

Progettazione DB per catene di negozi di componenti per PC

Fiorellino Andrea, Bernardi Gabriele, Vergnani Mauro, Rebeca Sofroniou Spaletta

28 gennaio 2025

Indice

1	Analisi Requisiti	3
1.1	Linguaggio naturale	3
1.3	Strutturazione dei requisiti	5
1.4	Eliminazione delle ambiguità presenti (è stata effettuata una nuova intervista)	7
1.5	Specifiche operazioni	8
1.2	Glossario dei termini	9
2	Progettazione Concettuale	10
2.1	Identificazione delle entità (Bottom-Up)	10
2.2	Sviluppo delle entità (Bottom-Up)	11
2.2.1	Prodotti	11
2.2.2	Controparti	12
2.2.3	Ordini	13
2.2.4	Dipendenti	14
2.2.5	Fatture	15
2.2.6	Restituzioni	16
2.2.7	Negozi	16
2.2.8	Magazzini	17
2.3	Identificazione delle relazioni (Top-Down)	18
2.3.1	Identificazione Relazioni dipendente (Top-Down)	19
2.3.2	Identificazione Relazioni negozio (Top-Down)	20
2.3.3	Identificazione Relazioni ordine (Top-Down)	21
2.3.4	Identificazione Relazioni prodotto (Top-Down)	22
2.3.5	Identificazione Relazioni fattura (Top-Down)	23
2.3.6	Identificazione Relazioni reso (Top-Down)	24
2.3.7	Identificazione Relazioni controparte (Top-Down)	24
2.4	Schema ER finale	25
3	Progettazione Logica	26
3.1	Ristrutturazione dello schema concettuale	26
3.1.1	Rimozione delle gerarchie	26
3.1.2	Rimozione della ridondanza	27
3.2	Business Rules	27
3.3	Identificazione delle chiavi	27
3.4	Parsing del modello relazionale e dizionario dei dati	28
3.5	Tavole dei volumi e delle operazioni	32
3.5.1	Tavola dei volumi	32
3.5.2	Tavola delle operazioni	33
4	Codifica SQL	34
4.1	Creazione delle tabelle	34
4.2	Inserimenti in tabelle	38
4.3	Mostrare le tabelle	42

4.4	Rimuovere dalle tabelle	44
4.5	Aggiornare le tabelle	45
5	Testing	46
5.1	Link al sito che permette di testare il database	46

Capitolo 1

Analisi Requisiti

1.1 Linguaggio naturale

Si desidera progettare un database per la gestione di una **catena di negozi** specializzata nella vendita di componenti per PC (schede madri, memorie RAM, schede video, processori, ecc.).

Il database deve consentire di rappresentare e gestire dati relativi a prodotti, controparti, ordini, categorie di componenti, dipendenti, magazzini, fatture, nonché eventuali resi e rimborsi. Tutto ciò deve consentire l'identificazione del negozio e di conseguenza anche questo deve essere presente nel database.

Per quanto riguarda i *prodotti* offerti, troviamo i vari componenti che servono per assemblare un PC; ogni componente è rappresentato da un codice identificativo, nome, descrizione generale e prezzo di vendita; sono inoltre richieste alcune informazioni specifiche per tutte e sole le varie categorie di prodotti trattati:

- **RAM:** tipo di tecnologia (DDR4, DDR5, ecc.), capacità (es. 8 GB, 16 GB), velocità (es. 3200 MHz), tensione operativa, latenza.
- **Schede video:** marca, modello, tipo di memoria (es. GDDR6), quantità di memoria, numero di porte (es. HDMI, DisplayPort), potenza richiesta, compatibilità con DirectX o OpenGL.
- **Processori:** architettura (es. x86, ARM), numero di core, frequenza base, frequenza boost, cache, compatibilità con socket.
- **Schede madri:** tipo di socket, chipset, formato (es. ATX, Micro-ATX), numero di slot RAM, numero di porte PCIe, supporto per velocità della RAM, connettività (es. Wi-Fi, Bluetooth).
- **Memorie di archiviazione:** tipo (SSD, HDD, NVMe), capacità (es. 1 TB, 512 GB), velocità di lettura/scrittura, interfaccia (es. SATA, PCIe).

Per quanto riguarda le *controparti*, troviamo due tipi di controparti; ogni controparte è rappresentato da un codice identificativo, il nome, l'indirizzo, il numero di telefono; inoltre vi sono richieste alcune informazioni specifiche per tutte e solo le due categorie di controparti trattate:

- **Fornitori:** Lista dei prodotti forniti.
- **Clienti:** Cognome, indirizzo mail, eventuali resi e rimborsi.

Ogni vendita genera una *fattura* contenente un Numero univoco della fattura, data di emissione, dati del cliente, elenco dei prodotti venduti (con quantità e prezzo unitario), importo totale, IVA applicata e modalità di pagamento; ogni ricevuta potrebbe essere collegata a eventuali richieste di

reso o rimborso tenendo quindi traccia della motivazione.

Per quanto riguarda gli *ordini*, troviamo due tipi di ordini; ogni ordine è rappresentato da un codice dell'ordine, la data della transazione, il negozio presente nella trattativa e l'elenco dei prodotti (con quantità); inoltre vi sono richieste alcune informazioni specifiche per tutte e solo le due categorie di ordini trattati:

- **Vendite:** Cliente coinvolto, dipendente responsabile.
- **Acquisti:** Data prevista di consegna.

Per quanto riguarda lo staff, ogni *dipendente* è caratterizzato da: Numero di matricola, nome, cognome, posizione lavorativa, numero di telefono, data di assunzione, e informazioni sui negozi di assegnazione.

