

TotalGym C.S

Simone Nicolae Mija
Nathaniele Rios Tomines
Data Consegna: 30/06/2021

Introduzione

Il centro sportivo TotalGym ha la necessità di modernizzarsi in seguito all'aumento dei clienti, con l'obiettivo di fornire ai propri dipendenti un sistema più efficiente di gestione dei propri abbonati e fornire servizi innovativi a quest'ultimi, con lo scopo finale di migliorare l'intera organizzazione aziendale attraverso un sistema informatico (database). L'impianto è provvisto di una piscina semiolimpionica, una palestra con sei sale di cui una sala attrezzi, due campi per il calcio a5 e tre campi da tennis in sintetico polivalenti; inoltre vanta di 400 tesserati e oltre 30 istruttori agonistici dedicati ad ogni specifico sport. L'associazione fornisce, oltre ai comuni corsi, anche attrezzatura sportiva, certificati medici e, su prenotazione, campi da calcio e/o tennis. Si vuole inoltre creare un'applicazione che permetta allo staff di iscrivere un determinato nuovo cliente, o modificare alcune informazioni di un iscritto del centro sportivo.

Descrizione nel dettaglio

Si vuole progettare una base di dati del centro sportivo TotalGym C.S. dove vogliamo rappresentare i dati dei clienti e dello staff.

I **clienti** registrati al centro sportivo saranno identificati con il proprio codice fiscale e saranno forniti anche informazioni sul *proprio nome, cognome, Città di residenza, indirizzo di residenza, sesso, data di nascita ed un'immagine scannerizzata del documento*.

Per chi è maggiorenne se è in possesso di un **auto** e di una **patente**, bisogna memorizzare la *targa della macchina, modello e codice patente*: ciò permette l'uso del parcheggio agli abbonati.

Il cliente sarà registrato da uno **membro dello staff** e più clienti possono venir registrati dallo stesso membro dello staff.

Inoltre, una volta iscritto, il cliente avrà una tessera che permetterà l'accesso alla struttura e un **abbonamento**. Quest'ultimo ha una *data di scadenza*, che viene usata per permettere o meno l'accesso all'associazione sportiva, e un *determinato tipo*, che indica i corsi che si possono seguire.

Ogni abbonamento è associato ad un singolo cliente e un cliente può avere più di un abbonamento.

Il cliente potrà visualizzare sia la **fattura** da pagare che quelli già pagati, dove ogni fattura avrà il proprio codice di fattura, la *data* e il *prezzo dell'abbonamento*.

Al pagamento di una fattura, la scadenza dell'abbonamento verrà rinnovata.

Il cliente inoltre dovrà avere un **certificato medico** valido ogni volta che andrà al centro sportivo; se quest'ultimo scade, l'accesso alla struttura verrà vietato.

A causa dell'emergenza del covid e secondo le nuove regole, per evitare assembramenti all'interno della struttura, ogni cliente deve prenotarsi ad un determinato corso attraverso il sito dell'associazione sportiva.

Un cliente può avere più **prenotazioni** in giorni diversi, ma ne può avere una al giorno: questo permette l'accesso alla struttura a tutte le persone.

Ogni membro dello **staff** sarà identificato con un codice unico per ogni membro e saranno salvati all'interno della base di dati: codice fiscale, *nome, cognome, residenza e anni di lavoro all'interno della TotalGym C.S.*

Tra i membri dello staff possiamo avere anche dei Trainer che si occupano della gestione dei corsi, dove ciascun trainer deve gestire almeno un corso.

Dei **Trainer** abbiamo informazioni sul proprio codice fiscale, *nome, cognome e la sua specializzazione* (Esempio: Trainer della palestra).

Una persona può seguire giornalmente un corso e un corso può essere seguito da più persone.

I **corsi** avranno informazioni sul *quanti clienti possono ospitare*: questi ultimi vengono eseguiti dai trainer.

Inoltre per ogni cliente saranno salvati anche i corsi che ha seguito durante tutta la sua permanenza nel centro sportivo.

Ogni cliente, se ha l'abbonamento della sala pesi, possono scegliere una specifica **scheda di allenamento**, unica e personale per ogni cliente; quando quest'ultima scade dovrà essere assegnata un'altra scheda da parte del trainer; solo i trainer che lavorano nella sala pesi, possono creare ai clienti apposite schede di allenamento. In una scheda di allenamento saranno disponibili *informazioni sul trainer che ha realizzato la scheda, il titolo, una semplice descrizione, l'insieme di esercizi e la sua durata*. Ogni scheda di allenamento sarà composta da più esercizi e un esercizio può appartenere a più schede di allenamento. In un **esercizio** è necessario salvare informazioni sul *nome dell'esercizio, il numero di serie e il numero di ripetizioni*.

