studi in Informatica**  
**UNIVERSITÀ DELLA CALABRIA**

**PROGETTO PER IL CORSO DI**  
**BASI DI DATI (9 CFU)**

**DOCENTE: PROF. P. RULLO**

**LABORATORIO:**  
**ING. G. LABOCETTA**  
**DOTT.SSA D. ANGILICA**

**SISTEMA INFORMATIVO**  
**PER LA GESTIONE DI UN FITNESS CLUB.**

*213959, Oliverio Antonio Davide*

**TEMATICA PROGETTUALE**

La progettazione del sistema informatico in esame riguarda la gestione di un Fitness Club.

**SPECIFICA DELLE OPERAZIONI SUI DATI PREVISTE**

1. Inserire tutti i dati relativi ad un nuovo cliente.
2. Modificare l'orario di una lezione.
3. Eliminare i dati di un determinato medico.
4. Visualizzare tutte le lezioni tenute da un determinato istruttore in un dato giorno della settimana.
5. Visualizzare il numero di abbonamenti sottoscritti da ciascun cliente.
6. Visualizzare il cliente che ha sottoscritto l'abbonamento con il numero massimo di corsi.
7. Visualizzare il cliente che ha prenotato lo stesso campetto due volte in una settimana.
8. Visualizzare la sala in cui si svolge il minor numero di attività sportive.
9. Visualizzare gli istruttori che hanno solo il brevetto di nuoto.
10. Visualizzare i medici che sono anche clienti del centro ma non seguono attività che si svolgono in un determinato orario.
11. Controllare che le lezioni per attività sportive svolte in sala abbiano come struttura di svolgimento la sala e idem per il campo.
12. Snellire le prenotazioni spostando le vecchie di oltre un anno in un'entità separata (Prenotazioni\_Storiche). Fare questa operazione ogni fine anno.

## 1. RACCOLTA ED ANALISI DEI REQUISITI

	Requisiti Richiesti
1	Per il centro di fitness lavora uno staff, costituito da personale amministrativo, medici e da un gruppo di istruttori. Ogni membro dello staff è caratterizzato da nome, cognome, sesso, luogo e data di nascita.
2	
3	
4	Un medico è caratterizzato inoltre dalla data di conseguimento della abilitazione alla professione e dalla data di conseguimento del titolo.
5	
6	Un istruttore può avere il titolo di diploma ISEF o di laurea in Scienze Motorie e può avere conseguito uno o più brevetti. Un brevetto è identificato da un titolo, da un livello, una descrizione, è emesso da un ente sportivo e dà una qualifica per l'insegnamento di un determinato tipo di attività sportiva (nuoto, tennis, body-building...). Si vuole conoscere tutti brevetti posseduti da un istruttore e la data di conseguimento degli stessi. Il database deve tenere traccia dei clienti (o soci) del centro di fitness. Un cliente è caratterizzato da nome, cognome, sesso, luogo e data di nascita.
7	
8	
9	
10	
11	Un abbonamento è caratterizzato da una data di inizio e da una durata di validità e fornisce l'iscrizione del cliente a una o più attività sportive. Le attività sportive possono essere svolte in sala o in un campo. Il costo dell'abbonamento dipende dal costo delle singole attività sportive scelte dal cliente. Alla scadenza del proprio abbonamento, il cliente può rinnovare lo stesso o sottoscriverne uno nuovo (ad esempio, con attività sportive diverse). Il cliente deve sottoporsi a una visita medica, effettuata da uno dei medici del centro, sia al momento dell'iscrizione che del rinnovo dell'abbonamento.
12	
13	
14	
15	
16	Un'attività sportiva è caratterizzata da un nome univoco e da una descrizione. Si vuole memorizzare, inoltre, il numero massimo di partecipanti e il numero di iscritti.
17	
18	Il centro di fitness si compone di un certo numero di sale, di diversa tipologia (sala fitness, sala corsi), da uno o più spogliatoi maschili e femminili e da un certo numero di campi da gioco.
19	
20	Per sale e per campetti, si vuole memorizzare il nome, univoco all'interno del fitness center, una descrizione e la capienza (espressa in numero di persone).
21	
22	Per tutti i tipi di attività sportiva sono previste una o più lezioni settimanali, ognuna delle quali è caratterizzata da un giorno, un orario di inizio e una durata, espressa in minuti.
23	
24	Per ogni lezione, si vuole memorizzare la sala o il campetto in cui si svolge e l'istruttore che la supervisiona.
25	
26	Un cliente, inoltre, può prenotare un campetto (negli orari in cui questo non è occupato per le lezioni).
27	
28	A tal fine, è di interesse conoscere anche il costo per la prenotazione di ogni campetto da gioco.
29	
30	
31	
32	
33	
34	
35	

## 1.1. Eliminazioni delle Ambiguità

Linea	Termine	Sinonimi	Motivazione Correzione
5	Titolo	Titolo di Laurea	Titolo è generico
11	Soci	Clienti	Ease of use: termine secondario non necessario
24	campi da gioco	Campi	Ease of use: indicano la stessa cosa
28	Giorno	giorno settimanale	Giorno può confondersi con un numero, invece, con giorno settimanale: Lunedì, Martedì, ecc ecc
30	Campetto	Campo	Ease of use: indicano la stessa cosa
34,35	campo da gioco	Campo	Ease of use: indicano la stessa cosa

## 1.2. Ristrutturazione dei Requisiti Richiesti

Specifiche ristrutturate
<p>Ogni <b>persona</b> (cliente e staff) del centro, è caratterizzata da:</p> <ul style="list-style-type: none"><li>• Codice identificativo (Codice Fiscale)</li><li>• Nome</li><li>• Cognome</li><li>• Sesso</li><li>• Luogo di nascita</li><li>• Data di nascita</li></ul> <p>All'interno del centro fitness lavora uno <b>staff</b>, costituito da:</p> <ul style="list-style-type: none"><li>• <b>Personale Amministrativo</b></li><li>• <b>Medici</b>, aventi:<ul style="list-style-type: none"><li>◦ Data rilascio abilitazione</li><li>◦ Data rilascio laurea</li><li>◦ Flag di Esercizio ( a scopo progettuale, indica se ancora il medico esercita all'interno del centro Fitness)</li></ul></li><li>• <b>Istruttori</b>, aventi:<ul style="list-style-type: none"><li>◦ Diploma ISEF o Laurea in Scienze Motorie</li><li>◦ Uno, o più brevetti</li></ul></li></ul>

### Un **Brevetto**:

- E' identificato da:
  - Codice identificativo (a scopo progettuale)
  - Titolo
  - Livello
  - Descrizione
- E' emesso da un ente sportivo
- Da una qualifica per l'insegnamento di un'attività sportiva
- Si vuole conoscere tutti i brevetti posseduti da un' istruttore e la data di conseguimento degli stessi

### Un' **Attività Sportiva** è caratterizzata da:

- Luogo di Svolgimento (campo o sala)
- Costo (Indica il costo dell'inserimento all'interno di un eventuale abbonamento)
- Nome univoco
- Descrizione
- Numero massimo di partecipanti
- Numero di Iscritti

### Un **Abbonamento** è caratterizzato da:

- Data inizio
- Durata di validità (espressa in giorni)
- Si riferisce ad un singolo cliente
- Fornisce l'iscrizione ad una o più attività sportive
- Costo (calcolato sommando i costi delle singole attività sportive scelte)
- Alla scadenza dell'abbonamento il cliente può rinnovare lo stesso o sottoscriverne un altro

Un cliente deve sottoporsi ad una visita medica (a cui è associata una data di scadenza) effettuata da uno dei medici del centro, sia al momento dell'iscrizione che del rinnovo dell'abbonamento. (La visita va ripetuta solo nel caso in cui la precedente sia scaduta).

Il centro Fitness, è composto da un certo numero di strutture (campi e sale), e da un certo numero di spogliatoi.

### Le **Sale** sono caratterizzate da:

- Tipologia (sala fitness, sala yoga)
- Nome univoco
- Descrizione
- Capienza (numero massimo di persone)

### I **Campi** sono caratterizzati da:

- Nome univoco
- Descrizione
- Capienza (numero massimo di persone)
- Costo di prenotazione (Orario)

Uno **Spogliatoio** è caratterizzato da un codice univoco.

Per tutti i tipi di attività sportive sono previste una o più **Lezioni settimanali**, le quali sono strutturate da:

- Sport associato
- Giorno settimanale
- Ora inizio
- Sala o Campo in cui si svolge
- Istruttore che supervisiona
- Durata

E' interesse del FitnessCLub memorizzare le prenotazioni vecchie di oltre un anno separatamente dalle altre.

### 1.3. Raffinamento delle Specifiche e Individuazione dei Concetti Base

Dopo aver eliminato tutte le fonti di ambiguità presenti nel testo, si può eseguire un'analisi più approfondita dei requisiti al fine di individuare i concetti di base, ossia le entità coinvolte nella realtà di interesse. I requisiti del sistema vengono, pertanto, suddivisi in ulteriori sotto parti ognuna delle quali raggruppa le informazioni relative ad uno specifico concetto.

#### Frasi di carattere generale

Il centro di fitness si compone di un certo numero di strutture (sale e campi) e da un certo numero di spogliatoi. Si vuole conoscere tutti i brevetti posseduti da un istruttore e la data di conseguimento degli stessi. Il database deve tenere traccia dei clienti del centro di fitness. Alla scadenza del proprio abbonamento, il cliente può rinnovare lo stesso o sottoscriverne uno nuovo (ad esempio, con attività sportive diverse). È di interesse conoscere la data di prenotazione, l'orario di inizio, la durata e il costo per la prenotazione di ogni campetto da gioco. È interesse del Centro Fitness memorizzare le prenotazioni vecchie di oltre un anno separatamente dalle altre.

