

Fitness Club

Università della Calabria
Dip. di Matematica e Informatica

Corso di laurea in informatica
a.a. 2020/2021

Progetto per il Corso di
Basi di dati

Gruppo <13>

<209466, Aiello, Simone>

<209429, Camarda, Alessia Donata>

<213533, Copelli, Francesco>

<213536, Critelli, Anna>

<210438, Yoshovski, Stefan Antonov>

Docente

Prof. P. Rullo

Esercitatori

Ing. G. Labocchetta

Ing. Marco Mastratise

1 Tematica Progettuale

Si vuole progettare un sistema informativo per la gestione di un Fitness Club.

2 Raccolta e Analisi dei Requisiti

	Requisiti Richiesti
1 2 3	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.
4 5	Un medico è caratterizzato inoltre dalla data di conseguimento della abilitazione alla professione e dalla data di conseguimento del titolo.
6 7 8 9 10 11	Un istruttore può avere il titolo di diploma ISEF o di laurea in Scienze Motorie 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 i brevetti posseduti da un istruttore e la data di conseguimento degli stessi.
12 13	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.
14 15 16 17	Un abbonamento è caratterizzato da una data di inizio e da una durata di validità ed 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.
18 19 20 21	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.
22 23 24 25 26	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. 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.
27 28 29 30 31	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). 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.
32 33 34 35	Per ogni lezione, si vuole memorizzare la sala o il campetto in cui si svolge e l'istruttore che la supervisiona. Un cliente, inoltre, può prenotare un campetto (negli orari in cui questo non è occupato per le lezioni). A tal fine, è di interesse conoscere anche il costo per la prenotazione di ogni campetto da gioco.

2.2 *Analisi dei requisiti*

2.2.1 Eliminazione delle Ambiguità

Linea	Termine	Sinonimi	Motivazione Correzione
1 12 24	Fitness Club	Centro di fitness	Uguale significato
20	Fitness Club	Centro	Uguale significato
28	Fitness Club	Fitness Center	Uguale significato
12	Clienti	Soci	Uguale significato
26	Campi da gioco	Campetti	Uguale significato
16	Campi da gioco	Campi	Uguale significato

2.2.2 Ristrutturazione dei Requisiti Richiesti

	Specifiche ristrutturate
1 2 3	Per il fitness club 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.
4 5	Un medico, oltre agli attributi specifici dello staff, è caratterizzato dalla data di conseguimento della abilitazione alla professione e dalla data di conseguimento del titolo.
6 7 8 9 10 11	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 i brevetti posseduti da un istruttore e la data di conseguimento degli stessi.
12 13	Un cliente del fitness club è caratterizzato da nome, cognome, sesso, luogo e data di nascita.
14 15 16	Un abbonamento è caratterizzato da una data di inizio e da una durata di validità e dà accesso al cliente a una o più attività sportive. Il costo dell'abbonamento dipende dal costo delle singole attività sportive scelte dal cliente.
17 18 19 20	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 fitness club, sia al momento dell'iscrizione che del rinnovo dell'abbonamento.
21 22 23 24 25 26	Un'attività sportiva è caratterizzata da un nome univoco e da una descrizione e può essere svolta o in sala o in un campo da gioco. Si vuole memorizzare, inoltre, il numero massimo di partecipanti e il numero di iscritti. Il fitness club 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.
27 28	Per sale e per campi da gioco, si vuole memorizzare il nome, univoco all'interno del fitness club, una descrizione e la capienza (espressa in numero di persone).
29 30 31	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.
32 33 34 35	Per ogni lezione, si vuole memorizzare la sala o il campo da gioco in cui si svolge e l'istruttore che la supervisiona. Un cliente, inoltre, può prenotare un campo da gioco (negli orari in cui questo non è occupato per le lezioni). A tal fine, è di interesse conoscere anche il costo per la prenotazione di ogni campo da gioco.

2.2.3 Raffinamento delle Specifiche e Individuazione dei Concetti di Base

	Frasi di carattere generale
23	Il fitness club 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.

	Frasi relative allo staff
1	Lo staff è costituito da: <ul style="list-style-type: none">- Personale amministrativo- Medici- Istruttori
2 3	Ogni membro dello staff è caratterizzato da: <ul style="list-style-type: none">- Nome- Cognome- Sesso- Luogo di Nascita- Data di Nascita

	Frasi relative ai medici
4	Un medico oltre agli attributi dello staff è caratterizzato da: <ul style="list-style-type: none">- Data di conseguimento dell'abilitazione alla professione- Data di conseguimento di tale titolo
18	Un medico effettua una visita medica ai clienti del fitness al momento dell'iscrizione e ad ogni rinnovo dell'abbonamento.

	Frasi relative all'istruttore
6	Un istruttore oltre agli attributi dello staff deve avere: <ul style="list-style-type: none">- Titolo di diploma ISEF o di Laurea in Scienze Motorie- Almeno un brevetto

	Frasi relative al brevetto
10	Un brevetto è caratterizzato da: <ul style="list-style-type: none">- Titolo- Livello- Descrizione- Ente Sportivo che lo ha rilasciato- Qualifica per l'insegnamento di un'attività sportiva- Data di Conseguimento

	Frase relative al Cliente
12	Un Cliente è caratterizzato da: <ul style="list-style-type: none"> - Nome - Cognome - Sesso - Luogo di Nascita - Data di Nascita
18	Un cliente sottoscrive un abbonamento, alla cui scadenza può essere rinnovato.
18	Un cliente si sottopone a visita medica sia al momento dell'iscrizione che del rinnovo dell'abbonamento.
33	Un cliente, inoltre, può prenotare un campo da gioco (negli orari in cui questo non è occupato per le lezioni).

	Frase relative all'abbonamento
14	Un abbonamento è caratterizzato da: <ul style="list-style-type: none"> - Data di inizio - Durata di validità - Abilitazione allo svolgimento di alcune Attività Sportive - Costo dipendente dalle singole attività sportive scelte

	Frase relative alle attività
21	Un'attività sportiva è caratterizzata da: <ul style="list-style-type: none"> - Nome univoco - Descrizione - Numero massimo di partecipanti - Numero di iscritti
15	Le attività sportive possono essere svolte in sala o in un campo.
29	Per tutti i tipi di attività sportiva sono previste una o più lezioni settimanali.

	Frase relative ai Campi da Gioco
27	Un campo da gioco è caratterizzato da: <ul style="list-style-type: none"> - Nome - Descrizione - Capienza (espressa in numero di persone) - Costo prenotazione
33	Un campo da gioco può essere prenotato da un cliente (negli orari in cui questo non è occupato per le lezioni).

	Frase relative alle Sale
27	Una Sala è caratterizzata da: <ul style="list-style-type: none"> - Nome - Descrizione - Tipologia - Capienza (espressa in numero di persone)

	Frase relative alle Lezioni
29	Le lezioni sono caratterizzate da: <ul style="list-style-type: none"> - Giorno - Orario di inizio - Durata (espressa in minuti)
32	Per ogni lezione, si vuole memorizzare la sala o il campetto in cui si svolge e l'istruttore che la supervisiona.

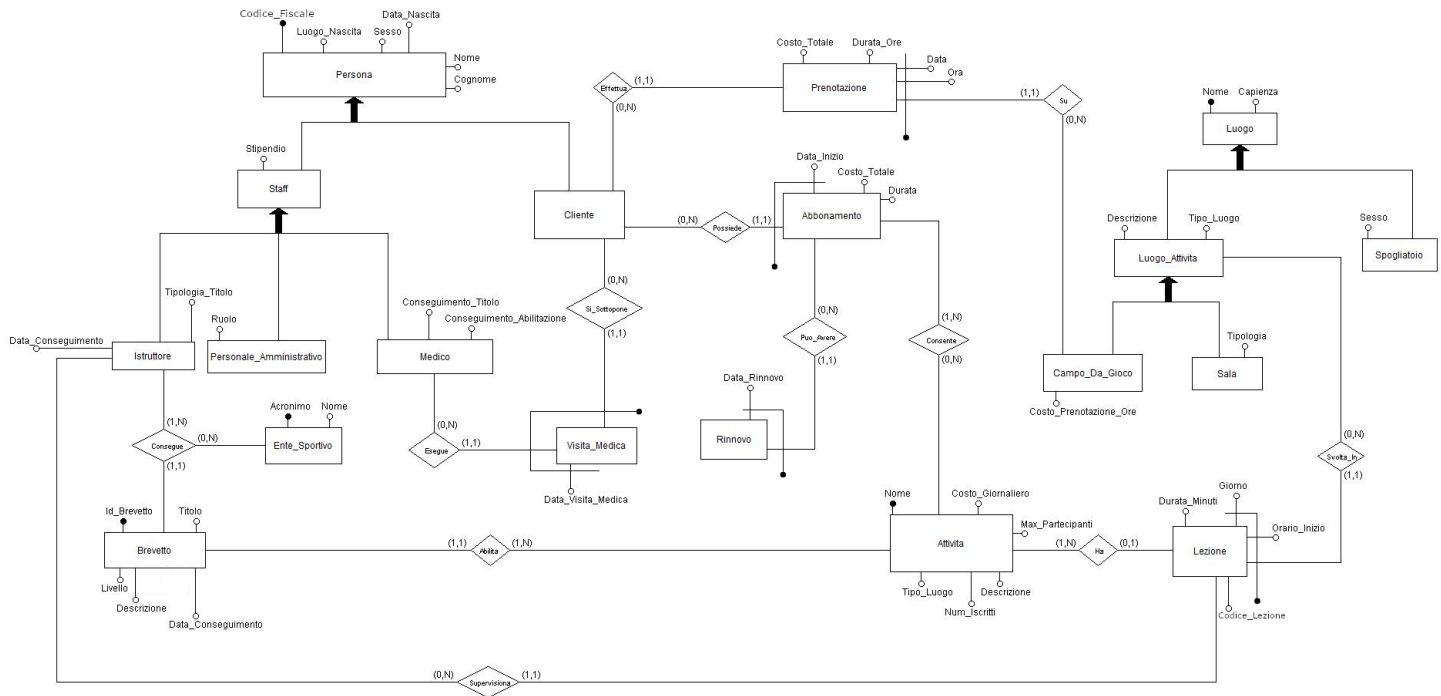
2.3 *Specifica delle Operazioni sui dati previste*