All'interno dello staff vi sono anche i **Gestori**, che hanno la possibilità di creare tornei non agonistici a squadre (esempio: torneo di calcio), singoli (esempio: torneo di tennis singolo) ed acquistare attrezzature dai fornitori.

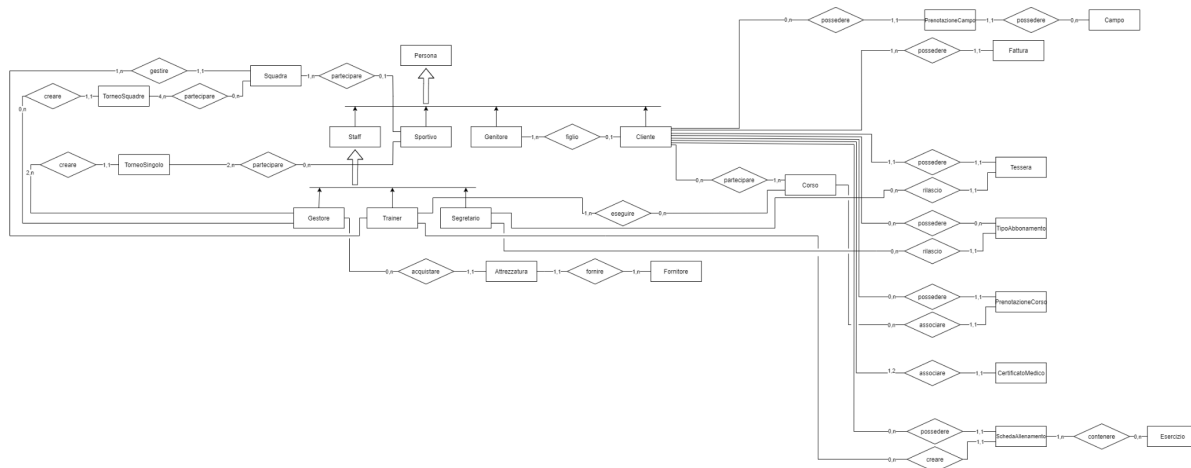
Per ogni **prodotto** che acquistano infatti, viene inserito nel database le informazioni del bene comprato e del fornitore.

Oltre ai comuni clienti, vi sono anche gli **sportivi**, persone che competono nei tornei e possono far parte di una squadra atletica di un determinato sport. Uno sportivo può anche non appartenere a nessuna squadra (es: i nuotatori).

Una **Squadra** è composta da un *gruppo di sportivi*, è unica e gestita da un singolo *trainer*.

Schema E/R

In seguito all'analisi della descrizione dettagliata, riusciamo a ricavare il seguente schema E/R

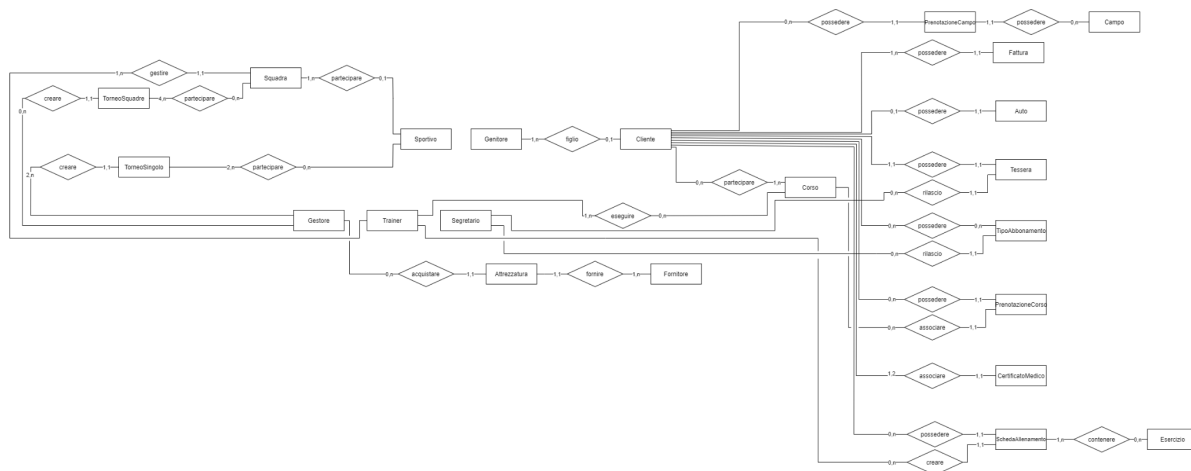


(consultare file ER Diagram.png)

Per motivi di visibilità/ordine, si è evitato di mettere gli attributi nello schema, visto che verranno specificati nel dizionario dei termini.

