

Laboratorio
Corso di Basi di Dati e Web
A.A. 2019/2020

Progetto "Catena di supermercati"

Gruppo composto da:

Martina Baga martina.baga@studenti.unimi.it

Emanuela Elli emanuela.elli@studenti.unimi.it

Data di consegna: *27/07/2020*

Studenti che intendono partecipare alla prova:

Martina Baga

Emanuela Elli

Il progetto è già stato consegnato in precedenza? No

Se sì, indica tutte le date degli appelli in cui il progetto è stato consegnato:

Se sì, indica brevemente (tre righe al massimo) le modifiche apportate rispetto all'ultimo progetto consegnato:

1. Progettazione concettuale

In questa sezione vanno riportate le conclusioni della progettazione concettuale. Non siamo interessati ai vari passaggi con i quali arrivate a costruire lo schema concettuale. Vogliamo lo schema concettuale "finale".

1.1 Schema ER

Lo schema concettuale proposto deve utilizzare al meglio i costrutti presenti nel modello ER (Entità, associazioni, gerarchie di generazioni, identificazione interne ed esterne, attributi multi-valore e composti). Lo schema ER deve essere facilmente leggibile. "A colpo d'occhio" deve essere chiaro quello che lo schema vuole rappresentare. Entità e associazioni devono avere nomi distinti e di facile comprensione.

Lo schema ER deve contenere tutte le entità ed associazioni che verranno utilizzate nel modello logico e poi nell'implementazione vera e propria. Non è ammesso che nelle fasi successive vengano aggiunte nuove informazioni. Se in una fase successiva del progetto vi accorgete di esservi "dimenticati qualcosa", dovete rimettere mano allo schema concettuale.

Schema ER in allegato.

1.2 Vincoli di dominio

In questa sezione vanno riportati e numerati tutti i vincoli che devono essere garantiti nella base di dati e che non si possono direttamente desumere dallo schema ER. Si noti che non è necessario riportare i vincoli di chiave e/o di cardinalità. Questi si desumono direttamente dallo schema ER.

La numerazione dei vincoli permette una più facile identificazione nelle fasi successive della progettazione ed implementazione.

Vincoli impliciti dello schema ER

- *Si memorizzano le informazioni dei lavoratori relative solo ai responsabili e agli impiegati (gerarchia totale ed esclusiva) che lavorano in un solo reparto di un supermercato.*
- *Un supermercato è composto almeno da un reparto in cui deve esserci un solo responsabile e almeno un impiegato.*
- *Un supermercato per completezza si identifica con la via e il numero civico in cui si trova ma supponiamo che non ci siano due supermercati nella stessa via perché si trovano tutti nella stessa città e appartengono alla stessa catena.*

Il lavoratore:

- *può essere un responsabile di uno e un solo reparto oppure essere un impiegato sempre di un solo reparto;*
- *ogni impiegato ha una specifica mansione;*
- *ad ogni lavoratore è associato almeno un turno;*
- *ogni lavoratore ha un solo specifico livello che ne determina lo stipendio (il livello è determinato dall'anzianità di servizio);*

- un responsabile/impiegato può essere licenziato e sostituito successivamente (metodo della storicizzazione).

Altri vincoli:

- Un prodotto può essere venduto (oppure no) da un reparto di un supermercato.

- Il reparto vende almeno un prodotto.

- Si mantengono le informazioni anche di quei prodotti che non sono venduti dal reparto del supermercato.

- Ogni supermercato ha obbligatoriamente un orario di apertura e di chiusura che si distingue in orario ordinario, nel caso in cui venga specificato l'attributo `giornoSettimana`, e in orario straordinario, nel caso in cui venga specificato l'attributo `dataStraordinario`.

- Ogni supermercato ha una lista di oggetti premio (almeno uno) che va a costituire il catalogo.

- Un oggetto premio può essere inserito nel database e non appartenere ancora al "catalogo" di un supermercato quindi la lista degli oggetti premio può variare nel tempo.

- Ogni supermercato ha almeno un magazzino dove riporre i prodotti (almeno uno).

- Un prodotto potrebbe essere contenuto nel magazzino oppure venduto direttamente nel reparto (ad esempio prodotti freschi).

- Il supermercato può fare un ordine ad un fornitore (o più fornitori) di uno o più prodotti.

- Il supermercato può fare più ordini dallo stesso fornitore per lo stesso prodotto in date diverse (in formato timestamp).

- Il fornitore rifornisce i prodotti richiesti dal supermercato con una stima del tempo medio di consegna, un prezzo e un codice di prodotto differente dal codice interno usato dal supermercato.

- Un prodotto può essere composto da uno o più prodotti del supermercato/fornitore.

- Un sottoinsieme dei prodotti sono le materie prime del supermercato.

- Ogni supermercato, quando vende i prodotti ai clienti, rilascia uno scontrino fiscale che indica anche i prodotti acquistati e le rispettive quantità. Inoltre viene specificato il numero della tessera nel caso in cui il cliente sia registrato.

- Il cliente può aderire al programma di fidelizzazione e nel caso entra in possesso di una tessera per accumulare i punti per ottenere un premio.

Vincoli che dipendono dalla semantica

- I prodotti, prima che la loro quantità scenda al di sotto della soglia minima (di prodotti presenti nel supermercato) devono essere ordinati nuovamente dal/dai fornitore/i.

- Ogni volta che il cliente ritira un premio vengono scalati i corrispondenti punti dalla tessera.

- Ogni volta che un cliente compra dei prodotti a cui sono associati dei punti, questi vengono aggiunti ai punti totali della tessera del cliente.

- Ogni volta che viene acquistato un prodotto viene aggiornata la disponibilità del prodotto in magazzino.

- I lavoratori di un certo reparto lavorano per il supermercato in cui si trova il medesimo reparto.

- *L'attributo giornoSettimana dell'entità orario viene omissso nel caso in cui si tratti di orarioStraordinario.*

2. Progettazione logica

2.1 Schema ER ristrutturato

In questa sezione va riportato lo schema ER ristrutturato secondo la metodologia presentata a lezione e contenuta nel libro di testo. Questo schema, a parte la ristrutturazione, deve essere equivalente a quello presentato nel Paragrafo 1.1.

Schema ER ristrutturato in allegato.

2.2 Vincoli di dominio

La ristrutturazione dello schema può portare ad introdurre nuovi vincoli sullo schema. Indicare tutti i vincoli di dominio (compresi quelli già definiti nella progettazione concettuale) evidenziando quali sono stati aggiunti in fase di progettazione logica.