#### Frasi relative a Persone (Staff e Clienti)

Ogni **persona** (cliente e staff) del centro, è caratterizzata da:

- Codice identificativo (Codice Fiscale)
- Nome
- Cognome
- Sesso
- Luogo di nascita
- Data di nascita

### Frasi relative a Staff

All'interno del centro fitness lavora uno **staff**, costituito da:

- **Personale Amministrativo**
- **Medici**, aventi:
  - Data rilascio abilitazione
  - Data rilascio laurea
  - Esercita ( a scopo progettuale, indica se ancora il medico esercita all'interno del centro Fitness)
- **Istruttori**, aventi:
  - Diploma ISEF o Laurea in Scienze Motorie
  - Uno, o più brevetti

### Frasi relative a Medico

**Medici**, aventi:

- Data rilascio abilitazione
- Data rilascio laurea
- Esercita

Il cliente deve sottoporsi a **una visita medica** (a cui viene associata una data di scadenza), effettuata da uno dei medici del centro, sia al momento dell'iscrizione che del rinnovo dell'abbonamento. (La visita va effettuata nuovamente solo se l'ultima visita è scaduta.)

### Frasi relative ad Istruttore

**Istruttori**, aventi:

- Diploma ISEF o Laurea in Scienze Motorie
- Uno, o più brevetti

### Frasi relative ad Attività Sportiva

Un'**Attività Sportiva** è caratterizzata da:

- Luogo di Svolgimento (campo o sala)
- Costo (Indica il costo dell'inserimento all'interno di un eventuale abbonamento)
- Nome univoco
- Descrizione
- Numero massimo di partecipanti
- Numero di Iscritti

### **Frase relative a Brevetto**

#### **Un Brevetto:**

- E' identificato da:
  - Codice identificativo (a scopo progettuale)
  - Titolo
  - Livello
  - Descrizione
- E' emesso da un ente sportivo
- Da una qualifica per l'insegnamento di un'attività sportiva
- Si vuole conoscere tutti i brevetti posseduti da un'istruttore e la data di conseguimento degli stessi

### **Frase relative ad Abbonamento**

#### **Un Abbonamento** è caratterizzato da:

- Data inizio
- Durata di validità (espressa in giorni)
- Si riferisce ad un singolo cliente
- Fornisce l'iscrizione ad una o più attività sportive
- Costo (calcolato sommando i costi delle singole attività sportive scelte)
- Alla scadenza dell'abbonamento il cliente può rinnovare lo stesso o sottoscriverne un altro
- Validità (scopo progettuale)

### **Frase relative a Cliente**

Il cliente può acquistare un abbonamento.

Il cliente deve sottoporsi ad una visita medica (a cui è associata una data di scadenza), effettuata da uno dei medici del centro, sia al momento dell'iscrizione che del rinnovo dell'abbonamento. (la visita va effettuata solo se la precedente è scaduta).

Un cliente, inoltre, può prenotare un campetto (negli orari in cui questo non è occupato per le lezioni).

### **Frase relative a Strutture**

Il centro di fitness si compone di un certo numero di strutture (sale e campi) e da un certo numero di spogliatoi.

### Frasi relative a Sala

Le **Sale** sono caratterizzate da:

- Tipologia (sala fitness, sala yoga)
- Nome univoco
- Descrizione
- Capienza (numero massimo di persone)

### Frasi relative a Campo

I **Campi** sono caratterizzati da:

- Nome univoco
- Descrizione
- Capienza (numero massimo di persone)
- Costo di prenotazione (Orario)

### Frasi relative a Spogliatoio

Uno **Spogliatoio** è caratterizzato da un codice univoco.

### Frasi relative a Lezione

Per tutti i tipi di attività sportive sono previste una o più **Lezioni settimanali**, le quali sono strutturate da:

- Sport associato
- Giorno settimanale
- Ora inizio
- Sala o Campo in cui si svolge
- Istruttore che supervisiona
- Durata

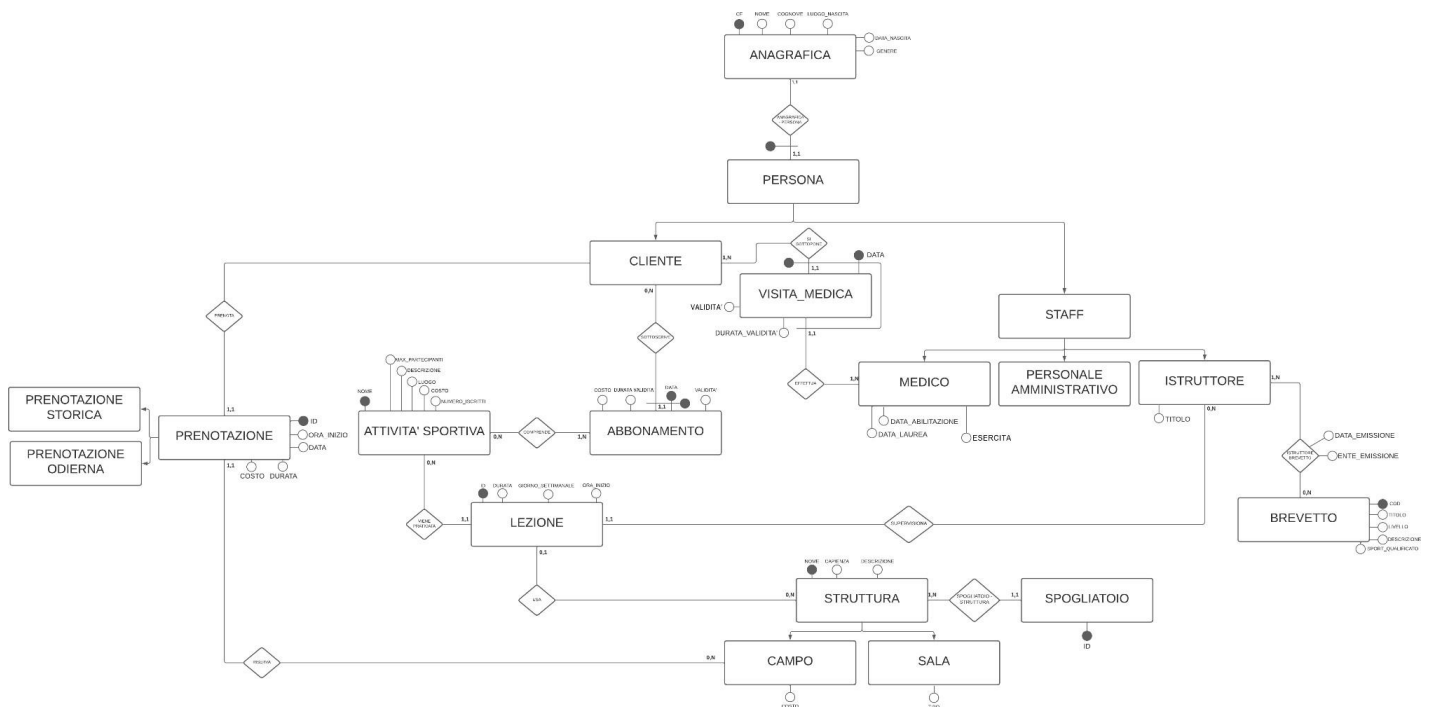
### Considerazioni aggiunte a scopo progettuale:

1. Aggiunto "Esercita" in Medico, per far sì che, in caso si voglia eliminare dalla base di dati un medico, invece di eliminare i dati del suddetto medico, si cambia il valore della flag, così per mantenere traccia delle informazioni della persona la quale ha eseguito visite mediche, per ragioni di sicurezza ed eventuali prove legali.
2. Aggiunta "Validità" in Abbonamento, che esplicita se in quel momento l'abbonamento è ancora valido.
3. Aggiunta "Validità" in Visita Medica, che esplicita se in quel momento la visita medica è ancora valida



## 2. PROGETTAZIONE CONCETTUALE

### 2.1. Schema ER



### 2.2. Documentazione dello Schema ER

#### 2.2.1. Dizionario dei Dati

Entità	Descrizione	Attributi	Identificatore
ANAGRAFICA	Entità che rappresenta i dati anagrafici di ogni componente del Fitness Club, tra cui i clienti, medici, personale amministrativo e gli istruttori.	CF (Codice Fiscale) Nome Cognome Genere Luogo_nascita Data_nascita	CF
CLIENTE	Entità che rappresenta il Cliente all'interno del Fitness Club. Specializzazione di Persona.	CF (Codice Fiscale) Nome Cognome Genere Luogo_nascita Data_nascita	CF

MEDICO	Entità che rappresenta i medici all'interno del Fitness Club. Specializzazione di Staff.	CF (Codice Fiscale) Nome Cognome Genere Luogo_nascita Data_nascita Data_abilitazione Data_laurea Esercita	CF
AMMINISTRAZIONE	Entità che rappresenta i medici all'interno del Fitness Club. Specializzazione di Staff.	CF (Codice Fiscale) Nome Cognome Genere Luogo_nascita Data_nascita	CF
ISTRUTTORE	Entità che rappresenta gli Istruttori all'interno del Fitness Club. Specializzazione di Staff.	CF (Codice Fiscale) Nome Cognome Genere Luogo_nascita Data_nascita Titolo	CF
BREVETTO	Entità che rappresenta i brevetti possedibili dagli istruttori del Fitness Club.	Cod Titolo Livello Descrizione Sport_qualificato	Cod
VISITA MEDICA	Entità che rappresenta una visita medica effettuata da un cliente con un medico.	Data_visita Durata_validità Validità	Data_visita (componente di chiave composta)

ABBONAMENTO	Entità che rappresenta un abbonamento acquistato da un cliente.	Data_inizio Costo Durata_validità Validità	Data_inizio (componente di chiave composta)
ATTIVITA' SPORTIVA	Entità che rappresenta un'attività sportiva praticabile all'interno del Fitness Club	Nome Descrizione Luogo Costo Max_partecipanti Numero_iscritti	Nome
LEZIONE	Entità che rappresenta un lezione seguita da un istruttore su di una determinata attività sportiva.	ID Durata Giorno_settimanale Ora_inizio	ID
PRENOTAZIONE ODIERNA	Entità che rappresenta una prenotazione di un campo effettuata da un cliente.	Id Ora_inizio Data Costo Durata	Id
PRENOTAZIONE STORICA	Entità che rappresenta una prenotazione di un campo effettuata da un cliente a distanza di un anno dalla data odierna	Id Ora_inizio Data Costo Durata	Id