Ristrutturazione Schema E/R

Analizzando lo schema E/R principale, notiamo che non vi sono relazioni ternarie o maggiori e non vi sono relazioni ricorsive; quindi basta semplicemente rimuovere le relazioni ISA/Generalizzazioni (attraverso l'accorpamento) e ridefinire l'attributo multivalore "Auto" come un'entità per ottenere la ristrutturazione dello Schema E/R



(consultare file ER rimodellato.png)

Dizionario dello Schema Ristrutturato

Per comodità, visto che ogni relazione viene usata soltanto per collegare le entità tra di loro e che quest'ultime non hanno attributi, abbiamo definito nel dizionario soltanto le entità

Entità	Descrizione	Attributi
Campo	Campo del centro sportivo	(<u>cod</u> ,nome,tipa,costo_ora,max_persone)
Fattura	Determinata fattura in seguito ad un pagamento di un abbonamento	(<u>cod</u> ,data,importo,cliente_associato)
Auto	Auto di un determinato cliente	(<u>targa</u> ,modello,marca,clienteAssociato)
Tessera	Tessera di un determinato cliente	(<u>cod</u> ,data_rilascio,cliente_associato,segretario_a ssociato)
Abbonamento	Indica l'abbonamento di un determinato cliente	(<u>nome</u> ,descrizione,data_creazione,tipaAbbonam ento)
PrenotazioneCorso	Prenotazione ad un determinato corso	(<u>cod</u> ,dataora_inizio,dataora_fine,cliente_associat o,corso_associato)
CertificatoMedico	Certificato medico di un determinato cliente	(<u>cod</u> ,tipo,data_scadenza,medico,cliente_associat o)
Esercizio	Esercizio fisico	(<u>cod</u> ,titolo,descrizione,categoria,image_url)
SchedaAllenamento	Scheda di allenamento contenente gli esercizi	(<u>cod</u> ,titolo,descrizione,durata,cod_fiscale_cliente, cod_fiscale_trainer)
Corso	Corso del centro sportivo	(<u>nome</u> ,max_persone,sala)
Cliente	Cliente del centro sportivo	(<u>cod_fiscale</u> ,nome,cognome, sesso,data_nascita,l uogo_nascita,cod_patente,image_url,genitore_a ssociato)
Fornitore	Un determinato fornitore che ha fornito l'attrezzatura	(<u>p_iva</u> ,nome)
Genitore	Genitore associato ad un cliente minorenne	(<u>cod_fiscale</u> ,nome cognome,sesso,data_nascita,luogo_nascita,imag e_url)
Attrezzatura	Attrezzatura comprata da parte del gestore	(<u>cod</u> ,nome,descrizione,p_iva_fornitore,cod_fiscal e_gestore)
Segretario	Segretario appartenente al centro sportivo	(<u>codice_fiscale</u> ,nome,cognome,sesso,data_nasci ta,luogo_nascita,data_lavoro,image_url)
Sportivo	Sportivo appartenente al centro sportivo	(<u>cod_fiscale</u> ,data_nascita,nome,cognome,sesso, squadra_partecipazione)

Trainer	Trainer appartenente al centro sportivo	(<u>cod_fiscale</u> , nome, cognome, sesso, specializzazione, data_lavoro)
Gestore	Gestore del centro sportivo	(<u>codice_fiscale</u> , nome, cognome, sesso, data_nascita, luogo_nascita, data_lavoro, image_url)
Squadra	Squadra del centro sportivo	(<u>cod</u> , nome, sport, trainer_associato)
TorneoSquadre	Torneo squadre di un determinato sport	(<u>cod</u> , nome, descrizione, sport, gestore_creatore)
TorneoSingolo	Torneo singolo di un determinato sport	(<u>cod</u> , nome, descrizione, sport, gestore_creatore)