Vincoli:

- *L'insieme dei responsabili è disgiunto da quello degli impiegati cioè non può esistere una stessa persona che sia responsabile e allo stesso tempo impiegato di quel reparto.*
- *Materia prima è un valore booleano di prodotto che indica se è effettivamente una materia prima (true) oppure no (false).*
- *L'attributo dataStraordinario dell'entità orario viene omissso se l'attributo giornoSettimana è specificato, viceversa se dataStraordinario viene definito allora giornoSettimana viene omissso.*
- *Se mailAvviso è false allora si può procedere con il ritiro, viceversa se mailAvviso è true allora non si può procedere con il ritiro.*
- *Se la disponibilità di PremioSpeciale è false non può essere prenotato dal cliente, viceversa se la disponibilità è true può essere prenotato dal cliente*
- *Il cliente può prenotare il premio speciale, in modalità acquisto con punti, sse i suoi punti superano (o sono pari) al numero di punti necessario per acquistare il premio speciale*

2.3 Modello relazionale

Riportare il modello relazionale.

Turno(numT, inizioT, fineT)

Livello(numL, stipendio)

Responsabile(CE, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, numTurno, numLivello)

FOREIGN KEY (numTurno) REFERENCES Turno (numT)

FOREIGN KEY (numLivello) REFERENCES Livello (numL)

Impiegato(CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, mansione, numTurno, numLivello)

FOREIGN KEY (numTurno) REFERENCES Turno (numT)

FOREIGN KEY (numLivello) REFERENCES Livello (numL)

Supermercato(via, numCiv, nome)

Reparto(nome, viaSupermercato, numCivSupermercato)

FOREIGN KEY (viaSupermercato, numCivSupermercato) REFERENCES Supermercato (via, numCiv)

Magazzino(CAP, via, numCiv, città, viaSupermercato, numCivSupermercato)

FOREIGN KEY (viaSupermercato, numCivSupermercato) REFERENCES Supermercato (via, numCiv)

Prodotto(codiceInterno, materiaPrima, dataScadenza, nome, categoria, prezzoAlPubblico, punti)

punti può essere NULL

Fornitore(partitaIva, ragioneSociale, email, telefono, CAP, via, numCiv, città, modalitàPagamento)

OggettoPremio(codicePremio, puntiPremio)

ClienteRegistrato(numeroTessera, nome, cognome, CF, CAP, via, numCiv, città, telefono, email, numeroPunti)

numeroPunti può essere NULL

Orario(ID, viaSupermercato, numCivSupermercato, oraInizio, oraFine, giornoSettimana, dataStraordinario)

FOREIGN KEY (viaSupermercato, numCivSupermercato) REFERENCES Supermercato (via, numCiv)

giornoSettimana può essere NULL

dataStraordinario può essere NULL

Scontrino(data, numeroScontrino, viaSupermercato, numCivSupermercato, numeroTesseraCliente)

FOREIGN KEY (viaSupermercato, numCivSupermercato) REFERENCES Supermercato (via, numCiv)

FOREIGN KEY (numeroTesseraCliente) REFERENCES ClienteRegistrato (numeroTessera)

numeroTesseraCliente può essere NULL

Ordine(dataOrdine, codiceInternoProdotto, partitaIVAforitore, viaSupermercato, numCivSupermercato, quantitàO)

FOREIGN KEY (codiceInternoProdotto) REFERENCES Prodotto (codiceInterno)

FOREIGN KEY (partitaIVAforitore) REFERENCES Fornitore (partitaIva)

FOREIGN KEY (viaSupermercato, numCivSupermercato) REFERENCES Supermercato (via, numCiv)

Supervisione(CFresponsabile, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato)

FOREIGN KEY (CFresponsabile) REFERENCES Responsabile (CF)

FOREIGN KEY (nomeReparto) REFERENCES Reparto (nome)

FOREIGN KEY (viaSupermercato, numCivSupermercato) REFERENCES Supermercato (via, numCiv)

dataFine può essere NULL

Lavora(CFimpiegato, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato)

FOREIGN KEY (CFimpiegato) REFERENCES Impiegato (CF)

FOREIGN KEY (nomeReparto) REFERENCES Reparto (nome)

FOREIGN KEY (viaSupermercato, numCivSupermercato) REFERENCES Supermercato (via, numCiv)

dataFine può essere NULL

Stoccaggio(codiceInternoProdotto, numCivMagazzino, viaMagazzino, CAPMagazzino, cittàMagazzino, quantità, sogliaMinima)

FOREIGN KEY (codiceInternoProdotto) REFERENCES Prodotto (codiceInterno)

FOREIGN KEY (numCivMagazzino, viaMagazzino, CAPMagazzino, cittàMagazzino) REFERENCES Magazzino (numCiv, via, CAP, città)

Rifornisce(partitaIVAfornitore, codiceInternoProdotto, stimaDiConsegna, prezzoF, codiceF)

FOREIGN KEY (partitaIvaFornitore) REFERENCES Fornitore (partitaIva)

FOREIGN KEY (codiceInternoProdotto) REFERENCES Prodotto (codiceInterno)

Composizione(composto, componente, quantità)

FOREIGN KEY (composto) REFERENCES Prodotto (codiceInterno)

FOREIGN KEY (componente) REFERENCES Prodotto (codiceInterno)

PS(codiceInternoProdotto, numeroScontrino, viaSupermercato, numCivSupermercato, quantità)

FOREIGN KEY (codiceInternoProdotto) REFERENCES Prodotto (codiceInterno)

FOREIGN KEY (numeroScontrino) REFERENCES Scontrino (numeroScontrino)

FOREIGN KEY (viaSupermercato, numCivSupermercato) REFERENCES Supermercato (via, numCiv)

Vende(codiceInternoProdotto, nomeReparto, viaSupermercato, numCivSupermercato)

FOREIGN KEY (codiceInternoProdotto) REFERENCES Prodotto (codiceInterno)

FOREIGN KEY (nomeReparto) REFERENCES Reparto (nome)

FOREIGN KEY (viaSupermercato, numCivSupermercato) REFERENCES Supermercato (via, numCiv)

Catalogo(viaSupermercato, numCivSupermercato, codicePremio)

FOREIGN KEY (viaSupermercato, numCivSupermercato) REFERENCES Supermercato (via, numCiv)

FOREIGN KEY (codicePremio) REFERENCES OggettoPremio (codicePremio)

PremiSpeciali(ID, codicePremio, disponibilità, puntiModalitàPagamento, denaroModalitàPagamento)

FOREIGN KEY (codicePremio, REFERENCES OggettoPremio (codicePremio,

Prenotazione(modalitàAcquisto, ID, codicePremio, numeroTessera, mailAvviso)

FOREIGN KEY (ID ,codicePremio, puntiPremio) REFERENCES PremiSpeciali (ID ,codicePremio, puntiPremio)

FOREIGN KEY (numeroTessera) REFERENCES ClienteRegistrato(numeroTessera)

mailAvviso può essere NULL

Ritiro(dataRitiro, modalitàAcquisto, ID, codicePremio, numeroTessera, numeroScontrino, viaSupermercato, numCivSupermercato)

