

PROGETTO SUPERMERCATO

di

Ricci Francesco Gabriele - 179399

Orlando Daniele – 180583

SOMMARIO

1. Passi di Progettazione
2. Requisiti
3. Individuazione dei concetti rilevanti
4. Contestualizzazione dei requisiti
5. Glossario dei termini
6. Note sui termini
7. Ristrutturazione dei requisiti
8. Schema concettuale con modello E-R
9. Concetti eletti ad entità
10. Tabella delle entità
11. Documentazione delle associazioni
12. Tabella delle associazioni
13. Regole di vincolo
14. Regole di derivazione
15. Tabella dei volumi
16. Operazioni
17. Tavola delle operazioni
18. Analisi delle operazioni
19. Qualità dello schema concettuale
20. Ristrutturazione dello schema ER
 - a) Eliminazione delle generalizzazioni
 - b) Partizionamento / Accorpamento di entità relazioni
 - c) Scelta degli identificatori primari
21. Analisi delle ridondanze
22. Considerazioni sulla ristrutturazione
23. Schema concettuale con modello E-R ristrutturato
24. Traduzione al modello relazionale
25. Vincoli d'integrità referenziale
26. Descrizione tabelle
27. Implementazione
 - a) Creazione del database
 - b) Popolazione delle tabelle
 - c) Operazioni sul database
 - i. Esempi di esecuzione

PASSI DI PROGETTAZIONE

1. Definizione formale, analisi e strutturazione dei requisiti;
2. Progettazione concettuale utilizzando il modello Entità-Relazione;
3. Formalizzazione dei vincoli;
4. Progettazione logica;
5. Analisi dei volumi e degli accessi;
6. Ristrutturazione ed ottimizzazione dello schema ER;
7. Traduzione nel modello relazionale;

Implementazione del progetto.

REQUISITI

Si vuole realizzare una base di dati per l'organizzazione di un supermercato.

Sebbene quanto richiesto sia stato semplificato e non copra interamente tutti gli aspetti di gestione, il sistema realizzato rappresenterà una base reale e significativa.

Il supermercato è strutturato in una serie di reparti. Ogni reparto ha un nome, un responsabile e una serie di impiegati, ognuno con una mansione specifica. Di ogni impiegato vogliamo memorizzare tutti i principali dati anagrafici (nome, indirizzo, telefono, codice fiscale, data di nascita,...) oltre alla data di assunzione.

Ogni impiegato lavora in un determinato reparto ed ha un particolare livello (che ne determina lo stipendio insieme all'anzianità di servizio).

Il supermercato mette in vendita una serie di prodotti, dei quali vogliamo memorizzare nome, genere ("preparazione alimentare", "prodotto di profumeria",...), il prezzo al pubblico, il tempo massimo entro il quale il prodotto deve essere venduto (ad es. se il prodotto è deperibile, il tempo potrebbe essere di solo qualche giorno), il reparto in cui è posto in vendita, la quantità di prodotto presente in magazzino e la soglia minima (determinata dall'esperienza, o da un'apposita interrogazione) al di sotto della quale il prodotto va riordinato. A ciascun prodotto viene inoltre assegnato un codice interno. Alcuni prodotti non sono acquistati direttamente dai fornitori, ma "assemblati" o "preparati" direttamente nel supermercato (ad esempio, delle confezioni regalo di profumi in offerta speciale, o un piatto precotto) a partire da una serie di materie prime (che a loro volta sono prodotti disponibili nel magazzino). Per questi prodotti "composti" vogliamo conoscere anche tutti i prodotti "ingredienti" e la quantità di essi necessaria alla loro preparazione.

La base di dati contiene una lista dei fornitori a cui il supermercato fa riferimento.

Per ogni fornitore vogliamo conoscere la ragione sociale, la partita IVA, la modalità di pagamento richiesta (es. bonifico a 60 giorni), oltre all'indirizzo e al recapito telefonico e di fax. Ogni fornitore può fornire, a un particolare prezzo, uno o più dei prodotti venduti dal supermercato, identificandoli con un codice interno (diverso da quello usato dal magazzino del supermercato!) da indicare negli ordini.

Per quel che riguarda i clienti del supermercato, alcuni di essi possono essere "fidelizzati" perché hanno richiesto una speciale tessera, che permette loro di accumulare punti con gli acquisti e ottenere sconti o regali. I clienti con tessera sono registrati nella base di dati con i loro dati anagrafici e il numero di tessera, e a ciascuno è associato il numero di punti correntemente accumulati

Per semplificare, supponiamo che i "premi" delle raccolte punti siano a loro volta prodotti in vendita nel supermercato. Per ciascun premio, oltre al prodotto associato, vogliamo conoscere il numero di punti necessari ad ottenerlo. I punti si ottengono acquistando particolari prodotti. E' necessario mantenere una lista dei prodotti che partecipano alla raccolta punti, unitamente al numero di punti ottenibili con il loro acquisto. Ogni volta che il cliente ritira un premio, vengono scalati i corrispondenti punti dalla sua tessera.

I registratori di cassa del nostro supermercato sono direttamente interfacciati con la base i dati (come succede molto spesso oggiogiorno). Ogni volta che un prodotto viene venduto, la base di dati registra i dati della vendita (numero di scontrino, prodotto venduto, prezzo al quale è stato venduto) unitamente al codice del cliente al quale è stato venduto, se questo è titolare di una tessera.

Contemporaneamente, vengono aggiornate la disponibilità del prodotto in magazzino e il numero di punti del cliente (se il prodotto in questione dava diritto a dei punti-raccolta).

Le operazioni previste sulla base di dati sono:

- *Fornire le istruzioni per la creazione del DB e degli oggetti che lo costituiscono;*
- *Per ogni relazione individuata, fornire le istruzioni di inserimento, modifica ed eliminazione delle istanze;*
- *Modifica del responsabile e degli impiegati di un reparto;*
- *Determinazione delle vendite per un reparto in un particolare periodo;*
- *Determinazione dei prodotti più venduti in un determinato reparto;*
- *Modifica del prezzo di un prodotto;*
- *Modifica dei dati riguardanti le scorte di prodotto disponibili;*
- *Per i prodotti “composti”, specifica degli “ingredienti” e delle quantità necessarie alla preparazione;*
- *Inclusione o esclusione di un prodotto dalla raccolta punti e indicazione del numero di punti raccolta forniti dal prodotto;*
- *Determinazione dei prodotti sotto scorta;*
- *Lista dei fornitori dai quali un determinato prodotto può essere acquistato, ordinati in base al prezzo richiesto;*
- *Modifica del numero di punti necessari ad ottenere un determinato premio;*
- *Verifica dei premi attualmente disponibili;*
- *Inserimento/Modifica dei prodotti forniti da un fornitore;*
- *Modifica del reparto di assegnazione di un impiegato e del suo livello;*
- *Modifica del numero di punti assegnati al cliente;*
- *Determinazione dei premi a cui un cliente ha diritto;*
- *Ritiro di un premio da parte di un cliente;*
- *Determinazione dei prodotti più acquistati da un cliente;*
- *Determinazione della spesa totale effettuata da un cliente in un determinato periodo.*

INDIVIDUAZIONE DEI CONCETTI RILEVANTI

Si vuole realizzare una base di dati per l'organizzazione di un supermercato.

Sebbene quanto richiesto sia stato semplificato e non copra interamente tutti gli aspetti di gestione, il sistema realizzato rappresenterà una base reale e significativa.

Il supermercato è strutturato in una **serie** di **reparti**. **Ogni** reparto ha un **nome**, un **responsabile** e una serie di **impiegati**, **ognuno** con una **mansione specifica**. **Di ogni** impiegato vogliamo memorizzare tutti i principali **dati anagrafici** (**nome**, **indirizzo**, **telefono**, **codice fiscale**, **data di nascita**,...) oltre alla **data di assunzione**.

Ogni impiegato **lavora** in **un determinato reparto** ed **ha un particolare livello** (che ne determina lo **stipendio** insieme all'**anzianità di servizio**).

Il supermercato **mette in vendita** una **serie** di **prodotti**, dei quali vogliamo memorizzare **nome**, **genere** ("preparazione alimentare", "prodotto di profumeria",...), il **prezzo al pubblico**, il **tempo massimo** **entro** il quale il prodotto **deve** essere venduto (ad es. **se il prodotto è deperibile**, il tempo potrebbe essere di solo qualche giorno), il **reparto** in cui è posto in vendita, la **quantità** di prodotto presente in magazzino e la **soglia minima** (**determinata dall'esperienza**, o da un'apposita interrogazione) al di sotto della quale il prodotto **va riordinato**. **A ciascun** prodotto viene inoltre assegnato un **codice interno**. **Alcuni** prodotti non sono acquistati direttamente dai fornitori, ma "assemblati" o "preparati" direttamente nel supermercato (ad esempio, delle confezioni regalo di profumi in offerta speciale, o un piatto precotto) **a partire da** una serie di **materie prime** (che **a loro volta** sono prodotti disponibili nel magazzino). **Per questi prodotti "composti"** vogliamo conoscere anche **tutti i prodotti "ingredienti"** e la **quantità** di essi necessaria alla loro preparazione.

La base di dati contiene **una lista** dei **fornitori** a cui il supermercato fa riferimento.

Per ogni fornitore vogliamo conoscere la **ragione sociale**, la **partita IVA**, la **modalità di pagamento** richiesta (es. bonifico a 60 giorni), oltre all'**indirizzo** e al **recapito telefonico** e di fax. **Ogni fornitore può fornire**, **a un particolare prezzo**, **uno o più** dei **prodotti venduti dal supermercato**, identificandoli con un **codice interno** (**diverso da quello usato dal magazzino del supermercato!**) **da indicare** negli **ordini**.





Per quel che riguarda i **clienti** del supermercato, **alcuni** di essi possono essere "**fidelizzati**" perché hanno richiesto una speciale tessera, che permette loro di accumulare **punti con gli acquisti** e **ottenere sconti** o **regali**. I clienti **con tessera** sono registrati nella base di dati con i loro **dati anagrafici** e il **numero di tessera**, e a ciascuno è associato il **numero di punti** correntemente accumulati.

Per semplificare, supponiamo che i "**premi**" delle **raccolte punti** **siano a loro volta prodotti in vendita nel supermercato**. **Per ciascun** premio, oltre al **prodotto associato**, vogliamo conoscere il **numero di punti** necessari ad ottenerlo. I punti si ottengono acquistando **particolari** prodotti. E' necessario mantenere una **lista** dei **prodotti che partecipano alla raccolta punti**, unitamente al numero di **punti ottenibili** con il loro acquisto. **Ogni volta** che il cliente **ritira** un premio, **vengono scalati** i corrispondenti punti dalla sua tessera.