CAMPO	Entità che rappresenta i campi utilizzabili per attività sportive all'interno del Fitness Club.	Nome Descrizione Capienza Costo	Nome
SALA	Entità che rappresenta le sale utilizzabili per attività sportive e/o altro all'interno del Fitness Club.	Nome Descrizione Capienza Tipo	Nome
SPOGLIATOIO	Entità che rappresenta gli spogliatoi utilizzabili dal cliente all'interno del Fitness Club.	Id	Id
PRENOTAZIONI STORICHE	Entità che rappresenta tutte quelle prenotazioni fatte dai clienti che, dalla data di scadenza, è passato un anno o più.	Id Cliente Campo Ora_inizio Data Costo Durata	Id
STAFF	Entità che rappresenta i dati anagrafici del personale del Fitness Club. (Medici, Pers. Amministrativo, Istruttori). Specializzazione di Persona.	CF (Codice Fiscale) Nome Cognome Genere Luogo_nascita Data_nascita	CF

## 2.2.2. Descrizione Entità

Anagrafica			
<b>Descrizione</b>	Entità che rappresenta i dati anagrafici di ogni componente del Fitness Club, tra cui i clienti, medici, personale amministrativo e gli istruttori.		
<b>Nome attributo</b>	<b>Tipo di Dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
CF	Alfanumerico	Chiave primaria	Codice fiscale della persona
Nome	Testo	Non nullo	Nome della persona
Cognome	Testo	Non nullo	Cognome della Persona
Genere	Carattere	Non nullo. Value $\in \{M,F,A\}$	Indica il genere della persona, maschile, femminile o altro
Luogo_nascita	Testo	Non nullo	Luogo di nascita della Persona
Data_nascita	Data	Non nullo	Data di nascita della Persona

Cliente			
<b>Descrizione</b>	Entità che rappresenta il Cliente all'interno del Fitness Club. Specializzazione di Persona.		
<b>Nome Attributo</b>	<b>Tipo di dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
CF	Alfanumerico	Chiave primaria	Codice fiscale del cliente, riferito alla chiave primaria di Persona.
Nome	Testo	Non nullo	Nome del cliente
Cognome	Testo	Non nullo	Cognome del cliente
Genere	Carattere	Non nullo. Value $\in \{M,F,A\}$	Indica il genere del cliente, maschile o femminile (M o F)
Luogo_nascita	Testo	Non nullo.	Luogo di nascita del cliente
Data_nascita	Data	Non nullo	Data di nascita del cliente

Staff			
<b>Descrizione</b>	Entità che rappresenta i dati anagrafici del personale del Fitness Club. (Medici, Pers. Amministrativo, Istruttori). Specializzazione di Persona.		
<b>Nome Attributo</b>	<b>Tipo di dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
CF	Alfanumerico	Chiave primaria	Codice fiscale del membro dello staff, riferito alla chiave primaria di Persona.
Nome	Testo	Non nullo	Nome del membro dello staff
Cognome	Testo	Non nullo	Cognome del membro dello Staff
Genere	Carattere	Non nullo. Value $\in \{M,F,A\}$	Indica il genere del membro dello staff, maschile o femminile (M o F)
Luogo_nascita	Testo	Non nullo.	Luogo di nascita del membro dello staff
Data_nascita	Data	Non nullo	Data di nascita del membro dello staff

Medico			
<b>Descrizione</b>	Entità che rappresenta i medici all'interno del Fitness Club. Specializzazione di Staff.		
<b>Nome Attributo</b>	<b>Tipo di dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
CF	Alfanumerico	Chiave primaria	Codice fiscale del medico, riferito alla chiave primaria di Staff.
Nome	Testo	Non nullo	Nome del medico
Cognome	Testo	Non nullo	Cognome del medico
Genere	Carattere	Non nullo. Value $\in \{M,F,A\}$	Indica il genere del medico, maschile o femminile (M o F)
Luogo_nascita	Testo	Non nullo.	Luogo di nascita del medico
Data_nascita	Data	Non nullo	Data di nascita del medico
Data_abilitazione	Data	Non nullo	Data rilascio abilitazione del medico
Data_laurea	Data	Non nullo	Data conseguimento laurea del medico
Esercita	Booleano	Non nullo	Rappresenta l'attività del medico
Validità	Booleano	Non nullo	Indica se il medico esercita o meno nel centro fitness

Amministrazione			
<b>Descrizione</b>	Entità che rappresenta i medici all'interno del Fitness Club. Specializzazione di Staff.		
<b>Nome Attributo</b>	<b>Tipo di dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
CF	Alfanumerico	Chiave primaria	Codice fiscale del membro dell'amministrazione, riferito alla chiave primaria di Staff.
Nome	Testo	Non nullo	Nome del membro dell'amministrazione
Cognome	Testo	Non nullo	Cognome del membro dell'amministrazione
Genere	Carattere	Non nullo. Value $\in \{M,F,A\}$	Indica il genere del membro dell'amministrazione, maschile o femminile (M o F)
Luogo_nascita	Testo	Non nullo.	Luogo di nascita del membro dell'amministrazione
Data_nascita	Data	Non nullo	Data di nascita del membro dell'amministrazione



Istruttore			
<b>Descrizione</b>	Entità che rappresenta gli Istruttori all'interno del Fitness Club. Specializzazione di Staff.		
<b>Nome Attributo</b>	<b>Tipo di dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
CF	Alfanumerico	Chiave primaria	Codice fiscale dell'istruttore, riferito alla chiave primaria di Staff.
Nome	Testo	Non nullo	Nome dell'istruttore
Cognome	Testo	Non nullo	Cognome dell'istruttore
Genere	Carattere	Non nullo. Value $\in$ {M,F,A}	Indica il genere dell'istruttore, maschile o femminile (M o F)
Luogo_nascita	Testo	Non nullo.	Luogo di nascita dell'istruttore
Data_nascita	Data	Non nullo	Data di nascita dell'istruttore
Titolo	Testo	Non nullo. Solo 'Diploma ISEF' o 'Laura inscienze motorie'	Titolo di conseguimento del diploma ISEF o Laurea in scienze Motorie

Brevetto			
<b>Descrizione</b>	Entità che rappresenta i brevetti possedibili dagli istruttori del Fitness Club.		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Cod	Numerico	Chiave primaria, autoincremento	Codice che identifica univocamente un brevetto
Titolo	Testo	Non nullo	Titolo univoco del brevetto
Livello	Testo	Non nullo	Indica il livello di validità del brevetto. Per esempio validità Nazionale o regionale ecc ecc
Descrizione	Testo	Nessun vincolo	
Sport_qualificato	Testo	Non nullo	Sport specifico che il brevetto qualifica

Visita Medica			
<b>Descrizione</b>	Entità che rappresenta una visita medica effettuata da un cliente con un medico.		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Data_visita	Data	Chiave primaria composta	Data in cui la visita è stata effettuata
Durata_validità	Numerico	Non nullo Valore intero $30 \leq \text{valore} \leq 356$	Durata di validità della visita, espressa in giorni
Validità	Booleano	Non nullo	Indica se la visita medica è ancora valida o no

Abbonamento			
<b>Descrizione</b>	Entità che rappresenta un abbonamento acquistato da un cliente		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Data_inizio	Data	Chiave primaria composta	Chiave primaria composta che identifica un abbonamento
Costo	Numerico	Non nullo $\text{Value} \geq 0$	Costo, calcolato come somma dei costi delle attività sportive comprese
Durata_validità	Numerico	Non nullo $\text{Value} \geq 1$	Durata di validità dell'abbonamento, espressa in giorni
Validità	Booleano	Non nullo	Positiva o negativa validità dell'abbonamento

Attività sportiva			
<b>Descrizione</b>	Rappresenta un'attività sportiva praticabile all'interno del Fitness Club		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Nome	Testo	Chiave primaria	Nome che identifica l'attività sportiva
Descrizione	Testo	Nessun vincolo	Descrizione opzionale dell'attività sportiva
Luogo	Booleano	Non nullo	Luogo in cui si può svolgere l'attività sportiva True = Campo False = Sala
Costo	Numerico	Non nullo, value>=0	Costo dell'attività sportiva
Max_partecipanti	Numerico	Non nullo Value > 0	Numero massimo di partecipanti all'attività sportiva
Numero_iscritti	Numerico	Non nullo, value>=0	Numero attuale di iscritti all'attività sportiva

Lezione			
<b>Descrizione</b>	Rappresenta una lezione seguita da un istruttore per una determinata attività sportiva		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Id	Numerico	Chiave primaria	Identificativo della lezione
Durata	Time	Non nulla Value >= 00:30:00	Durata della lezione
Giorno_settimanale	Numerico	Non nulla valore $\in$ { 1 = domenica, 2 = lunedì, 3 = martedì, 4 = mercoledì, 5 = giovedì, 6 = venerdì, 7 = sabato }(per corrispondere alla funzione daysofweek())	Giorno della settimana nel quale si svolge la lezione
Ora_inizio	Orario	Non nulla	Ora nella quale inizia la lezione

Prenotazione Storica			
<b>Descrizione</b>	Rappresenta una prenotazione di un campo effettuata da un cliente		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Id	Numerico	Chiave primaria, autoincremento	Identificativo di una prenotazione
Ora_inizio	Orario	Non nullo	Orario della prenotazione
Data	Data	Non nullo Almeno un anno precedente alla data odierna	Data della prenotazione
Costo	Numerico	Non nullo Valore>=0	Costo della prenotazione
Durata	Time	Non nulla Value >= 00:30:00	Durata della prenotazione

Prenotazione Odierna			
<b>Descrizione</b>	Rappresenta una prenotazione di un campo effettuata da un cliente		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Id	Numerico	Chiave primaria, autoincremento	Identificativo di una prenotazione
Ora_inizio	Orario	Non nullo	Orario della prenotazione
Data	Data	Non nullo	Data della prenotazione
Costo	Numerico	Non nullo Valore>=0	Costo della prenotazione
Durata	Time	Non nulla Value >= 00:30:00	Durata della prenotazione.