FOREIGN KEY (modalitàAcquisto, ID, codicePremio, numeroTessera) REFERENCES Prenotazione (modalitàAcquisto, ID, codicePremio, numeroTessera)

FOREIGN KEY (numeroScontrino, viaSupermercato, numCivSupermercato) REFERENCES Scontrino (numeroScontrino, viaSupermercato, numCivSupermercato)

2.4 Codice SQL

Riportare il codice SQL di creazione e alterazione delle tabelle per attuare i vincoli individuati (in particolare vincoli di primary e foreign key, not null, unique e check).

```
CREATE TABLE Turno (  
    numT DECIMAL(1,0) PRIMARY KEY,  
    inizioT TIME NOT NULL,  
    fineT TIME NOT NULL,  
    CONSTRAINT Turno_numT_check CHECK (numT >= 0)  
);
```

```
CREATE TABLE Livello (  
    numL DECIMAL(1,0) PRIMARY KEY,  
    stipendio DECIMAL(8,2) NOT NULL UNIQUE,  
    CONSTRAINT Livello_stipendio_check CHECK (stipendio > 0),  
    CONSTRAINT Livello_numL_check CHECK (numL >= 0)  
);
```

```
CREATE TABLE Responsabile (  
    CF CHAR(16) PRIMARY KEY,  
    dataDiNascita DATE NOT NULL,  
    CAP DECIMAL(5,0) NOT NULL,
```

```

    via CHAR(20) NOT NULL,
    numCiv DECIMAL(3,0) NOT NULL,
    città CHAR(20) NOT NULL,
    nome CHAR(20) NOT NULL,
    cognome CHAR(20) NOT NULL,
    email CHAR(254) NOT NULL UNIQUE,
    numTurno DECIMAL(1,0) REFERENCES Turno(numT) ON UPDATE CASCADE,
    numLivello DECIMAL(1,0) REFERENCES Livello(numL) ON UPDATE CASCADE,
    CONSTRAINT Responsabile_numCiv_check CHECK (numCiv > 0),
    CONSTRAINT Responsabile_CAP_check CHECK (CAP > 0)
);

```

```

CREATE TABLE Supermercato (
    via CHAR(20),
    numCiv DECIMAL(3,0),
    nome CHAR(60) NOT NULL,
    PRIMARY KEY (via, numCiv),
    CONSTRAINT Supermercato_numCiv_check CHECK (numCiv > 0)
);

```

```

CREATE TABLE Reparto (
    nome CHAR(20),
    viaSupermercato CHAR(20),
    numCivSupermercato DECIMAL(3,0),
    FOREIGN KEY (viaSupermercato, numCivSupermercato)
    REFERENCES Supermercato (via, numCiv)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    PRIMARY KEY(nome, viaSupermercato, numCivSupermercato)
);

```

```

CREATE TABLE Impiegato (
    CF CHAR(16) PRIMARY KEY,
    dataDiNascita DATE NOT NULL,
    CAP DECIMAL(5,0) NOT NULL,
    via CHAR(20) NOT NULL,
    numCiv DECIMAL(3,0) NOT NULL,
    città CHAR(20) NOT NULL,
    nome CHAR(20) NOT NULL,
    cognome CHAR(20) NOT NULL,
    email CHAR(254) NOT NULL UNIQUE,
    mansione CHAR(20) NOT NULL,
    numTurno DECIMAL(1,0) REFERENCES Turno(numT) ON UPDATE CASCADE,

```



```
numLivello DECIMAL(1,0) REFERENCES Livello(numL) ON UPDATE CASCADE,  
CONSTRAINT Impiegato_numCiv_check CHECK (numCiv > 0),  
CONSTRAINT Impiegato_CAP_check CHECK (CAP > 0)  
);
```

```
CREATE TABLE Magazzino (  
    CAP DECIMAL(5,0),  
    via CHAR(20),  
    numCiv DECIMAL(3,0),  
    città CHAR(20),  
    viaSupermercato CHAR(20),  
    numCivSupermercato DECIMAL(3,0),  
    FOREIGN KEY (viaSupermercato, numCivSupermercato)  
        REFERENCES Supermercato(via, numCiv)  
        ON UPDATE CASCADE,  
    PRIMARY KEY(CAP, via, numCiv, città),  
    CONSTRAINT Magazzino_numCiv_check CHECK (numCiv > 0),  
    CONSTRAINT Magazzino_CAP_check CHECK (CAP > 0)  
);
```

```
CREATE TABLE Prodotto (  
    codiceInterno CHAR(10) PRIMARY KEY,  
    materiaPrima BOOLEAN NOT NULL DEFAULT FALSE,  
    dataScadenza DATE NOT NULL,  
    nome CHAR(20) NOT NULL,  
    categoria CHAR(20) NOT NULL,  
    prezzoAlPubblico DECIMAL(8,2) NOT NULL,  
    punti DECIMAL(4,0),  
    CONSTRAINT Prodotto_prezzoAlPubblico_check CHECK (prezzoAlPubblico >= 0),  
    CONSTRAINT Prodotto_punti_check CHECK (punti >= 0)  
);
```

```
CREATE TABLE Fornitore (  
    partitalva CHAR(27) PRIMARY KEY,  
    ragioneSociale CHAR(30) NOT NULL,  
    email CHAR(254) NOT NULL UNIQUE,  
    telefono CHAR(11) NOT NULL UNIQUE,  
    CAP DECIMAL(5,0) NOT NULL,  
    via CHAR(20) NOT NULL,  
    numCiv DECIMAL(3,0) NOT NULL,  
    città CHAR(20) NOT NULL,  
    modalitàPagamento CHAR(20) NOT NULL,
```

CHECK (modalitàPagamento in ('bonifico', 'assegno', 'contanti', 'bollettino bancario', 'carta di credito'))),

CONSTRAINT Fornitore_numCiv_check CHECK (numCiv > 0),

CONSTRAINT Fornitore_CAP_check CHECK (CAP > 0)

);

CREATE TABLE OggettoPremio (

codicePremio CHAR(10) PRIMARY KEY,

puntiPremio DECIMAL(3,0) NOT NULL,

CONSTRAINT OggettoPremio_puntiPremio_check CHECK (puntiPremio > 0)

);

CREATE TABLE ClienteRegistrato (

numeroTessera CHAR(20) PRIMARY KEY,

nome CHAR(20) NOT NULL,

cognome CHAR(20) NOT NULL,

CF CHAR(16) UNIQUE NOT NULL,

CAP DECIMAL(5,0) NOT NULL,

via CHAR(20) NOT NULL,

numCiv DECIMAL(3,0) NOT NULL,

città CHAR(20) NOT NULL,

telefono CHAR(11) UNIQUE NOT NULL,

email CHAR(254) UNIQUE NOT NULL,

numeroPunti DECIMAL(6,0),

CONSTRAINT ClienteRegistrato_numCiv_check CHECK (numCiv > 0),

CONSTRAINT ClienteRegistrato_CAP_check CHECK (CAP > 0),

CONSTRAINT ClienteRegistrato_numeroPunti_check CHECK (numeroPunti >= 0)