Glossario dei Termini

Termini	Descrizione	Sinonimi	Collegamenti
Genitore	Persona che rappresenta un genitore del cliente minorenni	Parantela	Cliente
Cliente	Persona che è iscritto al centro sportivo	Partecipante	PrenotazioneCampo,Fattura,Auto, Tessera,TipoAbbonamento, PrenotazioneCorso,CertificatoMedico, SchedaAllenamento
PrenotazioneCampo	Prenotazione di un campo da parte di un cliente		Cliente,Campo
Campo	Campo sportivo dove un cliente può prenotare		PrenotazioneCampo
Corso	Corso sportivo dove un Cliente può partecipare		Cliente,PrenotazioneCorso
Fattura	La fattura dell'abbonamento che avviene ogni mese		Cliente
Auto	Macchina o Moto del cliente (solo maggiorenni e con patente)	Veicolo	Cliente
Tessera	Tessera del cliente rilasciato da un Segretario		Cliente,Segretario
TipoAbbonamento	Tipo di abbonamento (es.: trimestrale,annuale) del Cliente rilasciato dal Gestore		Cliente,Segretario
PrenotazioneCorso	Il Cliente partecipa ad un Corso, per andare fisicamente al corso ha bisogno di una prenotazione		Cliente,Corso
CertificatoMedico	Certificato medico di un cliente. Lo stesso cliente può avere al massimo 2 certificati medici, dove la scadenza sarà il certificato più recente		Cliente
SchedaAllenamento	Scheda di allenamento del Cliente, Scritto ad hoc da un Trainer		Cliente,Trainer
Esercizio	Tipo di esercizio che compone una scheda di allenamento		SchedaAllenamento
Segretario	Persona dello staff che è incaricato nel rilasciare Tipo di abbonamento e Tessera	Impiegato	TipoAbbonamento,Tessera

Trainer	Persona dello staff che prepara una scheda di allenamento per il cliente, organizza una squadra e fa i corsi di competenza		Corso,Squadra,SchedaAllenamento
Gestore	Persona dello staff che è incaricato di comprare Attrezzature e creare i tornei (sia individuali che a squadre)	Organizzatore	Attrezzatura, TorneoSingolo, TorneoSquadre
Attrezzatura	Prodotto comprato dal Gestore e fornito dal Fornitore	Prodotto	Gestore,Fornitore
Fornitore	Azienda che fornisce i prodotti		Attrezzatura
Sportivo	Persona che partecipa ad un torneo singolo o ad una squadra	Atleta	TorneoSingolo,Squadra
TorneoSingolo	Torneo composto da singoli atleti		Gestore,Sportivo
Squadra	Organizzazione di più atleti coordinati da un Trainer		Trainer,Gestore,Sportivo
TorneoSquadre	Torneo composto da più squadre		Gestore

Modello Relazionale

La ristrutturazione dello schema E/R ci permette di eseguire il modello relazionale:

Trainer(cod_fiscale, nome, cognome, sesso, data_nascita, anni_lavoro, specializzazione)
Corso(nome, max_persones)
Campo(cod, tipo, costo_ora, max_persones)
Genitore(cod_fiscale, nome, cognome, sesso, data_nascita, luogo_nascita, image_url,)
Cliente(cod_fiscale, nome, cognome, sesso, data_nascita, luogo_nascita, cod_patente, image_url, cod_genitore)
Gestore(cod_fiscale, nome, cognome, sesso, data_nascita, anni_lavoro)
Fornitore(p_iva, nome)
Sportivo (cod_fiscale, nome, cognome, sesso, data_)
Esercizio (titolo, categoria, ripetizioni, serie)
Squadra(cod, nome, cod_fiscale_Trainer)
Eseguiere_TC(corso, cod_fiscale_Trainer)
Partecipazione_CC(cod_fiscale_cliente, corso)
Partecipare_SS(squadra, cod_fiscale_Sportivo)
PrenotazioneCampo(cod, dataora_inizio, dataora_fine, campoAssociato, clienteAssociato)
Tessera (cod, cliente_associato, data_rilascio)
CertificatoMedico(cod, data_scadenza, medico, cliente_associato)
PrenotazioneCorso(cod, dataora_inizio, dataora_fine, cliente_associato)
Fattura (cod, data, importo, clienteAssociato)
Auto (targa , modello, marca, clienteAssociato)
TipoAbbonamento
(categoria, scadenza, descrizione, cliente_associato, gestore_associato, tipo_abbonamento)
SchedaAllenamento(cod, titolo, descrizione, durata, cod_fiscale_Cliente, cod_fiscale_Gestore)
Torneo (cod, nome, categoria, data_inizio, data_fine, cod_fiscale_Gestore)
Attrezzatura (cod, nome, cod_fiscale, p_iva_associato)
Partecipare_SqT(squadra, torneo)
Partecipare_SpT(sportivo, torneo)
Contenere_ScEs(cod_SchedaAllenamento, titolo_Esercizio)