Struttura			
<b>Descrizione</b>	Rappresenta le strutture all'interno del FitnessClub (campi, sale)		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Nome	Testo	Chiave primaria	Nome univoco della struttura
Descrizione	Testo	Nessun vincolo	Descrizione facoltativa della struttura
Capienza	Numerico	Non nulla Valore>=0	Capienza massima di persone della struttura

Sala			
<b>Descrizione</b>	Rappresenta le sale utilizzabili all'interno del FitnessClub		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Nome	Testo	Chiave primaria	Nome univoco della sala
Descrizione	Testo	Nessun vincolo	Descrizione facoltativa della sala
Capienza	Numerico	Non nulla Valore>=0	Capienza massima di persone della sala
Tipo	Testo	Non nulla	Indica la tipologia della sala

Campo			
<b>Descrizione</b>	Rappresenta i campi utilizzabili all'interno del FitnessClub		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Nome	Testo	Chiave primaria	Nome univoco del campo
Descrizione	Testo	Nessun vincolo	Descrizione facoltativa del campo
Capienza	Numerico	Non nulla Valore>=0	Capienza massima di persone del campo
Costo	Numerico	Non nullo Valore>=0	Costo del campo in caso di prenotazione

Spogliatoio			
<b>Descrizione</b>	Rappresenta gli spogliatoi utilizzabili all'interno del FitnessClub		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Id	Numerico	Chiave primaria, autoincremento	Identificativo dello spogliatoio

Prenotazioni storiche			
<b>Descrizione</b>	Entità che rappresenta prenotazioni datate di oltre un anno dalla data odierna. Nasce per una questione di ottimizzazione, snellendo quelle che saranno le tuple di Prenotazione.		
<b>Nome Attributo</b>	<b>Tipo dato</b>	<b>Vincolo</b>	<b>Descrizione</b>
Id	Numerico	Chiave primaria, autoincremento	Identificativo di una prenotazione
Ora_inizio	Orario	Non nullo	Orario della prenotazione
Data	Data	Non nullo	Data della prenotazione
Costo	Numerico	Non nullo Valore>=0	Costo della prenotazione
Durata	Time	Non nulla Value >= 00:30:00	Durata della prenotazione.
CF_cliente	Alfanumerico	Non nullo	Codice fiscale del cliente cha ha effettuato la prenotazione
Campo	Testo	Non nullo	Nome univoco del campo prenotato

### 2.2.3. Descrizione Relazioni

Relazione PRENOTA	
Descrizione	Consente di correlare i Clienti con le prenotazioni da loro effettuate
Entità Coinvolte	
Entità	Cardinalità
CLIENTE	0,N
PRENOTAZIONE ODIERNA	1,1

Relazione PRENOTA	
Descrizione	Consente di correlare i Clienti con le prenotazioni storiche da loro effettuate
Entità Coinvolte	
Entità	Cardinalità
CLIENTE	0,N
PRENOTAZIONE STORICA	1,1

Relazione ISTRUTTORE-BREVETTO	
Descrizione	Consente di correlare gli Istruttori con i Brevetti da loro conseguiti
Entità Coinvolte	
Entità	Cardinalità
ISTRUTTORE	1,N
BREVETTO	0,N

Relazione USA
---------------

Descrizione	Consente di correlare la lezione e la struttura che usa per la sua funzione
Entità Coinvolte	
Entità	Cardinalità
LEZIONE	0,1
SALA	0,N

Relazione USA	
Descrizione	Consente di correlare la lezione e la struttura che usa per la sua funzione
Entità Coinvolte	
Entità	Cardinalità
LEZIONE	0,1
CAMPO	0,N

Relazione VIENE-PRATICATA	
Descrizione	Consente di correlare un'attività sportiva e la lezione in cui viene praticata
Entità Coinvolte	
Entità	Cardinalità
ATTIVITA' SPORTIVA	0,N
LEZIONE	1,1

Relazione CAMPO-SPOGLIATOIO	
Descrizione	Consente di correlare i Campi agli spogliatoi previsti
Entità Coinvolte	
Entità	Cardinalità
SPOGLIATOIO	1,1
CAMPO	1,N

Relazione SALA-SPOGLIATOIO	
Descrizione	Consente di correlare le Sale agli spogliatoi previsti



Entità Coinvolte	
Entità	Cardinalità
SPOGLIATOIO	1,1
SALA	1,N

Relazione SUPERVISIONA	
Descrizione	Consente di correlare gli Istruttori con le lezioni da loro supervisionate
Entità Coinvolte	
Entità	Cardinalità
ISTRUTTORE	0,N
LEZIONE	1,1

Relazione ANAGRAFICA-CLIENTE	
Descrizione	Consente di correlare le informazioni anagrafiche con i relativi clienti
Entità Coinvolte	
Entità	Cardinalità
ANAGRAFICA	0,1
CLIENTI	1,1

Relazione ANAGRAFICA-MEDICO	
Descrizione	Consente di correlare le informazioni anagrafiche con i relativi Medici
Entità Coinvolte	
Entità	Cardinalità
ANAGRAFICA	0,1
MEDICO	1,1

Relazione ANAGRAFICA-PERSONALE AMMINISTRATIVO	
Descrizione	Consente di correlare le informazioni anagrafiche con i relativi membri del Personale Amministrativo
Entità Coinvolte	
Entità	Cardinalità
ANAGRAFICA	0,1
PERSONALE AMMINISTRATIVO	1,1

Relazione ANAGRAFICA-ISTRUTTORE
---------------------------------

Descrizione	Consente di correlare le informazioni anagrafiche con i relativi Istruttori
Entità Coinvolte	
Entità	Cardinalità
ANAGRAFICA	0,1
ISTRUTTORE	1,1

Relazione SOTTOSCRIVE	
Descrizione	Consente di correlare i Clienti ai relativi abbonamenti sottoscritti
Entità Coinvolte	
Entità	Cardinalità
CLIENTE	0,N
ABBONAMENTO	1,1

Relazione SI SOTTOPONE	
Descrizione	Consente di correlare i Clienti alle visite medica a cui si è sottoposto
Entità Coinvolte	
Entità	Cardinalità
CLIENTE	0,N
VISITA MEDICA	1,1

Relazione EFFETTUA	
Descrizione	Consente di correlare i Medici alle visite medica che ha effettuato
Entità Coinvolte	
Entità	Cardinalità
MEDICO	0,N
VISITA MEDICA	1,1

Relazione RISERVA	
Descrizione	Consente di correlare le Prenotazioni ai rispettivi campi riservati
Entità Coinvolte	
Entità	Cardinalità
PRENOTAZIONE ODIERNA	1,1
CAMPO	0,N

Relazione RISERVA	
-------------------	--

Descrizione	Consente di correlare le Prenotazioni storiche ai rispettivi campi riservati
Entità Coinvolte	
Entità	Cardinalità
PRENOTAZIONE STORICA	1,1
CAMPO	0,N

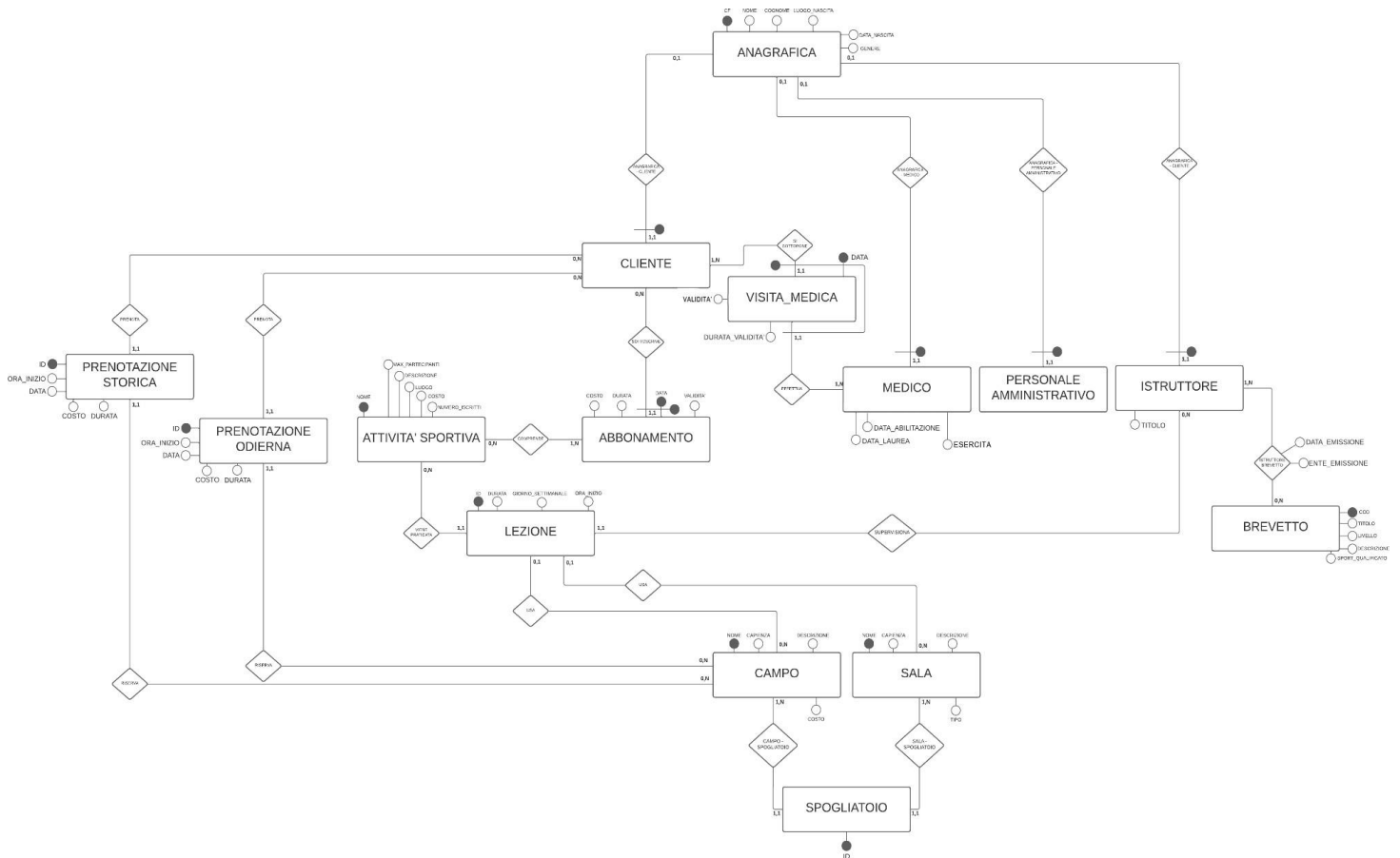
Relazione COMPRENDE	
Descrizione	Consente di correlare gli Abbonamenti con le rispettive Attività sportive comprese
Entità Coinvolte	
Entità	Cardinalità
ABBONAMENTO	1,N
ATTIVITA' SPORTIVE	0,N