);

CREATE TABLE Orario (

ID CHAR(5),

viaSupermercato CHAR(20),

numCivSupermercato DECIMAL(3,0),

oraInizio TIME NOT NULL,

oraFine TIME NOT NULL,

giornoSettimana CHAR(9),

dataStraordinario DATE,

CHECK (giornoSettimana in ('lunedì', 'martedì', 'mercoledì', 'giovedì', 'venerdì', 'sabato', 'domenica'))),

FOREIGN KEY (viaSupermercato, numCivSupermercato)

REFERENCES Supermercato(via, numCiv)

ON DELETE CASCADE

ON UPDATE CASCADE,

```

PRIMARY KEY(ID, viaSupermercato, numCivSupermercato)
);

CREATE TABLE Scontrino (
    data TIMESTAMP NOT NULL,
    numeroScontrino CHAR(10),
    viaSupermercato CHAR(20),
    numCivSupermercato DECIMAL(3,0),
    numeroTesseraCliente CHAR(20) REFERENCES ClienteRegistrato(numeroTessera) ON DELETE
    CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (viaSupermercato, numCivSupermercato)
        REFERENCES Supermercato(via, numCiv)
        ON UPDATE CASCADE
        ON DELETE CASCADE,
    PRIMARY KEY(numeroScontrino, viaSupermercato, numCivSupermercato)
);

CREATE TABLE Ordine (
    dataOrdine TIMESTAMP,
    codiceInternoProdotto CHAR(10) REFERENCES Prodotto(codiceInterno) ON UPDATE
    CASCADE,
    partitaIVAfornitore CHAR(27) REFERENCES Fornitore(partitaIVA) ON UPDATE CASCADE,
    viaSupermercato CHAR(20),
    numCivSupermercato DECIMAL(3,0),
    quantitàO DECIMAL(4,0) NOT NULL,
    PRIMARY KEY(dataOrdine, codiceInternoProdotto, partitaIVAfornitore, viaSupermercato,
    numCivSupermercato),
    FOREIGN KEY (viaSupermercato, numCivSupermercato)
        REFERENCES Supermercato(via, numCiv)
        ON UPDATE CASCADE
        ON DELETE CASCADE,
    CONSTRAINT Ordine_quantitàO_check CHECK (quantitàO > 0)
);

CREATE TABLE Supervisione (
    CFresponsabile CHAR(16) REFERENCES Responsabile(CF),
    dataInizio DATE,
    dataFine DATE,
    nomeReparto CHAR(20),
    viaSupermercato CHAR(20),
    numCivSupermercato DECIMAL(3,0),
    FOREIGN KEY (nomeReparto, viaSupermercato, numCivSupermercato)
        REFERENCES Reparto (nome, viaSupermercato, numCivSupermercato)

```

*ON UPDATE CASCADE,
PRIMARY KEY (CFresponsabile, dataInizio)
);*

*CREATE TABLE Lavora (
CFimpiegato CHAR(16) REFERENCES Impiegato(CF),
dataInizio DATE,
dataFine DATE,
nomeReparto CHAR(20),
viaSupermercato CHAR(20),
numCivSupermercato DECIMAL(3,0),
FOREIGN KEY (nomeReparto, viaSupermercato, numCivSupermercato)
REFERENCES Reparto(nome, viaSupermercato, numCivSupermercato)
ON UPDATE CASCADE,
PRIMARY KEY (CFimpiegato, dataInizio)
);*

*CREATE TABLE Stoccaggio (
codiceInternoProdotto CHAR(10) REFERENCES Prodotto(codiceInterno) ON UPDATE
CASCADE,
numCivMagazzino DECIMAL(3,0),
viaMagazzino CHAR(20),
CAPMagazzino DECIMAL(5,0),
cittàMagazzino CHAR(20),
quantità DECIMAL(5,0) NOT NULL,
sogliaMinima DECIMAL(5,0) NOT NULL,
FOREIGN KEY (viaMagazzino, numCivMagazzino, CAPMagazzino, cittàMagazzino)
REFERENCES Magazzino (via, numCiv, CAP, città)
ON DELETE CASCADE
ON UPDATE CASCADE,
PRIMARY KEY(codiceInternoProdotto, numCivMagazzino, viaMagazzino, CAPMagazzino,
cittàMagazzino),
CONSTRAINT Stoccaggio_quantità_check CHECK (quantità >= 0),
CONSTRAINT Stoccaggio_sogliaMinima_check CHECK (sogliaMinima >= 0)
);*

*CREATE TABLE Rifornisce (
partitaIVAfornitore CHAR(27) REFERENCES Fornitore(partitaIVA) ON UPDATE CASCADE,
codiceInternoProdotto CHAR(10) REFERENCES Prodotto(codiceInterno) ON UPDATE
CASCADE,
stimaDiConsegna CHAR(20) NOT NULL,
prezzoF DECIMAL(8,2) NOT NULL,
codiceF CHAR(20) NOT NULL,*

```
PRIMARY KEY(partitaIVAfornitore, codiceInternoProdotto),  
CONSTRAINT Rifornisce_prezzoF_check CHECK (prezzoF >= 0)  
);
```

```
CREATE TABLE Composizione (  
composto CHAR(10) REFERENCES Prodotto(codiceInterno) ON UPDATE CASCADE,  
componente CHAR(10) REFERENCES Prodotto(codiceInterno) ON UPDATE CASCADE,  
quantità DECIMAL(5,0) NOT NULL,  
PRIMARY KEY(composto, componente),  
CONSTRAINT Composizione_quantità_check CHECK (quantità > 0)  
);
```

```
CREATE TABLE PS (  
codiceInternoProdotto CHAR(10) REFERENCES Prodotto(codiceInterno) ON UPDATE  
CASCADE,  
numeroScontrino CHAR(10),  
viaSupermercato CHAR(20),  
numCivSupermercato DECIMAL(3,0),  
quantità DECIMAL(5,0) NOT NULL,  
FOREIGN KEY (numeroScontrino, viaSupermercato, numCivSupermercato)  
REFERENCES Scontrino(numeroScontrino, viaSupermercato, numCivSupermercato)  
ON UPDATE CASCADE,  
PRIMARY KEY (codiceInternoProdotto, numeroScontrino, viaSupermercato,  
numCivSupermercato),  
CONSTRAINT PS_quantità_check CHECK (quantità > 0)  
);
```

```
CREATE TABLE Vende (  
codiceInternoProdotto CHAR(10) REFERENCES Prodotto(codiceInterno) ON UPDATE  
CASCADE,  
nomeReparto CHAR(20),  
viaSupermercato CHAR(20),  
numCivSupermercato DECIMAL(3,0),  
FOREIGN KEY (nomeReparto, viaSupermercato, numCivSupermercato)  
REFERENCES Reparto(nome, viaSupermercato, numCivSupermercato)  
ON UPDATE CASCADE,  
PRIMARY KEY(codiceInternoProdotto, nomeReparto, viaSupermercato, numCivSupermercato)  
);
```

```
CREATE TABLE Catalogo(  
viaSupermercato CHAR(20),  
numCivSupermercato DECIMAL(3,0),  
FOREIGN KEY (viaSupermercato, numCivSupermercato)
```