Integrità referenziali:

- cod_fiscale_Trainer in Squadra e Trainer
- cod_fiscale_Trainer in Eseguiere_TC e Trainer
- cod_fiscale_cliente in Partecipazione_CC e Cliente
- corso in Partecipazione_CC e Corso
- squadra in Partecipare_SS e Squadra
- cod_fiscale_Sportivo Partecipare_SS e Sportivo
- clienteAssociato in PrenotazioneCampo e Cliente
- campoAssociato in PrenotazioneCampo e Campo
- cliente_associato in Tessera e Cliente
- cliente_associato in CertificatoMedico e Cliente

- ## Database Diagram



Rv4: Una scheda di allenamento è creato da un trainer che fa un corso di sala pesi

Rv5: un cliente non può iscriversi ad un corso se quest'ultimo ha raggiunto il massimo delle persone

Per risolvere ciò, abbiamo usato i Triggers attraverso l'uso del SQL

Rv1:

```
CREATE TRIGGER Max2TimesPerDay
BEFORE INSERT
ON PrenotazioneCorso
FOR EACH ROW
BEGIN
    IF new.dataora_inizio > new.dataora_fine
    THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'ERROR:
dataora_inizio > dataora_fine';
    END IF;
END;
```

Rv2:

```
CREATE TRIGGER DriverLicenseCheck
BEFORE INSERT
ON Cliente
FOR EACH ROW
BEGIN
    IF DATEDIFF(CURDATE(), new.data_nascita)/365 < 18 AND new.cod_patente
IS NOT NULL
    THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'ERROR: driver
license under of age';
    END IF;
END;
```

Rv3:

```
CREATE TRIGGER DurationUnder1WeekCheck
BEFORE INSERT
ON SchedaAllenamento
FOR EACH ROW
BEGIN
    IF DATEDIFF(new.durata, CURDATE()) < 7
    THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'ERROR: durata under
1 week';
    END IF;
END;
```

Rv4:

```
CREATE TRIGGER CreatedBySalaPesiTrainer
BEFORE INSERT
ON SchedaAllenamento
FOR EACH ROW
BEGIN
    IF (select count(*)
        from TrainerCorso
        WHERE TrainerCorso.cod_fiscale_Trainer = new.cod_fiscale_trainer
AND TrainerCorso.corso = 'sala_pesi')
```

```

) < 1 THEN
    SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'ERROR: trainer is
not doing course sala pesi';
END IF;
END;

```

Rv5:

```

CREATE TRIGGER IfCourseIsFullCheck
BEFORE INSERT
ON ClienteCorso
FOR EACH ROW
BEGIN
    IF (select count(*)
        from TrainerCorso
        WHERE corso associato = new.corso associato
        ) > (select max persone
            from Corso
            WHERE nome = new.corso associato) THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'ERROR: Max persone
reached for the course';
    END IF;
END;

```

Tavola dei Volumi

Concetto	Tipo	Volume
Cliente	E	3000
Corso	E	100
PrenotazioneCampo	E	30000
Campo	E	100
Fattura	E	30000
Auto	E	3000
Tessera	E	3000
TipoAbbonamento	E	20
PrenotazioneCorso	E	3000
CertificatoMedico	E	6000
SchedaAllenamento	E	3000
Esercizio	E	200
Genitore	E	3000
Sportivo	E	3300