### 2.3. Vincoli non espressi dallo Schema ER

Regole di Vincolo
La generalizzazione Persona è totale e sovrapposta, con specializzazioni: Cliente e Staff.
La generalizzazione Staff è totale e sovrapposta, con specializzazioni: Medico, Personale Amministrativo e Istruttore.
La generalizzazione Struttura è totale e disgiunta, con specializzazioni: Campo e Sala.
La generalizzazione Prenotazione è totale e disgiunta, con specializzazioni,
Un Istruttore deve obbligatoriamente possedere almeno uno tra: Diploma ISEF o Laurea in Scienze Motorie.
Il costo dell'abbonamento è calcolato come somma dei singoli costi delle attività sportive presenti nell'abbonamento stesso.
Una visita medica ha una durata minima espressa in giorni, indicante il suo periodo di validità.
Un campo non può essere prenotato da più di un cliente negli stessi orari.
Un istruttore non può supervisionare più lezioni durante gli stessi orari.
Una lezione non può essere svolta, nello stesso orario, sia in campo che in sala.
Una lezione che è associata ad una attività sportiva deve avere la struttura in cui si svolge coerente al "Luogo" dell'attività sportiva. Ad esempio, se l'attività è per sala allora la lezione associata deve essere svolta in una sala.
Prenotazione e lezioni hanno una durata minima di 30 minuti.
Costo di una prenotazione o di un abbonamento deve essere $\geq 0$ .
La durata di un abbonamento deve essere $> 0$ .

### **3. PROGETTAZIONE LOGICA**

#### **3.1. Ristrutturazione dello Schema ER**



### 3.2. Schema Logico

**Anagrafica (CF, Nome, Cognome, Luogo\_nascita, Data\_nascita, Genere)**

**Cliente (CF\*)**

Con vincolo d'integrità referenziale tra:  
attributo CF e la relazione Anagrafica

**Medico (CF\*, Data\_abilitazione, Data\_laurea, Esercizio)**

Con vincolo d'integrità referenziale tra:  
attributo CF e la relazione Anagrafica

**Istruttore (CF\*, Titolo)**

Con vincolo d'integrità referenziale tra:  
attributo CF e la relazione Anagrafica

**Personale\_Amministrativo (CF\*)**

Con vincolo d'integrità referenziale tra:  
attributo CF e la relazione Anagrafica

**Prenotazione\_Odierna (ID, Ora\_inizio, Data, Costo, Durata, Cliente\*, Campo\*)**

Con vincolo d'integrità referenziale tra:  
attributo Cliente e la relazione Cliente  
attributo Campo e la relazione Campo

**Prenotazione\_Storica(ID, Ora\_inizio, Data, Costo, Durata, Cliente\*, Campo\*)**

Con vincolo d'integrità referenziale tra:  
attributo Cliente e la relazione Cliente  
attributo Campo e la relazione Campo

**Abbonamento (Data, Cliente\*, Costo, Durata\_validità, Validità)**

Con vincolo d'integrità referenziale tra:  
attributo Cliente e la relazione Cliente

**Visita\_Medica (Data, Cliente\*, Medico\*, Durata\_validità, Validità)**

Con vincolo d'integrità referenziale tra:  
attributo Cliente e la relazione Cliente  
attributo Medico e la relazione Medico

**Brevetto (Cod, Titolo, Livello, Descrizione, Sport\_qualificato)**

**Attività\_Sportiva (Nome, Max\_partecipanti, Descrizione, Luogo, Costo, Numero\_iscritti)**

**Lezione (ID, Durata, Giorno\_settimanale, Ora\_inizio, Istruttore\*, Nome\_campo\*, Nome\_sala\*, Attività\_sportiva\*)**

Con vincolo d'integrità referenziale tra:  
attributo Istruttore e la relazione Istruttore  
attributo Attività\_sportiva e la relazione Attività\_Sportiva  
attributo ID\_sala e la relazione Sala  
attributo ID\_campo e la relazione Campo

**Campo (Nome, Descrizione, Capienza, Costo)**

**Sala (Nome, Descrizione, Capienza, Tipo)**

**Spogliatoio (ID, Nome\_campo\*, Nome\_sala\*)**

Con vincolo d'integrità referenziale tra:  
attributo ID\_campo e la relazione Campo,  
attributo ID\_sala e la relazione Sala

**Abbonamento\_Actività (Nome attività\*, Cliente\*, Data abbonamento\*)**

Con vincolo d'integrità referenziale tra:

attributo Nome\_attività e la relazione Attività\_Sportiva

attributo Cliente, Data\_abbonamento e la relazione Abbonamento

**Istruttore\_Brevetto (Istruttore\*, Brevetto\*, Data emissione, Ente emissione)**

Con vincolo d'integrità referenziale tra:

attributo Istruttore e la relazione Istruttore

attributo Brevetto e la relazione Brevetto

### 3.3. Tabelle di trascrizione dei dati

Anagrafica			
Descrizione	Rappresenta la traduzione dell'entità Anagrafica		
Nome Attributo	Tipo dato	Vincolo	Descrizione
CF	CHAR(16)	PRIMARY KEY	Codice fiscale, identificativo, della persona
Nome	VARCHAR(60)	NOT NULL	Nome della persona
Cognome	VARCHAR(60)	NOT NULL	Cognome della persona
Luogo_nascita	VARCHAR(60)	NOT NULL	Luogo di nascita della persona
Data_nascita	DATE	NOT NULL	Data di nascita della persona
Genere	CHAR(1)	Value $\in$ {M,F,A} NOT NULL	Indica il genere della persona (Maschio, Femmina, Altro)

Personale_Amministrativo			
Descrizione	Rappresenta la traduzione dell'entità Personale_Amministrativo		
Nome Attributo	Tipo dato	Vincolo	Descrizione
CF	CHAR(16)	PRIMARY KEY, FOREIGN KEY	Codice fiscale, identificativo, del membro del Personale amministrativo, riferito alla chiave primaria di ANAGRAFICA

Medico			
Descrizione	Rappresenta la traduzione dell'entità Personale_Amministrativo		
Nome Attributo	Tipo dato	Vincolo	Descrizione
CF	CHAR(16)	PRIMARY KEY, FOREIGN KEY	Codice fiscale, identificativo, del Medico, riferito alla chiave primaria di ANAGRAFICA
Data_abilitazione	DATE	NOT NULL	Data conseguimento abilitazione del Medico
Data_laurea	DATE	NOT NULL	Data conseguimento laurea del Medico
Esercita	BIT	NOT NULL	Rappresenta l'attività del medico

Cliente			
Descrizione	Rappresenta la traduzione dell'entità Cliente		
Nome Attributo	Tipo dato	Vincolo	Descrizione
CF	CHAR(16)	PRIMARY KEY, FOREIGN KEY	Codice fiscale, identificativo, del Cliente, riferito alla chiave primaria di ANAGRAFICA



Visita_Medica			
Descrizione	Rappresenta la traduzione dell'entità Cliente		
Nome Attributo	Tipo dato	Vincolo	Descrizione
Cliente	CHAR(16)	PRIMARY KEY FOREIGN KEY	Codice fiscale, identificativo, del Cliente, riferito alla chiave primaria di CLIENTE
Medico	CHAR(16)	PRIMARY KEY FOREIGN KEY	Codice fiscale, identificativo, del Medico, riferito alla chiave primaria di MEDICO
Data	DATE	PRIMARY KEY	Data della visita
Durata_validità	INTEGER UNSIGNED	NOT NULL 30<= VALUE<=356	Durata di validità della visita, espressa in giorni
Validità	BIT	NOT NULL DEAFULT B'1'	Determina se la vitisa medica è ancora valida o no

Istruttore			
Descrizione	Rappresenta la traduzione dell'entità Istruttore		
Nome Attributo	Tipo dato	Vincolo	Descrizione
CF	CHAR(16)	PRIMARY KEY, FOREIGN KEY	Codice fiscale, identificativo, dell'Istruttore, riferito alla chiave primaria di ANAGRAFICA
Titolo	VARCHAR(22)	NOT NULL Value = 'DIPLOMA_ISEF' or Value = 'LAUREA_SCIENZE_MOTORIE'	Titolo conseguito dall'istruttore

Brevetto			
Descrizione	Rappresenta la traduzione dell'entità Spogliatoio		
Nome Attributo	Tipo dato	Vincolo	Descrizione
Cod	BIGINT(20) UNSIGNED	PRIMARY KEY	Codice identificativo del brevetto
Titolo	VARCHAR(100)	NOT NULL	Titolo del brevetto
Livello	VARCHAR(50)	NOT NULL	Indica il livello di validità, per esempio se Nazionale, Regionale ecc ecc
Descrizione	VARCHAR(100)	NESSUN VINCOLO	Descrizione facoltativa del brevetto
Sport_qualificato	VARCHAR(30)	NOT NULL	Sport abilitato dal Brevetto