REFERENCES Supermercato(via, numCiv)
ON UPDATE CASCADE,
codicePremio CHAR(20) REFERENCES OggettoPremio(codicePremio)
ON UPDATE CASCADE,
PRIMARY KEY(viaSupermercato, numCivSupermercato, codicePremio)
);

INSERT INTO Turno (numT, inizioT, fineT) VALUES (1, '07:00:00', '13:30:00');
INSERT INTO Turno (numT, inizioT, fineT) VALUES (2, '13:30:00', '20:00:00');
INSERT INTO Turno (numT, inizioT, fineT) VALUES (3, '05:30:00', '08:30:00');
INSERT INTO Turno (numT, inizioT, fineT) VALUES (4, '11:00:00', '16:00:00');

INSERT INTO Livello (numL, stipendio) VALUES (1, 1000.00);
INSERT INTO Livello (numL, stipendio) VALUES (2, 2000.00);
INSERT INTO Livello (numL, stipendio) VALUES (3, 3000.00);

INSERT INTO Responsabile (CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, numTurno, numLivello) VALUES ('RSSMRA81T15F205Z', '1981-12-15', 20122, 'via Pace', 21, 'Milano', 'Mario', 'Rossi', 'mario.rossi@gmail.com', 4, 3);
INSERT INTO Responsabile (CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, numTurno, numLivello) VALUES ('FBRBNC90A51A944C', '1990-01-11', 27100, 'via Roma', 1, 'Pavia', 'Bianca', 'Fabrizi', 'bianca.fabrizi@gmail.com', 4, 3);
INSERT INTO Responsabile (CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, numTurno, numLivello) VALUES ('FNTGLI71T45D612B', '1971-12-05', 20090, 'via Tre Castelli', 90, 'Buccinasco', 'Giulia', 'Fanti', 'giuliafanti71@alice.it', 4, 3);
INSERT INTO Responsabile (CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, numTurno, numLivello) VALUES ('PRDSMN70S07E801P', '1970-11-07', 20013, 'via Rosmini', 3, 'Magenta', 'Simone', 'Parodi', 'simoneparodi@gmail.com', 4, 3);
INSERT INTO Responsabile (CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, numTurno, numLivello) VALUES ('GVZPTR68T07E801A', '1968-12-07', 20013, 'via Verdi', 3, 'Magenta', 'Pietro', 'Gavizi', 'pietro.gavizi68@gmail.com', 4, 3);
INSERT INTO Responsabile (CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, numTurno, numLivello) VALUES ('LRSNLO93D16D969M', '1993-04-16', 20013, 'via Carducci', 16, 'Vigevano', 'Noel', 'Larosa', 'noel.larosa@gmail.com', 4, 3);

INSERT INTO Supermercato (via, numCiv, nome) VALUES ('via Lorenteggio', 219, 'Esselunga Lorenteggio');
INSERT INTO Supermercato (via, numCiv, nome) VALUES ('via Andrea Solari', 29, 'Esselunga Solari');
INSERT INTO Supermercato (via, numCiv, nome) VALUES ('via G. Washington', 53, 'Esselunga Washington');

INSERT INTO Reparto (nome, viaSupermercato, numCivSupermercato) VALUES ('ortofrutta', 'via Lorenteggio', 219);

INSERT INTO Reparto (nome, viaSupermercato, numCivSupermercato) VALUES ('pescheria', 'via Andrea Solari', 29);

INSERT INTO Reparto (nome, viaSupermercato, numCivSupermercato) VALUES ('cartoleria', 'via G. Washington', 53);

INSERT INTO Reparto (nome, viaSupermercato, numCivSupermercato) VALUES ('panetteria', 'via G. Washington', 53);

INSERT INTO Impiegato (CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, mansione, numTurno, numLivello) VALUES ('GLLMRC60C23A944L', '1960-03-23', 20094, 'via Raffaello', 22, 'Corsico', 'Marco', 'Gellini', 'marco.gellini@gmail.com', 'panettiere', 2, 1);

INSERT INTO Impiegato (CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, mansione, numTurno, numLivello) VALUES ('DSNLSU19R51D612E', '1998-02-28', 20089, 'via Cervino', 10, 'Rozzano', 'Luisa', 'De Santis', 'luisadesantis@alice.it', 'provviste scaffali', 3, 1);

INSERT INTO Impiegato (CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, mansione, numTurno, numLivello) VALUES ('NVLMMS88A20H623N', '1988-01-20', 20089, 'via Roma', 1, 'Rozzano', 'Massimiliano', 'Novella', 'novella.massimiliano@gmail.com', 'addetto al banco', 1, 1);

INSERT INTO Impiegato (CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, mansione, numTurno, numLivello) VALUES ('GVLMRA61C23A944U', '1961-03-23', 20090, 'via Celoria', 29, 'Assago', 'Mauro', 'Gavilli', 'gavillimauro@gmail.com', 'provviste scaffali', 3, 1);

INSERT INTO Impiegato (CF, dataDiNascita, CAP, via, numCiv, città, nome, cognome, email, mansione, numTurno, numLivello) VALUES ('RSSVNT99C50L682W', '1999-03-10', 20090, 'via Lazio', 20, 'Cormano', 'Valentina', 'Rossi', 'rossivalentina@gmail.com', 'panettiere', 1, 1);

INSERT INTO Magazzino (CAP, via, numCiv, città, viaSupermercato, numCivSupermercato) VALUES (20096, 'SP Cassanese', 1, 'Pioltello', 'via Andrea Solari', 29);

INSERT INTO Magazzino (CAP, via, numCiv, città, viaSupermercato, numCivSupermercato) VALUES (00155, 'viale P. Togliatti', 121, 'Roma', 'via G. Washington', 53);

INSERT INTO Magazzino (CAP, via, numCiv, città, viaSupermercato, numCivSupermercato) VALUES (20090, 'SP 161', 9, 'Settala', 'via Lorenteggio', 219);