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 tutti 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 dato 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 una entità separata (PRENOTAZIONI_STORICHE). Fare questa operazione ogni fine anno

3 Progettazione Concettuale

3.1 Schemi E-R

Figura 1- Schema E-R



3.2 Documentazione dello Schema E-R

3.2.1 Dizionario dei Dati

Entità	descrizione	attributi	identificatore
Persona	Individuo che può accedere al Fitness club.	Codice_Fiscale Nome Cognome Sesso Luogo_Nascita Data_Nascita	Codice_Fiscale
Staff	Personale del Fitness club addetto ai lavori.	(Attributi derivati da persona) Codice_Fiscale Stipendio	Codice_Fiscale
Cliente	Persona che può accedere alle attività del Fitness club.	(Attributi derivati da persona)	Codice_Fiscale
Istruttore	Persona che insegna e supervisiona le attività del Fitness club.	(Attributi derivati da staff) Tipologia_titolo Data_conseguimento	Codice_Fiscale
Personale_Amministrativo	Persona che cura i lati economici e amministrativi del Fitness club.	(Attributi derivati da staff) Ruolo	Codice_Fiscale
Medico	Persona abilitata all'effettuazione delle visite sui clienti e al rilascio dei certificati.	(Attributi derivati da staff) Conseguimento_abilitazione Conseguimento_titolo	Codice_Fiscale
Brevetto	Attestato ufficiale della capacità di un individuo a esercitare determinate mansioni specifiche.	Id_brevetto Titolo Livello Descrizione Data_conseguimento	Id_brevetto
Ente_Sportivo	Un ente pubblico o privato abilitato a rilasciare brevetti.	Acronimo Nome	Acronimo

Visita_Medica	Visita medica che consente di svolgere le attività sportive.	Data_Visita_Medica	Chiave esterna mista (Data_Visita_Medica, chiave primaria di Medico e chiave primaria di Cliente)
Prenotazione	Pagamento per aver riservato un campo in una data e ora specifici.	Data Ora Durata_ore Costo_Totale	Chiave esterna mista (Data, ora e chiave primaria di Campo)
Abbonamento	Contratto per cui, pagando una determinata somma, si può, per un tempo determinato, usufruire dei servizi del Fitness Club.	Data_inizio Costo_Totale Durata_giorni	Data_inizio, Codice_Fiscale
Rinnovo	Rinnovo di un abbonamento	Data_rinnovo	Chiave esterna mista (Data_rinnovo e chiave primaria di abbonamento)
Attivita	Attività fisica pianificata ed eseguita regolarmente.	Nome Descrizione Max_Partecipanti Num_iscritti Costo_giornaliero Tipo_luogo	Nome
Lezione	Incontri settimanali tra istruttori e clienti per svolgere un'attività	Giorno Orario_Inizio Durata_minuti CodiceLezione	Chiave esterna mista (Giorno, Orario_Inizio e chiave primaria di Luogo)
Luogo	Ubicazione fisica degli spazi all'interno del Fitness Club.	Nome Capienza	Nome
Spogliatoio	Luogo destinato alla deposizione o al cambio degli indumenti.	(Attributi derivati da Luogo) Nome Sesso	Nome

Luogo_Activita	Luogo destinato allo svolgimento delle lezioni del Fitness Club.	(Attributi derivati da Luogo) Nome Descrizione Tipo_luogo	Nome
Sala	Un luogo attrezzato per praticare sport al chiuso (indoor).	(Attributi derivati da Nome Luogo_Activita) Tipologia	Nome
Campo_Da_Gioco	Luogo attrezzato per praticare sport all'esterno (outdoor) del Fitness Club.	(Attributi derivati da Luogo_Activita) Nome Costo_prenotazione_ore	Nome

3.2.2 Descrizione Entità

Persona			
Descrizione	Individuo che può accedere al Fitness Club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Codice_Fiscale	Alfanumerico	Chiave primaria	Identificatore univoco della persona
Nome	Alfabetico	Obbligatorio	Nome anagrafico.
Cognome	Alfabetico	Obbligatorio	Cognome anagrafico.
Sesso	Alfabetico	Obbligatorio	Sesso biologico
Luogo_nascita	Alfabetico	Obbligatorio	Luogo di nascita anagrafico
Data_nascita	Data	Obbligatorio	Data di nascita anagrafica

Staff			
Descrizione	Personale del Fitness club addetto ai lavori.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Codice_Fiscale	Alfanumerico	(Attributo derivato da Persona) Chiave primaria	Identificatore univoco della persona
Stipendio	Numerico	Obbligatorio	Compenso monetario erogato allo staff in base alla mansione svolta.

Cliente			
Descrizione	Persona che può accedere alle attività del Fitness club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Codice_Fiscale	Alfanumerico	(Attributo derivato da Persona) Chiave primaria	Identificatore univoco della persona

Istruttore			
Descrizione	Persona che insegna e supervisiona le attività del Fitness club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Codice_Fiscale	Alfanumerico	(Attributo derivato da Persona) Chiave primaria	Identificatore univoco della persona
Tipologia_titolo	Alfabetico	Obbligatorio	Titolo conseguito di tipo ISEF o laurea
Data_conseguimento	Data	Opzionale	Data di conseguimento del titolo ISEF o laurea

Personale_Amministrativo			
Descrizione	Persona abilitata all'effettuazione delle visite sui clienti e al rilascio dei certificati.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Codice_Fiscale	Alfanumerico	(Attributo derivato da Persona) Chiave primaria	Identificatore univoco della persona
Ruolo	Alfabetico	Opzionale	Ruolo assunto dal personale amministrativo all'interno del fitness club

Medico			
Descrizione	Persona abilitata all'effettuazione delle visite sui clienti e al rilascio dei certificati.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Codice_Fiscale	Alfanumerico	(Attributo derivato da Persona) Chiave primaria	Identificatore univoco della persona
Conseguimento_Abitolazione	Data	Obbligatorio	Data di conseguimento dell'abilitazione medica
Conseguimento_titolo	Data	Obbligatorio	Data di conseguimento al titolo di medico

Brevetto			
Descrizione	Attestato ufficiale della capacità di un individuo a esercitare determinate mansioni specifiche.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Id_brevetto	Numerico	Chiave Primaria	Identificatore univoco del brevetto
Titolo	Alfabetico	Obbligatorio	Abilitazione all'attività
Livello	Alfanumerico	Obbligatorio	Grado che identifica il titolo
Descrizione	Alfabetico	Opzionale	Spiegazione del titolo conseguito
Data_conseguimento	Data	Obbligatorio	Data di conseguimento del brevetto

Ente_Sportivo			
Descrizione	Un ente pubblico o privato abilitato a rilasciare brevetti.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Acronimo	Alfabetico	Chiave Primaria	Abbreviazione del Nome
Nome	Alfabetico	Obbligatorio	Nome esteso dell'ente Sportivo

Visita_Medica			
Descrizione	Visita medica che consente di svolgere le attività sportive.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Data_Visita_Medica	Data	Chiave Primaria (con Codice_Fiscale (Medico) e Codice_Fiscale (Cliente))	Data visita medica
Codice_Fiscale (Medico)	Alfanumerico	(Attributo derivato da Medico) Chiave Primaria (con Data_Visita_Medica e Codice_Fiscale (Cliente))	Identificatore univoco della persona
Codice_Fiscale (Cliente)	Alfanumerico	(Attributo derivato da Cliente) Chiave Primaria (con Data_Visita_Medica e Codice_Fiscale (Medico))	Identificatore univoco della persona

Prenotazione			
Descrizione	Pagamento per aver riservato un campo in una data e ora specifici.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Data	Data	Chiave Primaria (con Ora e Nome)	Data prenotazione
Ora	Numerico	Chiave Primaria (con Data e Nome)	Ora prenotazione
Nome	Alfanumerico	(Attributo derivato da Campo Da Gioco) Chiave Primaria (con Data e Ora)	Nome identificativo del campo da gioco
Durata_ore	Numerico	Obbligatorio	Durata della prenotazione espresso in ore
Costo_Totale	Numerico	Opzionale Default 0	Costo totale del campo prenotato

Abbonamento			
Descrizione	Contratto per cui, pagando una determinata somma, si può, per un tempo determinato, usufruire dei servizi del Fitness Club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Data_inizio	Data	Chiave Primaria (con Codice_Fiscale)	Data inizio dell'abbonamento sottoscritto
Codice_Fiscale	Numerico	(Attributo derivato da Cliente) Chiave Primaria (con Data_inizio)	Identificatore univoco della persona
Costo_Totale	Numerico	Opzionale Default 0	Costo totale delle singole attività sottoscritte nell'abbonamento
Durata_giorni	Numerico	Obbligatorio	Durata dell'abbonamento espressa in giorni

Rinnovo			
Descrizione	Rinnovo di un abbonamento		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Data_rinnovo	Data	Chiave Primaria (con Data_inizio e Codice_Fiscale)	Data rinnovo dell'abbonamento
Data_inizio	Data	(Attributo derivato da Abbonamento) Chiave Primaria (con Data_rinnovo e Codice_Fiscale)	Data inizio dell'abbonamento sottoscritto
Codice_Fiscale	Alfanumerico	(Attributo derivato da Cliente) Chiave Primaria (con Data_rinnovo e Data_inizio)	Identificatore univoco della persona