I registratori di cassa del nostro supermercato sono direttamente interfacciati con la base i dati (come succede molto spesso oggi). **Ogni volta** che un prodotto viene venduto, la base di dati registra i dati della **vendita** (numero di **scontrino**, **prodotto** venduto, **prezzo** al quale è stato venduto) unitamente al **codice del cliente** al quale è stato venduto, **se questo è titolare di una tessera**.

Contemporaneamente, **vengono aggiornate** la **disponibilità** del prodotto in **magazzino** e il **numero di punti del cliente** (**se il prodotto in questione dava diritto** a dei punti-raccolta).

LEGENDA:

-  : concetti primari;
-  : attributi dei concetti primari;
-  : valori/operazioni dedotte;
-  : vincoli/limitazioni/estensioni;

CONTESTUALIZZAZIONE DEI REQUISITI

Fraasi relative a (Macro Concetti):

- **reparto:**

- Il supermercato è strutturato in una **serie** di **reparti**. Ogni **reparto** ha un **nome**, un **responsabile** e una **serie** di **impiegati**, ognuno con una **mansione specifica**.
- Ogni **impiegato** **lavora** in un **determinato reparto** [...].
- Il supermercato mette in **vendita** una **serie** di **prodotti**, dei quali vogliamo memorizzare[...] il **reparto** in cui è posto in **vendita** [...].

- **impiegato:**

- Di ogni **impiegato** vogliamo memorizzare tutti i principali **dati anagrafici** (**nome**, **indirizzo**, **telefono**, **codice fiscale**, **data di nascita**,...) oltre alla **data di assunzione**.
- Ogni **impiegato** **lavora** in un **determinato reparto** ed ha un particolare **livello** (che ne determina lo **stipendio** insieme all'**anzianità di servizio**).

- **prodotto:**

- Il supermercato mette in **vendita** una **serie** di **prodotti**, dei quali vogliamo memorizzare **nome**, **genere** ("preparazione alimentare", "prodotto di profumeria",...), il **prezzo** al pubblico, il **tempo massimo entro** il quale il **prodotto** **deve** essere **venduto** (ad es. se il **prodotto** è deperibile, il tempo potrebbe essere di solo qualche giorno), il **reparto** in cui è posto in **vendita**, la **quantità** di **prodotto** presente in **magazzino** e la **soglia minima** (**determinata dall'esperienza**, o da un'apposita interrogazione) al di sotto della quale il **prodotto** **va riordinato**.
- A **ciascun** **prodotto** viene inoltre assegnato un **codice interno**. **Alcuni** **prodotti** non sono acquistati direttamente dai fornitori, ma **"assemblati"** o **"preparati"** direttamente nel supermercato (ad esempio, delle confezioni regalo di profumi in offerta speciale, o un piatto precotto) a partire da una **serie** di **materie prime** (che a loro volta sono **prodotti** disponibili nel **magazzino**). **Per questi prodotti** **"composti"** vogliamo conoscere **anche tutti i prodotti** "ingredienti" e la **quantità** di essi necessaria alla loro preparazione.
- Ogni **fornitore** **può fornire**, a un particolare **prezzo**, **uno o più** dei **prodotti** venduti dal supermercato, identificandoli con un **codice interno** (**diverso** da quello usato dal **magazzino** del supermercato!) **da indicare negli ordini**.
- Per semplificare, supponiamo che i **"premi"** delle **raccolte punti** siano a loro volta **prodotti** in **vendita** nel supermercato. Per ciascun **premio**, oltre al **prodotto** associato, vogliamo conoscere il **numero di punti** necessari ad **ottenerlo**. I punti si **ottengono acquistando** **particolari prodotti**. E' necessario mantenere una **lista** dei **prodotti** che partecipano alla **raccolta punti**, unitamente al numero di **punti ottenibili** con il loro **acquisto**.

- **fornitore:**

- La base di dati contiene una **lista** dei **fornitori** a cui il supermercato fa riferimento. **Per ogni fornitore** vogliamo conoscere la **ragione sociale**, la **partita IVA**, la **modalità di pagamento** richiesta (es. bonifico a 60 giorni), oltre all'**indirizzo** e al **recapito telefonico** e di **fax**. Ogni **fornitore** **può fornire**, a un particolare **prezzo**, **uno o più** dei **prodotti** **venduti** dal supermercato, identificandoli con un **codice interno** (**diverso** da quello usato dal **magazzino** del supermercato!) **da indicare negli ordini**.

- **cliente:**

- Per quel che riguarda i **clienti** del supermercato, **alcuni** di essi **possono** essere “**fidelizzati**” perché hanno richiesto una speciale **tessera**, che permette loro di accumulare **punti** con gli **acquisti** e ottenere **sconti o regali**. I **clienti con tessera** sono registrati nella base di dati con i loro **dati anagrafici** e il **numero di tessera**, e a **ciascuno** è associato il **numero di punti correntemente accumulati**.
- **Ogni volta** che un **prodotto** viene **venduto**, la base di dati registra i dati della **vendita** (**numero di scontrino**, **prodotto venduto**, **prezzo** al quale è stato **venduto**) unitamente al **codice del cliente** al quale è stato **venduto**, **se** questo è titolare di una **tessera**.

- **vendita:**

- **Ogni volta** che un **prodotto** viene **venduto**, la base di dati registra i dati della **vendita** (**numero di scontrino**, **prodotto venduto**, **prezzo** al quale è stato **venduto**) unitamente al **codice del cliente** al quale è stato **venduto**, **se** questo è titolare di una **tessera**.

- **premio:**

- Per semplificare, supponiamo che i “**premi**” delle **raccolte punti** siano a loro volta **prodotti** in **vendita** nel supermercato. Per ciascun **premio**, oltre al **prodotto** associato, vogliamo conoscere il **numero di punti** necessari ad **ottenerlo**. I punti si **ottengono acquistando particolari prodotti**. E’ necessario mantenere una **lista** dei **prodotti** che partecipano alla **raccolta punti**, unitamente al numero di **punti ottenibili** con il loro **acquisto**. **Ogni volta** che il **cliente ritira** un **premio**, **vengono scalati** i corrispondenti **punti** dalla sua **tessera**.

LEGENDA:

- **[abc]** : riferimenti interni;
- **[abc]** : riferimenti esterni;
- **[abc]** : attributi dei concetti primari;
- **[abc]** : valori/operazioni dedotte;
- **[abc]** : vincoli/limitazioni/estensioni;

GLOSSARIO DEI TERMINI

TERMINE	DESCRIZIONE	SINONIMI	COLLEGAMENTI
reparto	Zona del supermercato che contiene prodotti con caratteristiche comuni		impiegato, prodotto
impiegato	Personale del supermercato che lavora in un certo reparto	responsabile	reparto
prodotto	Merce generica venduta dal supermercato	ingrediente, materia prima	reparto, vendita, fornitore, premio
fornitore	Azienda che fornisce parte dei prodotti che vengono prodotti nel supermercato (ad esclusione di quelli composti che vengono prodotti all'interno del supermercato stesso).		prodotto
cliente	Persona che effettua acquisti all'interno del supermercato		vendita, premio
vendita	Transazione effettuata tra un cliente ed il supermercato	transazione, acquisto	cliente, prodotto
premio	Bene vinto tramite la raccolta punti	regalo, vincita	prodotto, cliente

NOTE SUI TERMINI

Con {“*cliente*”} si intende

una persona fisica (e non giuridica) che direttamente (e non mediante conto terzi) usufruisce dei servizi offerti dall'azienda, senza alcuna limitazione sulla natura dello stesso.

Con {“*fidelizzati*”} si intende

determinare un gruppo di clienti di cui si sono ottenute informazioni mediante la stipulazione di un contratto fra le due parti (cliente e azienda).

Con {“*assemblati*”, “*preparati*”, “*composti*”} si intendono

i prodotti realizzati dall'unione di altri prodotti.

Con {“*prodotti*”, “*ingredienti*”, “*materie prime*”} si intendono

prodotti sotto il controllo dell'azienda debitamente catalogati ed organizzati.

Con {“*tessera*”} si intende

un qualsiasi metodo, riconosciuto dall'azienda come metodo fidato, di identificazione della persona, fatta specie il cliente.

Con {“*fornitore*”} si intende

identificare un'azienda convenzionata col supermercato a cui è possibile fare riferimento per l'approvvigionamento di beni e servizi.

Con {“*magazzino*”} si intende

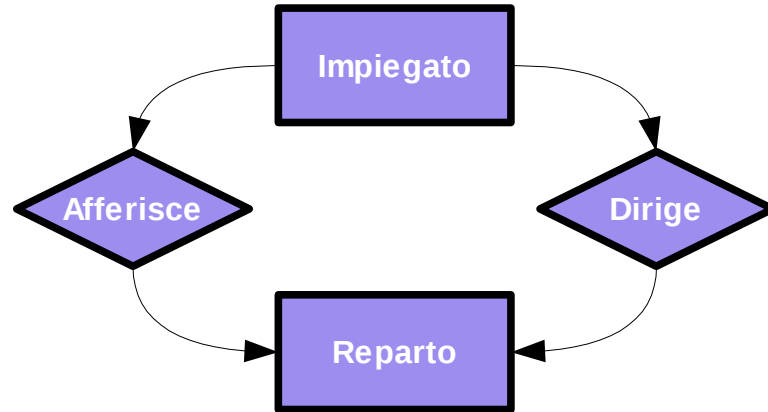
una struttura fisica (vicina e raggiungibile dal supermercato) e logica atta a contenere beni materiali di proprietà del supermercato stesso.

Con {“*raccolta punti*”} si intende

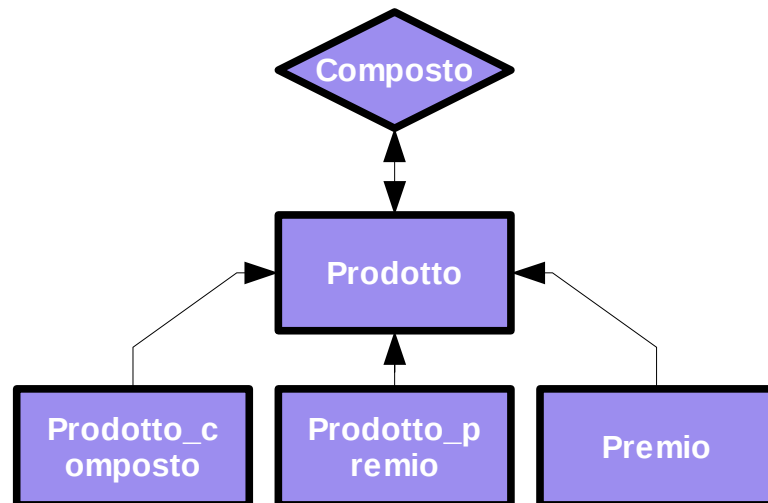
una campagna promozionale, a cui sono associati prodotti offerti in vendita e prodotti offerti in regalo, accessibili dal cliente.