Staff	E	200
Persona	E	9500
Gestore	E	190
Trainer	E	190
ToneoSingolo	E	100
Squadra	E	32
TorneoSquadre	E	16
Attrezzatura	E	500
Fornitore	E	5
Possedere_PrencampCam	R	30000
Possedere_CliPrencamp	R	30000
Possedere_CliFat	R	30000
Possedere_CliAut	R	3000
Possedere_CliTes	R	3000
Rilascio_TesSeg	R	3000
Possedere_CliTipoabb	R	60000
Rilascio_TipabbSeg	R	20
Possedere_CliPrencor	R	3000
Associare_PrencorCor	R	3000
Associare_CliCer	R	6000
Possedere_CliSchedall	R	3000
Creare_SchedallTrainer	R	3000
Contenere_SchedallEser	R	600000
Partecipare_CliCor	R	300000
Eseguire_CorTra	R	19000
Fornire	R	500
Aquistare	R	500
Partecipare_SqSp	R	3300
Partecipare_TorsingSp	R	330000
Partecipare_TorsqSq	R	512
Gestire_SqTr	R	32
Creare_TorsqGes	R	16
Creare_TorsingGes	R	100

Operazioni SQL

Op1: I clienti maggiorenni che non hanno una macchina registrata

```
CREATE VIEW ClientiConAuto AS
SELECT clienteAssociato FROM Auto;

SELECT * FROM Cliente
WHERE DATEDIFF(CURDATE(), Cliente.data_nascita)/365 >= 18 AND
Cliente.cod_fiscale NOT IN (SELECT * FROM ClientiConAuto)
```

Op2: dato il codice fiscale di un gestore si vuole ricavare una lista dei nomi dei fornitori che hanno fornito l'attrezzatura per quel gestore

```
SELECT Fornitore.nome
FROM Fornitore
WHERE Fornitore.p_iva IN (SELECT Attrezzatura.p_iva_fornitore FROM
Attrezzatura WHERE Attrezzatura.cod_fiscale_gestore =
'ZHLVML80C12Z155E')
```

Op3: lista dei gestori che ha comprato N attrezzature

```
SELECT Gestore.codice_fiscale, Gestore.nome, Gestore.cognome,
COUNT(*) AS n
FROM Gestore, Attrezzatura
WHERE Attrezzatura.cod_fiscale_gestore = Gestore.codice_fiscale
GROUP BY Gestore.codice_fiscale
HAVING n = 1
```

Op4: nome e cognome dei trainer che gestiscono almeno una squadra e che hanno almeno realizzato una scheda di allenamento

```
SELECT Trainer.nome, Trainer.cognome
FROM Trainer
WHERE Trainer.cod_fiscale IN (SELECT Squadra.trainer_associato FROM
Squadra)
AND Trainer.cod_fiscale IN (SELECT
SchedaAllenamento.cod_fiscale_trainer FROM SchedaAllenamento)
```

Op5: Il numero dei clienti minorenni prenotati in un corso in una data iniziale specifica

```
SELECT COUNT(*)
FROM Cliente
```

```
WHERE DATEDIFF(CURDATE(), Cliente.data_nascita)/365 < 18
AND Cliente.cod_fiscale IN
    (SELECT PrenotazioneCorso.cliente_associato
    FROM PrenotazioneCorso
    WHERE PrenotazioneCorso.dataora_inizio = '2021-06-29
16:20:41')
```

Op6:Dato il codice fiscale del cliente e il codice di un campo si vuole vedere la sua prenotazione del campo più recente

```
SELECT PrenotazioneCampo.*
FROM PrenotazioneCampo
WHERE ClienteAssociato = 'PNNNLG71P55A960W' AND
    CampoAssociato=
'A2321' AND
    dataora_inizio IN
    (Select MIN(dataora_inizio ) FROM PrenotazioneCampo )
```

Op7:Lista dei genitori che hanno due figli minorenni iscritti

```
CREATE VIEW genitoriConDueFigli AS
SELECT genitore_associato,COUNT(*) AS n
FROM Cliente
WHERE DATEDIFF(CURDATE(), Cliente.data_nascita)/365 < 18
GROUP BY genitore_associato
HAVING n= 2;
```

```
SELECT Genitore.*
FROM Genitore
WHERE Genitore.cod_fiscale IN (SELECT genitore_associato FROM
genitoriConDueFigli)
```

Op8:tutti i nomi dei gestori con il numero totale degli acquisti effettuati

```
SELECT Gestore.nome,COUNT(Attrezzatura.cod)
FROM Gestore,Attrezzatura
WHERE Attrezzatura.cod_fiscale_gestore = Gestore.codice_fiscale
GROUP BY Gestore.codice_fiscale
```