Attività			
Descrizione	Attività fisica pianificata ed eseguita regolarmente.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Nome	Alfabetico	Chiave Primaria	Nome identificativo dell'attività
Descrizione	Alfabetico	Opzionale	Descrizione dettagliata dell'attività
Max_Partecipanti	Numerico	Obbligatorio	Numero massimo di persone che possono iscriversi all'attività
Num_iscritti	Numerico	Obbligatorio	Numero attuale di persone iscritte all'attività
Costo_giornaliero	Numerico	Obbligatorio	Costo singolo giornaliero dell'attività
Tipo_luogo	Alfabetico	Obbligatorio	Tipologia del luogo dove si svolge l'attività

Luogo			
Descrizione	Ubicazione fisica degli spazi all'interno del Fitness Club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Nome	Alfanumerico	Chiave Primaria	Nome identificativo del luogo
Capienza	Numerico	Obbligatoria	Numero massimo di persone che il luogo può contenere

Spogliatoio			
Descrizione	Luogo destinato alla deposizione o al cambio degli indumenti.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Nome	Alfanumerico	(Attributo derivato da Luogo) Chiave Primaria	Nome identificativo dello spogliatoio
Sesso	Alfabetico	Obbligatorio	Il sesso per il quale è destinato lo spogliatoio

Luogo_Activita			
Descrizione	Luogo destinato allo svolgimento delle lezioni del Fitness Club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Nome	Alfanumerico	(Attributo derivato da Luogo) Chiave Primaria	Nome identificativo del luogo dell'attività
Descrizione	Alfabetico	Opzionale	Descrizione dettagliata del luogo
Tipo_luogo	Alfabetico	Obbligatorio	Tipologia del luogo

Sala			
Descrizione	Un luogo attrezzato per praticare sport al chiuso (indoor).		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Nome	Alfanumerico	(Attributo derivato da Luogo_Activita) Chiave Primaria	Nome identificativo della sala
Tipologia	Alfabetico	Opzionale	Tipo di sala destinata a una determinata attività

Campo_Da_Gioco			
Descrizione	Luogo attrezzato per praticare sport all'esterno (outdoor) del Fitness Club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Nome	Alfanumerico	(Attributo derivato da Luogo_Activita) Chiave Primaria	Nome identificativo del campo da gioco
Costo_prenotazione_ore	Numerico	Obbligatorio	Compenso da erogare per prenotare ad ore il campo.

Lezione			
Descrizione	Incontri settimanali tra istruttori e clienti per svolgere un'attività		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Orario_Inizio	Numerico	Chiave Primaria con (Giorno e Nome)	Ora di inizio della lezione
Giorno	Alfabetico	Chiave Primaria con (Nome e Orario_inizio)	Giorno della settimana in cui viene svolta la lezione
Nome	Alfanumerico	(Attributo derivato da Luogo_Activita) Chiave Primaria (con Orario_inizio e Giorno)	Nome identificativo del luogo dell'attività
Durata_minuti	Numerico	Obbligatorio	Durata in minuti della lezione
CodiceLezione	Numerico	Obbligatorio e Unico	Codice univoco della lezione

3.2.3 Descrizione Relazioni

Si_sottopone	
Descrizione	Ottenimento del certificato medico del cliente
Entità Coinvolte	
Entità	Cardinalità
Cliente	(0,n)
Visita_Medica	(1,1)

Esegue	
Descrizione	Visita Medica effettuata dal medico che permette di sottoscrivere abbonamenti
Entità Coinvolte	
Entità	Cardinalità
Visita_Medica	(1,1)
Medico	(0,n)

Consegue	
Descrizione	Istruttore consegue brevetto abilitante ad una determinata attività rilasciato da un ente sportivo
Entità Coinvolte	
Entità	Cardinalità
Istruttore	(1,n)
Brevetto	(1,1)
Ente_sportivo	(0,n)

Supervisiona	
Descrizione	Istruttore supervisiona una lezione
Entità Coinvolte	
Entità	Cardinalità
Istruttore	(0,n)
Lezione	(1,1)

Abilita	
Descrizione	Abilitazione all'insegnamento di un'attività
Entità Coinvolte	
Entità	Cardinalità
Brevetto	(1,1)
Attività	(1,n)

Consente	
Descrizione	Consente lo svolgimento di una o più attività
Entità Coinvolte	
Entità	Cardinalità
Abbonamento	(1,n)
Attività	(0,n)

Possiede	
Descrizione	Sottoscrizione di un abbonamento da parte di un cliente
Entità Coinvolte	
Entità	Cardinalità
Cliente	(0,n)
Abbonamento	(1,1)

Puo_avere	
Descrizione	Esistenza per un abbonamento di un rinnovo
Entità Coinvolte	
Entità	Cardinalità
Abbonamento	(0,n)
Rinnovo	(1,1)

Effettua	
Descrizione	Il cliente effettua una prenotazione su un campo da gioco
Entità Coinvolte	
Entità	Cardinalità
Cliente	(0,n)
Prenotazione	(1,1)

Su	
Descrizione	Prenotazione di un cliente su un campo da gioco
Entità Coinvolte	
Entità	Cardinalità
Prenotazione	(1,1)
Campo_Da_Gioco	(0,n)

Ha	
Descrizione	Lezione programmata per una determinata attività
Entità Coinvolte	
Entità	Cardinalità
Attività	(1,n)
Lezione	(0,1)

Svolta_In	
Descrizione	Svolgimento di una lezione in un luogo
Entità Coinvolte	
Entità	Cardinalità
Lezione	(1,1)
Luogo_Actività	(0,n)

3.2.4 Vincoli non espressi dallo schema E/R

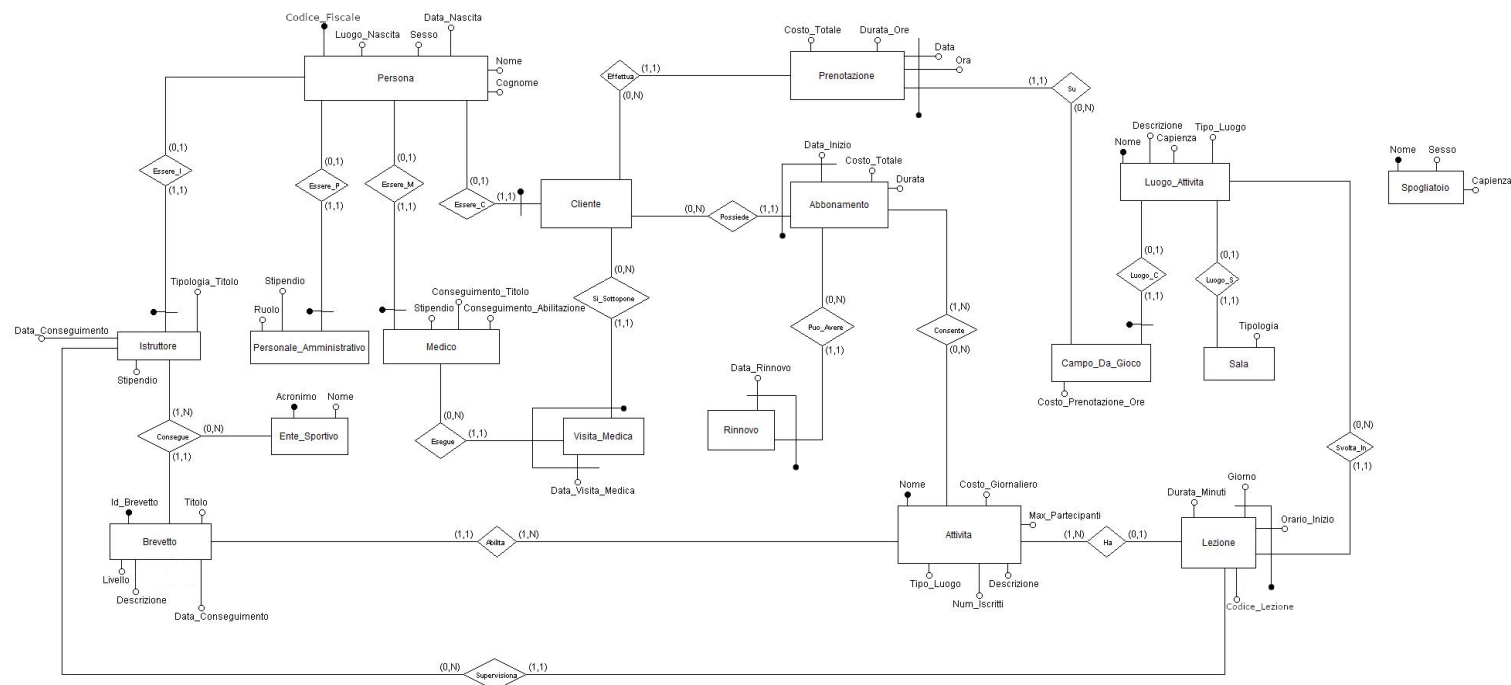
Tabella 1. Vincoli non espressi dallo schema E-R

Regole di Vincolo
Un cliente deve sottoporsi ad una visita medica prima di un abbonamento/rinnovo.
Un'attività di sala/campo non può essere svolta in campo/sala.
Uno stesso istruttore non può partecipare contemporaneamente a due o più lezioni.
Un cliente non può prenotare un campo ancora occupato dalla prenotazione precedente o da una lezione.
Una lezione non può svolgersi nello stesso luogo in cui si sta svolgendo già un'altra lezione.
Un abbonamento si può rinnovare solo dopo la sua scadenza.
Un cliente non può avere più abbonamenti con le stesse attività.
Un istruttore può supervisionare solo le lezioni delle attività per le quali ha un brevetto.
Una lezione non può sovrapporsi ad un'altra in termini di durata.
Un'attività deve tener conto del numero di iscritti che hanno un abbonamento in corso di validità.
Il costo totale di un abbonamento è la somma dei costi delle singole attività abilitate.
Il costo della prenotazione del campo da gioco è calcolato in base alle ore della prenotazione per il costo ad ore del campo.