RISTRUTTURAZIONE DEI REQUISITI

- **Reparto**, è un settore del supermercato che contiene dei prodotti comuni e dove lavorano un dirigente ed una serie di impiegati.
- **Impiegato**, è un lavoratore del supermercato impiegato all'interno di uno specifico reparto del supermercato. Ad ogni impiegato è associato un determinato livello, che insieme alla data di assunzione (quindi all'anzianità) ne determina la retribuzione.



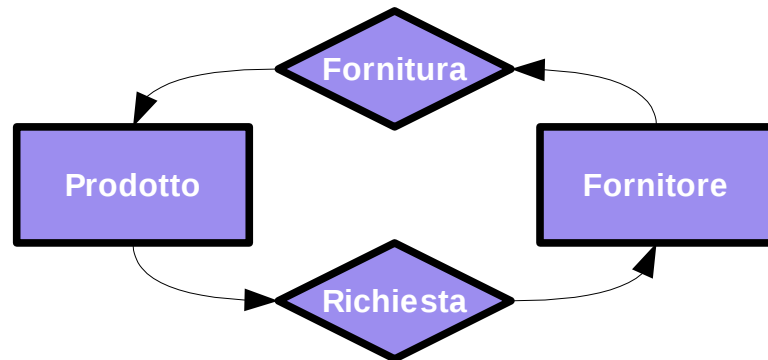
- **Prodotto**, all'interno del supermercato abbiamo una serie di prodotti identificati tramite una serie di attributi generici quali nome, genere, prezzo al pubblico; ed altri necessari a compiere determinate operazioni come la scadenza per determinare il tempo massimo entro il quale il prodotto deve essere venduto, la rimanenza in magazzino e la soglia minima per valutare il riordino dei prodotti in magazzino. Un prodotto può anche essere un prodotto composto da altri prodotti (materie prime) oppure un premio da ritirare con la raccolta punti.
- **Prodotto_composto**, mentre la maggior parte dei prodotti viene acquistata da dei fornitori esterni, alcuni di questi vengono prodotti assemblando delle materie prime all'interno del supermercato stesso. Per tali prodotti vogliamo conoscere anche tutti i prodotti "ingredienti" e la quantità degli stessi necessari alla preparazione del prodotto finale.
- **Prodotto_premio**, i punti si ottengono mediante l'acquisto di alcuni prodotti all'interno del supermercato, ma non tutti i prodotti forniscono dei punti al fine della raccolta premi. Abbiamo quindi la necessità di mantenere all'interno della nostra base di dati una lista di prodotti che concorrono alla raccolta punti insieme ai punti acquisiti con il loro acquisto.



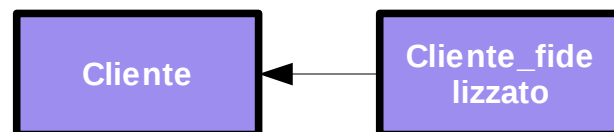
- **Premio**, i premi che si possono vincere tramite la raccolta punti altro non sono che dei prodotti che si vendono normalmente all'interno del supermercato. Tale lista conterrà tutti i prodotti ottenibili come premi, con relativi punti necessari. Ogni qual volta un cliente ritira

un premio, previa verifica dei punti acquisiti, questi gli verranno scalati direttamente dalla sua tessera.

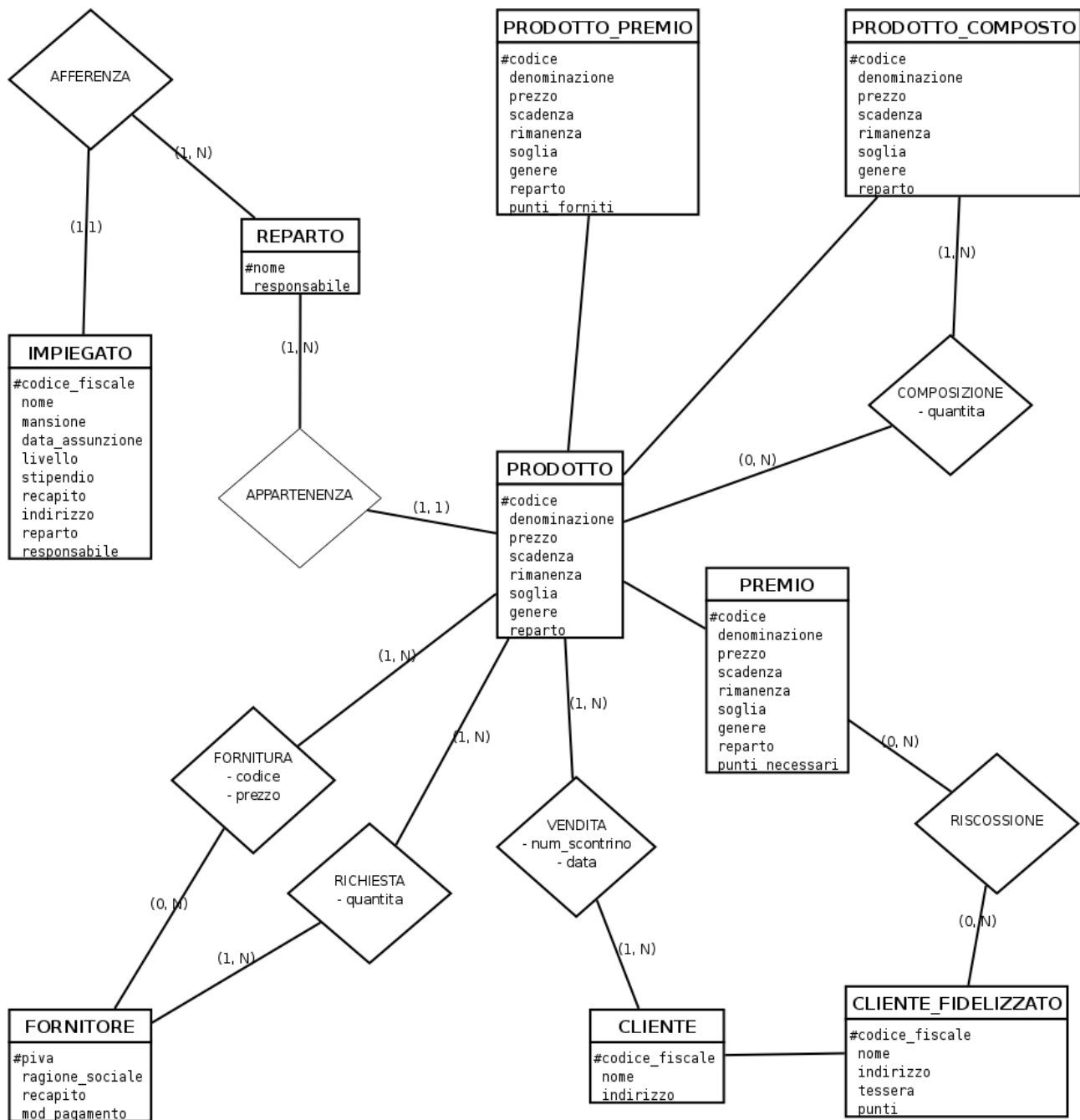
- **Fornitore**, per ogni fornitore vogliamo conoscere la ragione sociale, la partita iva, il metodo di pagamento, indirizzo, recapito telefonico e fax. Ogni fornitore può fornire al supermercato uno o più prodotti, ad un proprio prezzo (per definizione inferiore a quello di vendita), identificandolo mediante un codice interno che è differente da quello del nostro magazzino.



- **Cliente**, persone che effettuano acquisti all'interno del supermercato vengono definiti cliente e i loro dati anagrafici vengono salvati sulla base di dati per utilizzi futuri.
- **Cliente_fidelizzato**, un cliente viene definito fidelizzato se decide di partecipare alla raccolta punti. Per i clienti fidelizzati vengono registrati nella base di dati i loro dati anagrafici e il numero della loro tessera relativamente alla quale vengono aggiunti ad ogni acquisto o decurtati ad ogni ritiro i punti al suo interno.



SCHEMA CONCETTUALE CON MODELLO E-R



CONCETTI ELETTI AD ENTITÀ

- **Impiegato:** si è scelto di promuovere ad entità il concetto di “impiegato” poiché si ha la necessità di tenere traccia di tutte le informazioni anagrafiche del personale più le informazioni di assunzione;
- **Reparto:** si è scelto di promuovere ad entità il concetto di “reparto” poiché si ha la necessità di organizzare i prodotti secondo categorie ben definite e poiché si ha bisogno di collocare un impiegato/responsabile per ogni reparto;
- **Prodotto:** si è scelto di promuovere ad entità il concetto di “prodotto” poiché si deve gestire il magazzino e quindi si ha la necessità di conoscere tutte le informazioni tipiche di un prodotto;
- **Prodotto_composto:** si è scelto di promuovere ad entità il concetto di “prodotto composto” poiché dobbiamo gestire i prodotti formati dall'unione di altri prodotti;
- **Prodotto_premio:** si è scelto di promuovere ad entità il concetto di “prodotto premio” poiché dobbiamo permettere ad un prodotto di poter assegnare dei punti ad un cliente, da utilizzare in una campagna promozionale;
- **Premio:** si è scelto di promuovere ad entità il concetto di “premio” poiché dobbiamo gestire i premi messi a disposizione per una campagna promozionale;
- **Fornitore:** si è scelto di promuovere ad entità il concetto di “fornitore” poiché il supermercato deve avere una lista di distributori di merci a cui potersi appoggiare per ottenere i prodotti di cui ha bisogno;
- **Cliente:** si è scelto di promuovere ad entità il concetto di “cliente” poiché il supermercato può aver bisogno di conoscere lo storico degli acquisti di un cliente;
- **Cliente_fidelizzato:** si è scelto di promuovere ad entità il concetto di “cliente fidelizzato” per permettere ad un cliente di partecipare ad una campagna promozionale.