Per quanto riguarda i negozi, ogni *negozio* è caratterizzato da: la denominazione, l'indirizzo della sede, codice identificativo

Per quanto concerne le sedi di stoccaggio della merce si desidera avere un codice del magazzino, indirizzo, lista dei prodotti presenti con quantità disponibile per ciascun prodotto in ogni sede, e quali negozi rifornisce.

1.3 Strutturazione dei requisiti

- **Fraasi di carattere generale:** Si desidera progettare un database per la gestione di una catena di negozi specializzata nella vendita di componenti per PC (schede madri, memorie RAM, schede video, processori, ecc.).
Il database deve consentire di rappresentare e gestire dati relativi a prodotti, fornitori, clienti, ordini, categorie di componenti, dipendenti, magazzini, fatture, nonché eventuali resi e rimborsi.
- **Fraasi relative alle fatture:** Ogni vendita genera una fattura contenente un Numero univoco della fattura, data di emissione, dati del cliente, elenco dei prodotti venduti (con quantità e prezzo unitario), importo totale, IVA applicata e modalità di pagamento.
- **Fraasi relative a resi e rimborsi:** ogni ricevuta potrebbe essere collegata a eventuali richieste di reso o rimborso tenendo quindi traccia della motivazione.
- **Fraasi relative alle controparti:** Per quanto riguarda le controparti, troviamo due tipi di controparti; ogni controparte è rappresentato da un codice identificativo, il nome, l'indirizzo, il numero di telefono; inoltre vi sono richieste alcune informazioni specifiche per tutte e sol le due categorie di controparti trattati:
- **Fraasi relative ai fornitori:** Lista dei prodotti forniti.
- **Fraasi relative ai clienti:** Cognome, indirizzo mail, eventuali resi e rimborsi.
- **Fraasi relative agli ordini:** Per quanto riguarda gli ordini, troviamo due tipi di ordini; ogni ordine è rappresentato da un codice dell'ordine, la data della transazione, il negozio presente nella trattativa, elenco dei prodotti (con quantità); inoltre vi sono richieste alcune informazioni specifiche per tutte e solo le due categorie di ordini trattati:
- **Fraasi relative agli acquisti:** Data prevista di consegna.
- **Fraasi relative alle vendite:** Cliente coinvolto, dipendente responsabile, prezzo unitario dei prodotti venduti.
- **Fraasi relative ai prodotti:** Per quanto riguarda i prodotti offerti , troviamo i vari componenti che servono per assemblare un PC; ogni componente è rappresentato da un codice identificativo, nome, descrizione generale e prezzo di vendita; sono inoltre richieste alcune informazioni specifiche per tutte e sole le varie categorie di prodotti trattati:
- **Fraasi relative alle RAM:** Tipo di tecnologia (DDR4, DDR5, ecc.), capacità (es. 8 GB, 16 GB), velocità (es. 3200 MHz), tensione operativa, latenza.
- **Fraasi relative alle Schede Video:** Marca, modello, tipo di memoria (es. GDDR6), quantità di memoria, numero di porte (es. HDMI, DisplayPort), potenza richiesta, compatibilità con DirectX o OpenGL.
- **Fraasi relative ai Processori:** Architettura (es. x86, ARM), numero di core, frequenza base, frequenza boost, cache, compatibilità con socket.
- **Fraasi relative alle Schede Madri:** Tipo di socket, chipset, formato (es. ATX, Micro-ATX), numero di slot RAM, numero di porte PCIe, supporto per velocità della RAM, connettività (es. Wi-Fi, Bluetooth)..
- **Fraasi relative alle Memorie di Archiviazioni:** Tipo (SSD, HDD, NVMe), capacità (es. 1 TB, 512 GB), velocità di lettura/scrittura, interfaccia (es. SATA, PCIe).

- **Frase relative alle Sedi di Stoccaggio:** Per quanto concerne le sedi di stoccaggio della merce si desidera avere un codice del magazzino, indirizzo, lista dei prodotti presenti con quantità disponibile per ciascun prodotto in ogni sede, e quali negozi rifornisce.
- **Frase relative ai negozi:** Per quanto riguarda i negozi, ogni *negozio* è caratterizzato da: la denominazione, l'indirizzo della sede, codice identificativo
- **Frase relative allo staff:** Per quanto riguarda lo staff, ogni dipendente è caratterizzato da: Numero di matricola, nome, cognome, posizione lavorativa, numero di telefono, data di assunzione, e informazioni sui negozi di assegnazione.

1.4 Eliminazione delle ambiguità presenti (è stata effettuata una nuova intervista)

- I Clienti non sono obbligati ad aver acquistato qualcosa in passato per essere nel database, i clienti che si sono registrati sono sempre inseriti nel database in quanto possibili acquirenti e non per forza con ordini già eseguiti;
- Negli ordini esistono sempre e solo due entità, la controparte e il negozio;
- Ogni dipendente può lavorare al massimo per 1 negozio contemporaneamente ma ha la possibilità di cambiare il negozio.