4 Progettazione Logica

4.1 Ristrutturazione dello Schema E-R

Figura 2- Schema E-R ristrutturato



4.2 Documentazione dello Schema Logico

4.2.1 Descrizione Entità

Persona			
Descrizione	Individuo che può accedere al Fitness Club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Codice_Fiscale	CHAR(16)	Primary Key	Identificatore univoco della persona
Nome	VARCHAR(40)	NOT NULL	Nome anagrafico.
Cognome	VARCHAR(40)	NOT NULL	Cognome anagrafico.
Sesso	ENUM('M','F')	NOT NULL	Sesso biologico
Luogo_nascita	VARCHAR(40)	NOT NULL	Luogo di nascita anagrafico
Data_nascita	DATE	NOT NULL	Data di nascita anagrafica

Cliente			
Descrizione	Persona che può accedere alle attività del Fitness club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Codice_Fiscale	CHAR(16)	Primary Key (integrità referenziale con Persona)	Identificatore univoco della persona

Istruttore			
Descrizione	Persona che insegna e supervisiona le attività del Fitness club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Codice_Fiscale	CHAR(16)	Primary Key (integrità referenziale con Persona)	Identificatore univoco della persona
Tipologia_titolo	ENUM('Diploma ISEF', 'Laurea in Scienze Motorie')	NOT NULL	Titolo conseguito di tipo ISEF o laurea
Data_conseguimento	DATE		Data di conseguimento del titolo ISEF o laurea
Stipendio	DOUBLE	NOT NULL	Compenso monetario erogato allo staff in base alla mansione svolta

Personale_Amministrativo			
Descrizione	Persona abilitata all'effettuazione delle visite sui clienti e al rilascio dei certificati.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Codice_Fiscale	CHAR(16)	Primary Key (integrità referenziale con Persona)	Identificatore univoco della persona
Stipendio	DOUBLE	NOT NULL	Compenso monetario erogato allo staff in base alla mansione svolta
Ruolo	VARCHAR(20)		Ruolo assunto dal personale amministrativo all'interno del Fitness Club

Medico			
Descrizione	Persona abilitata all'effettuazione delle visite sui clienti e al rilascio dei certificati.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Codice_Fiscale	CHAR(16)	Primary Key (integrità referenziale con Persona)	Identificatore univoco della persona
Conseguimento_Abitolazione	DATE	NOT NULL	Data di conseguimento dell'abilitazione medica
Conseguimento_titolo	DATE	NOT NULL	Data di conseguimento al titolo di medico
Stipendio	DOUBLE	NOT NULL	Compenso monetario erogato allo staff in base alla mansione svolta

Brevetto			
Descrizione	Attestato ufficiale della capacità di un individuo a esercitare determinate mansioni specifiche.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Id_brevetto	INT	Primary Key	Identificatore univoco del brevetto
Titolo	VARCHAR(20)	NOT NULL	Abilitazione all'attività
Livello	CHAR(2)	NOT NULL	Grado che identifica il titolo
Descrizione	VARCHAR(100)		Spiegazione del titolo conseguito
Data_conseguimento	DATE	NOT NULL	Data di conseguimento del brevetto

Ente_Sportivo			
Descrizione	Un ente pubblico o privato abilitato a rilasciare brevetti.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Acronimo	VARCHAR(10)	Primary Key	Abbreviazione del Nome
Nome	VARCHAR(40)	NOT NULL	Nome esteso dell'ente Sportivo

Visita_Medica			
Descrizione	Visita medica che consente di svolgere le attività sportive.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Data_Visita_Medica	DATE	Primary Key (con Codice_Fiscale (Medico) e Codice_Fiscale (Cliente))	Data visita medica
Codice_Fiscale (Medico)	CHAR(16)	Primary Key (con Data_Visita_Medica e Codice_Fiscale (Cliente)) (integrità referenziale con Medico)	Identificatore univoco della persona
Codice_Fiscale (Cliente)	CHAR(16)	Primary Key (con Data_Visita_Medica e Codice_Fiscale (Medico)) (integrità referenziale con Cliente)	Identificatore univoco della persona

Prenotazione			
Descrizione	Pagamento per aver riservato un campo in una data e ora specifici.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Data	DATE	Primary Key (con Ora e Nome)	Data prenotazione
Ora	TIME	Primary Key (con Data e Nome)	Ora prenotazione
Nome	VARCHAR(20)	Primary Key (con Ora e Data) (integrità referenziale con Campo_Da_Gioco)	Nome identificativo del campo da gioco
Durata_ore	DECIMAL(2, 1)	NOT NULL	Durata della prenotazione espresso in ore
Costo_Totale	DOUBLE	DEFAULT 0	Costo totale del campo prenotato

Abbonamento			
Descrizione	Contratto per cui, pagando una determinata somma, si può, per un tempo determinato, usufruire dei servizi del Fitness Club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Data_inizio	DATE	Primary Key (con Codice_Fiscale)	Data inizio dell'abbonamento sottoscritto
Codice_Fiscale (Cliente)	CHAR(16)	Primary Key (con Data_inizio) (integrità referenziale con Cliente)	Identificatore univoco della persona
Costo_Totale	DOUBLE	DEFAULT 0	Costo totale delle singole attività sottoscritte nell'abbonamento
Durata_giorni	INT	NOT NULL	Durata dell'abbonamento espressa in giorni

Rinnovo			
Descrizione	Rinnovo di un abbonamento		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Data_rinnovo	DATE	Primary Key(con Data_inizio e Codice_Fiscale)	Data rinnovo dell'abbonamento
Data_inizio	DATE	Primary Key (con Data_rinnovo e Codice_Fiscale) (integrità referenziale con Abbonamento)	Data inizio dell'abbonamento sottoscritto
Codice_Fiscale (Cliente)	CHAR(16)	Primary Key (con Data_rinnovo e Data_inizio) (integrità referenziale con Cliente)	Identificatore univoco della persona

Attività			
Descrizione	Attività fisica pianificata ed eseguita regolarmente.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Nome	VARCHAR(30)	Primary Key	Nome identificativo dell'attività
Descrizione	VARCHAR(100)		Descrizione dettagliata dell'attività
Max_Partecipanti	INT	NOT NULL	Numero massimo di persone che possono iscriversi all'attività
Num_iscritti	INT	NOT NULL	Numero attuale di persone iscritte all'attività
Costo_giornaliero	DOUBLE	NOT NULL	Costo singolo giornaliero dell'attività
Tipo_Luogo	ENUM (‘S’,‘C’)	NOT NULL	Tipologia del luogo dove si svolge l'attività

Spogliatoio			
Descrizione	Luogo destinato alla deposizione o al cambio degli indumenti.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Nome	VARCHAR(30)	Primary Key	Nome identificativo dello spogliatoio
Capienza	INT	NOT NULL	Numero massimo di persone che il luogo può contenere
Sesso	ENUM('M','F')	NOT NULL	Il sesso per il quale è destinato lo spogliatoio

Luogo_Activita			
Descrizione	Luogo destinato allo svolgimento delle lezioni del Fitness Club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Nome	VARCHAR(30)	Primary Key	Nome identificativo del luogo dell'attività
Descrizione	VARCHAR(100)		Descrizione dettagliata del luogo
Tipo_luogo	ENUM('S','C')	NOT NULL	Tipologia del luogo
Capienza	INT	NOT NULL	Numero massimo di persone che il luogo può contenere

Sala			
Descrizione	Un luogo attrezzato per praticare sport al chiuso (indoor).		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Nome	VARCHAR(30)	Primary Key (integrità referenziale con Luogo_Activita)	Nome identificativo della sala
Tipologia	VARCHAR(30)		Tipo di sala destinata a una determinata attività

Campo_Da_Gioco			
Descrizione	Luogo attrezzato per praticare sport all'esterno del Fitness Club.		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Nome	VARCHAR(30)	Primary Key (integrità referenziale con Luogo_Activita)	Nome identificativo del campo da gioco
Costo_prenotazione_ore	DOUBLE	NOT NULL	Compenso da erogare per prenotare ad ore il campo.

Lezione			
Descrizione	Incontri settimanali tra istruttori e clienti per svolgere un'attività		
Nome Attributo	Tipo di dato	Vincolo	Descrizione
Orario_Inizio	TIME	Primary Key (con Giorno e Nome)	Ora di inizio della lezione
Giorno	ENUM('Lunedì', 'Martedì', 'Mercoledì', 'Giovedì', 'Venerdì', 'Sabato', 'Domenica')	Primary Key (con Nome e Orario_inizio)	Giorno della settimana in cui viene svolta la lezione
Nome	VARCHAR(30)	Primary Key (con Orario_inizio e Giorno) (integrità referenziale con Luogo_Activita)	Nome identificativo del luogo dell'attività
Durata_minuti	INT	NOT NULL	Durata in minuti della lezione
CodiceLezione	INT	NOT NULL, UNIQUE	Codice univoco della lezione

4.2.2 Descrizione Relazioni

Essere_C	
Descrizione	La possibilità di una persona di essere un cliente
Entità Coinvolte	
Entità	Cardinalità
Persona	(0,1)
Cliente	(1,1)

Essere_I	
Descrizione	La possibilità di una persona di essere un istruttore
Entità Coinvolte	
Entità	Cardinalità
Persona	(0,1)
Istruttore	(1,1)

Essere_M	
Descrizione	Possibilità di una persona di essere un medico
Entità Coinvolte	
Entità	Cardinalità
Persona	(0,1)
Medico	(1,1)

Essere_P	
Descrizione	Possibilità di una persona di essere parte del personale amministrativo
Entità Coinvolte	
Entità	Cardinalità
Persona	(0,1)
Personale_Ammministrativo	(1,1)

Si_sottopone	
Descrizione	Ottenimento del certificato medico del cliente
Entità Coinvolte	
Entità	Cardinalità
Cliente	(0,n)
Visita_Medica	(1,1)