TABELLA DELLE ENTITA'

ENTITA'	ATTRIBUTI
Impiegato	codice_fiscale , mansione, data_assunzione, livello, stipendio, recapito, indirizzo, reparto
Reparto	nome , responsabile
Prodotto	codice , denominazione, prezzo, scadenza, rimanenza, soglia, genere, reparto
Prodotto_composto	codice , denominazione, prezzo, scadenza, rimanenza, soglia, genere, reparto
Prodotto_premio	codice , denominazione, prezzo, scadenza, rimanenza, soglia, genere, reparto, punti_forniti
Premio	codice , denominazione, prezzo, scadenza, rimanenza, soglia, genere, reparto, punti_necessari
Fornitore	piva , ragione_sociale, recapito, mod_pagamento
Cliente	codice_fiscale , nome, indirizzo
Cliente_fidelizzato	codice_fiscale , nome, indirizzo, tessera, punti

n.b.: In grassetto sono riportate le chiavi primarie

DOCUMENTAZIONE DELLE ASSOCIAZIONI

- **Afferenza:** si è scelto di creare la relazione “afferenza” fra “Impiegato” e “Reparto” per permettere di associare ad un “reparto” un “impiegato”;
- **Appartenenza:** si è scelto di creare la relazione “appartenenza” fra “Prodotto” e “Reparto” poiché, come per gli “impiegati”, anche i “prodotti” debbono essere associati ad un “reparto”;
- **Composizione:** si è scelto di creare l'associazione “composizione” tra “Prodotto” e “Prodotto_composto” poiché i requisiti impongono che un prodotto (“prodotto_composto”) possa essere formato da più “prodotti”;
- **Fornitura:** si è scelto di creare l'associazione “fornitura” tra “Fornitore” e “Prodotto” poiché si ha la necessità di associare al codice esterno del fornitore il codice interno del supermercato, unitamente al prezzo richiesto dal fornitore;
- **Richiesta:** si è scelto di creare l'associazione “richiesta” tra “Fornitore” e “Prodotto” poiché vogliamo tenere traccia degli ordini effettuati dal supermercato verso i proprio fornitori;
- **Vendita:** si è scelto di creare l'associazione “vendita” tra “Prodotto” e “Cliente” per mantenere uno storico degli acquisti effettuati da un cliente;
- **Riscossione:** si è scelto di creare l'associazione “riscossione” tra “Cliente_fidelizzato” e “Premio” per tenere traccia dei premi vinti da un cliente che partecipa alla campagna promozionale.

TABELLA DELLE ASSOCIAZIONI

ASSOCIAZIONE	ENTITA' COINVOLTE	ATTRIBUTI
Afferenza	Impiegato, Reparto	
Appartenenza	Prodotto, Reparto	
Composizione	Prodotto, Prodotto_composto	quantita
Fornitura	Fornitore, Prodotto	codice, prezzo
Richiesta	Fornitore, Prodotto	quantita
Vendita	Prodotto, Cliente	num_scontrino, data_vendita
Riscossione	Cliente_fidelizzato, Premio	

REGOLE DI VINCOLO

RV1: Si assume che un impiegato non possa lavorare in più reparti;

RV2: Lo stipendio di un impiegato è sempre maggiore di 0 e il suo livello cresce partendo da 0 nel momento in cui viene assunto;

RV3: Il responsabile di un reparto deve necessariamente essere un dipendente dello stesso;

RV4: Un prodotto può trovarsi in un unico reparto;

RV5: Un prodotto deve sempre avere i tre attributi prezzo, punti_forniti, punti_necessari maggiori o uguali a 0;

RV6: Un prodotto che non è possibile vincere tramite la raccolta punti ha l'attributo punti_necessari uguale a 0 per indicare la non partecipazione;

RV7: I prodotti forniti da un fornitore si intendono necessariamente anche disponibili dallo stesso;

RV8: Un prodotto può essere inserito solamente una volta nel medesimo scontrino;

RV9: Un cliente fidelizzato possiede una sola tessera nella quale i punti sono sempre maggiori o uguali a 0 e tramite la quale può ritirare il medesimo premio una sola volta;

RV10: Un prodotto non può essere composto di un prodotto del quale è a sua volta componente;

RV11: Il prezzo richiesto dal fornitore deve essere minore, al massimo uguale, al prezzo di vendita al pubblico;

RV12: Un cliente può richiedere uno stesso premio una sola volta;

REGOLE DI DERIVAZIONE

RD1: L'anzianità di un dipendente si ricava dalla differenza tra la data odierna e la data di assunzione;

RD2: Gli importi dei prodotti si ottengono moltiplicando la quantità per il costo unitario dei prodotti;

RD3: Il totale dello scontrino si ottiene mediante la somma degli importi di tutti i prodotti che ne fanno parte;

RD4: Un prodotto composto deve avere almeno 2 ingredienti;

RD5: I dati relativi alle rimanenze in magazzino devono essere aggiornati contestualmente all'emissione dello scontrino;

RD6: I punti di una tessera devono essere aggiornati ad ogni emissione di scontrino o ritiro di premio.

TAVOLA DEI VOLUMI

DENOMINAZIONE	TIPOLOGIA	VOLUME
Impiegato	Entità	45 u
Reparto	Entità	15 u
Prodotto	Entità	1500 u
Prodotto_premio	Entità	1250 u
Premio	Entità	300 u
Prodotto_composto	Entità	50 u
Fornitore	Entità	20 u
Cliente	Entità	15000 u
Cliente_fidelizzato	Entità	4400 u
Afferenza	Relazione	45 u
Appartenenza	Relazione	1500 u
Composizione	Relazione	250 u
Fornitura	Relazione	150 a settimana
Richiesta	Relazione	5000 u
Vendita	Relazione	5.000 al giorno
Riscossione	Relazione	500 al mese

OPERAZIONI

- Op.1) Modifica del responsabile e degli impiegati di un reparto;
- Op.2) Determinazione delle vendite per un reparto in un particolare periodo;
- Op.3) Determinazione dei prodotti più venduti in un determinato reparto;
- Op.4) Modifica del prezzo di un prodotto;
- Op.5) Modifica dei dati riguardanti le scorte di prodotto disponibili;
- Op.6) Per i prodotti “composti”, specifica degli “ingredienti” e delle quantità necessarie alla preparazione;
- Op.7) Inclusione o esclusione di un prodotto dalla raccolta punti e indicazione del numero di punti raccolta forniti dal prodotto;
- Op.8) Determinazione dei prodotti sotto scorta;
- Op.9) Lista dei fornitori dai quali un determinato prodotto può essere acquistato, ordinati in base al prezzo richiesto;
- Op.10) Modifica del numero di punti necessari ad ottenere un determinato premio;
- Op.11) Verifica dei premi attualmente disponibili;
- Op.12) Inserimento/Modifica dei prodotti forniti da un fornitore;
- Op.13) Modifica del reparto di assegnazione di un impiegato e del suo livello;
- Op.14) Modifica del numero di punti assegnati al cliente;
- Op.15) Determinazione dei premi a cui un cliente ha diritto;
- Op.16) Ritiro di un premio da parte di un cliente;
- Op.17) Determinazione dei prodotti più acquistati da un cliente;
- Op.18) Determinazione della spesa totale effettuata da un cliente in un determinato periodo.

TAVOLA DELLE OPERAZIONI

OPERAZIONE	TIPOLOGIA	FREQUENZA
Operazione 1	I	1 volta a semestre
Operazione 2	B	1 volta al mese
Operazione 3	B	1 volta al mese
Operazione 4	I	1 volta ogni 15 giorni
Operazione 5	I	5.000 al giorno
Operazione 6	B	1 volta a bimestre
Operazione 7	B	1 volta a bimestre
Operazione 8	I	1 volta al giorno
Operazione 9	B	1 volta a settimana
Operazione 10	I	1 volta al mese
Operazione 11	I	1 volta a settimana
Operazione 12	B	1 volta a settimana
Operazione 13	I	1 volta all'anno
Operazione 14	B	Ogni volta che un cliente fidelizzato acquista un prodotto che partecipa alla raccolta punti
Operazione 15	B	1 volta a settimana
Operazione 16	B	5 volte al giorno
Operazione 17	I	1 volta al mese
Operazione 18	B	1 volta al mese

ANALISI DELLE OPERAZIONI

1. Modifica del responsabile e degli impiegati di un reparto

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Reparto	Entità	1	Scrittura
Appartenenza	Relazione	1	Scrittura

2. Determinazione delle vendite per un reparto in un particolare periodo

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Vendita	Relazione	N	Lettura
Appartenenza	Relazione	1	Lettura

3. Determinazione dei prodotti più venduti in un determinato reparto.

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Vendita	Relazione	N	Lettura
Appartenenza	Relazione	N	Lettura

4. Modifica del prezzo di un prodotto

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Prodotto	Entità	1	Scrittura

5. Modifica dei dati riguardanti le scorte di prodotto disponibili

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Prodotto	Entità	1	Scrittura

6. Per i prodotti “composti”, specifica degli “ingredienti” e delle quantità necessarie alla preparazione

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Composizione	Relazione	1	Lettura

7. Inclusione o esclusione di un prodotto dalla raccolta punti e indicazione del numero di punti raccolta forniti dal prodotto

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Prodotto_premio	Entità	1	Scrittura
Prodotto_premio	Entità	1	Lettura

8. Determinazione dei prodotti sotto scorta

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Prodotto	Entità	N	Lettura

9. Lista dei fornitori dai quali un determinato prodotto può essere acquistato, ordinati in base al prezzo richiesto

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Richiesta	Relazione	N	Lettura

10. Modifica del numero di punti necessari ad ottenere un determinato premio

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Premio	Entità	1	Scrittura

11. Verifica dei premi attualmente disponibili

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Premio	Entità	N	Lettura

12. Inserimento/Modifica dei prodotti forniti da un fornitore

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Prodotto	Relazione	N	Scrittura

13. Modifica del reparto di assegnazione di un impiegato e del suo livello

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Afferenza	Relazione	1	Scrittura
Impiegato	Entità	1	Scrittura

14. Modifica del numero di punti assegnati al cliente

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Cliente_fidelizzato	Relazione	1	Scrittura

15. Determinazione dei premi a cui un cliente ha diritto

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Cliente_fidelizzato	Entità	1	Lettura
Premio	Entità	N	Lettura

16. Ritiro di un premio da parte di un cliente

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Riscossione	Relazione	1	Scrittura
Cliente_fidelizzato	Entità	1	Scrittura
Prodotto	Entità	1	Scrittura

17. Determinazione dei prodotti più acquistati da un cliente

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Vendita	Relazione	N	Lettura

18. Determinazione della spesa totale effettuata da un cliente in un determinato periodo

DENOMINAZIONE	TIPOLOGIA	NUM. ACCESSI	TIPO ACCESSO
Vendita	Relazione	N	Lettura
Prodotto	Entità	N	Lettura

QUALITÀ DELLO SCHEMA CONCETTUALE

Lo schema concettuale risulta:

CORRETTO: utilizza correttamente i costrutti dei diagrammi ER e non si sono individuati errori sintattici o semantici.

COMPLETO: il diagramma copre tutti i requisiti analizzati e tutte le operazioni definite.

LEGGIBILE: i nomi dei concetti espressi sono stati derivati direttamente dai requisiti cercando quindi di usare un lessico consistente. Anche da un punto di vista grafico il diagramma risulta esteticamente chiaro.

MINIMALE: non esistono cicli né altre ripetizioni.