1.5 Specifica operazioni

- inserire un ordine d'acquisto
- inserire un ordine di vendita
- inserire un membro dello staff di un negozio
- rimuovere un membro dello staff di un negozio
- inserire un nuovo magazzino
- rimuovere un magazzino
- modificare la quantità di un prodotto in un magazzino
- inserire una fattura
- inserire una controparte
- inserire un cliente
- inserire un fornitore
- inserire una RAM
- inserire una Scheda Video
- inserire un Processore
- inserire una Scheda Madre
- inserire una Memoria d'archiviazione
- inserire un rimborso
- mostrare la lista dei dipendenti che lavorano in un negozio specifico
- mostrare i clienti che hanno acquistato in un negozio specifico
- mostrare i prodotti disponibili in un magazzino di un negozio specifico
- mostrare gli acquisti di un cliente presso un determinato negozio
- mostrare tutti gli acquisti di tutti i clienti presso un determinato negozio
- mostrare gli ordini evasi da un dipendente specifico
- mostrare il costo totale d'acquisto di un tipo di prodotto
- mostrare l'importo totale di vendita di un tipo di prodotto
- mostrare tutti i dettagli di un magazzino
- mostrare tutti i magazzini
- mostrare tutti i negozi
- mostrare i dettagli di un negozio
- mostrare la fattura associata a un ordine specifico

1.2 Glossario dei termini

Termine	Descrizione	Sinonimi	Collegamenti
Componenti	Componenti hardware per PC come RAM, processori, schede madri, ecc.	Prodotti	Fornitori, Ordini, Magazzini
Fornitori	Rappresentanti delle aziende che forniscono i prodotti ai negozi, con dettagli su recapiti e prodotti offerti.	-	Acquisti, Componenti, Controparti
Ordini	Ordine generico di acquisto o vendita, con informazioni sui prodotti coinvolti e sul negozio che svolge l'ordine	-	Prodotti, Negozi
Acquisti	Richieste di prodotti inviate ai fornitori. Contengono data, stato e dettagli prodotti.	-	Fornitori, Prodotti, Ordini
Clienti	Persone o aziende che acquistano prodotti dal negozio. Identificati con codice cliente e contatti.	Acquirenti	Vendite, Resi, Controparti
Vendite	Transazioni commerciali che includono cliente, prodotti, fattura e responsabile.	Transazioni	Fatture, Clienti, Dipendenti, Ordini
Fatture	Documenti generati per ogni vendita con dettagli su importo, IVA e modalità di pagamento.	Ricevute	Vendite, Resi
Resi	Richieste di restituzione, rimborso o sostituzione di prodotti, con motivazione e stato.	Restituzioni, rimborsi	Fatture
Dipendenti	Persone che lavorano nei negozi, identificati da matricola, nome e posizione lavorativa.	Staff	Vendite, Negozi
Magazzini	Sedi di stoccaggio dei prodotti con dettagli su quantità e ubicazione e sui negozi riforniti.	Sedi di Stoccaggio	Componenti, Negozi
Controparti	Entità con cui facciamo lo scambio di merce con denaro.	-	-
Negozi	punti vendita fisici e virtuali, specializzati in componenti per PC	-	Ordini, Vendite, Dipendenti, Magazzini

Tabella 1.1: Glossario dei termini utilizzati nella progettazione del database.

Capitolo 2

Progettazione Concettuale

2.1 Identificazione delle entità (Bottom-Up)

- Controparte
- Fornitore
- Cliente
- Prodotto
- RAM
- Scheda Video
- Memoria d'Archiviazione
- Scheda Madre
- Processore
- Fattura
- Ordine
- Acquisti
- Vendite
- Rimborso
- Magazzino
- Dipendenti
- Negozio