Esegue	
Descrizione	Visita Medica effettuata dal medico che permette di sottoscrivere abbonamenti
Entità Coinvolte	
Entità	Cardinalità
Visita_Medica	(1,1)
Medico	(0,n)

Consegue	
Descrizione	Istruttore consegue brevetto abilitante ad una determinata attività rilasciato da un ente sportivo
Entità Coinvolte	
Entità	Cardinalità
Istruttore	(1,n)
Brevetto	(1,1)
Ente_sportivo	(0,n)

Supervisiona	
Descrizione	Istruttore supervisiona una lezione
Entità Coinvolte	
Entità	Cardinalità
Istruttore	(0,n)
Lezione	(1,1)

Abilita	
Descrizione	Abilitazione all'insegnamento di un'attività
Entità Coinvolte	
Entità	Cardinalità
Brevetto	(1,1)
Attività	(1,n)

Consente	
Descrizione	Consente lo svolgimento di una o più attività
Entità Coinvolte	
Entità	Cardinalità
Abbonamento	(1,n)
Attività	(0,n)

Possiede	
Descrizione	Sottoscrizione di un abbonamento da parte di un cliente
Entità Coinvolte	
Entità	Cardinalità
Cliente	(0,n)
Abbonamento	(1,1)

Puo_avere	
Descrizione	Esistenza per un abbonamento di un rinnovo
Entità Coinvolte	
Entità	Cardinalità
Abbonamento	(0,n)
Rinnovo	(1,1)

Effettua	
Descrizione	Il cliente effettua una prenotazione su un campo da gioco
Entità Coinvolte	
Entità	Cardinalità
Cliente	(0,n)
Prenotazione	(1,1)

Su	
Descrizione	Prenotazione di un cliente su un campo da gioco
Entità Coinvolte	
Entità	Cardinalità
Prenotazione	(1,1)
Campo_Da_Gioco	(0,n)

Ha	
Descrizione	Lezione programmata per una determinata attività
Entità Coinvolte	
Entità	Cardinalità
Attività	(1,n)
Lezione	(0,1)

Svolta_In	
Descrizione	Svolgimento di una lezione in un luogo
Entità Coinvolte	
Entità	Cardinalità
Lezione	(1,1)
Luogo_Actività	(0,n)

Luogo_S	
Descrizione	Il rispettivo luogo è una sala
Entità Coinvolte	
Entità	Cardinalità
Luogo_Activita	(0,1)
Sala	(1,1)

Luogo_C	
Descrizione	Il rispettivo luogo è un campo da gioco
Entità Coinvolte	
Entità	Cardinalità
Luogo_Activita	(0,1)
Campo_Da_Gioco	(1,1)

4.3 Traduzione dello Schema E-R

4.3.1 Traduzione Entità

Persona(Codice_Fiscale, Nome, Cognome, Sesso, Luogo_Nascita, Data_Nascita)

Medico(Codice_Fiscale*, Consegimento_Abilitazione, Consegimento_Titolo, Stipendio)
Esiste un vincolo di integrità referenziale fra l'attributo Codice_Fiscale e la relazione Persona.

Personale Amministrativo(Codice_Fiscale*, Ruolo, Stipendio)
Esiste un vincolo di integrità referenziale fra l'attributo Codice_Fiscale e la relazione Persona.

Istruttore(Codice_Fiscale*, Tipologia_Titolo, Data_Consegimento, Stipendio)
Esiste un vincolo di integrità referenziale fra l'attributo Codice_Fiscale e la relazione Persona.

Cliente(Codice_Fiscale*)
Esiste un vincolo di integrità referenziale fra l'attributo Codice_Fiscale e la relazione Persona.

Ente_Sportivo(Acronimo, Nome)

Attività(Nome, Descrizione, Max_Partecipanti, Num_Iscritti, Costo_Giornaliero, Tipo_Luogo)

Brevetto(Id_Brevetto, Ente_Sportivo*, Istruttore*, Attività*, Titolo, Livello, Descrizione, Data_Consegimento)
Esiste un vincolo di integrità referenziale fra l'attributo Ente_Sportivo e la relazione Ente_Sportivo.
Esiste un vincolo di integrità referenziale fra l'attributo Istruttore e la relazione Istruttore.
Esiste un vincolo di integrità referenziale fra l'attributo Attività e la relazione Attività.

Visita Medica(Data_Visita_Medica, Medico*, Cliente*)
Esiste un vincolo di integrità referenziale fra l'attributo Medico e la relazione Medico.
Esiste un vincolo di integrità referenziale fra l'attributo Cliente e la relazione Cliente.

Abbonamento(Data_Inizio_Cliente*, Costo_Totale, Durata_Giorni)
Esiste un vincolo di integrità referenziale fra l'attributo Cliente e la relazione Cliente.

Rinnovo(Data_Rinnovo, (Abbonamento_data_inizio, Cliente_Abbonamento*))
Esiste un vincolo di integrità referenziale fra l'attributo Abbonamento e la relazione Abbonamento.

Luogo_Actività(Nome, Descrizione, Tipo_Luogo, Capienza)

Spogliatoio(Nome, Capienza, Sesso)

Sala(Luogo_Actività*, Tipologia)
Esiste un vincolo di integrità referenziale fra l'attributo Luogo_Actività e la relazione Luogo_Actività.

Campo_Da_Gioco(Luogo_Actività*, Costo_Prenotazione_Ore)
Esiste un vincolo di integrità referenziale fra l'attributo Luogo_Actività e la relazione Luogo_Actività.

Prenotazione(Data, Ora, Campo_Da_Gioco*, Durata_Ore, Costo_Totale, Cliente*)
Esiste un vincolo di integrità referenziale fra l'attributo Campo_Da_Gioco e la relazione Campo_Da_Gioco.
Esiste un vincolo di integrità referenziale fra l'attributo Cliente e la relazione Cliente.

Lezione(Giorno, Orario_Inizio, Luogo_Actività*, Durata_Minuti, CodiceLezione, Attività*, Istruttore*)
Esiste un vincolo di integrità referenziale fra l'attributo Luogo_Actività e la relazione Luogo_Actività.
Esiste un vincolo di integrità referenziale fra l'attributo Attività e la relazione Attività.
Esiste un vincolo di integrità referenziale fra l'attributo Istruttore e la relazione Istruttore.

4.3.2 Traduzione Relazioni

Consente(Attività*,(Abbonamento Data Inizio,Abbonamento Cliente*))

Esiste un vincolo di integrità referenziale fra l'attributo Attività e la relazione Attività.

Esiste un vincolo di integrità referenziale fra l'attributo Abbonamento e la relazione Abbonamento.

5 Progettazione Fisica

5.1 Definizione dello schema della base di dati

Tabella Persona

```
CREATE TABLE `Persona` (  
  `Codice_Fiscale` char(16) NOT NULL,  
  `Nome` varchar(40) NOT NULL,  
  `Cognome` varchar(40) NOT NULL,  
  `Sesso` ENUM(' M ', ' F ') NOT NULL,  
  `Luogo_Nascita` varchar(40) NOT NULL,  
  `Data_Nascita` date NOT NULL,  
  PRIMARY KEY (`Codice_Fiscale`)  
);
```

Tabella Medico

```
CREATE TABLE `Medico` (  
  `Codice_Fiscale` char(16) NOT NULL,  
  `Conseguimento_Abititazione` date NOT NULL,  
  `Conseguimento_Titolo` date NOT NULL,  
  `Stipendio` double NOT NULL,  
  PRIMARY KEY (`Codice_Fiscale`),  
  FOREIGN KEY (`Codice_Fiscale`) REFERENCES  
  `Persona`(`Codice_Fiscale`)  
);
```

Tabella Personale Amministrativo

```
CREATE TABLE `Personale_Ammministrativo` (  
  `Codice_Fiscale` char(16) NOT NULL,  
  `Stipendio` double NOT NULL,  
  `Ruolo` varchar(20) DEFAULT NULL,  
  PRIMARY KEY (`Codice_Fiscale`),  
  FOREIGN KEY (`Codice_Fiscale`) REFERENCES `Persona`  
  (`Codice_Fiscale`)  
);
```

Tabella Istruttore

```
CREATE TABLE `Istruttore` (  
  `Codice_Fiscale` char(16) NOT NULL,  
  `Tipologia_Titolo` enum('Diploma ISEF', 'Laurea in Scienze  
Motorie') NOT NULL,  
  `Data_Conseguimento` date DEFAULT NULL,  
  `Stipendio` double NOT NULL,  
  PRIMARY KEY (`Codice_Fiscale`),  
  FOREIGN KEY (`Codice_Fiscale`) REFERENCES `Persona`  
  (`Codice_Fiscale`)  
);
```

Tabella Cliente

```
CREATE TABLE `Cliente` (  
    `Codice_Fiscale` char(16) NOT NULL,  
    PRIMARY KEY (`Codice_Fiscale`),  
    FOREIGN KEY (`Codice_Fiscale`) REFERENCES `Persona`  
    (`Codice_Fiscale`)  
);
```

Tabella Ente Sportivo

```
CREATE TABLE `Ente_Sportivo` (  
    `Acronimo` varchar(10) NOT NULL,  
    `Nome` varchar(40) NOT NULL,  
    PRIMARY KEY (`Acronimo`)  
);
```

Tabella Attivita

```
CREATE TABLE `Attivita` (  
    `Nome` varchar(30) NOT NULL,  
    `Descrizione` varchar(100) DEFAULT NULL,  
    `Max_Partecipanti` int(10) unsigned NOT NULL,  
    `Num_Iscritti` int(10) unsigned NOT NULL,  
    `Costo_Giornaliero` double NOT NULL,  
    `Tipo_Luogo` enum('S','C') NOT NULL,  
    PRIMARY KEY (`Nome`)  
);
```