Istruttore_Brevetto			
Descrizione	Rappresenta la traduzione dell'entità Spogliatoio		
Nome Attributo	Tipo dato	Vincolo	Descrizione
CF	CHAR(16)	PRIMARY KEY FOREIGN KEY	Codice fiscale, identificativo, dell'Istruttore, riferito alla chiave primaria di ISTRUTTORE
Cod	BIGINT UNSIGNED	PRIMARY KEY FOREIGN KEY	Codice identificativo del brevetto, riferito alla chiave primaria di BREVETTO
Data_emissione	VARCHAR(50)	NOT NULL	Indica la data in cui l'Istruttore consegue il Brevetto
Ente_emissione	VARCHAR(50)	NOT NULL	L'ente che emette il Brevetto

Campo			
Descrizione	Rappresenta la traduzione dell'entità Campo		
Nome Attributo	Tipo dato	Vincolo	Descrizione
Nome	VARCHAR(30)	PRIMARY KEY	Nome identificativo del campo
Descrizione	VARCHAR(100)	NESSUN VINCOLO	Descrizione facoltativa del campo
Capienza	INTEGER UNSIGNED	NOT NULL VALUE > 0	Capienza massima del campo
Costo	DOUBLE	NOT NULL VALUE >= 0	Costo del campo in caso di prenotazione

Sala			
Descrizione	Rappresenta la traduzione dell'entità Sala		
Nome Attributo	Tipo dato	Vincolo	Descrizione
Nome	VARCHAR(30)	PRIMARY KEY	Nome identificativo del campo
Descrizione	VARCHAR(100)	NESSUN VINCOLO	Descrizione facoltativa del campo
Capienza	INTEGER UNSIGNED	NOT NULL VALUE > 0	Capienza massima del campo
Tipo	VARCHAR(100)	NOT NULL	Tipologia della sala

Spogliatoio			
Descrizione	Rappresenta la traduzione dell'entità Spogliatoio		
Nome Attributo	Tipo dato	Vincolo	Descrizione
ID	BIGINT UNSIGNED	PRIMARY KEY AUTO INCREMENT	Id identificativo del campo
Nome_campo	VARCHAR(30)	FOREIGN KEY	Indica il campo a cui lo spogliatoio si può riferire. In caso faccia parte di una sala, il valore sarà NULL, riferito alla chiave primaria di CAMPO
Nome_sala	VARCHAR(30)	FOREIGN KEY	Indica la sala a cui lo spogliatoio si può riferire. In caso faccia parte di un campo, il valore sarà NULL, riferito alla chiave primaria di SALA

Prenotazione_Odierna			
Descrizione	Rappresenta la traduzione dell'entità Prenotazione		
Nome Attributo	Tipo dato	Vincolo	Descrizione
ID	BIGINT UNSIGNED	PRIMARY KEY, AUTO INCREMENT	Identificativo della prenotazione
Ora_inizio	TIME	NOT NULL	Orario d'inizio della prenotazione
Data	DATE	NOT NULL	Data della prenotazione
Costo	DOUBLE	NOT NULL VALUE >= 0	Costo della prenotazione
Durata	TIME	NOT NULL VALUE => 30	Durata prenotazione.
Cliente	CHAR(16)	NOT NULL FOREIGN KEY	Codice fiscale della persona che ha effettuato la prenotazione, riferita alla chiave primaria di CLIENTE
Campo	VARCHAR(30)	NOT NULL FOREIGN KEY	Nome del campo prenotato, riferito alla chiave primaria di CAMPO

Prenotazione_Storica			
Descrizione	Rappresenta la traduzione dell'entità Prenotazione		
Nome Attributo	Tipo dato	Vincolo	Descrizione
ID	BIGINT UNSIGNED	PRIMARY KEY, AUTO INCREMENT	Identificativo della prenotazione
Ora_inizio	TIME	NOT NULL	Orario d'inizio della prenotazione
Data	DATE	NOT NULL Almeno un anno precedente alla data odierna	Data della prenotazione
Costo	DOUBLE	NOT NULL VALUE >= 0	Costo della prenotazione
Durata	TIME	NOT NULL VALUE => 30	Durata prenotazione.
Cliente	CHAR(16)	NOT NULL FOREIGN KEY	Codice fiscale della persona che ha effettuato la prenotazione, riferita alla chiave primaria di CLIENTE
Campo	VARCHAR(30)	NOT NULL FOREIGN KEY	Nome del campo prenotato, riferito alla chiave primaria di CAMPO

Abbonamento			
Descrizione	Rappresenta la traduzione dell'entità Abbonamento		
Nome Attributo	Tipo dato	Vincolo	Descrizione
Data	DATE	PRIMARY KEY	Data della sottoscrizione dell'abbonamento
Cliente	CAHR(16)	PRIMARY KEY FOREIGN KEY	Cliente che ha sottoscritto l'abbonamento
Costo	DOUBLE	NOT NULL VALUE >= 0	Costo totale dell'abbonamento
Durata	INTEGER UNSIGNED	NOT NULL VALUE >= 1	Durata di validità dell'abbonamento espressa in giorni
Validità	BIT	NOT NULL	Indicatore di validità dell'abbonamento, scaduto o no

Attività Sportiva			
Descrizione	Rappresenta la traduzione dell'entità Attività_Sportiva		
Nome Attributo	Tipo dato	Vincolo	Descrizione
Nome	VARCHAR(30)	PRIMARY KEY	Nome univoco dell'attività sportiva
Max_partecipanti	INTEGER UNSIGNED	NOT NULL VALUE > 0	Numero massimo di persone che possono partecipare all'attività sportiva
Descrizione	VARCHAR(100)	NESSUN VINCOLO	Descrizione facoltativa dell'attività sportiva
Luogo	BIT	NOT NULL	0 = L'attività si svolge in una sala 1 = L'attività si svolge in un campo
Costo	DOUBLE	NOT NULL VALUE >= 0	Costo dell'attività sportiva necessaria al calcolo del costo degli abbonamenti
Numero_iscritti	INTEGER	NOT NULL VALUE >= 0	Numero attuale di iscritto all'attività sportiva, aggiornata ad ogni nuovo abbonamento, o ad ogni abbonamento scaduto.

Abbonamento_Actività			
Descrizione	Rappresenta la traduzione dell'entità Abbonamento_Actività		
Nome Attributo	Tipo dato	Vincolo	Descrizione
Data_abbonamento	DATE	PRIMARY KEY FOREIGN KEY	Data della sottoscrizione dell'abbonamento
Cliente	CAHR(16)	PRIMARY KEY FOREIGN KEY	Cliente che ha sottoscritto l'abbonamento
Nome_attività	VARCHAR(30)	PRIMARY KEY FOREIGN KEY	Nome univoco dell'attività sportiva

Lezione			
Descrizione	Rappresenta la traduzione dell'entità Abbonamento_Actività		
Nome Attributo	Tipo dato	Vincolo	Descrizione
ID	INTEGER UNSIGNED	PRIMARY KEY, AUTO INCREMENT	Identifativo della lezione
Durata	TIME	NOT NULL VALUE => 30	Durata Lezione.
Giorno_settimanale	INTEGER(1)	Non nulla valore $\in$ { 1 = domenica, 2 = lunedì, 3 = martedì, 4 = mercoledì, 5 = giovedì, 6 = venerdì, 7 = sabato }(per corrispondere alla funzione daysofweek())	Indica il giorno della settimana in cui la lezione si svolge
Ora_inizio	TIME	NOT NULL	Orario d'inizio della lezione
Istruttore	CHAR(16)	NOT NULL FOREIGN KEY	Istruttore che supervisiona la lezione, riferita alla chiave primaria di Istruttore
Nome_campo	VARCHAR(30)	FOREIGN KEY	Nome identificativo del campo usato dalla lezione, riferita alla chiave primaria di CAMPO, può essere null
Nome_sala	VARCHAR(30)	FOREIGN KEY	Nome identificativo della sala usata dalla lezione, riferita alla chiave primaria di SALA, può essere null
Attività_sportiva	VARCHAR(30)	NOT NULL FOREIGN KEY	Nome univoco dell'attività sportiva praticata durante la lezione, riferita alla chiave primari di ATTIVITA_SPORTIVA