2.2 Sviluppo delle entità (Bottom-Up)

2.2.1 Prodotti

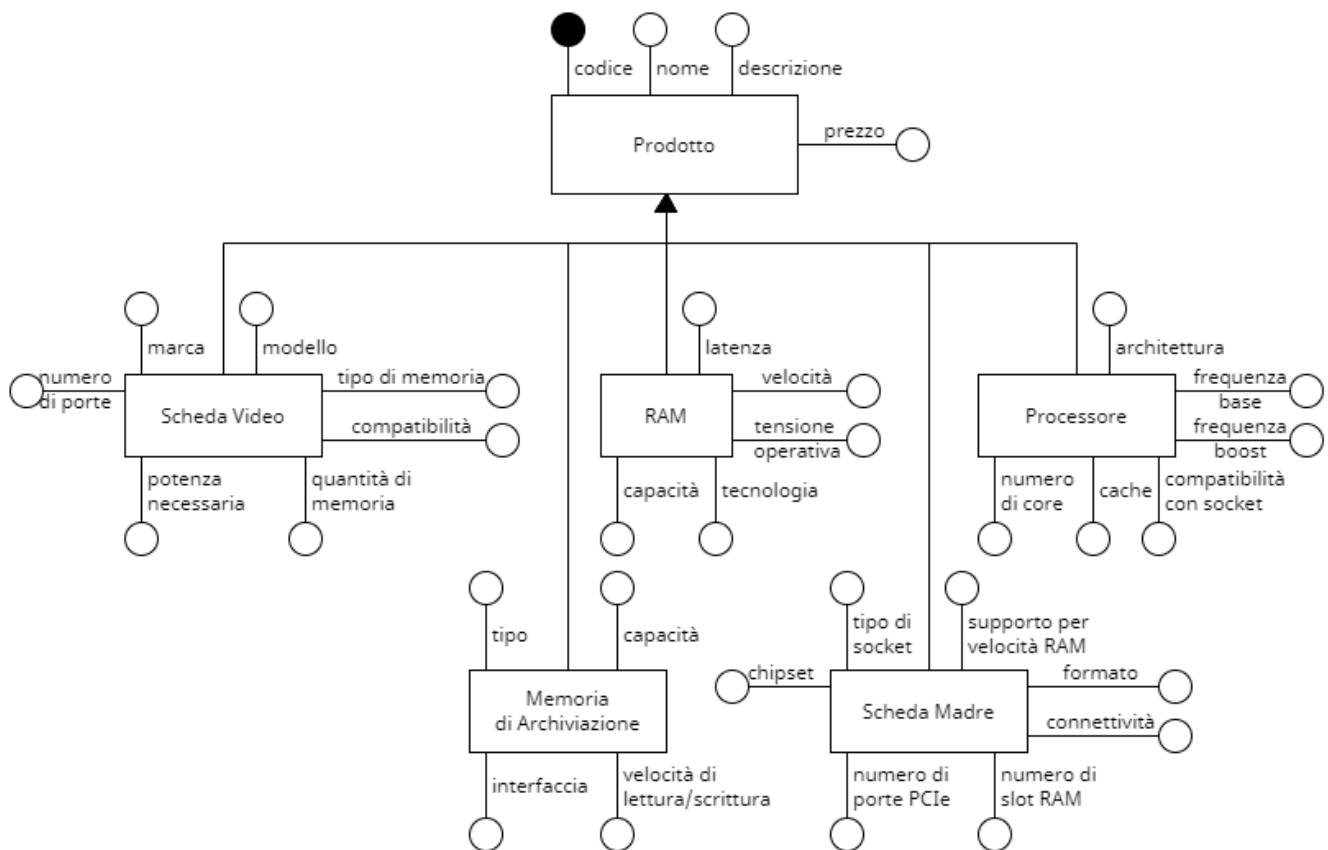


Figura 2.1: Diagramma iniziale Bottom-Up dei Prodotti

2.2.2 Controparti

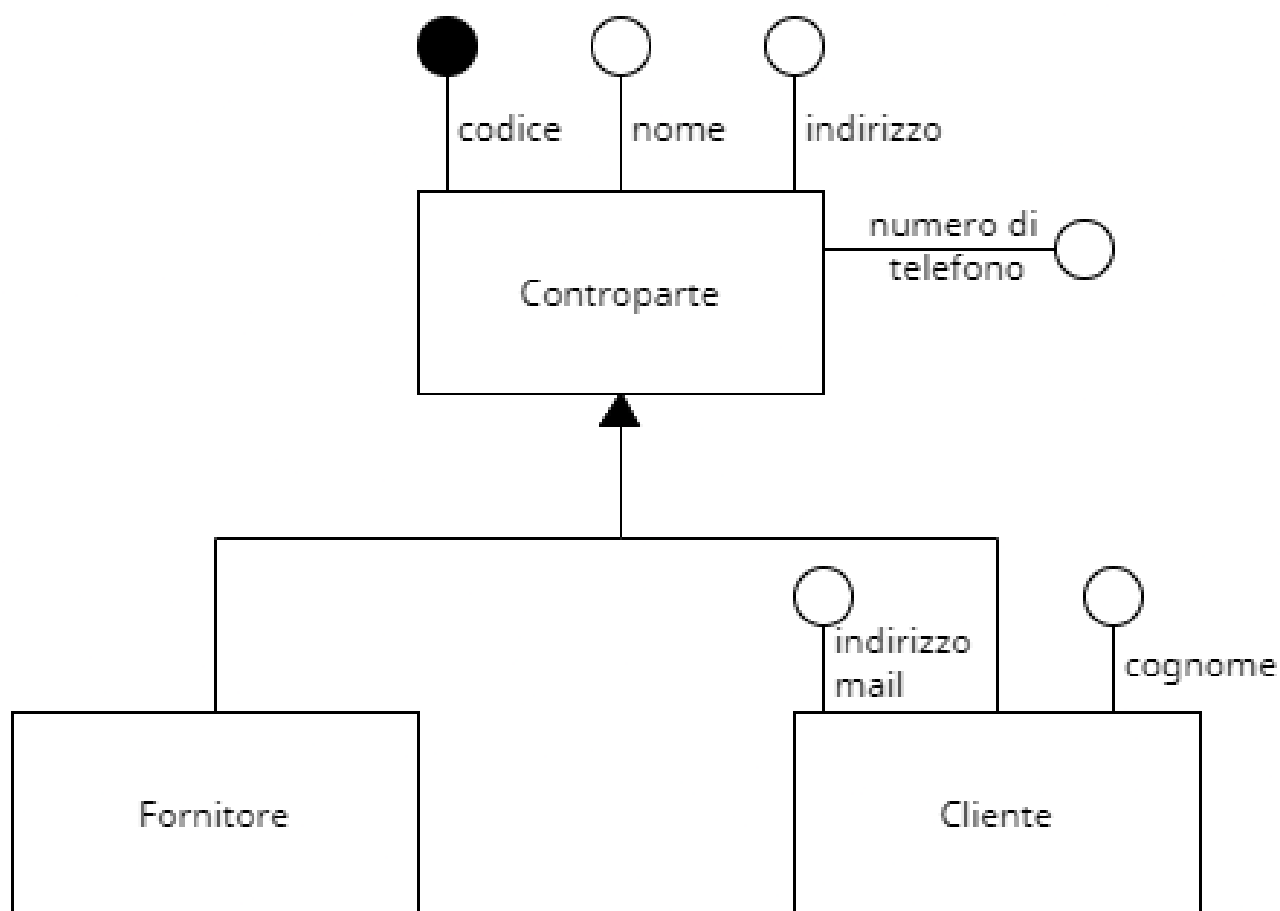


Figura 2.2: Diagramma iniziale Bottom-Up delle Controparti