INSERT INTO Prodotto (codiceInterno, materiaPrima, dataScadenza, nome, categoria, prezzoAlPubblico, punti) VALUES ('ESS101025', 'TRUE', '2020-06-05', 'farina', 'drogheria', 1.40, 2);

INSERT INTO Prodotto (codiceInterno, materiaPrima, dataScadenza, nome, categoria, prezzoAlPubblico, punti) VALUES ('ESS101667', 'FALSE', '2020-05-05', 'pane', 'panificati', 1.00, 2);

INSERT INTO Prodotto (codiceInterno, materiaPrima, dataScadenza, nome, categoria, prezzoAlPubblico, punti) VALUES ('ESS101007', 'TRUE', '2025-07-08', 'sogliola', 'pescheria', 2.68, 10);

INSERT INTO Prodotto (codiceInterno, materiaPrima, dataScadenza, nome, categoria, prezzoAlPubblico, punti) VALUES ('ESS101332', 'FALSE', '2021-11-18', 'set di penne', 'cartoleria', 5.00, 5);

INSERT INTO Prodotto (codiceInterno, materiaPrima, dataScadenza, nome, categoria, prezzoAlPubblico, punti) VALUES ('ESS101555', 'TRUE', '2020-10-03', 'pesca', 'ortofrutta', 0.60, 5);

INSERT INTO Prodotto (codiceInterno, materiaPrima, dataScadenza, nome, categoria, prezzoAlPubblico, punti) VALUES ('ESS102222', 'FALSE', '2020-07-31', 'pasta fresca', 'panificati', 4.99, 7);

INSERT INTO Prodotto (codiceInterno, materiaPrima, dataScadenza, nome, categoria, prezzoAlPubblico, punti) VALUES ('ESS222025', 'TRUE', '2020-08-05', 'uova', 'drogheria', 3.00, 5);

INSERT INTO Fornitore (partitaIva, ragioneSociale, email, telefono, CAP, via, numCiv, città, modalitàPagamento) VALUES ('12345670157', 'Fratelli Ceccarelli', 'info@fratelliceccarelli.it', '025743092', 20151, 'via Gallarate', 355, 'Milano', 'bonifico');

INSERT INTO Fornitore (partitaIva, ragioneSociale, email, telefono, CAP, via, numCiv, città, modalitàPagamento) VALUES ('12345670165', 'Walter FoodService', 'info@walterfoodservice.it', '0299010606', 20030, 'via P. Togliatti', 12, 'Milano', 'assegno');

INSERT INTO Fornitore (partitaIva, ragioneSociale, email, telefono, CAP, via, numCiv, città, modalitàPagamento) VALUES ('12345670132', 'AGEA', 'info@agea.it', '06494991', 00185, 'via Palestro', 81, 'Roma', 'bollettino bancario);

INSERT INTO OggettoPremio (codicePremio, puntiPremio) VALUES ('WW90M74NN2', 334);

INSERT INTO OggettoPremio (codicePremio, puntiPremio) VALUES ('DV90N62632', 987);

INSERT INTO OggettoPremio (codicePremio, puntiPremio) VALUES ('NV75K5541B', 654);

INSERT INTO ClienteRegistrato (numeroTessera, nome, cognome, CF, CAP, via, numCiv, città, telefono, email, numeroPunti) VALUES ('0500654350006', 'Astolfo', 'Albanese', 'LBNSLF08B24F205R', 20126, 'via Garian', 41, 'Milano', 20146, 'albenese.a@gmail.com', 235);

INSERT INTO ClienteRegistrato (numeroTessera, nome, cognome, CF, CAP, via, numCiv, città, telefono, email, numeroPunti) VALUES ('05006543544002', 'Cristiana', 'Franco', 'FRNCST08M69H501B', 20165, 'via Gregorio VII', 122, 'Roma', '3336562844', 'cri.franco@icloud.com', 500);

INSERT INTO ClienteRegistrato (numeroTessera, nome, cognome, CF, CAP, via, numCiv, città, telefono, email, numeroPunti) VALUES ('0522609950006', 'Priamo', 'Bonetti', 'BNTPRM07S29L736V', 30173, 'Via Altinia', 131, 'Venezia VE', '3214336779', 'piamo.bonetti@tiscali.it', 1329);

INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oraInizio, oraFine, giornoSettimana, dataStraordinario) VALUES ('01', 'via Lorenteggio', 219, '09:00:00', '18:30:00', 'lunedì', NULL);


```

INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('02', 'via Lorenteggio', 219, '09:00:00', '18:30:00',
'martedì', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('03', 'via Lorenteggio', 219, '09:00:00', '18:30:00',
'mercoledì', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('04', 'via Lorenteggio', 219, '09:00:00', '18:30:00',
'giovedì', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('05', 'via Lorenteggio', 219, '09:00:00', '18:30:00',
'venerdì', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('06', 'via Andrea Solari', 29, '08:00:00', '18:00:00',
'lunedì', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('07', 'via Andrea Solari', 29, '08:00:00', '18:00:00',
'martedì', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('08', 'via Andrea Solari', 29, '08:00:00', '18:00:00',
'mercoledì', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('09', 'via Andrea Solari', 29, '08:00:00', '18:00:00',
'giovedì', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('10', 'via Andrea Solari', 29, '08:00:00', '18:00:00',
'veenerdì', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('11', 'via Andrea Solari', 29, '09:00:00', '18:00:00',
'sabato', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('12', 'via Andrea Solari', 29, '10:00:00', '13:00:00',
'domenica', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('13', 'via G. Washington', 53, '08:30:00', '18:00:00',
'lunedì', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('14', 'via G. Washington', 53, '08:30:00', '18:00:00',
'martedì', NULL);
INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine,
giornoSettimana, dataStraordinario) VALUES ('15', 'via G. Washington', 53, '08:30:00', '18:00:00',
'mercoledì', NULL);

```

INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine, giornoSettimana, dataStraordinario) VALUES ('16', 'via G. Washington', 53, '08:30:00', '18:00:00', 'giovedì', NULL);

INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine, giornoSettimana, dataStraordinario) VALUES ('17', 'via G. Washington', 53, '08:30:00', '18:00:00', 'venerdì', NULL);

INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine, giornoSettimana, dataStraordinario) VALUES ('18', 'via G. Washington', 53, '08:30:00', '12:30:00', 'sabato', NULL);

INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine, giornoSettimana, dataStraordinario) VALUES ('19', 'via G. Washington', 53, '10:00:00', '12:30:00', NULL, '2020-04-25');

INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine, giornoSettimana, dataStraordinario) VALUES ('20', 'via Andrea Solari', 29, '09:30:00', '12:30:00', NULL, '2020-04-25');