RISTUTTURAZIONE DELLO SCHEMA E-R

Passiamo a questo punto a ristrutturare lo schermo E-R mediante i tre classici passi:

l'eliminazione delle generalizzazioni, il partizionamento o accorpamento di entità e relazioni, la scelta degli identificatori primari.

Eliminazione delle generalizzazioni

La prima eliminazione di generalizzazione che possiamo effettuare è sicuramente quella relativa all'entità "Prodotto".

L'entità "Prodotto_premio" può essere facilmente inglobata nel padre aggiungendo a quest'ultimo l'attributo punti_forniti, che avrà il vincolo di valore 0 se il prodotto in questione non partecipa alla raccolta punti.

L'entità "Premio" può essere inglobata nel padre aggiungendogli l'attributo punti_necessari.

L'entità "Prodotto_composto" può essere inglobata nel padre facendo diventare la relazione componente una relazione ricorsiva composto-componente. La cardinalità di tale relazione dovrà essere però modificata da (1,N);(0,N) a (0,N);(0,N) per permettere ad un prodotto di non essere necessariamente composto.

Per quanto riguarda la generalizzazione "Cliente"->"Cliente_fidelizzato", questa viene risolta inglobando gli attributi tessera e punti nell'entità "Cliente" perché una riflessione attenta ci porta a capire che un cliente inserito nella base di dati altro non può essere che fidelizzato altrimenti non potremmo avere i suoi dati per una serie di motivi logici e legali.

Partizionamento / Accorpamento di entità e relazioni

Abbiamo ritenuto opportuno inserire una nuova entità "Scontrino" per contenere il numero dello scontrino (numero), la data di emissione (data), ed il codice cliente, togliendoli dalla relazione "Vendita". Tale operazione, oltre a semplificare lo schema, permette una notevole ottimizzazione dello spazio utilizzato in quanto queste informazioni non verranno ripetute in ogni record di acquisto.

Per eseguire un'ulteriore ottimizzazione all'entità "Scontrino" è stato inserito il campo "tot_spesa" per avere a disposizione il totale dello scontrino senza dover accedere all'entità "Prodotto" e fare n operazioni di somma. In queste condizioni però, un prodotto può essere inserito una sola volta in uno scontrino, per ovviare a tale limitazione abbiamo inserito l'attributo "quantità" alla relazione "Vendita".

Scelta degli identificatori primari

Gli identificatori primari per le relazioni binarie sono date dalle chiave composta generata dalle chiavi delle due entità alla quale è collegata.

Per quanto riguarda l'entità "Scontrino", è necessario considerare la chiave di "Cliente_fidelizzato" un attributo semplice della relazione che verrà impostato a NULL se il prodotto in questione è stato acquistato da un cliente non fidelizzato.

ANALISI DELLE RIDONDANZE

- IMPIEGATO.livello

questo attributo potrebbe essere deducibile dall'attributo data_assunzione, ma per la leggibilità abbiamo deciso di mantenerlo.

- IMPIEGATO.reparto

attributo ridondante in quanto può essere facilmente ricavato tramite la relazione AFFERENZA. Oltre ad uno svantaggio in termini di spazio, il mantenimento di questo attributo causerebbe la perdita di tempo nell'aggiornamento.

- IMPIEGATO.stipendio

questo attributo potrebbe essere deducibile dall'attributo livello e/o data_assunzione, ma per semplicità e generalità abbiamo deciso di mantenerlo.

- PRODOTTO.reparto

attributo ridondante in quanto può essere facilmente ricavato tramite la relazione APPARTENENZA. Oltre ad uno svantaggio in termini di spazio, il mantenimento di questo attributo causerebbe la perdita di tempo nell'aggiornamento.

- SCONTRINO.tot_spesa

attributo ridondante in quanto può essere ricavato accedendo a VENDITA e PRODOTTO. Tale soluzione però risulta molto più onerosa ed abbiamo quindi ritenuto opportuno mantenere tale attributo.

CONSIDERAZIONI SULLA RISTRUTTURAZIONE

- La creazione dell'entità “Scontrino”, se da un lato ha portato con se un leggero overhead per le operazioni di ricerca sulla relazione “Vendita” (a causa del Join), dall'altra parte ha comportato una drastica riduzione delle informazioni duplicate per record di acquisto effettuato per ogni scontrino.

Se a questo aggiungiamo che il nuovo attributo “quantita” comporta una netta riduzione di record per prodotti identici di una stessa vendita (cosa estremamente frequente nei supermercati), risulta evidente la riduzione del footprint richiesto dalla base di dati, con un probabile miglioramento (a dispetto del Join) delle prestazioni sulle interrogazioni, grazie alla drastica riduzione dei record presenti nella relazione “Vendita”.

Se consideriamo la stima fatta nella Tabella dei Volumi circa la relazione “Vendita” (segnata come critica in rossa), anche una diminuzione media di 2X sul tempo di ricerca rappresenta un notevole miglioramento.

- L'eliminazione od il mantenimento di “Scontrino”.tot_spesa” può rappresentare un punto su cui soffermarsi per valutare pro e contro delle diverse scelte implementative.

Il suo mantenimento comporta evidenti vantaggi in termini di tempo computazionale ed accessi alla base dati, eliminando la necessità di ricalcolare la sommatoria dei prodotti componenti una determinata vendita.

Lo svantaggio pagato è l'inevitabile ridondanza di informazione apportata con conseguente aumento delle dimensioni della base di dati.

Data l'operazione 18, che fornisce una funzionalità molto potente nella interrogazione dello storico e che si presta ad un utilizzo tutt'altro che saltuario, probabilmente l'aumento in termini di spazio occupato non rappresenta un ostacolo così importante se paragonato al vantaggio in termini di tempo richiesto al completamento della suddetta operazione.

La stima del tempo necessario a calcolare il totale della spesa può essere approssimata ad un $O(n)$, dove n rappresenta il numero di prodotti disponibili all'interno del supermercato, paragonata ad un $O(1)$ (quindi in tempo costante) per la restituzione del campo totale.

- L'accorpamento delle entità figlie di “Prodotto”, in un'unica entità, oltre che a semplificare la struttura del modello E-R, ha portato con se anche miglioramenti in termini di efficienza.

L'iniziale struttura, infatti, implicava una ripetizione delle operazioni di modifica ed inserimento per mantenere aggiornate e coerenti tutte le liste di prodotti.

Con l'accorpamento in un'unica entità, nel caso migliore, si riesce ad avere una riduzione dei tempi richiesti dal database di 3X.

Considerata l'operazione 5 che opera sull'entità “Prodotto” e figlie con carichi giornalieri di riguardo (segnati in rosso nella Tabella dei Volumi), il miglioramento prestazionale apportato da questa modifica risulta considerevole.

```

    erDiagram
        REPARTO ||--o{ IMPIEGATO : "APPARTENENZA"
        REPARTO ||--o{ IMPIEGATO : "AFFERENZA"
        IMPIEGATO ||--o{ PRODOTTO : "FORNITURA"
        FORNITORE ||--o{ PRODOTTO : "RICHIESTA"
        FORNITORE ||--o{ PRODOTTO : "VENDITA"
        CLIENTE ||--o{ PRODOTTO : "COMPOSIZIONE"
        CLIENTE ||--o{ SCONTRINO : "RISCOSSIONE"
        SCONTRINO ||--o{ PRODOTTO : "RICHIESTA"
        SCONTRINO ||--o{ PRODOTTO : "VENDITA"

        REPARTO {
            string id PK
            string impiegato_id FK
        }
        IMPIEGATO {
            string id-codf PK
            string nome
            string indirizzo
            string telefono
            string mansione
            string data_assunzione
            string livello
            string stipendio
        }
        FORNITORE {
            string id-piva PK
            string ragione_sociale
            string recapito
            string mod_pagamento
            string fax
            string indirizzo
            string telefono
        }
        PRODOTTO {
            string id PK
            string nome
            float prezzo
            string scadenza
            float rimanenza
            float soglia
            string genere
            string reparto_id FK
            string punti_forniti
            string punti_vincita
        }
        CLIENTE {
            string id-codf PK
            string nome
            string indirizzo
            string citta
            string tessera
            string punti
        }
        SCONTRINO {
            string id PK
            string cliente_id FK
            float tot_spesa
            string data
        }

        REPARTO }..(1,N) IMPIEGATO : APPARTENENZA
        REPARTO }..(1,N) IMPIEGATO : AFFERENZA
        IMPIEGATO }..(1,N) PRODOTTO : FORNITURA
        FORNITORE }..(1,N) PRODOTTO : RICHIESTA
        FORNITORE }..(1,N) PRODOTTO : VENDITA
        CLIENTE }..(0,N) PRODOTTO : COMPOSIZIONE
        CLIENTE }..(0,N) SCONTRINO : RISCOSSIONE
        SCONTRINO }..(1,N) PRODOTTO : RICHIESTA
        SCONTRINO }..(1,N) PRODOTTO : VENDITA
  
```

TRADUZIONE AL MODELLO RELAZIONALE

IMPIEGATO (**id-codf**, nome, indirizzo, telefono, mansione, data_assunzione, livello, stipendio)

REPARTO (**id**, impiegato_id)

PRODOTTO (**id**, nome, prezzo, scadenza, rimanenza, soglia, genere, reparto_id, punti_forniti, punti_vincita)

FORNITORE (**id-piva**, ragione_sociale, recapito, mod_pagamento, fax, indirizzo, telefono)

CLIENTE (**id-codf**, nome, indirizzo, tessera, punti)

SCONTRINO (**id**, cliente_id, tot_spesa, data)

AFFERENZA (**impiegato_id**, **reparto_id**)

APPARTENENZA (**prodotto_id**, **reparto_id**)

COMPOSIZIONE (**id**, **prodotto_id**, quantita)

FORNITURA (**id**, **fornitore_id**, prodotto_id, prezzo)

RICHIESTA (**fornitore_id**, **prodotto_id**, quantita)

VENDITA (**scontrino_id**, **prodotto_id**, quantita)

RISCOSSIONE (**id**, cliente_id, prodotto_id)