2.2.3 Ordini

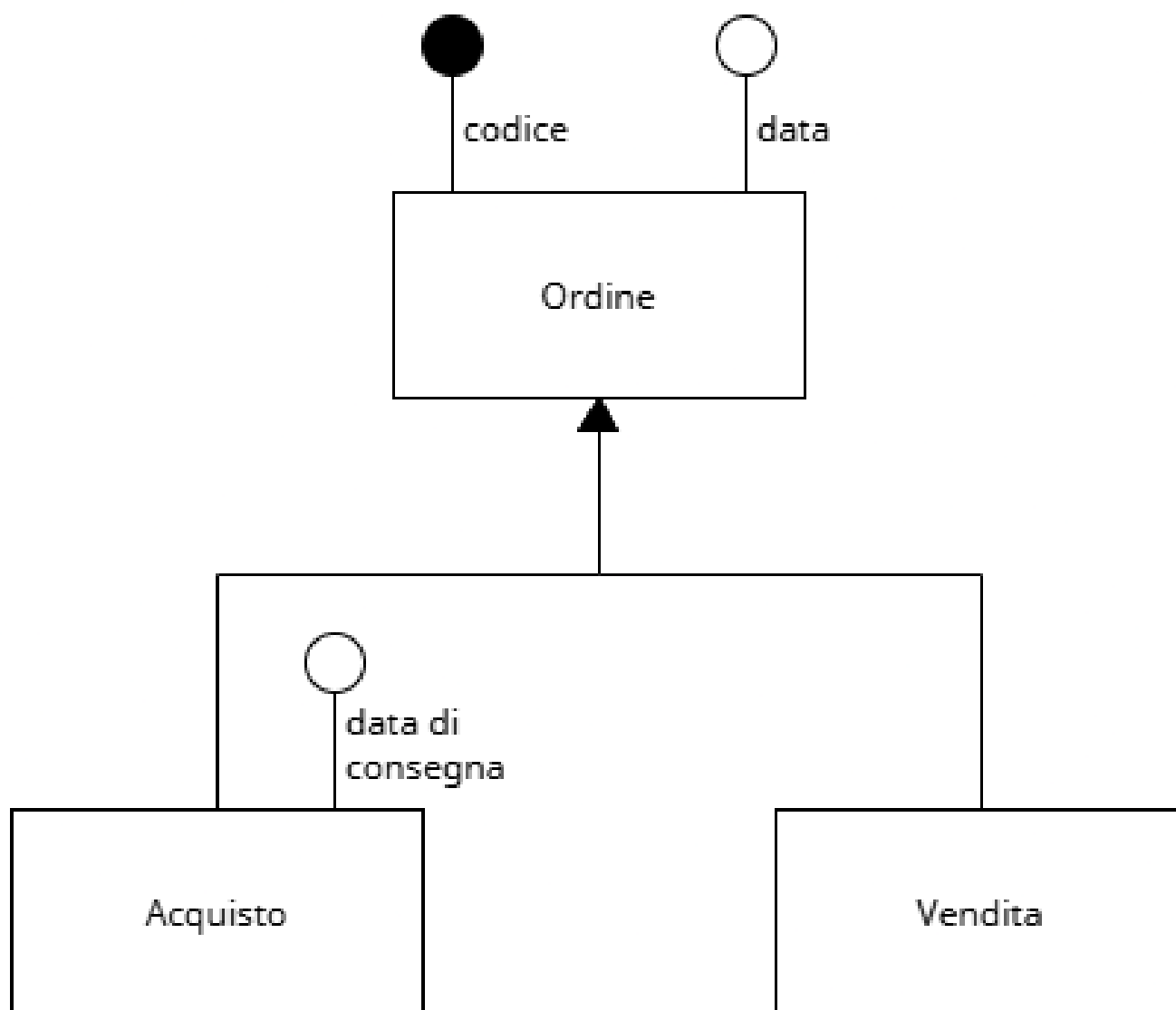


Figura 2.3: Diagramma iniziale Bottom-Up degli Ordini

2.2.4 Dipendenti

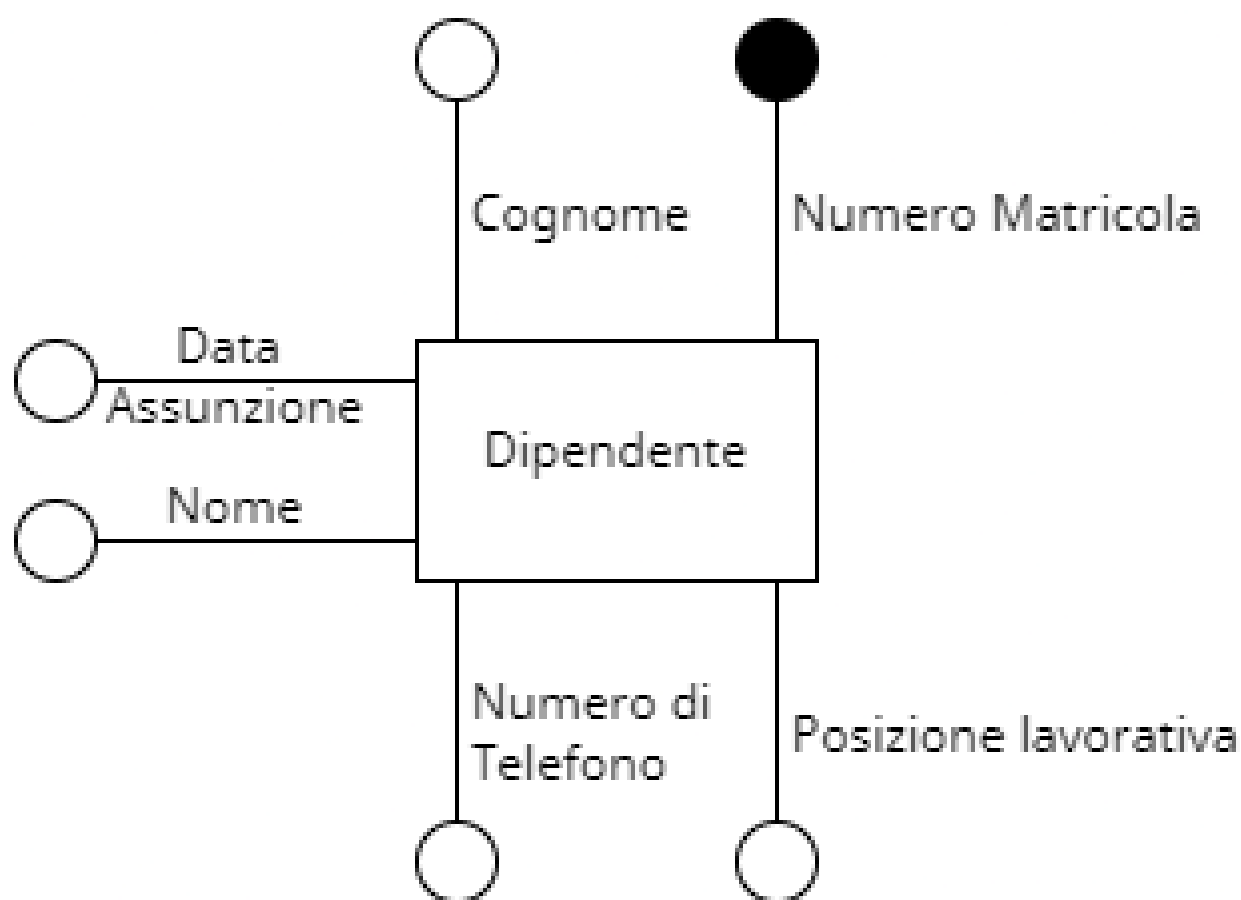


Figura 2.4: Diagramma iniziale Bottom-Up dei Dipendenti

2.2.5 Fatture