INSERT INTO Orario (ID, viaSupermercato, numCivSupermercato, oralnizio, oraFine, giornoSettimana, dataStraordinario) VALUES ('21', 'via Lorenteggio', 219, '09:30:00', '12:30:00', NULL, '2020-06-02');

INSERT INTO Scontrino (data, numeroScontrino, viaSupermercato, numCivSupermercato, numeroTesseraCliente) VALUES ('2020-06-12 09:22:09', '0004531976', 'via Andrea Solari', 29, '0500654350006');

INSERT INTO Scontrino (data, numeroScontrino, viaSupermercato, numCivSupermercato, numeroTesseraCliente) VALUES ('2020-06-12 17:30:46', '0004532406', 'via Andrea Solari', 29, '0500654350006');

INSERT INTO Scontrino (data, numeroScontrino, viaSupermercato, numCivSupermercato, numeroTesseraCliente) VALUES ('2020-06-02 11:09:07', '0003706005', 'via G. Washington', 53, '0522609950006');

INSERT INTO Scontrino (data, numeroScontrino, viaSupermercato, numCivSupermercato, numeroTesseraCliente) VALUES ('2020-06-20 12:01:56', '0003706850', 'via G. Washington', 53, '0522609950006');

INSERT INTO Scontrino (data, numeroScontrino, viaSupermercato, numCivSupermercato, numeroTesseraCliente) VALUES ('2020-06-18 14:56:23', '0021819716', 'via Lorenteggio', 219, NULL);

INSERT INTO Scontrino (data, numeroScontrino, viaSupermercato, numCivSupermercato, numeroTesseraCliente) VALUES ('2020-03-19 10:56:23', '0021802791', 'via Lorenteggio', 219, NULL);

INSERT INTO Ordine (dataOrdine, codiceInternoProdotto, partitalVAFornitore, viaSupermercato, numCivSupermercato, quantitàO) VALUES ('2020-03-20', 'ESS101332', '12345670157', 'via Lorenteggio', 219, 200);

INSERT INTO Ordine (dataOrdine, codiceInternoProdotto, partitaIVAforitore, viaSupermercato, numCivSupermercato, quantitàO) VALUES ('2020-03-20', 'ESS101555', '12345670165', 'via Andrea Solari', 29, 120);

INSERT INTO Ordine (dataOrdine, codiceInternoProdotto, partitaIVAforitore, viaSupermercato, numCivSupermercato, quantitàO) VALUES ('2020-05-02', 'ESS101332', '12345670132', 'via G. Washington', 53, 400);

INSERT INTO Supervisione (CFresponsabile, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato) VALUES ('RSSMRA81T15F205Z', '2019-09-19', NULL, 'ortofrutta', 'via Lorenteggio', 219);

INSERT INTO Supervisione (CFresponsabile, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato) VALUES ('FBRBNC90A51A944C', '2020-01-25', NULL, 'pescheria', 'via Andrea Solari', 29);

INSERT INTO Supervisione (CFresponsabile, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato) VALUES ('FNTGLI71T45D612B', '2015-02-12', NULL, 'cartoleria', 'via G. Washington', 53);

INSERT INTO Supervisione (CFresponsabile, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato) VALUES ('PRDSMN70S07E801P', '2019-06-13', NULL, 'panetteria', 'via G. Washington', 53);

INSERT INTO Supervisione (CFresponsabile, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato) VALUES ('GVZPTR68T07E801A', '2018-05-18', '2019-06-12', 'panetteria', 'via G. Washington', 53);

INSERT INTO Supervisione (CFresponsabile, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato) VALUES ('LRSNLO93D16D969M', '2018-05-18', '2019-09-18', 'ortofrutta', 'via Lorenteggio', 219);

INSERT INTO Lavora (CFimpiegato, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato) VALUES ('NVLM88A20H623N', '2013-09-13', NULL, 'ortofrutta', 'via Lorenteggio', 219);

INSERT INTO Lavora (CFimpiegato, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato) VALUES ('GVL61C23A944U', '2018-08-11', NULL, 'pescheria', 'via Andrea Solari', 29);

INSERT INTO Lavora (CFimpiegato, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato) VALUES ('RSSVNT99C50L682W', '2019-10-01', '2020-01-20', 'panetteria', 'via G. Washington', 53);

INSERT INTO Lavora (CFimpiegato, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato) VALUES ('DSNLSU19R51D612E', '2017-05-04', NULL, 'cartoleria', 'via G. Washington', 53);

INSERT INTO Lavora (CFimpiegato, dataInizio, dataFine, nomeReparto, viaSupermercato, numCivSupermercato) VALUES ('GLLMRC60C23A944L', '2015-01-21', NULL, 'panetteria', 'via G. Washington', 53);

INSERT INTO Stoccaggio (codiceInternoProdotto, numCivMagazzino, viaMagazzino, CAPMagazzino, cittàMagazzino, quantità, sogliaMinima) VALUES ('ESS101007', 1, 'SP Cassanese', 20096, 'Pioltello', 100, 150);

INSERT INTO Stoccaggio (codiceInternoProdotto, numCivMagazzino, viaMagazzino, CAPMagazzino, cittàMagazzino, quantità, sogliaMinima) VALUES ('ESS101025', 1, 'SP Cassanese', 20096, 'Pioltello', 400, 100);

INSERT INTO Stoccaggio (codiceInternoProdotto, numCivMagazzino, viaMagazzino, CAPMagazzino, cittàMagazzino, quantità, sogliaMinima) VALUES ('ESS101332', 121, 'viale P Togliatti', 00155, 'Roma', 500, 200);

INSERT INTO Stoccaggio (codiceInternoProdotto, numCivMagazzino, viaMagazzino, CAPMagazzino, cittàMagazzino, quantità, sogliaMinima) VALUES ('ESS101555', 9, 'SP 161', 20090, 'Settala', 600, 250);

INSERT INTO Rifornisce (partitaIVAforitore, codiceInternoProdotto, stimaDiConsegna, prezzoF, codiceF) VALUES ('12345670157', 'ESS101332', '3 giorni', '4.00', '8076809573016');

INSERT INTO Rifornisce (partitaIVAforitore, codiceInternoProdotto, stimaDiConsegna, prezzoF, codiceF) VALUES ('12345670165', 'ESS101007', '8 giorni', '2.30', '80019886');

INSERT INTO Rifornisce (partitaIVAforitore, codiceInternoProdotto, stimaDiConsegna, prezzoF, codiceF) VALUES ('12345670132', 'ESS101555', '1 giorno', '0.35', '8052440350209');

INSERT INTO Composizione (composto, componente, quantità) VALUES ('ESS101667', 'ESS101025', 2);

INSERT INTO Composizione (composto, componente, quantità) VALUES ('ESS102222', 'ESS222025', 2);

INSERT INTO Composizione (composto, componente, quantità) VALUES ('ESS102222', 'ESS101025', 1);