## 4. PROGETTAZIONE FISICA

### 4.1. Definizione dello schema della base di dati

```
create database FitnessClub;
```

```
use FitnessClub;
```

```
CREATE TABLE anagrafica (  
    CF char(16) NOT NULL,  
    Nome varchar(60) NOT NULL,  
    Cognome varchar(60) NOT NULL,  
    Luogo_nascita varchar(60) NOT NULL,  
    Data_nascita date NOT NULL,  
    Genere char(1) NOT NULL,  
    PRIMARY KEY (CF),  
    CONSTRAINT `check_valore_genere` CHECK (Genere in ('M','A','F'))  
);
```

```
CREATE TABLE cliente (  
    CF char(16) NOT NULL,  
    PRIMARY KEY (CF),  
    FOREIGN KEY (CF) REFERENCES anagrafica (CF) ON DELETE CASCADE  
);
```

```
CREATE TABLE istruttore (  
    CF char(16) NOT NULL,  
    Titolo varchar(22) NOT NULL,  
    PRIMARY KEY (CF),  
    FOREIGN KEY (CF) REFERENCES anagrafica (CF) ON DELETE CASCADE,  
    CONSTRAINT `check_tipo_titolo` CHECK (  
        Titolo = 'DIPLOMA_ISEF' or Titolo = 'LAUREA_SCIENZE_MOTORIE'  
    )  
);
```



```
CREATE TABLE medico (  
  CF char(16) NOT NULL,  
  Data_abilitazione date NOT NULL,  
  Data_laurea date NOT NULL,  
  Esercita bit(1) NOT NULL DEFAULT b'1',  
  PRIMARY KEY (CF),  
  FOREIGN KEY (CF) REFERENCES anagrafica (CF) ON DELETE CASCADE  
);
```

```
CREATE TABLE personale_amministrativo (  
  CF char(16) NOT NULL,  
  PRIMARY KEY (CF),  
  FOREIGN KEY (CF) REFERENCES anagrafica (CF) ON DELETE CASCADE  
);
```

```
CREATE TABLE abbonamento (  
  Data date NOT NULL,  
  Cliente char(16) NOT NULL,  
  Costo double NOT NULL,  
  Durata_validita int(4) NOT NULL,  
  Validita bit(1) NOT NULL,  
  PRIMARY KEY (Data, Cliente),  
  FOREIGN KEY (Cliente) REFERENCES cliente (CF),  
  CONSTRAINT `check_valore_costo_abbonamento` CHECK (Costo >= 0),  
  CONSTRAINT `check_durata_validita_abbonamento` CHECK (Durata_validita >= 1)  
);
```

```
CREATE TABLE attivita_sportiva (  
  Nome varchar(30) NOT NULL,  
  Max_partecipanti int(3) unsigned NOT NULL,  
  Descrizione varchar(100) DEFAULT NULL,  
  Luogo bit(1) NOT NULL,  
  Costo double NOT NULL,  
  Numero_iscritti int(3) NOT NULL,  
  PRIMARY KEY (Nome),  
  CONSTRAINT check_max_partecipanti CHECK (Max_partecipanti > 0),  
  CONSTRAINT `check_valore_costo_attivita_sportiva` CHECK (Costo >= 0),  
  CONSTRAINT `check_numero_iscritti` CHECK (Numero_iscritti >= 0),  
  CONSTRAINT `check_luogo` CHECK (Luogo = 1 or Luogo = 0)  
);
```

```
CREATE TABLE abbonamento_attivita (  
  Data_abbonamento date NOT NULL,  
  Cliente char(16) NOT NULL,  
  Nome_attivita varchar(30) NOT NULL,  
  PRIMARY KEY (Data_abbonamento, Cliente, Nome_attivita),  
  FOREIGN KEY (Cliente) REFERENCES abbonamento (Cliente),  
  FOREIGN KEY (Data_abbonamento) REFERENCES abbonamento (Data),  
  FOREIGN KEY (Nome_attivita) REFERENCES attivita_sportiva (Nome)  
);
```

```

CREATE TABLE brevetto (
  Cod bigint(20) unsigned NOT NULL
  AutoINCREMENT
  ,Titolo varchar(100) NOT NULL,
  Livello varchar(50) NOT NULL,
  Descrizione varchar(100) DEFAULT NULL,
  Sport_qualificato varchar(30) NOT NULL,
  PRIMARY KEY (Cod)
);

```

```

CREATE TABLE istruttore_brevetto (
  CF char(16) NOT NULL,
  Cod bigint(20) unsigned NOT NULL,
  Data_emissione date NOT NULL,
  Ente_emissione varchar(50) NOT NULL,
  PRIMARY KEY (CF,Cod),
  FOREIGN KEY (CF) REFERENCES istruttore (CF),
  FOREIGN KEY (Cod) REFERENCES brevetto (Cod)
);

```

```

CREATE TABLE campo (
  Nome varchar(30) NOT NULL,
  Descrizione varchar(100) DEFAULT NULL,
  Capienza int(10) unsigned NOT NULL,
  Costo double NOT NULL,
  PRIMARY KEY (Nome),
  CONSTRAINT `check_valore_capienza_campo` CHECK (Capienza > 0),
  CONSTRAINT `check_valore_costo_campo` CHECK (Costo >= 0)
);

```

```

CREATE TABLE sala (
  Nome varchar(30) NOT NULL,
  Descrizione varchar(100) DEFAULT NULL,
  Capienza int(10) unsigned NOT NULL,
  Tipo varchar(100) NOT NULL,
  PRIMARY KEY (Nome),
  CONSTRAINT `check_valore_capienza_sala` CHECK (Capienza > 0)
);

```

```

CREATE TABLE spogliatoio (
  ID bigint(10) NOT NULL AUTO_INCREMENT,
  Nome_campo varchar(30),
  Nome_sala varchar(30),
  PRIMARY KEY (ID),
  FOREIGN KEY (Nome_campo) REFERENCES campo (Nome),
  FOREIGN KEY (Nome_sala) REFERENCES sala (Nome)
);

```

```

CREATE TABLE lezione (
  ID bigint(10) NOT NULL AUTO_INCREMENT,
  Durata time NOT NULL,
  Giorno_settimanale integer(1) NOT NULL,
  Ora_inizio time NOT NULL,
  Nome_campo varchar(30) DEFAULT NULL,
  Nome_sala varchar(30) DEFAULT NULL,
  Attivita_sportiva varchar(30) NOT NULL,
  Istruttore char(16) DEFAULT NULL,
  PRIMARY KEY (ID),
  FOREIGN KEY (Nome_campo) REFERENCES campo (Nome),
  FOREIGN KEY (Nome_sala) REFERENCES sala (Nome),
  FOREIGN KEY (Attivita_sportiva) REFERENCES attivita_sportiva (Nome),
  FOREIGN KEY (Istruttore) REFERENCES istruttore (CF),
  CONSTRAINT `check_valore_durata` CHECK (Durata >= 30),
  CONSTRAINT `check_giorno_settimanale` CHECK (
    Giorno_settimanale in (1,2,3,4,5,6,7)
  )
);

```

```

CREATE TABLE visita_medica (
  Cliente char(16) NOT NULL,
  Medico char(16) NOT NULL,
  Data date NOT NULL,
  Durata_validita int(10) unsigned NOT NULL,
  Validita bit(1) NOT NULL DEFAULT b'1',
  PRIMARY KEY (Cliente,Medico,Data),
  FOREIGN KEY (Cliente) REFERENCES cliente (CF) ON DELETE CASCADE,
  FOREIGN KEY (Medico) REFERENCES medico (CF) ON DELETE CASCADE,
  CONSTRAINT `check_durata_validita_visita_medica` CHECK (
    Durata_validita >= 30 and Durata_validita <= 365
  )
);

```

```

CREATE TABLE prenotazione_odierna (
  ID bigint(10) NOT NULL AUTO_INCREMENT,
  Ora_inizio time NOT NULL,
  Data date NOT NULL,
  Costo double NOT NULL,
  Durata time NOT NULL,
  Cliente char(16) NOT NULL,
  Campo varchar(30) NOT NULL,
  PRIMARY KEY (ID),
  FOREIGN KEY (Cliente) REFERENCES cliente (CF),
  FOREIGN KEY (Campo) REFERENCES campo (Nome),
  CONSTRAINT `check_valore_costo_prenotazione_odierna` CHECK (Costo >= 0),
  CONSTRAINT `check_valore_durata_prenotazione_odierna` CHECK (Durata >= 30)
);

```

```
CREATE TABLE prenotazione_storica (  
  ID bigint(10) NOT NULL AUTO_INCREMENT,  
  Ora_inizio time NOT NULL,  
  Data date NOT NULL,  
  Costo double NOT NULL,  
  Durata time NOT NULL,  
  Cliente char(16) NOT NULL,  
  Campo varchar(30) NOT NULL,  
  PRIMARY KEY (ID),  
  FOREIGN KEY (Cliente) REFERENCES cliente (CF),  
  FOREIGN KEY (Campo) REFERENCES campo (Nome),  
  CONSTRAINT `check_valore_costo_prenotazione_storica` CHECK (Costo >= 0),  
  CONSTRAINT `check_valore_durata_prenotazione_storica` CHECK (Durata >= 30)  
);
```

## 4.2. Definizione delle interrogazioni per la visualizzazione dei dati

### 1. *Inserire tutti i dati relativi ad un nuovo cliente*

**DELIMITER \$\$**

```
CREATE PROCEDURE `CaricaClienti`(in CF_CLIENTE char(16),  
                                in NOME varchar(60),  
                                in COGNOME varchar(60),  
                                in LUOGO_NASCITA varchar(60),  
                                in DATA_NASCITA date,  
                                in GENERE char(1)
```

**)**

**begin**

**if not exists** (select \* from anagrafica where CF=CF\_Cliente)

**THEN insert into anagrafica** ( CF, Nome, Cognome, Luogo\_nascita,Data\_nascita, Genere)  
**values** ( CF\_CLIENTE, NOME, COGNOME, LUOGO\_NASCITA,DATA\_NASCITA, GENERE);

**end if;**

**insert into cliente** (CF) **values** (CF\_CLIENTE);

**end\$\$**

**DELIMITER ;**

## 2. Modificare l'orario di una lezione

```
DELIMITER $$
CREATE PROCEDURE `ModificaOrarioLezione` (
    id_lezione bigint, new_time time
)
BEGIN
    if not exists (select * from lezione where ID = id_lezione) then
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'La lezione che si vuole modificare non esiste';
    else
        set @giorno_sett = (select Giorno_settimanale from lezione where ID = id_lezione);
        set @durata_lezione = (select Durata from lezione where ID=id_lezione);
        set @ora_fine = addtime(new_time, @durata_lezione);

        if (select Nome_campo from lezione where ID = id_lezione) is not null then
            set @nome_luogo = (select Nome_campo from lezione where ID = id_lezione);
        end if;

        if not exists (
            select * from lezione
            where ID!=id_lezione and
            (Giorno_settimanale = @giorno_sett) and (
                (new_time < Ora_inizio and Ora_inizio < @ora_fine) or
                (new_time < addtime(Ora_inizio, Durata) and
                    addtime(Ora_inizio, Durata) < @ora_fine ) or
                (Ora_inizio < new_time and @ora_fine < addtime(Ora_inizio, Durata)) or
                (new_time < Ora_inizio and addtime(Ora_inizio, Durata) < @ora_fine)
            )
        ) and not exists (
            select * from Prenotazione_odierna po
            where dayofweek(po.Data) = @giorno_sett and (
                (new_time < Ora_inizio and Ora_inizio < @ora_fine) or
                (new_time < addtime(Ora_inizio, Durata) and
                    addtime(Ora_inizio, Durata) < @ora_fine ) or
                (Ora_inizio < new_time and @ora_fine < addtime(Ora_inizio, Durata)) or
                (new_time < Ora_inizio and addtime(Ora_inizio, Durata) < @ora_fine)
            ) and (Campo = @nome_luogo)
        ) then
            update lezione set Ora_inizio = new_time where ID=id_lezione;
        else
            SIGNAL SQLSTATE '45000'
            SET MESSAGE_TEXT = 'Questo orario non è disponibile';
        end if;
    end if;
END$$

DELIMITER ;
```