VINCOLI D'INTEGRITÀ REFERENZIALE

- AFFERENZA.impiegato_id
è una chiave esterna con vincolo d'integrità referenziale verso
IMPIEGATO.id-codf
- AFFERENZA.reparto_id
è una chiave esterna con vincolo d'integrità referenziale verso
REPARTO.id
- APPARTENENZA.prodotto_id
è una chiave esterna con vincolo d'integrità referenziale verso
PRODOTTO.id
- APPARTENENZA.reparto_id
è una chiave esterna con vincolo d'integrità referenziale verso
REPARTO.id
- PRODOTTO.reparto_id
è una chiave esterna con vincolo d'integrità referenziale verso
REPARTO.id
- COMPOSIZIONE.id
è una chiave esterna con vincolo d'integrità referenziale verso
PRODOTTO.id
- COMPOSIZIONE.prodotto_id
è una chiave esterna con vincolo d'integrità referenziale verso
PRODOTTO.id
- FORNITURA.fornitore_id
è una chiave esterna con vincolo d'integrità referenziale verso
FORNITORE.id-piva
- FORNITURA.prodotto_id
è una chiave esterna con vincolo d'integrità referenziale verso
PRODOTTO.id
- REPARTO.impiegato_id
è una chiave esterna con vincolo d'integrità referenziale verso
IMPIEGATO.id
- RICHIESTA.fornitore_id
è una chiave esterna con vincolo d'integrità referenziale verso
FORNITORE.id-piva
- RICHIESTA.prodotto_id
è una chiave esterna con vincolo d'integrità referenziale verso
PRODOTTO.id
- RISCOSSIONE.cliente_id

è una chiave esterna con vincolo d'integrità referenziale verso
CLIENTE.id-piva

- RISCOSSIONE.prodotto_id
è una chiave esterna con vincolo d'integrità referenziale verso
PRODOTTO.id

- VENDITA.prodotto_id
è una chiave esterna con vincolo d'integrità referenziale verso
PRODOTTO.id

- VENDITA.scontrino_id
è una chiave esterna con vincolo d'integrità referenziale verso
SCONTRINO.id

- SCONTRINO.cliente_id
è una chiave esterna con vincolo d'integrità referenziale verso
CLIENTE.id-codf

DESCRIZIONE TABELLE

IMPIEGATO : Tabella contenente i dati anagrafici e aziendali di ogni lavoratore del supermercato

id-codf	Identificativo codice fiscale
nome	Nome e cognome dell'impiegato
indirizzo	Indirizzo dell'impiegato
telefono	Recapito telefonico
mansione	Mansione svolta
data_assunzione	Data di assunzione
livello	Livello raggiunto dall'impiegato
stipendio	Stipendio dell'impiegato

REPARTO: Tabella contenente la lista dei reparti del supermercato

id	Nome del reparto
impiegato_id	Codice fiscale del responsabile

PRODOTTO : Tabella contenente tutti i prodotti in vendita nel supermercato, sia quelli forniti dai fornitori, sia quelli composti all'interno del supermercato stesso

id	Codice prodotto
nome	Nome del prodotto
prezzo	Costo del prodotto
scadenza	Data di scadenza del prodotto
rimanenza	Quantità presente nel magazzino
soglia	Quantità minima per effettuare il riordino in magazzino
genere	Genere del prodotto
reparto_id	Id del reparto in cui si trova il prodotto
punti_forniti	Punti forniti dall'acquisto del prodotto
punti_vincita	Punti necessari per ricevere il prodotto come premio

FORNITORE: Tabella contenente la rubrica del supermercato relativamente ai fornitori

id-piva	Identificativo fornitore partita iva
ragione_sociale	Ragione sociale del fornitore
recapito	Referente aziendale
mod_pagamento	Modalità di pagamento accettate

fax	Fax fornitore
indirizzo	Indirizzo del fornitore
telefono	Recapito telefonico

CLIENTE: Tabella relativa ai clienti fidelizzati del supermercato

id-codf	Identificativo cliente codice fiscale
nome	Nome e cognome cliente
indirizzo	Indirizzo del cliente
tessera	Numero della tessera punti
punti	Saldo dei punti

SCONTRINO: Tabella relativa agli scontrini emessi dal supermercato

id	Numero progressivo scontrino
cliente_id	Identificativo cliente (codice fiscale)
tot_spesa	Totale della spesa relativa allo scontrino
data	Data emissione scontrino

AFFERENZA: Tabella che associa gli impiegati al loro reparto

impiegato_id	Identificativo impiegato (codice fiscale)
reparto_id	Identificativo reparto (nome reparto)

APPARTENENZA: Tabella che associa i prodotti al reparto in cui sono sistemati

prodotto_id	Identificativo prodotto (codice interno)
reparto_id	Identificativo reparto (nome reparto)

COMPOSIZIONE: Tabella che lega i prodotti composti ai loro componenti

id	Codice interno prodotto composto
prodotto_id	Codice interno componente
quantita	Quantità necessaria del prodotto componente

FORNITURA: Tabella che lega i codici prodotto dei fornitori ai codici prodotto all'interno del supermercato

id	Codice prodotto fornitore
fornitore_id	Identificativo fornitore (partita iva)
prodotto_id	Codice prodotto interno al supermercato
prezzo	Prezzo richiesto dal fornitore

RICHIESTA: Tabella che lega i codici prodotto all'interno del supermercato al fornitore con relativo costo

fornitore_id	Identificativo fornitore (partita iva)
prodotto_id	Codice prodotto interno al supermercato
quantita	Quantità di prodotto richiesto al fornitore

VENDITA: Tabella che associa ad uno scontrino i prodotti relativi a tale vendita

scontrino_id	Identificativo scontrino (numero progressivo)
prodotto_id	Identificativo prodotto (codice interno)
quantita	Quantità prodotto acquistata

RISCOSSIONE: Tabella contenente i premi ritirati dai clienti fidelizzati

id	Numero progressivo riscossione
cliente_id	Identificativo cliente (codice fiscale)
prodotto_id	Prodotto ritirato come premio

IMPLEMENTAZIONE

DBMS utilizzato: FIREBIRD

Op. 0.a - Creazione del database e delle relazioni:

```
CREATE TABLE Impiegato (  
    id-codf CHAR(16),  
    nome VARCHAR(30),  
    indirizzo VARCHAR(20),  
    telefono VARCHAR(11),  
    data_assunzione TIMESTAMP,  
    mansione VARCHAR(20),  
    livello INTEGER,  
    stipendio DECIMAL,  
    data_assunzione TIMESTAMP,  
    PRIMARY KEY (id-codf)  
);  
  
CREATE TABLE Reparto (  
    id VARCHAR(20),  
    impiegato_id CHAR(16),  
    PRIMARY KEY (id),  
    FOREIGN KEY (impiegato_id) REFERENCES Impiegato(id-codf)  
);  
  
CREATE TABLE Prodotto (  
    id INTEGER,  
    nome VARCHAR (30),  
    genere VARCHAR(20),  
    prezzo DECIMAL,  
    scadenza TIMESTAMP,  
    rimanenza INTEGER,  
    soglia INTEGER,  
    punti_forniti INTEGER,  
    punti_vincita INTEGER,  
    reparto_id VARCHAR (20),  
    PRIMARY KEY (id),  
    FOREIGN KEY (reparto_id) REFERENCES Reparto(id)  
);  
  
CREATE TABLE Fornitore (  
    id-piva CHAR(11),  
    ragione_sociale VARCHAR(20),  
    mod_pagamento VARCHAR(20),  
    fax VARCHAR(11),  
    indirizzo VARCHAR(20),  
    telefono VARCHAR(11),  
    PRIMARY KEY (id-piva)  
);
```

```

CREATE TABLE Cliente (
    id-codf CHAR(16),
    nome VARCHAR (30),
    indirizzo VARCHAR(20),
    citta VARCHAR(20),
    tessera INTEGER,
    punti INTEGER,
    PRIMARY KEY (id-codf)
);

CREATE TABLE Scontrino (
    id INTEGER,
    cliente_id CHAR(16),
    tot_spesa DECIMAL,
    data TIMESTAMP,
    PRIMARY KEY (id),
    FOREIGN KEY (cliente_id) REFERENCES Cliente(id-codf)
);

CREATE TABLE Afferenza (
    impiegato_id CHAR(16),
    reparto_id VARCHAR (20),
    PRIMARY KEY(impiegato_id, reparto_id),
    FOREIGN KEY (impiegato_id) REFERENCES Impiegato(id-codf),
    FOREIGN KEY (reparto_id) REFERENCES Reparto(id)
);

CREATE TABLE Composizione (
    id INTEGER,
    prodotto_id INTEGER,
    quantita INTEGER, PRIMARY KEY (id, prodotto_id),
    FOREIGN KEY (id) REFERENCES Prodotto(id),
    FOREIGN KEY (prodotto_id) REFERENCES Prodotto(id)
);

CREATE TABLE Fornitura (
    id INTEGER,
    fornitore_id CHAR(11),
    prodotto_id INTEGER,
    prezzo FLOAT,
    PRIMARY KEY (id, fornitore_id),
    FOREIGN KEY (prodotto_id) REFERENCES Prodotto(id),
    FOREIGN KEY (fornitore_id) REFERENCES Fornitore(id-piva)
);

CREATE TABLE Richiesta (
    prodotto_id INTEGER,
    fornitore_id CHAR(11),
    quantita INTEGER,
    PRIMARY KEY (prodotto_id, fornitore_id),
    FOREIGN KEY (prodotto_id) REFERENCES Prodotto(id),

```

```

        FOREIGN KEY (fornitore_id) REFERENCES Fornitore(id-piva)
    );

CREATE TABLE Vendita (
    prodotto_id INTEGER,
    scontrino_id INTEGER,
    quantita INTEGER,
    PRIMARY KEY (prodotto_id, scontrino_id),
    FOREIGN KEY (prodotto_id) REFERENCES Prodotto(id),
    FOREIGN KEY (scontrino_id) REFERENCES Scontrino(id)
);

CREATE TABLE Riscossione (
    cliente_id CHAR(16),
    prodotto_id INTEGER,
    PRIMARY KEY (cliente_id, prodotto_id),
    FOREIGN KEY (cliente_id) REFERENCES Cliente(id-codf),
    FOREIGN KEY (prodotto_id) REFERENCES Prodotto(id)
);

```

Op. 0.b - Inserimento di un impiegato:

```

SET term !!;

CREATE PROCEDURE inserisci_impiegato (
    id-codf CHAR(16),
    nome VARCHAR(30),
    data_nascita TIMESTAMP,
    mansione VARCHAR(20),
    livello INTEGER,
    stipendio DECIMAL,
    data_assunzione TIMESTAMP,
    indirizzo VARCHAR(20),
    telefono INTEGER
) AS
BEGIN
    INSERT INTO Impiegato (
        id-codf, nome, data_nascita, mansione, livello,
        stipendio, data_assunzione, indirizzo, telefono
    ) VALUES (
        :id-codf, :nome, :data_nascita, :mansione, :livello,
        :stipendio, :data_assunzione, :indirizzo, :telefono
    );
END!!

SET term;!!