INSERT INTO PS (codiceInternoProdotto, numeroScontrino, viaSupermercato, numCivSupermercato, quantità) VALUES ('ESS101025', '0003706005', 'via G. Washington', 53, 7);

INSERT INTO PS (codiceInternoProdotto, numeroScontrino, viaSupermercato, numCivSupermercato, quantità) VALUES ('ESS101332', '0003706850', 'via G. Washington', 53, 7);

INSERT INTO PS (codiceInternoProdotto, numeroScontrino, viaSupermercato, numCivSupermercato, quantità) VALUES ('ESS101007', '0004531976', 'via Andrea Solari', 29, 12);

INSERT INTO PS (codiceInternoProdotto, numeroScontrino, viaSupermercato, numCivSupermercato, quantità) VALUES ('ESS101007', '0004532406', 'via Andrea Solari', 29, 12);

INSERT INTO PS (codiceInternoProdotto, numeroScontrino, viaSupermercato, numCivSupermercato, quantità) VALUES ('ESS101555', '0021819716', 'via Lorenteggio', 219, 4);

INSERT INTO PS (codiceInternoProdotto, numeroScontrino, viaSupermercato, numCivSupermercato, quantità) VALUES ('ESS101555', '0021802791', 'via Lorenteggio', 219, 4);

```

INSERT INTO Vende (codiceInternoProdotto, nomeReparto, viaSupermercato,
numCivSupermercato) VALUES ('ESS101555', 'ortofrutta', 'via Lorenteggio', '219');
INSERT INTO Vende (codiceInternoProdotto, nomeReparto, viaSupermercato,
numCivSupermercato) VALUES ('ESS101007', 'pescheria', 'via Andrea Solari', '29');
INSERT INTO Vende (codiceInternoProdotto, nomeReparto, viaSupermercato,
numCivSupermercato) VALUES ('ESS101332', 'cartoleria', 'via G. Washington', '53');
INSERT INTO Vende (codiceInternoProdotto, nomeReparto, viaSupermercato,
numCivSupermercato) VALUES ('ESS101667', 'panetteria', 'via G. Washington', '53');
INSERT INTO Vende (codiceInternoProdotto, nomeReparto, viaSupermercato,
numCivSupermercato) VALUES ('ESS102222', 'panetteria', 'via G. Washington', '53');
INSERT INTO Vende (codiceInternoProdotto, nomeReparto, viaSupermercato,
numCivSupermercato) VALUES ('ESS101025', 'panetteria', 'via G. Washington', '53');
INSERT INTO Vende (codiceInternoProdotto, nomeReparto, viaSupermercato,
numCivSupermercato) VALUES ('ESS222025', 'panetteria', 'via G. Washington', '53');

```

```

INSERT INTO Catalogo (viaSupermercato, numCivSupermercato, codicePremio) VALUES ('via
Lorenteggio', 219, 'WW90M74NN2');
INSERT INTO Catalogo (viaSupermercato, numCivSupermercato, codicePremio) VALUES ('via
Andrea Solari', 29, 'DV90N62632');
INSERT INTO Catalogo (viaSupermercato, numCivSupermercato, codicePremio) VALUES ('via
G. Washington', 53, 'NV75K5541B');

```

```

CREATE TABLE Login (
username CHAR(20),
password CHAR(20),
PRIMARY KEY (username, password)
);

```

```

INSERT INTO Login (username, password) VALUES ('amministratore', 'abc');

```

```

CREATE TABLE PremiSpeciali(
ID CHAR(10),
codicePremio CHAR(10),
disponibilità BOOLEAN NOT NULL,
puntiModalitàPagamento DECIMAL(6,0) NOT NULL,
denaroModalitàPagamento DECIMAL(6,0) NOT NULL,
PRIMARY KEY(ID, codicePremio),
FOREIGN KEY (codicePremio) REFERENCES OggettoPremio (codicePremio)
);

```

```

CREATE TABLE Prenotazione(
modalitàAcquisto CHAR(20),

```

```

ID CHAR(10),
codicePremio CHAR(10),
numeroTessera CHAR(20),
mailAvviso BOOLEAN NOT NULL,
PRIMARY KEY (modalitàAcquisto, ID, codicePremio, numeroTessera),
CHECK (modalitàAcquisto in ('punti', 'puntiEdenaro')),
FOREIGN KEY (ID ,codicePremio) REFERENCES PremiSpeciali (ID ,codicePremio),
FOREIGN KEY (numeroTessera) REFERENCES ClienteRegistrato(numeroTessera)

);

```

```

CREATE TABLE Ritiro(
dataRitiro DATE,
modalitàAcquisto CHAR(20),
ID CHAR(10),
codicePremio CHAR(10),
numeroTessera CHAR(20),
numeroScontrino CHAR(10),
viaSupermercato CHAR(20),
numCivSupermercato DECIMAL(3,0),
PRIMARY KEY (dataRitiro, modalitàAcquisto, ID, codicePremio, numeroTessera,
numeroScontrino, viaSupermercato, numCivSupermercato),
CHECK (modalitàAcquisto in ('punti', 'puntiEdenaro')),
FOREIGN KEY (modalitàAcquisto, ID, codicePremio, numeroTessera) REFERENCES
Prenotazione (modalitàAcquisto, ID, codicePremio, numeroTessera),
FOREIGN KEY (numeroScontrino, viaSupermercato, numCivSupermercato) REFERENCES
Scontrino (numeroScontrino, viaSupermercato, numCivSupermercato)
);

```

```

INSERT INTO PremiSpeciali(ID, codicePremio, disponibilità, puntiModalitàPagamento,
denaroModalitàPagamento) VALUES ('1233256dt', 'WW90M74NN2', 'TRUE', 250, 200)

```

3. Progettazione del sito

Rappresentare uno schema che descriva il flusso di navigazione all'interno del sito. Ogni pagina deve essere rappresentata da un rettangolo che contenga il nome (univoco) della pagina e l'elenco delle funzionalità principali. Indicare come l'utente può spostarsi tra una pagina e l'altra mediante delle frecce. Nel caso in cui alcuni link siano sempre disponibili (per esempio nell'intestazione della pagina), si può evitare di indicare sempre le relative frecce (che renderebbero illeggibile lo schema).

Schema del flusso di navigazione in allegato.

Nome del database PostgreSQL: progettosupermercato

QUERY IN SQL

```
SELECT DISTINCT numeroTessera  
FROM Prenotazione JOIN premispeciali  
WHERE disponibilità = 'FALSE'  
GROUP BY numeroTessera  
HAVING COUNT(*) <= 3
```

```
SELECT DISTINCT numeroTessera  
FROM Ritiro  
WHERE modalitàAcquisto = 'puntiEdenaro'  
GROUP BY numeroTessera
```