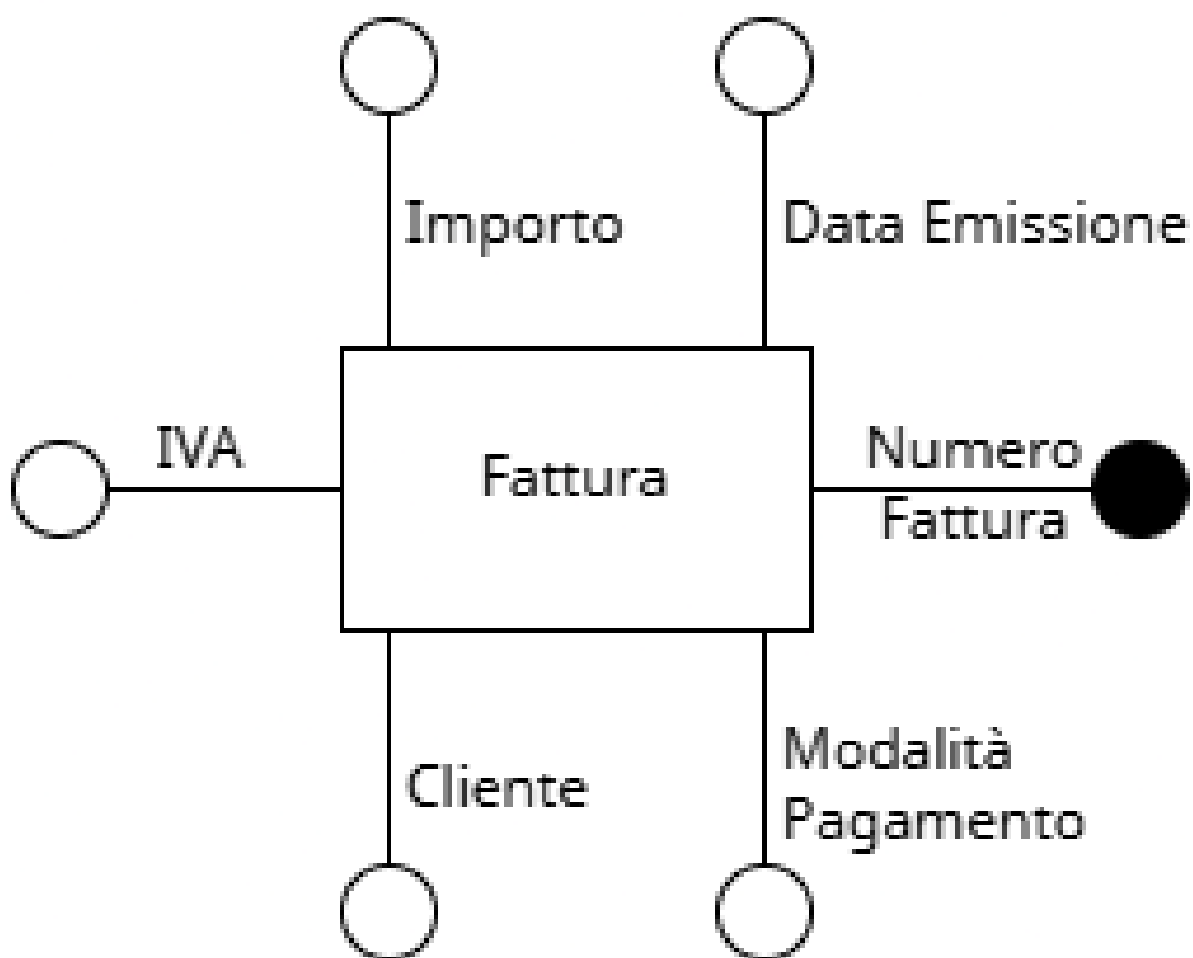


Figura 2.5: Diagramma iniziale Bottom-Up delle Fatture

2.2.6 Restituzioni

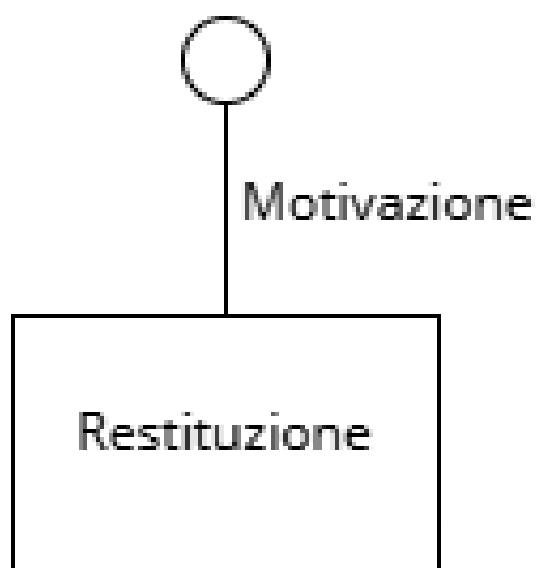


Figura 2.6: Diagramma iniziale Bottom-Up delle Restituzioni

2.2.7 Negozi