Tabella Brevetto

```
CREATE TABLE `Brevetto` (  
    `Id_Brevetto` int(10) unsigned NOT NULL AUTO_INCREMENT,  
    `Titolo` varchar(20) NOT NULL,  
    `Livello` char(2) NOT NULL,  
    `Descrizione` varchar(100) DEFAULT NULL,  
    `Data_Conseguimento` date NOT NULL,  
    `Ente_Sportivo` varchar(10) NOT NULL,  
    `Istruttore` char(16) NOT NULL,  
    `Attivita` varchar(30) NOT NULL,  
    PRIMARY KEY (`Id_Brevetto`),  
    FOREIGN KEY (`Attivita`) REFERENCES `Attivita` (`Nome`),  
    FOREIGN KEY (`Ente_Sportivo`) REFERENCES `Ente_Sportivo`  
    (`Acronimo`),  
    FOREIGN KEY (`Istruttore`) REFERENCES `Istruttore`  
    (`Codice_Fiscale`)  
);
```

Tabella Visita Medica

```
CREATE TABLE `Visita_Medica` (  
  `Data_Visita_Medica` date NOT NULL,  
  `Medico` char(16) NOT NULL,  
  `Cliente` char(16) NOT NULL,  
  PRIMARY KEY (`Data_Visita_Medica`,`Medico`,`Cliente`),  
  FOREIGN KEY (`Cliente`) REFERENCES `Cliente`  
    (`Codice_Fiscale`),  
  FOREIGN KEY (`Medico`) REFERENCES `Medico` (`Codice_Fiscale`)  
);
```

Tabella Abbonamento

```
CREATE TABLE `Abbonamento` (  
  `Data_Inizio` date NOT NULL,  
  `Cliente` char(16) NOT NULL,  
  `Costo_Totale` double NULL DEFAULT '0' ,  
  `Durata_Giorni` int(10) unsigned NOT NULL,  
  PRIMARY KEY (`Data_Inizio`,`Cliente`),  
  FOREIGN KEY (`Cliente`) REFERENCES `Cliente`  
    (`Codice_Fiscale`)  
);
```

Tabella Rinnovo

```
CREATE TABLE `Rinnovo` (  
  `Data_Rinnovo` date NOT NULL,  
  `Abbonamento_data_inizio` date NOT NULL,  
  `Cliente_Abbonamento` char(16) NOT NULL,  
  PRIMARY KEY (`Data_Rinnovo`,`Cliente_Abbonamento`,`Abbonamento_data_inizio`),  
  FOREIGN KEY (`Cliente_Abbonamento`) REFERENCES `Abbonamento`  
    (`Cliente`),  
  FOREIGN KEY (`Abbonamento_data_inizio`) REFERENCES  
    `Abbonamento` (`Data_Inizio`)  
);
```

Tabella Luogo Attivita

```
CREATE TABLE `Luogo_Activita` (  
  `Nome` varchar(30) NOT NULL,  
  `Descrizione` varchar(100) DEFAULT NULL,  
  `Tipo_Luogo` enum('S','C') NOT NULL,  
  `Capienza` int(10) unsigned NOT NULL,  
  PRIMARY KEY (`Nome`)  
);
```

Tabella Spogliatoio

```
CREATE TABLE `Spogliatoio` (  
  `Nome` varchar(30) NOT NULL,  
  `Capienza` int(10) unsigned NOT NULL,  
  `Sesso` ENUM('M','F') NOT NULL,  
  PRIMARY KEY (`Nome`)  
);
```

Tabella Sala

```
CREATE TABLE `Sala` (  
  `Nome` varchar(30) NOT NULL,  
  `Tipologia` varchar(30) DEFAULT NULL,  
  PRIMARY KEY (`Nome`),  
  FOREIGN KEY (`Nome`) REFERENCES `Luogo_Activita` (`Nome`)  
);
```

Tabella Campo Da Gioco

```
CREATE TABLE `Campo_Da_Gioco` (  
  `Nome` varchar(30) NOT NULL,  
  `Costo_Prenotazione_Ore` double NOT NULL,  
  PRIMARY KEY (`Nome`),  
  FOREIGN KEY (`Nome`) REFERENCES `Luogo_Activita` (`Nome`)  
);
```

Tabella Prenotazione

```
CREATE TABLE `Prenotazione` (  
  `Data` date NOT NULL,  
  `Ora` time NOT NULL,  
  `Campo_Da_Gioco` varchar(20) NOT NULL,  
  `Durata_Ore` decimal(2,1) NOT NULL,  
  `Costo_Totale` double NULL DEFAULT '0' ,  
  `Cliente` char(16) NOT NULL,  
  PRIMARY KEY (`Data`,`Ora`,`Campo_Da_Gioco`),  
  FOREIGN KEY (`Campo_Da_Gioco`) REFERENCES `Campo_Da_Gioco`  
  (`Nome`),  
  FOREIGN KEY (`Cliente`) REFERENCES `Cliente`  
  (`Codice_Fiscale`)  
);
```

Tabella Lezione

```
CREATE TABLE `Lezione` (  
  `Giorno` enum('Lunedì', 'Martedì', 'Mercoledì', 'Giovedì',  
  'Venerdì', 'Sabato', 'Domenica') NOT NULL,  
  `Orario_Inizio` time NOT NULL,  
  `Luogo_Activita` varchar(30) NOT NULL,  
  `Durata_Minuti` int(11) NOT NULL,  
  `Attivita` varchar(30) NOT NULL,  
  `Istruttore` char(16) NOT NULL,  
  `CodiceLezione` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  PRIMARY KEY (`Giorno`,`Orario_Inizio`,`Luogo_Activita`),  
  UNIQUE KEY `CodiceLezione_UNIQUE` (`CodiceLezione`),  
  FOREIGN KEY (`Attivita`) REFERENCES `Attivita` (`Nome`),  
  FOREIGN KEY (`Istruttore`) REFERENCES `Istruttore`  
  (`Codice_Fiscale`),  
  FOREIGN KEY (`Luogo_Activita`) REFERENCES `Luogo_Activita`  
  (`Nome`)  
);
```

Tabella **Consente**

```
CREATE TABLE `Consente` (  
  `Attivita` varchar(30) NOT NULL,  
  `Abbonamento_Data_Inizio` date NOT NULL,  
  `Abbonamento_Cliente` int(10) unsigned NOT NULL,  
  PRIMARY KEY  
  (`Attivita`,`Abbonamento_Data_Inizio`,`Abbonamento_Cliente`),  
  FOREIGN KEY (`Abbonamento_Data_Inizio`) REFERENCES  
  `Abbonamento` (`Data_Inizio`),  
  FOREIGN KEY (`Abbonamento_Cliente`) REFERENCES `Abbonamento`  
  (`Cliente`),  
  FOREIGN KEY (`Attivita`) REFERENCES `Attivita` (`Nome`)  
);
```

Tabella **Prenotazione Storica**

/ Per agevolare le operazioni sui dati è stata creata la tabella 'Prenotazione_Storica' in cui vengono spostate le vecchie prenotazioni di oltre un anno ad ogni fine anno */*

```
CREATE TABLE `Prenotazione_Storica` (  
  `Data` date NOT NULL,  
  `Ora` time NOT NULL,  
  `Campo_Da_Gioco` varchar(20) NOT NULL,  
  `Durata_Ore` decimal(2,1) NOT NULL,  
  `Costo_Totale` double NOT NULL,  
  `Cliente` char(16) NOT NULL,  
  PRIMARY KEY (`Data`,`Ora`,`Campo_Da_Gioco`),  
  FOREIGN KEY (`Campo_Da_Gioco`) REFERENCES `Campo_Da_Gioco`  
  (`Nome`),  
  FOREIGN KEY (`Cliente`) REFERENCES `Cliente` (`Codice_Fiscale`)  
);
```

5.2 *Trigger, Event Scheduler e Stored Procedure*

Function LezioniSovrapposte

```
DELIMITER $$
CREATE FUNCTION `LezioniSovrapposte`(`durata_nuova`
INT,`durata_esistente` INT,`ora_inizio_nuova`
TIME,`ora_inizio_esistente` TIME) RETURNS TINYINT(1) DETERMINISTIC BEGIN
DECLARE durata_secondi_nuova INT DEFAULT durata_nuova*60;
DECLARE durata_secondi_esistente INT DEFAULT durata_esistente*60;
DECLARE ora_fine_nuova_sec INT DEFAULT 0;
DECLARE ora_fine_esistente_sec INT DEFAULT 0;
DECLARE ora_inizio_nuova_sec INT DEFAULT time_to_sec(ora_inizio_nuova);
DECLARE ora_inizio_esistente_sec INT DEFAULT
time_to_sec(ora_inizio_esistente);
SET ora_fine_nuova_sec = time_to_sec(ora_inizio_nuova) +
durata_secondi_nuova;
SET ora_fine_esistente_sec = time_to_sec(ora_inizio_esistente) +
durata_secondi_esistente;
IF(
    (ora_inizio_nuova_sec < ora_inizio_esistente_sec AND
    ora_fine_nuova_sec < ora_inizio_esistente_sec)
    OR(ora_inizio_nuova_sec > ora_fine_esistente_sec)
    OR(ora_fine_esistente_sec = ora_inizio_nuova_sec)
    OR(ora_fine_nuova_sec = ora_inizio_esistente_sec)
)
THEN RETURN 0;
ELSE RETURN 1;
END IF;
/* 0 (non si sovrappongono)~ 1 (si sovrappongono)*/
END; $$
DELIMITER ;
```

Trigger LezioniInSovrapposizione

```
delimiter //
CREATE TRIGGER `LezioniInSovrapposizione` BEFORE INSERT ON `Lezione`
FOR EACH ROW BEGIN
    IF EXISTS (
        SELECT *
        FROM Lezione
        WHERE NEW.Giorno = Lezione.Giorno
        AND Lezione.Luogo_Attivita = NEW.Luogo_Attivita
        AND LezioniSovrapposte( Lezione.Durata_Minuti,
        NEW.Durata_Minuti, Lezione.Orario_Inizio,
        NEW.Orario_Inizio) = 1)
    THEN signal SQLSTATE '45000' SET message_text = 'Errore: Due
    lezioni non possono sovrapporsi';
    END IF;
END; //
delimiter ;
```