### 3. *Eliminare i dati di un determinato medico*

*Avendo creato:*

```
CREATE TRIGGER elimina_medico  
AFTER DELETE ON medico  
FOR EACH ROW  
DELETE FROM anagrafica WHERE old .CF=CF
```

*delete from medico where CF=CF;*

**#NB:** *non crea problemi con visita\_medica, poiché la foreign key che è collegata al CF del medico ha l'ON DELETE CASCADE*

### 4. *Visualizzare tutte le lezioni tenuta da un determinato istruttore, in un dato giorno della settimana*

```
DELIMITER $$  
CREATE PROCEDURE `LezioneIstruttoreGiorno`  
cf_istruttore char(16),  
giorno int(1)  
)  
begin  
select * from lezione  
where Giorno_settimanale = giorno and  
istruttore=cf_istruttore;  
END$$  
DELIMITER ;
```

### 5. *Visualizzare il numero di abbonamenti sottoscritti da ciascun cliente.*

```
CREATE VIEW NumeroAbbonamentiPerCliente AS  
select count(*) as NumeroAbbonamenti, Cliente  
from abbonamento  
group by Cliente;
```

*select \* from NumeroAbbonamentiPerCliente;*

6. **Visualizzare il cliente che ha sottoscritto l'abbonamento con il numero massimo di corsi.**

```
select Data_abbonamento, Cliente
from abbonamento_attivita aa
group by Data_abbonamento, Cliente
having count(*) = (
    select Max(numeroattivita)
    from (
        select count(*) as NumeroAttivita, Cliente
        from abbonamento_attivita
        group by Data_abbonamento, Cliente
    ) as N_Activita
);
```

7. **Visualizzare il cliente che ha prenotato lo stesso campetto due volte in una settimana.**

```
select Cliente, Settimana_prenotata
from (
    select count(*) as Conta_prenotazioni, yearweek(Data) as Settimana_prenotata, Cliente,
    Campo
    from prenotazione_odierna as po
    group by yearweek(Data), Campo
) as TP
where Conta_prenotazioni = 2;
```

8. **Visualizzare la sala in cui si svolge il minor numero di attività sportive.**

```
select l1.Nome_sala
from lezione l1
group by l1.Nome_sala
having count(*) <= all (
    select count(*)
    from lezione l2
    where l2.Nome_sala is not null
    group by l2.Nome_sala
);
```



**9. Visualizzare gli istruttori che hanno solo il brevetto di nuoto.**

```
(      select CF
      from istruttore_brevetto ib, brevetto b
      where b.Cod = ib.Cod and
            b.Sport_qualificato = 'Nuoto'
) except (
      select CF
      from istruttore_brevetto ib, brevetto b
      where b.Cod = ib.Cod and
            b.Sport_qualificato != 'Nuoto'
)
```

**Oppure possiamo usare la procedure per visualizzare qualsiasi determinato tipo di brevetto**

```
          DELIMITER $$
CREATE PROCEDURE `IstruttoreconBrevetto`(
Sport_qualificato varchar(30)
)
begin

select CF
from brevettiistruttore as b
where Sport_qualificato = b.Sport_qualificato;

END$$
DELIMITER ;
```

**10. Visualizzare i medici che sono anche clienti del centro ma non seguono attività che si svolgono in un determinato orario.**

```
CREATE PROCEDURE
MediciClienti_no_attivita(searched_
time time)
select Cliente
from prenotazione_odierna as po ,
Medico as m
where po.Cliente= m.CF and
Ora_inizio != searched_time;
```

**//non seguono quindi mettiamo !=**

11. **Controllare che le lezioni per attività sportive svolte in sala abbiano come struttura di svolgimento la sala e idem per il campo.**

```
delimiter $$
CREATE TRIGGER check_luogo_lezione
BEFORE INSERT ON lezione
FOR EACH ROW
BEGIN
    declare tipo BIT;

    if (new.Nome_campo is null and new.Nome_sala is null) then
        signal sqlstate '45000' set message_text = "Attenzione, alla Lezione deve essere
        associata almeno una tra sala e campo";
    end if;

    if (new.Nome_campo is not null and new.Nome_sala is not null) then
        signal sqlstate '45000' set message_text = "Attenzione, alla Lezione deve essere
        associata solo una sala o solo un campo";
    end if;

    set tipo = (
        select Luogo from attivita_sportiva as ats
        where new.Attivita_sportiva = ats.Nome
    );

    if (tipo = b'0' and new.Nome_sala is null) then
        signal sqlstate '45000' set message_text = "Questa lezione non può essere
        effettuata in un campo, reinserire.";
    end if;

    if (tipo = b'1' and new.Nome_campo is null) then
        signal sqlstate '45000' set message_text = "Questa lezione non può essere
        effettuata in una sala, reinserire.";
    end if;
end$$
delimiter ;
```

12. **Snellire le prenotazioni spostando le vecchie di oltre un anno in un'entità separata (Prenotazioni\_Storiche). Fare questa operazione ogni fine anno.**

```
delimiter $$
CREATE EVENT inseriscilnPrenotazioniStoriche
ON SCHEDULE EVERY 1 YEAR
ON COMPLETION PRESERVE
ENABLE
DO BEGIN
    insert into prenotazione_storica (
        select *
        from prenotazione_odierna as p
        where datediff(curdate(), data) > 365
    );

    DELETE FROM prenotazione_odierna p
    WHERE p.ID IN (
        SELECT ps.ID
        FROM prenotazione_storica as ps
    );
END$$
delimiter ;
```

13. **(Da traccia) Si vuole conoscere tutti i brevetti posseduti da un istruttore e la data di conseguimento degli stessi**

```
delimiter $$
CREATE PROCEDURE BrevettiIstruttore(cf_istruttore char(16))
begin
    select ib.Cod, b.Titolo
    from brevetto b, istruttore_brevetto ib
    where ib.CF = cf_istruttore and
    ib.Cod = b.Cod;
END$$
delimiter ;
```

**EXTRA**

## **1. Procedura Caricamento Istruttore**

**DELIMITER \$\$**

**CREATE DEFINER='root'@'localhost' PROCEDURE `Caricalistruttore`**

**in CF\_ISTRUTTORE char(16),**

**in NOME varchar(60),**

**in COGNOME varchar(60),**

**in LUOGO\_NASCITA varchar(60),**

**in DATA\_NASCITA date,**

**in GENERE char(1),**

**in TITOLO char(22)**

**)**

**begin**

**if not exists (select \* from anagrafica where**

**CF=CF\_ISTRUTTORE) THENinsert into anagrafica**

**(CF, Nome, Cognome, Luogo\_nascita, Data\_nascita,**

**Genere**

**) values ( CF\_ISTRUTTORE, NOME, COGNOME, LUOGO\_NASCITA,**

**DATA\_NASCITA, GENERE**

**);**

**end if;**

**if (TITOLO <> 'DIPLOMA\_ISEF' and TITOLO <> 'LAUREA\_SCIENZE\_MOTORIE') then**

**SIGNAL SQLSTATE '45000'**

**SET MESSAGE\_TEXT = 'Titolo inserito non valido';**

**else**

**insert into istruttore (CF, Titolo) values (CF\_ISTRUTTORE, TITOLO);**

**end if;**

**end\$\$**

**DELIMITER ;**

## 2. Procedura caricamento Medico

**DELIMITER \$\$**

**CREATE DEFINER='root'@'localhost' PROCEDURE `CaricaMedico` (**

**in CF\_MEDICO char(16),**

**in NOME varchar(60),**

**in COGNOME varchar(60),**

**in LUOGO\_NASCITA varchar(60),**

**in DATA\_NASCITA date,**

**in GENERE char(1),**

**in DATA\_ABILITAZIONE date,**

**in DATA\_LAUREA date**

**)**

**begin**

**if not exists (select \* from anagrafica where CF=CF\_MEDICO)**

**THEN insert into anagrafica ( CF, Nome, Cognome,**

**Luogo\_nascita,Data\_nascita, Genere**

**) values ( CF\_MEDICO, NOME, COGNOME, LUOGO\_NASCITA,DATA\_NASCITA,**

**GENERE**

**);**

**end if;**

**insert into medico (CF, Data\_abilitazione, Data\_laurea) values (CF\_MEDICO,**

**DATA\_ABILITAZIONE, DATA\_LAUREA);**

**end\$\$**

**DELIMITER ;**

- 3. Daily Event. Controlla giornalmente che non siano scaduti abbonamenti o visite mediche, in caso contrario modificherà la flag Validità all'interno delle rispettive occorrenze.**

```
delimiter $$
create event DailyCheck
on schedule every 1 day
on completion preserve
enable
do begin
    update abbonamento
    set Validita = 0
    where Data in (
        select Data from (select * from abbonamenti_scaduti) as abs
    );

    update visita_medica
    set Validita = 0
    where Data in (
        select Data from (select * from visite_mediche_scadute) as vm
    );
end $$
DELIMITER ;
```

- 4. Trigger che aumenta il numero di iscritti delle attività quando viene fatto un nuovo abbonamento**

```
delimiter $$
create trigger handler_nuovo_abbonamento
before INSERT on abbonamento_attivita
for each row
begin
    update attivita_sportiva as att
    set Numero_iscritti = Numero_iscritti + 1
    where new.Nome_attivita = att.Nome;
end $$
DELIMITER ;
```

- 5. View che visualizza tutti i brevetti degli Istruttore**

```
CREATE VIEW brevettIstruttore AS
select i.CF, b.Sport_qualificato
from istruttore_brevetto as i , brevetto as b
where i.Cod=b.Cod
```