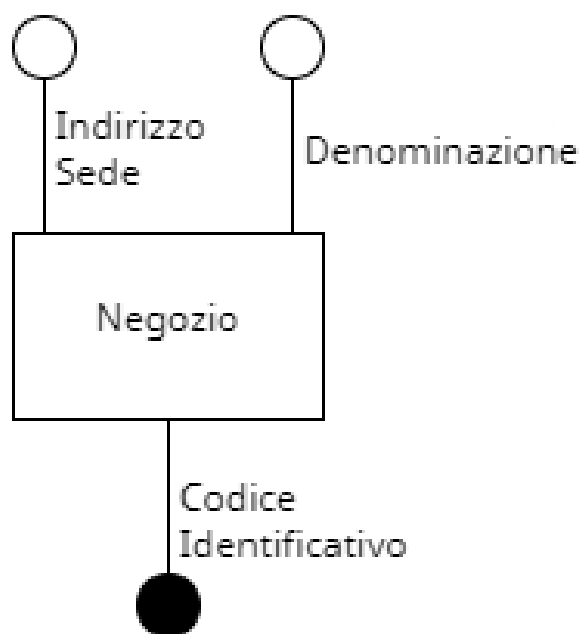


Figura 2.7: Diagramma iniziale Bottom-Up dei Negozi

2.2.8 Magazzini

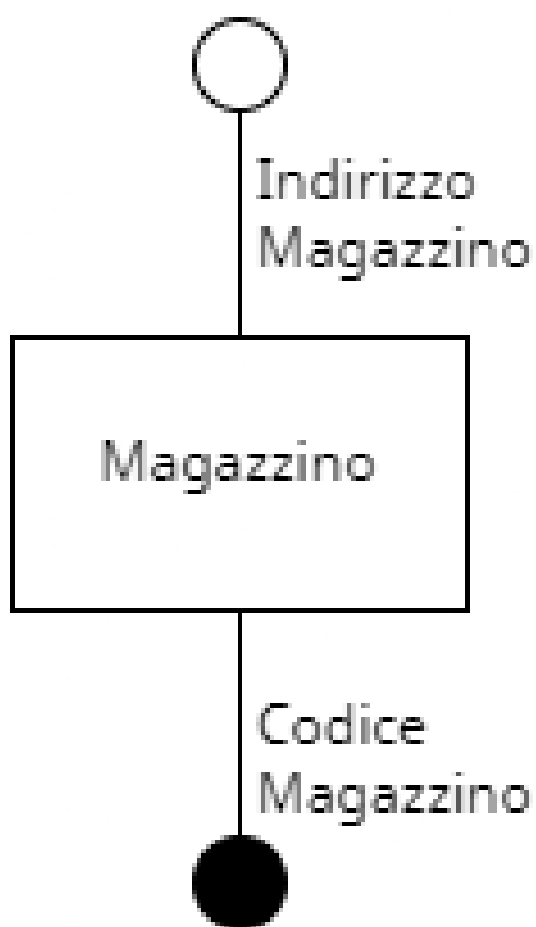


Figura 2.8: Diagramma iniziale Bottom-Up dei Magazzini

2.3 Identificazione delle relazioni (Top-Down)