Trigger LezioniInSovrapposizioneUpdate

```
delimiter //
CREATE TRIGGER `LezioniInSovrapposizioneUpdate` BEFORE UPDATE
  ON `Lezione` FOR EACH ROW BEGIN
  DECLARE OldGiorno ENUM(`Lunedì`, `Martedì`, `Mercoledì`, `Giovedì`,
    `Venerdì`, `Sabato`, `Domenica` ) DEFAULT OLD.Giorno;
  DECLARE OldOrario TIME DEFAULT OLD.Orario_Inizio;
  DECLARE OldDurata INT DEFAULT OLD.Durata_Minuti;
  IF EXISTS (
    SELECT *
    FROM Lezione
    WHERE Lezione.CodiceLezione != NEW.CodiceLezione
      AND NEW.Giorno = Lezione.Giorno
      AND Lezione.Luogo_Activita = NEW.Luogo_Activita
      AND LezioniSovrapposte( Lezione.Durata_Minuti,
        NEW.Durata_Minuti, Lezione.Orario_Inizio,
        NEW.Orario_Inizio ) = 1)
  THEN signal SQLSTATE '45000' SET message_text = 'Errore: Due lezioni
    non possono sovrapporsi';
  END IF;
END;//
delimiter ;
```

Trigger IncrementaPartecipanti

```
delimiter //
CREATE TRIGGER IncrementaPartecipanti BEFORE INSERT
  ON Consente FOR EACH ROW
  BEGIN
    UPDATE Attivita
    SET Num_Iscritti = Num_Iscritti + 1
    WHERE NEW.Attivita = Nome AND
      AbbonamentoValido(new.Abbonamento_Data_Inizio;
  END;//
delimiter ;
```


Trigger CalcoloCostoAbbonamento

```
delimiter //
CREATE TRIGGER CalcoloCostoAbbonamento BEFORE INSERT ON Consente
FOR EACH ROW BEGIN
DECLARE costoAttivita DOUBLE;

if (AbbonamentoValido(new.Abbonamento_Data_Inizio,
new.Abbonamento_Cliente, now()) != 1)
    then signal sqlstate '45000' set message_text = 'Abbonamento
    Scaduto o non valido';
end if;

SELECT Costo_Giornaliero INTO costoAttivita
FROM Attivita
WHERE NEW.Attivita = Nome;

UPDATE Abbonamento
SET Costo_Totale=Costo_Totale + costoAttivita*Durata_Giorni
WHERE NEW.Abbonamento_Data_Inizio = Abbonamento.Data_Inizio AND
NEW.Abbonamento_Cliente = Abbonamento.Cliente;
END; //
delimiter ;
```

Trigger PossibileRinnovo

```
delimiter //
CREATE TRIGGER PossibileRinnovo BEFORE INSERT
ON Rinnovo FOR EACH ROW
BEGIN
    IF (AbbonamentoValido(NEW.Abbonamento_data_inizio,
    NEW.Cliente_Abbonamento, NEW.Data_Rinnovo) != 0)
    THEN signal SQLSTATE '45000' SET message_text = 'Impossibile
    rinnovare';
    END IF;
END; //
delimiter ;
```

Function AbbonamentoValido

```
CREATE FUNCTION AbbonamentoValido (Data_Inizio DATE, Id_Cliente
CHAR(16), Data_Nuovo_Abbonamento DATE) RETURNS TINYINT(1)
BEGIN
    IF(NOT EXISTS (
        SELECT *
        FROM Abbonamento
        WHERE Abbonamento.Cliente = Id_Cliente
            AND Abbonamento.Data_Inizio = Data_Inizio))
    THEN RETURN 2;
    END IF;
    # Controlla se ci sono RINNOVI ancora validi
    IF(EXISTS (
        SELECT *
        FROM Abbonamento, Rinnovo
        WHERE Data_Inizio = Rinnovo.Abbonamento_data_inizio
            AND Id_Cliente = Rinnovo.Cliente_Abbonamento
            AND Abbonamento.Cliente = Id_Cliente
            AND Abbonamento.Data_Inizio = Data_Inizio
            AND date_add(Rinnovo.Data_Rinnovo,
INTERVAL(Abbonamento.Durata_Giorni) DAY) > Data_Nuovo_Abbonamento
        )
    )
    THEN RETURN 1;
    END IF;
    # Controlla se non ho mai rinnovato e il primo abbonamento è ancora
    valido
    IF (EXISTS (
        SELECT *
        FROM Abbonamento
        WHERE Data_Inizio = Abbonamento.Data_Inizio
            AND Id_Cliente = Abbonamento.Cliente
            AND date_add(Abbonamento.Data_Inizio, INTERVAL
Abbonamento.Durata_Giorni DAY) > Data_Nuovo_Abbonamento
        )
    )
    THEN RETURN 1;
    END IF;
    RETURN 0;
    END ;
```

Trigger UgualiAttivita

```
delimiter //
CREATE TRIGGER UgualiAttivita BEFORE INSERT ON Consente FOR EACH ROW
BEGIN
    IF(NOT EXISTS (
        SELECT *
        FROM Abbonamento
        WHERE Abbonamento.Cliente = NEW.Abbonamento_Cliente
        AND Abbonamento.Data_Inizio=NEW.Abbonamento_Data_Inizio
    )
    )
    THEN signal SQLSTATE '45000' SET message_text = 'Abbonamento
    inesistente';
    END IF;
    IF(EXISTS (
        SELECT *
        FROM Abbonamento
        WHERE Abbonamento.Cliente = NEW.Abbonamento_Cliente
        AND Abbonamento.Data_Inizio!=NEW.Abbonamento_Data_Inizio
        AND AbbonamentoValido(Abbonamento.Data_Inizio,
        Abbonamento.Cliente, NEW.Abbonamento_Data_Inizio) = 1
        AND NEW.Attivita IN (
            SELECT Attivita
            FROM Consente
            WHERE
            Abbonamento_Data_Inizio=Abbonamento.Data_Inizio
            AND Abbonamento_Cliente = Abbonamento.Cliente
        )
    )
    )
    THEN signal SQLSTATE '45000' SET message_text = 'Attivita già presente
    in un abbonamento';
    END IF;
END; //
delimiter ;
```

Trigger clienteHaFattoVisita

```
delimiter //
CREATE TRIGGER clienteHaFattoVisita BEFORE INSERT ON Abbonamento FOR
EACH ROW BEGIN
    IF NOT EXISTS(
        SELECT *
        FROM Visita_Medica
        WHERE NEW.Cliente = Visita_Medica.Cliente
        AND NEW.Data_Inizio = Visita_Medica.Data_Visita_Medica
    )
    THEN signal SQLSTATE '45000' SET message_text = 'visita medica non
    effettuata';
    END IF;
END; //
delimiter ;
```

Trigger clienteHaFattoVisitaRinnovo

```
delimiter //
CREATE TRIGGER clienteHaFattoVisitaRinnovo BEFORE INSERT ON Rinnovo FOR
EACH ROW BEGIN
    IF NOT EXISTS(
        SELECT *
        FROM Visita_Medica
        WHERE NEW.Cliente_Abbonamento = Visita_Medica.Cliente
        AND NEW.Data_Rinnovo =
Visita_Medica.Data_Visita_Medica
    )
THEN signal SQLSTATE '45000' SET message_text = 'visita medica non
effettuata';
END IF;
END; //
delimiter ;
```

Trigger IstruttorePerLezione

```
delimiter //
CREATE TRIGGER `IstruttorePerLezione` BEFORE INSERT ON `Lezione` FOR
EACH ROW BEGIN
    IF (EXISTS(
        SELECT *
        FROM Lezione
        WHERE Lezione.Istruttore = NEW.Istruttore
        AND Lezione.Giorno = NEW.Giorno
        AND LezioniSovrapposte(Lezione.Durata_Minuti,
NEW.Durata_Minuti, Lezione.Orario_Inizio,
NEW.Orario_Inizio) = 1
    )
    )
THEN signal SQLSTATE '45000' SET message_text = 'Un istruttore non può
svolgere due lezioni contemporaneamente';
END IF;
    IF( NOT EXISTS(
        SELECT *
        FROM Brevetto
        WHERE Brevetto.Istruttore = NEW.Istruttore
        AND Brevetto.Attivita = NEW.Attivita
    )
    )
THEN signal SQLSTATE '45000' SET message_text = 'Istruttore non
abilitato a svolgere tale attività';
END IF;
END; //
delimiter ;
```

Trigger **PrenotazioneCampoLibero**

```
delimiter //
CREATE TRIGGER `PrenotazioneCampoLibero` BEFORE INSERT ON
`Prenotazione` FOR EACH ROW BEGIN

DECLARE costoCampo DOUBLE;
    IF ( EXISTS (
        SELECT *
        FROM Lezione
        WHERE Lezione.Giorno = GiornoDellaSettimana(NEW.Data)
            AND Lezione.Luogo_Attivita = NEW.Campo_Da_Gioco
            AND (LezioniSovrapposte(Lezione.Durata_Minuti,
NEW.Durata_Ore*60,Lezione.Orario_Inizio,NEW.Ora)=1)
        )
    )
THEN signal SQLSTATE '45000' SET message_text = 'Campo già occupato da
una lezione!';
END IF;
IF ( EXISTS (
    SELECT *
    FROM Prenotazione
    WHERE Prenotazione.Data = NEW.Data
        AND Prenotazione.Campo_Da_Gioco = NEW.Campo_Da_Gioco
        AND LezioniSovrapposte(Prenotazione.Durata_Ore * 60,
NEW.Durata_Ore*60, Prenotazione.Ora, NEW.Ora) = 1
    )
)
THEN signal SQLSTATE '45000' SET message_text = 'Campo già occupato da
una prenotazione';
END IF;

SELECT Costo_Prenotazione_Ore INTO costoCampo
FROM Campo_Da_Gioco
WHERE Nome = NEW.Campo_Da_Gioco;
SET NEW.Costo_Totale = costoCampo * NEW.Durata_Ore;
END; //
delimiter ;
```