Op9:nome e cognome dei clienti che hanno il certificato medico scaduto (andando a vedere il certificato medico più recente perché ne abbiamo 2 per cliente)

```
CREATE VIEW CertificatoMedicoRecente AS
SELECT
CertificatoMedico.cliente_associato,MAX(CertificatoMedico.data_sca
denza) as minimum
FROM CertificatoMedico
```

```
GROUP BY CertificatoMedico.cliente_associato;
```

```
SELECT Cliente.nome, Cliente.cognome  
FROM Cliente, CertificatoMedicoRecente  
WHERE CertificatoMedicoRecente.cliente_associato =  
Cliente.cod_fiscale AND CertificatoMedicoRecente.minimum <  
CURDATE()
```

Op10: Il numero di squadre che partecipano ad un dato Torneo di squadre

```
SELECT COUNT(*)  
FROM Squadra, PartecipazioneTorneoSquadra  
WHERE PartecipazioneTorneoSquadra.torneo_squadra_associato = 1  
AND PartecipazioneTorneoSquadra.squadra_associata =  
Squadra.cod;
```

Op11: i nomi dei trainer che non hanno creato una scheda di allenamento

```
CREATE VIEW TrainersConSchedaAllenamento AS  
SELECT SchedaAllenamento.cod_fiscale_trainer  
FROM SchedaAllenamento;
```

```
SELECT Trainer.cod_fiscale  
FROM Trainer  
WHERE Trainer.cod_fiscale NOT IN (SELECT cod_fiscale_trainer FROM  
TrainersConSchedaAllenamento)
```

Op12: nome e cognome dei clienti che hanno più di una scheda di allenamento

```
CREATE VIEW CodFiscalePiuSchede AS  
SELECT cod_fiscale_cliente  
FROM SchedaAllenamento  
GROUP BY SchedaAllenamento.cod_fiscale_cliente  
HAVING COUNT(*) > 1;
```

```
SELECT nome, cognome  
FROM Cliente, CodFiscalePiuSchede  
WHERE cod_fiscale = CodFiscalePiuSchede.cod_fiscale_cliente
```

Op13: Nome degli esercizi che sono utilizzati da più di una scheda di allenamento

```
CREATE VIEW CodEserciziUsatiPiuVolte AS  
SELECT SchedaAllenamentoEsercizio.esercizio_associato  
FROM SchedaAllenamentoEsercizio  
GROUP BY esercizio_associato  
HAVING COUNT(*) > 1;
```



```
SELECT Esercizio.titolo
FROM Esercizio,CodEserciziUsatiPiuVolte
WHERE Esercizio.cod = CodEserciziUsatiPiuVolte.esercizio_associato
```

Op14:nome e cognome dei clienti che hanno prenotato sia per un corso che per un campo

```
CREATE VIEW PrenotatiCampo AS
SELECT cliente_associato
FROM PrenotazioneCorso;
```

```
CREATE VIEW PrenotatiCorso AS
SELECT clienteAssociato
FROM PrenotazioneCampo;
```

```
SELECT Cliente.cod_fiscale
FROM PrenotatiCampo,PrenotatiCorso,Cliente
WHERE Cliente.cod_fiscale = cliente_associato AND cod_fiscale =
clienteAssociato
```

Op15:Nome dei fornitori con il numero delle proprie attrezzature fornite

```
SELECT Fornitore.p_iva,Fornitore.nome, COUNT(*) AS n
FROM Fornitore,Attrezzatura
WHERE Attrezzatura.p_iva_fornitore = Fornitore.p_iva
GROUP BY Fornitore.p_iva
```

Op16: Nome e Cognome dei genitori che hanno un figlio con il certificato medico scaduto e dove il certificato medico è solo uno.

```
CREATE VIEW ContaCerificatiMedici AS
SELECT cliente_associato,count(*) AS n
FROM CertificatoMedico
GROUP BY cliente_associato;
```

```
CREATE VIEW ClienteConUnCertificatoMedicoScaduto AS
SELECT CertificatoMedico.cliente_associato
FROM CertificatoMedico
WHERE CertificatoMedico.cod IN
(
SELECT CertificatoMedico.cod
FROM CertificatoMedico,ContaCerificatiMedici
WHERE n=1
) AND CertificatoMedico.data_scadenza < CURDATE();
```

```
SELECT Genitore.nome,Genitore.cognome
FROM Cliente,Genitore
```

```
WHERE Cliente.genitore_associato = Genitore.cod_fiscale AND
Cliente.cod_fiscale IN (SELECT cliente_associato FROM
ClienteConUnCertificatoMedicoScaduto )
```

Tavola degli Accessi

In seguito alle operazioni SQL effettuate, stipuliamo una tavola degli accessi su alcune operazioni