```

Esempi di esecuzione:

```

EXECUTE PROCEDURE inserisci_impiegato ('aaaaaaaaaaaaaaaa', 'Pippo
Franco', 'GEN-01-

```



```
1900','nulla facente',2, 432.0, GEN-01-1990', 'via pippo 96',  
0123456789);  
EXECUTE PROCEDURE inserisci_impiegato ('bbbbbbbbbbbbbbb', 'Alex J',  
'GEN-01-1900', 'sabotatore', 8, 2000.0, 'GEN-01-1990', 'via roma  
01', 0123456789);
```

Op. 0.c - Inserimento di un reparto:

```
SET term !!;  
  
CREATE PROCEDURE inserisci_reparto (  
    id VARCHAR(20), impiegato_id CHAR(16)  
) AS  
BEGIN  
    INSERT INTO Reparto (id, impiegato_id)  
    VALUES (:id, :impiegato_id);  
END!!  
  
SET term;!!
```

Esempi di esecuzione:

```
EXECUTE PROCEDURE inserisci_reparto ('pesce', 'aaaaaaaaaaaaaaaa');  
EXECUTE PROCEDURE inserisci_reparto ('pesce', 'bbbbbbbbbbbbbbb');
```

Op. 0.d - Inserimento di un fornitore:

```
SET term !!;  
  
CREATE PROCEDURE inserisci_fornitore (  
    id-piva CHAR(11), ragione_sociale VARCHAR(20), mod_pagamento  
    VARCHAR(20), fax VARCHAR(11), indirizzo VARCHAR(20), telefono  
    VARCHAR(11)  
) AS  
BEGIN  
    INSERT INTO Fornitore (  
        id-piva, ragione_sociale, mod_pagamento, fax, indirizzo,  
telefono  
    ) VALUES (:id-piva, :ragione_sociale, :mod_pagamento,  
:fax, :indirizzo, :telefono);  
END!!  
  
SET term;!!
```

Esempi di esecuzione:

```
EXECUTE PROCEDURE inserisci_fornitore ('01234567891',  
'fornitore1', 'mazzetta', '0123456789', 'via inesistente 00',  
'0123456789');  
EXECUTE PROCEDURE inserisci_fornitore ('01234567892',  
'fornitore2', 'contanti', '0123456789', 'via fasulla 123',  
'0123456789');
```

Op. 0.e - Inserimento di un prodotto:

```
SET term !!;
```

```
CREATE PROCEDURE inserisci_prodotto (  
    id INTEGER, nome VARCHAR (30), genere VARCHAR(20), prezzo  
    DECIMAL, scadenza TIMESTAMP, rimanenza INTEGER, soglia INTEGER,  
    punti_forniti INTEGER, punti_vincita INTEGER, reparto_id VARCHAR  
    (20)  
    ) AS  
BEGIN  
    INSERT INTO Prodotto (  
        id, nome, genere, prezzo, scadenza, rimanenza, soglia,  
        punti_forniti, punti_vincita, reparto_id  
    ) VALUES (:id, :nome, :genere, :prezzo, :scadenza,  
    :rimanenza, :soglia, :punti_forniti, :punti_vincita, :reparto_id  
    );  
END!!
```

```
SET term;!!
```

Esempi di esecuzione:

```
EXECUTE PROCEDURE inserisci_prodotto (1111, 'aaa', 'tipo1', 1.00,  
'GEN-01-3000', 100, 10, 0, NULL, 'pesce');  
EXECUTE PROCEDURE inserisci_prodotto (1112, 'bbb', 'tipo2', 2.00,  
'GEN-01-3000', 100, 10, 0, NULL, 'pesce' );
```

Op. 0.f - Inserimento di un cliente fidelizzato:

```
SET term !!;
```

```
CREATE PROCEDURE inserisci_cliente (  
    id-codf CHAR(16), nome VARCHAR (30), indirizzo VARCHAR(20),  
    citta VARCHAR(20), tessera INTEGER, punti INTEGER  
    ) AS  
BEGIN  
    INSERT INTO Cliente (id-codf, nome, indirizzo, citta,  
    tessera, punti)  
    VALUES (:id-codf, :nome, :indirizzo, :citta, :tessera,  
    :punti);  
END!!
```

```
SET term;!!
```

Esempi di esecuzione:

```
EXECUTE PROCEDURE inserisci_cliente ('cccccccccccccccc', 'Ego  
Sum', 'via via 123', 'milano', 1000, 0);  
EXECUTE PROCEDURE inserisci_cliente ('dddddddddddddddd', 'Cogito  
Ergo Sum', 'via piazza 111', 'milano', 1001, 0);
```

Op. 0.g - Inserimento di uno scontrino:

```
SET term !!;

CREATE PROCEDURE inserisci_scontrino (
    id INTEGER, cliente_id CHAR(16), tot_spesa DECIMAL, data
TIMESTAMP
) AS
BEGIN
    INSERT INTO Scontrino (id, cliente_id, tot_spesa, data)
    VALUES (:id, :cliente_id, :tot_spesa, :data);
END!!

SET term;!!
```

Esempi di esecuzione:

```
EXECUTE PROCEDURE inserisci_scontrino (0001, 'cccccccccccccccc',
1.00, 'GEN-01-2009 00:00');
EXECUTE PROCEDURE inserisci_scontrino (0002, NULL, 2.00, 'GEN-01-
2009 00:00');
```

Op. 0.h - Inserimento di un'afferenza:

```
SET term !!;

CREATE PROCEDURE inserisci_afferenza (
    impiegato_id CHAR(16), reparto_id VARCHAR (20)
) AS
BEGIN
    INSERT INTO Afferenza (impiegato_id, reparto_id)
    VALUES (:impiegato_id, :reparto_id);
END!!

SET term;!!
```

Esempi di esecuzione:

```
EXECUTE PROCEDURE inserisci_afferenza ('aaaaaaaaaaaaaaaa',
'pesce');
EXECUTE PROCEDURE inserisci_afferenza ('bbbbbbbbbbbbbbbbbb',
'pesce');
```

Op. 0.i - Inserimento di una fornitura:

```
SET term !!;

CREATE PROCEDURE inserisci_fornitura (
    id INTEGER, fornitore_id CHAR(11), prodotto_id INTEGER
) AS
```

```
BEGIN
    INSERT INTO Fornitura (id, fornitore_id, prodotto_id, prezzo)
    VALUES (:id, :fornitore_id, :prodotto_id, :prezzo);
END!!

SET term;!!
```

Esempi di esecuzione:

```
EXECUTE PROCEDURE inserisci_fornitura (1234, '01234567891', 1111,
12.0);
EXECUTE PROCEDURE inserisci_fornitura (1235, '01234567892', 1112,
46.3);
```

Op. 0.j - Inserimento di una suddivisione:

```
SET term !!;

CREATE PROCEDURE inserisci_suddivisione (
    id INTEGER, prodotto_id INTEGER, quantita INTEGER
) AS
BEGIN
    INSERT INTO Composizione (id, prodotto_id, quantita)
    VALUES (:id, :prodotto_id, :quantita);
END!!

SET term;!!
```

Esempi di esecuzione:

```
EXECUTE PROCEDURE inserisci_suddivisione (1111, 1111, 1);
EXECUTE PROCEDURE inserisci_suddivisione (1112, 1112, 2);
```

Op. 0.k - Inserimento di una disponibilità:

```
SET term !!;

CREATE PROCEDURE inserisci_disponibilita (
    prodotto_id INTEGER, fornitore_id CHAR(11), quantita DECIMAL
) AS
BEGIN
    INSERT INTO Richiesta (prodotto_id, fornitore_id, quantita)
    VALUES (:prodotto_id, :fornitore_id, :quantita);
END!!

SET term;!!
```

Esempi di esecuzione:

```
EXECUTE PROCEDURE inserisci_disponibilita (1111, '01234567891',
8);
EXECUTE PROCEDURE inserisci_disponibilita (1112, '01234567892',
2);
```

Op. 0.l - Inserimento di una vendita:

```
SET term !!;

CREATE PROCEDURE inserisci_vendita (
    prodotto_id INTEGER, scontrino_id INTEGER, quantita INTEGER
) AS
    DECLARE VARIABLE punti_spesa INTEGER;
BEGIN
    INSERT INTO Vendita (prodotto_id, scontrino_id, quantita)
    VALUES (:prodotto_id, :scontrino_id, :quantita);

    SELECT punti_forniti FROM Prodotto WHERE id = :prodotto_id
    INTO punti_spesa;

    UPDATE Prodotto SET rimanenza = (rimanenza - :quantita);
    UPDATE Cliente SET punti = (punti + :punti_spesa);
END!!

SET term;!!
```

Esempi di esecuzione:

```
EXECUTE PROCEDURE inserisci_vendita (1111, 0001, 1);
EXECUTE PROCEDURE inserisci_vendita (1112, 0002, 1);
```

Op. 0.m - Inserimento di un ritiro di un premio da parte di un cliente fidelizzato:

```
SET term !!;

CREATE PROCEDURE inserisci_ritiro (cliente_id CHAR(16),
    prodotto_id INTEGER)
AS
BEGIN
    INSERT INTO Riscossione (cliente_id, prodotto_id)
    VALUES (:cliente_id, :prodotto_id);
END!!

SET term;!!
```

Esempio di esecuzione:

```
EXECUTE PROCEDURE inserisci_ritiro ('dddddddddddddddd', 1111);
```

Op. 1.a - Modifica del responsabile di un reparto:

```
SET term !!;

CREATE PROCEDURE modifica_resp_reparto (
```

```

        reparto_id VARCHAR (20), nuovo_resp CHAR (16)
    ) AS
BEGIN
    UPDATE Reparto SET impiegato_id = :nuovo_resp WHERE id =
:reparto_id;
END!!

```

SET term;!!

Esempio di esecuzione:

```

EXECUTE PROCEDURE modifica_resp_reparto ('pesce',
'aaaaaaaaaaaaaaaaaa');

```

Op. 1.b - Modifica dei dati di un impiegato:

SET term !!;

```

CREATE PROCEDURE modifica_imp_cf (id-codf CHAR(16), new_id-codf
CHAR(16))

```

AS

BEGIN

```

    UPDATE Impiegato SET id-codf = :new_id-codf WHERE id-codf
= :id-codf;

```

END!!

SET term;!!

SET term !!;

```

CREATE PROCEDURE modifica_imp_nome (id-codf CHAR(16), nome
VARCHAR(30))

```

AS

BEGIN

```

    UPDATE Impiegato SET nome = :nome WHERE id-codf = :id-codf;

```

END!!

SET term;!!

SET term !!;

```

CREATE PROCEDURE modifica_imp_datanascita (
    id-codf CHAR(16), data_nascita TIMESTAMP

```

) AS

BEGIN

```

    UPDATE Impiegato SET data_nascita = :data_nascita WHERE id-
codf = :id-codf;

```

END!!

SET term;!!