Event Scheduler CalcoloIscrittiPerAttivita

```
DELIMITER $$
CREATE EVENT `CalcoloIscrittiPerAttivita` ON SCHEDULE EVERY 1 WEEK
STARTS '2021-07-21 00:00:00' ON COMPLETION NOT PRESERVE ENABLE DO BEGIN

DECLARE numPartecipanti INT;
DECLARE nomeAttivita VARCHAR(30);
DECLARE finished INTEGER DEFAULT 0;
DECLARE curs CURSOR FOR
    SELECT Nome
    FROM Attivita;
DECLARE CONTINUE handler FOR NOT found SET finished = 1;
OPEN curs;
upLoop : loop FETCH curs INTO nomeAttivita;
    IF finished = 1
        THEN leave upLoop;
    END IF;

    SELECT COUNT(*) INTO numPartecipanti
    FROM Consente
    WHERE Attivita = nomeAttivita
        AND AbbonamentoValido(Abbonamento_Data_Inizio,
        Abbonamento_Cliente, now()) = 1;

    UPDATE Attivita
    SET Num_Iscritti = numPartecipanti
    WHERE Nome = nomeAttivita;
END loop upLoop;
END;$$
delimiter ;
```

Function GiornoDellaSettimana

```
delimiter //
CREATE FUNCTION GiornoDellaSettimana(giorno DATE) RETURNS VARCHAR(20)
DETERMINISTIC BEGIN

DECLARE giornoSettimana INT DEFAULT dayofweek(giorno);

IF(giornoSettimana = 1) THEN RETURN 'Domenica';
END IF;
IF(giornoSettimana = 2) THEN RETURN 'Lunedì';
END IF;
IF(giornoSettimana = 3) THEN RETURN 'Martedì';
END IF;
IF(giornoSettimana = 4) THEN RETURN 'Mercoledì';
END IF;
IF(giornoSettimana = 5) THEN RETURN 'Giovedì';
END IF;
IF(giornoSettimana = 6) THEN RETURN 'Venerdì';
END IF;
IF(giornoSettimana = 7) THEN RETURN 'Sabato';
END IF;

END;//
delimiter ;
```

Function InserisciAbbonamento

delimiter //

```
CREATE FUNCTION `InserisciAbbonamento`(Data_inizio Date, CF char(16),  
durata int, NomeAttivita varchar(30)) RETURNS int  
DETERMINISTIC
```

```
BEGIN
```

```
if(not exists(SELECT * from Attivita WHERE Nome = NomeAttivita))  
    then signal sqlstate '45000' set message_text = "Attività non  
    presente!";
```

```
end if;
```

```
INSERT INTO Abbonamento VALUES(Data_inizio, CF, 0, durata);
```

```
INSERT INTO Consente VALUES(NomeAttivita, Data_inizio, CF);
```

```
RETURN 1;
```

```
END; //
```

delimiter ;

5.3 Definizione delle interrogazioni per la visualizzazione dei dati

5.3.1 Script SQL-DML : analisi puntuali

Modificare l'orario di una lezione

```
UPDATE Lezione
SET Orario_Inizio = 'nuovo_orario (00:00:00)'
WHERE Luogo_Activita = 'nome_luogo_attivita'
      AND Giorno = 'Giorno_Della_Settimana'
      AND Orario_Inizio = 'vecchio_orario (00:00:00)';
```

Visualizzare tutti le lezioni tenute da un determinato istruttore in un dato giorno della settimana.

```
SELECT *
FROM Lezione
WHERE Giorno = 'Giorno_Della_Settimana'
      AND Istruttore = 'Codice_Fiscale_Istruttore';
```

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

```
SELECT Abbonamento_Cliente
FROM Consente
GROUP BY Abbonamento_Cliente, Abbonamento_Data_Inizio
HAVING COUNT(*) >= ALL (SELECT COUNT(*)
                        FROM Consente
                        GROUP BY Abbonamento_Cliente,
                                Abbonamento_Data_Inizio
                        )
```

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

```
SELECT DISTINCT Cliente
FROM prenotazione AS p1
WHERE EXISTS (SELECT *
              FROM prenotazione AS p2
              WHERE YEAR(p1.Data) = YEAR(p2.Data)
                    AND p1.Cliente = p2.Cliente
                    AND weekofyear(p1.Data) = weekofyear(p2.Data)
                    AND p1.Campo_Da_Gioco = p2.Campo_Da_Gioco
                    AND p1.Ora != p2.Ora
              );
```

Visualizzare gli istruttori che hanno solo il brevetto di nuoto.

```
SELECT DISTINCT Istruttore
FROM Brevetto AS B1
WHERE Attivita = 'Nuoto' AND
NOT EXISTS (SELECT Istruttore
            FROM Brevetto AS B2
            WHERE B2.Attivita != 'Nuoto'
            AND B1.Istruttore = B2.Istruttore
            );
```


Snellire le prenotazioni spostando le vecchie di oltre un anno in una entità separata (PRENOTAZIONI_STORICHE). Fare questa operazione ogni fine anno.

Event Scheduler **InserisciPrenotazioniStoriche**

```
delimiter //
CREATE EVENT `InserisciPrenotazioniStoriche` ON SCHEDULE EVERY 1
YEAR STARTS '2021-01-01 00:00:00'
DO BEGIN
INSERT INTO Prenotazione_Storica(SELECT *
                                FROM Prenotazione
                                WHERE date_add(Data, INTERVAL 1
                                YEAR) <= now());

DELETE FROM Prenotazione
WHERE date_add(Data, INTERVAL 1 YEAR) <= now();
END//
DELIMITER ;
```

5.3.2 Script SQL-DML : analisi aggregate

Inserire i dati relativi ad un nuovo cliente

```
insert into Persona values('codice_fiscale', 'nome', 'cognome',
'S' , 'Luogo_di_nascita', 'YYYY/MM/DD');
insert into Cliente values('codice_fiscale');
```

Eliminare i dati di un determinato medico.

Function **EliminaMedico**

```
delimiter //
CREATE FUNCTION EliminaMedico (cf CHAR(16)) RETURNS bool
DETERMINISTIC BEGIN

DELETE FROM visita_medica WHERE Medico = cf;
DELETE FROM medico WHERE Codice_Fiscale = cf;
IF(NOT EXISTS (SELECT * FROM Cliente
               WHERE Codice_Fiscale = cf )
)
THEN DELETE FROM persona WHERE Codice_Fiscale = cf;
RETURN 0;
END IF;
RETURN 1;
END ; //
```

```
SELECT EliminaMedico(CF_Medico);
```

Visualizzare il numero di abbonamenti sottoscritti da ciascun cliente.

```
SELECT COUNT(*), Cliente
FROM(SELECT Data_Rinnovo, Cliente_Abbonamento AS Cliente
FROM Rinnovo
UNION
SELECT Data_Inizio, Cliente
FROM Abbonamento
)AS Client
GROUP BY Cliente;
```

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

View AttivitaPerSala

```
CREATE VIEW AttivitaPerSala (Sala, Num_Activita) AS
SELECT L1.Luogo_Activita, COUNT(DISTINCT attivita)
FROM Lezione AS L1, Luogo_Activita
WHERE L1.Luogo_Activita = Luogo_Activita.nome
      AND Luogo_Activita.Tipo_Luogo = 'S'
GROUP BY L1.Luogo_Activita;
```

```
SELECT Sala
FROM AttivitaPerSala
WHERE Num_Activita = (SELECT MIN(Num_Activita)
                      FROM AttivitaPerSala);
```

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

View MediciClienti

```
CREATE VIEW MediciClienti(Codice_Fiscale, abbonamentoValido) AS
SELECT Cliente.Codice_Fiscale, Data_Inizio
FROM Cliente, Medico, Abbonamento
WHERE Cliente.Codice_Fiscale = Medico.Codice_Fiscale
      AND Abbonamento.Cliente = Cliente.Codice_Fiscale
      AND AbbonamentoValido(Data_Inizio, Cliente.Codice_Fiscale,
                              now()) = 1;
```

```
SELECT DISTINCT Codice_Fiscale
FROM MediciClienti, Consente, Lezione
WHERE Codice_Fiscale = Abbonamento_Cliente
      AND abbonamentoValido = Abbonamento_Data_Inizio
      AND Consente.Attivita = Lezione.Attivita
      AND (
          (time_to_sec(Orario_Inizio) + (Durata_Minuti*60)) <
          time_to_sec('Orario_interessato')
      );
```

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

```
Trigger SvolgimentoAttivita
delimiter //
CREATE TRIGGER SvolgimentoAttivita BEFORE INSERT ON Lezione FOR
EACH ROW BEGIN
DECLARE LuogoAttivita ENUM('S', 'C');
DECLARE LuogoLezione ENUM('S', 'C');

SELECT Tipo_Luogo INTO LuogoAttivita
FROM Attivita
WHERE Attivita.Nome = NEW.Attivita;
IF EXISTS (SELECT *
           FROM Sala
           WHERE NEW.Luogo_Activita = Sala.Nome
          )
    THEN SET LuogoLezione = 'S';
ELSE
    SET LuogoLezione = 'C';
END IF;

IF (LuogoAttivita != LuogoLezione)
    THEN signal SQLSTATE '45000' SET message_text = 'Errore:
    1\'attivita non si svolge nel luogo adatto';
END IF;
END; //
delimiter ;
```