Op1:	i clienti maggiorenni che non hanno una macchina registrata				
Concetto	tipo	costrutto	frequenza		
Cliente	L	E	1		
Possedere_CliTes	L	R	1		
Tessera	L	E	1		
Op2:	dato un gestore si vuole ricavare una lista dei nomi dei fornitori che hanno fornito l'attrezzatura				
Concetto	tipo	costrutto	frequenza		
Gestore	L	E	1		
Acquistare	L	R	500		
Attrezzatura	L	E	500		
Fornire	L	R	500		
Fornitore	L	E	500		

Op3:	lista dei gestori che hanno comprato 10 attrezzature					
Concetto	tipo	costrutto	frequenza			
Attrezzatura	L	E	500			
Acquistare	L	R	500			
Gestore	L	E	190			
Op4:	i nomi dei trainer che gestiscono almeno una squadra e che hanno almeno realizzato una scheda di allenamento					
Concetto	tipo	costrutto	frequenza			
Trainer	L	E	190			
Creare_SchedaAllTrainer	L	R	3000			
SchedeAllenamento	L	E	3000			
Gestire_SqTr	L	R	32			
Squadra	L	E	16			

Ristrutturazione del modello Relazionale

A seguito del tavolo degli accessi si è rilevato che non occorre ristrutturare il modello relazionale

Requisiti minimi

REQUISITI MINIMI		STATO LAVORO	MOTIVAZIONE
A.	Deve prendere in considerazione un dominio di interesse che consente di identificare almeno 12 entità concettuali principali. Questo senza contare le sotto-entità che compaiono nelle relazioni ISA e/o nelle generalizzazioni	Done	Sono state rilevate oltre 16 entità all'interno dello schema E/R, escludendo quelle che fanno parte delle generalizzazioni o degli ISA (pagina 4)
B.	Deve avere una complessità tale per cui lo schema ER contiene sia relazioni ISA che generalizzazioni.	Done	Vi sono relazioni ISA (Persona) e generalizzazioni (Staff) (pagina 4)
C.	Deve esserci una struttura sufficiente nelle relazioni. Questo comporta che se si vede il diagramma ER come un grafo (in cui i nodi sono le entità e gli archi sono dati dalla partecipazione delle entità nelle relazioni) questo deve contenere dei cicli.	Done	Creando un grafo, si può notare il ciclo: (Cliente, Corso, PrenotazioneCorso)
D.	Lo schema deve contenere vincoli di cardinalità sulla partecipazione delle entità a relazioni diverse dal valore predefinito (0, n).	Done	Vi sono parecchie relazioni (n,n) oppure (1,1) all'interno dello schema E/R (pagina 4)
E.	Lo schema deve contenere alcuni attributi facoltativi e alcuni multi-valore.	Done	Cliente ha l'attributo genitore che è di tipo opzionale Cliente ha l'attributo multivalore auto, che successivamente verrà convertito in (Vedere dizionario pagine 5-6)
F.	Devono inoltre esserci dei vincoli esterni (almeno 4) che non possono essere rappresentati nel modello ER.	Done	Vi sono 5 vincoli esterni, implementati con l'utilizzo dei triggers (pagina 10)
G.	Le specifiche devono includere un'indicazione dei volumi per le varie entità e relazioni (prestare attenzione alla coerenza tra i volumi e i vincoli di cardinalità dello schema ER).	Done	Pagina 12-13
H.	Le specifiche devono includere un carico di lavoro delle query e delle operazioni più comuni (almeno 15) che sono di interesse nel dominio modellato, con un'indicazione della loro frequenza. Le query devono avere i seguenti requisiti minimi:	Done	A partire da pagina 13
	I) Devono richiedere l'utilizzo degli operatori aggregati	Done	Le operazioni che ne fanno uso sono: op3, op5, op6, op7, op8, op9, op10, op12, op13, op15, op16
	II) Devono richiedere l'utilizzo di query annidate (di cui una con almeno tre livelli)	Done	Le operazioni che ne fanno uso sono: op1, op2, op4, op5, op6, op7, op9, op11, op12, op13, op14, op16
	III) La metà delle query deve anche essere scritta	Done	Le operazioni che coinvolgono op1, op7, op9,

utilizzando le viste		op11, op12, op13, op14, op16
IV) Devono coinvolgere almeno due tabelle	Done	Tutti le query coinvolgono almeno 2 tabelle