SET term !!;

```

CREATE PROCEDURE modifica_imp_mansione (id-codf CHAR(16), mansione
VARCHAR(20))
AS
BEGIN
    UPDATE Impiegato SET mansione = :mansione WHERE id-codf =
:id-codf;
END!!

SET term;!!

SET term !!;

CREATE PROCEDURE modifica_imp_stipendio (id-codf CHAR(16),
stipendio DECIMAL)
AS
BEGIN
    UPDATE Impiegato SET stipendio = :stipendio WHERE id-codf
= :id-codf;
END!!

SET term;!!

SET term !!;

CREATE PROCEDURE modifica_imp_data_ass (
    id-codf CHAR(16), data_assunzione TIMESTAMP
) AS
BEGIN
    UPDATE Impiegato SET data_assunzione = :data_assunzione WHERE
id-codf = :id-codf;
END!!

SET term;!!

SET term !!;

CREATE PROCEDURE modifica_imp_indirizzo (id-codf CHAR(16),
indirizzo VARCHAR(20))
AS
BEGIN
    UPDATE Impiegato SET indirizzo = :indirizzo WHERE id-codf
= :id-codf;
END!!

SET term;!!

SET term !!;

CREATE PROCEDURE modifica_imp_telefono (id-codf CHAR(16), telefono
VARCHAR(11))
AS
BEGIN

```

```
UPDATE Impiegato SET telefono = :telefono WHERE id-codf =  
:id-codf;  
END!!  
  
SET term;!!
```

Op. 2 - Determinazione delle vendite per un reparto in un particolare periodo:

```
SELECT Vendita.*  
FROM (  
    (Scontrino JOIN Vendita  
        ON Scontrino.id = Vendita.scontrino_id  
        AND Scontrino.data >= $data_inizio  
        AND Scontrino.data <= $data_fine  
    ) JOIN Prodotto  
        ON Vendita.prodotto_id = Prodotto.id  
        AND Prodotto.reparto_id = $reparto  
    )  
;
```

Op. 3 - Determinazione dei prodotti più venduti in un determinato reparto:

```
SELECT id, SUM(quantita)  
FROM (Vendita JOIN Prodotto  
    ON Vendita.prodotto_id = Prodotto.id  
    AND Prodotto.reparto_id = $reparto  
) GROUP BY id  
HAVING SUM (quantita) > 100  
;
```

Op. 4 - Modifica del prezzo di un prodotto:

```
SET term !!;  
  
CREATE PROCEDURE modifica_prezzo_prodotto (id INTEGER, prezzo  
DECIMAL)  
AS  
BEGIN  
    UPDATE Prodotto SET prezzo = :prezzo WHERE id = :id;  
END!!  
  
SET term;!!
```

```
Esempio di esecuzione:  
EXECUTE PROCEDURE modifica_prezzo_prodotto (1111, 10.00);
```


Op. 5 - Modifica dei dati riguardanti le scorte di prodotto disponibili:

```
SET term !!;
```

```
CREATE PROCEDURE modifica_quantita_prodotto (id INTEGER,  
nuova_quantita INTEGER)  
AS  
BEGIN  
    UPDATE Prodotto SET rimanenza = :nuova_quantita WHERE id =  
:id;  
END!!
```

```
SET term;!!
```

Esempio di esecuzione:

```
EXECUTE PROCEDURE modifica_quantita_prodotto(1112, 27);
```

Op. 6 - Per i prodotti "composti", specifica degli "ingredienti" e delle quantità necessarie alla preparazione:

```
SELECT prodotto_id, quantita FROM Composizione WHERE id = $id;
```

Op. 7.a - Inclusione o esclusione di un prodotto dalla raccolta punti:

```
SET term !!;
```

```
CREATE PROCEDURE inclusione_raccolta_punti (id INTEGER, punti  
INTEGER)  
AS  
BEGIN  
    UPDATE Prodotto SET punti_forniti = :punti WHERE id = :id;  
END!!
```

```
SET term;!!
```

Esempio di esecuzione:

```
EXECUTE PROCEDURE inclusione_raccolta_punti (1112, 50);
```

Op. 7.b - Indicazione del numero di puntiraccolta forniti dal prodotto:

```
SET term !!;
```

```
CREATE PROCEDURE punti_forniti (id INTEGER)  
    RETURNS(punti INTEGER)  
AS  
BEGIN
```

```
        SELECT punti_forniti FROM Prodotto WHERE id = :id INTO
:punti;
END!!
```

```
SET term;!!
```

Esempio di esecuzione:

```
EXECUTE PROCEDURE punti_forniti (1112);
```

Op. 8 - Determinazione dei prodotti sotto scorta:

```
SELECT id, id FROM Prodotto WHERE rimanenza >= soglia;
```

Op. 9 - Lista dei fornitori dai quali un determinato prodotto può essere acquistato, ordinati in base al prezzo richiesto:

```
SELECT fornitore_id FROM Fornitura WHERE prodotto_id = $product
ORDER BY prezzo;
```

Op. 10 - Modifica del numero di punti necessari ad ottenere un determinato premio:

```
SET term !!;
```

```
CREATE PROCEDURE modifica_punti_premio (id INTEGER, punti INTEGER)
AS
BEGIN
    UPDATE Prodotto SET punti_vincita = :punti WHERE id = :id;
END!!
```

```
SET term;!!
```

Esempio di esecuzione:

```
EXECUTE PROCEDURE modifica_punti_premio (1111, 1000);
```

Op. 11 - Verifica dei premi attualmente disponibili:

```
SELECT id, codice_interno FROM Prodotto WHERE punti_vincita > 0
AND rimanenza > 0;
```

Op. 12 - Inserimento/Modifica dei prodotti forniti da un fornitore:

```
SET term !!;
```

```
CREATE PROCEDURE modifica_fornitura (prod INTEGER, fornitore_id
CHAR(11))
```

```
AS
BEGIN
    UPDATE Fornitura SET fornitore_id = :fornitore_id WHERE
    prodotto_id = :prod;
END!!
```

```
SET term;!!
```

Esempio di esecuzione:

```
EXECUTE PROCEDURE modifica_fornitura (1111, 01234567891);
```

```
SET term !!;
```

```
CREATE PROCEDURE modifica_disponibilita (
    prod INTEGER, fornitore_id CHAR (11), prezzo_ric DECIMAL
) AS
BEGIN
    UPDATE Richiesta
        SET fornitore_id = :fornitore_id, costo = :prezzo_ric
        WHERE prodotto_id = :prod
    ;
END!!
```

```
SET term;!!
```

Esempio di esecuzione:

```
EXECUTE PROCEDURE modifica_disponibilita (1111, '01234567892', 2);
```

Op. 13.a - Modifica del reparto di assegnazione di un impiegato:

```
SET term !!;
```

```
CREATE PROCEDURE modifica_imp_reparto (
    impiegato_id CHAR(16), reparto_id VARCHAR (20))
AS
    DECLARE VARIABLE x CHAR(16);
BEGIN
    SELECT impiegato_id FROM Reparto WHERE id = :reparto_id
INTO :x;
    IF ( x <> :impiegato_id)
    THEN
        BEGIN
            UPDATE Afferenza
                SET reparto_id = :reparto_id
                WHERE impiegato_id = :impiegato_id
            ;
        END
    END!!
```

```
SET term;!!
```

```
Esempio di esecuzione:  
EXECUTE PROCEDURE modifica_imp_reparto ( 'aaaaaaaaaaaaaaaa',  
'pesce');
```

Op. 13.b - Modifica del livello di un impiegato:

```
SET term !!;  
CREATE PROCEDURE modifica_imp_livello (id-codf CHAR(16), livello  
INTEGER)  
AS  
BEGIN  
    UPDATE Impiegato SET livello = :livello WHERE id-codf = :id-  
codf;  
END!!  
  
SET term;!!
```

```
Esempio di esecuzione:  
EXECUTE PROCEDURE modifica_imp_livello ( 'aaaaaaaaaaaaaaaa', 30);
```

Op. 14 - Modifica del numero di punti assegnati al cliente:

```
SET term !!;  
  
CREATE PROCEDURE modifica_punti_cliente (id CHAR(16), punti  
INTEGER)  
AS  
BEGIN  
    UPDATE Cliente SET punti= :punti WHERE id-codf = :id;  
END!!  
  
SET term;!!
```

```
Esempio di esecuzione:  
EXECUTE PROCEDURE modifica_punti_cliente ( 'dddddddddddddddd',  
666);
```

Op. 15 - Determinazione dei premi a cui un cliente ha diritto:

```
SELECT Cliente.id, Prodotto.id  
    FROM (Cliente JOIN Prodotto  
        ON punti >= punti_vincita  
    ) WHERE id-codf = $id  
;
```

Op. 16 - Ritiro di un premio da parte di un cliente:

```
SET term !!;
```

```

CREATE PROCEDURE ritira_premio (cliente_id CHAR(16), premio
INTEGER)
AS
    DECLARE VARIABLE punti_totali INTEGER;
    DECLARE VARIABLE punti_da_sottrarre INTEGER;
BEGIN
    SELECT punti FROM Cliente WHERE id-codf = :id INTO
:punti_totali;
    SELECT punti_vincita FROM Prodotto WHERE id = :premio INTO
:punti_da_sottrarre;

    UPDATE Cliente SET punti = (:punti_totali -
:punti_da_sottrarre) WHERE id-codf = :id;
    INSERT INTO Riscossione (cliente_id, prodotto_id) VALUES
(:cliente_id, :premio);
    UPDATE Prodotto SET rimanenza = (rimanenza - 1) WHERE id =
:premio;
END!!

SET term;!!

```

Esempio di esecuzione:
EXECUTE PROCEDURE ritira_premio ('dddddddddddddddd', 1112);

Op. 17 - Determinazione dei prodotti più acquistati da un cliente:

```

SELECT prodotto_id, SUM(quantita)
FROM (Scontrino JOIN Vendita
      ON Scontrino.id = Vendita.scontrino_id
)
WHERE cliente_id = $id
GROUP BY prodotto_id
HAVING SUM (quantita) > 20;

```

Op. 18 - Determinazione della spesa totale effettuata da un cliente in un determinato periodo:

```

SET term !!;

CREATE PROCEDURE spesa_totale (
    cliente_id CHAR(16), data_inizio TIMESTAMP, data_fine
TIMESTAMP
) RETURNS(spesa_totale DECIMAL)
AS
BEGIN
    SELECT SUM(tot_spesa)
    FROM(Scontrino JOIN Vendita
        ON Scontrino.id = Vendita.scontrino_id
        AND cliente_id = :cliente_id
    )

```

```
                AND data >= :data_inizio
                AND data <= :data_fine
            ) INTO :spesa_totale
        ;
END!!

SET term;!!
```

Esempio di esecuzione:

```
EXECUTE PROCEDURE spesa_totale ( 'cccccccccccccccc', 'GEN-01-
1900', 'GEN-01-2001');
```