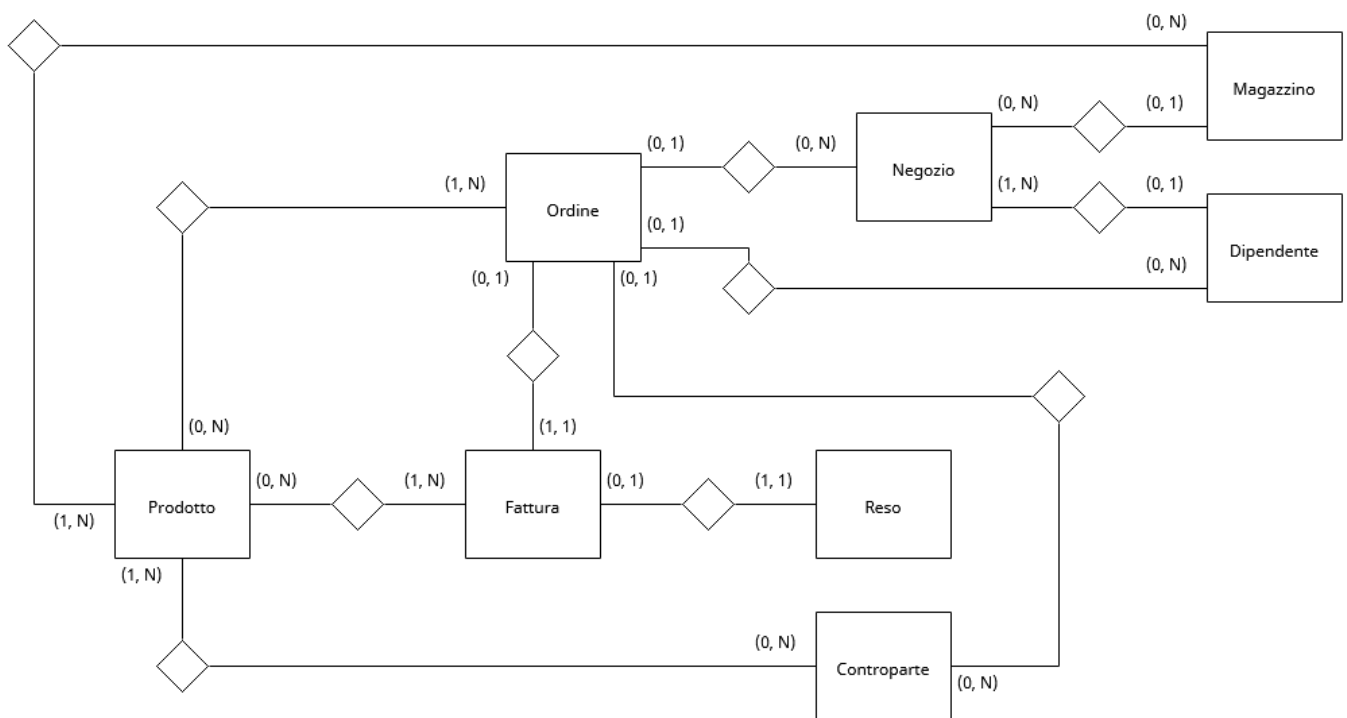


Figura 2.9: Diagramma iniziale Top-Down

2.3.1 Identificazione Relazioni dipendente (Top-Down)

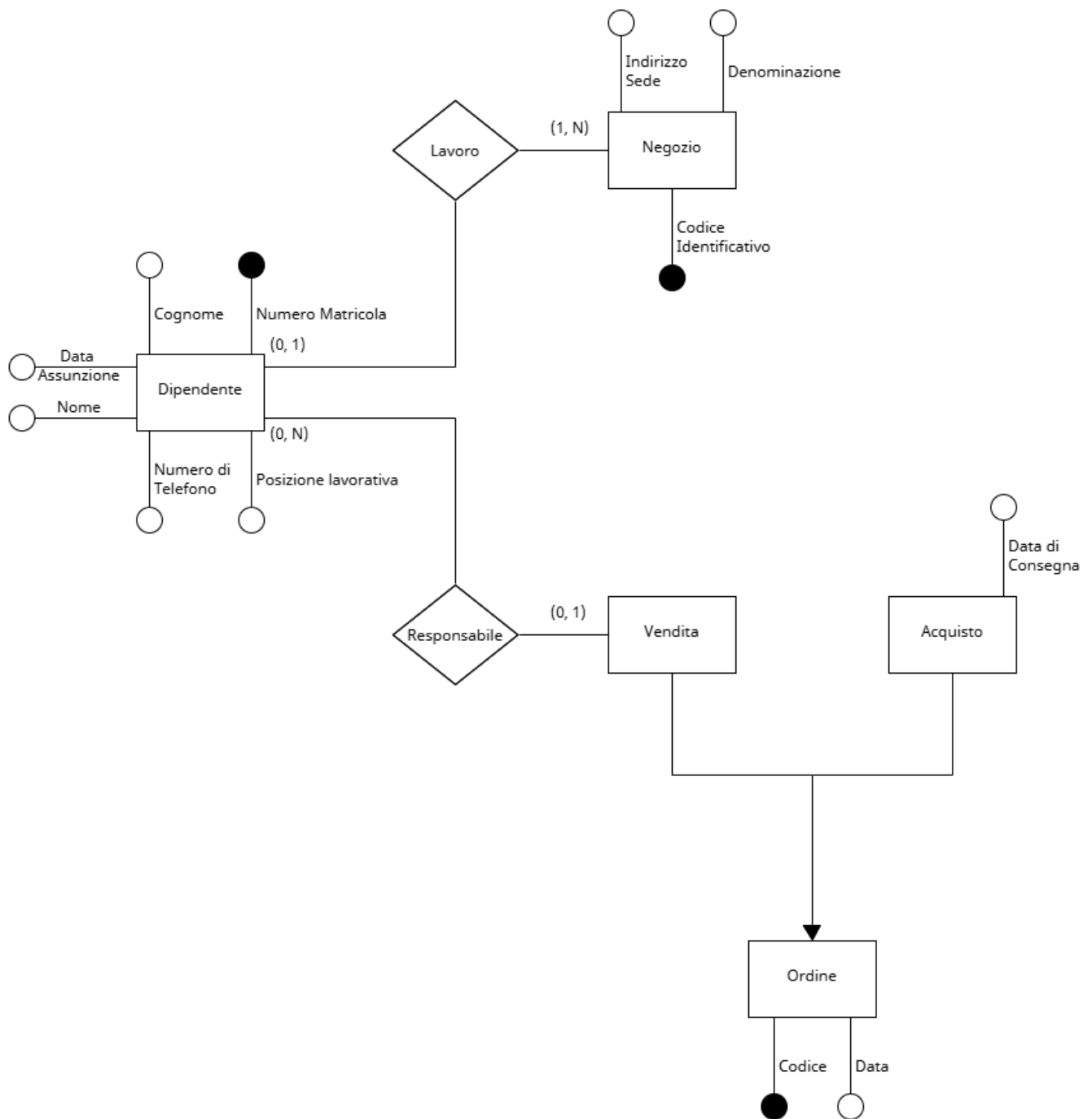


Figura 2.10: Diagramma dipendente Top-Down

2.3.2 Identificazione Relazioni negozio (Top-Down)

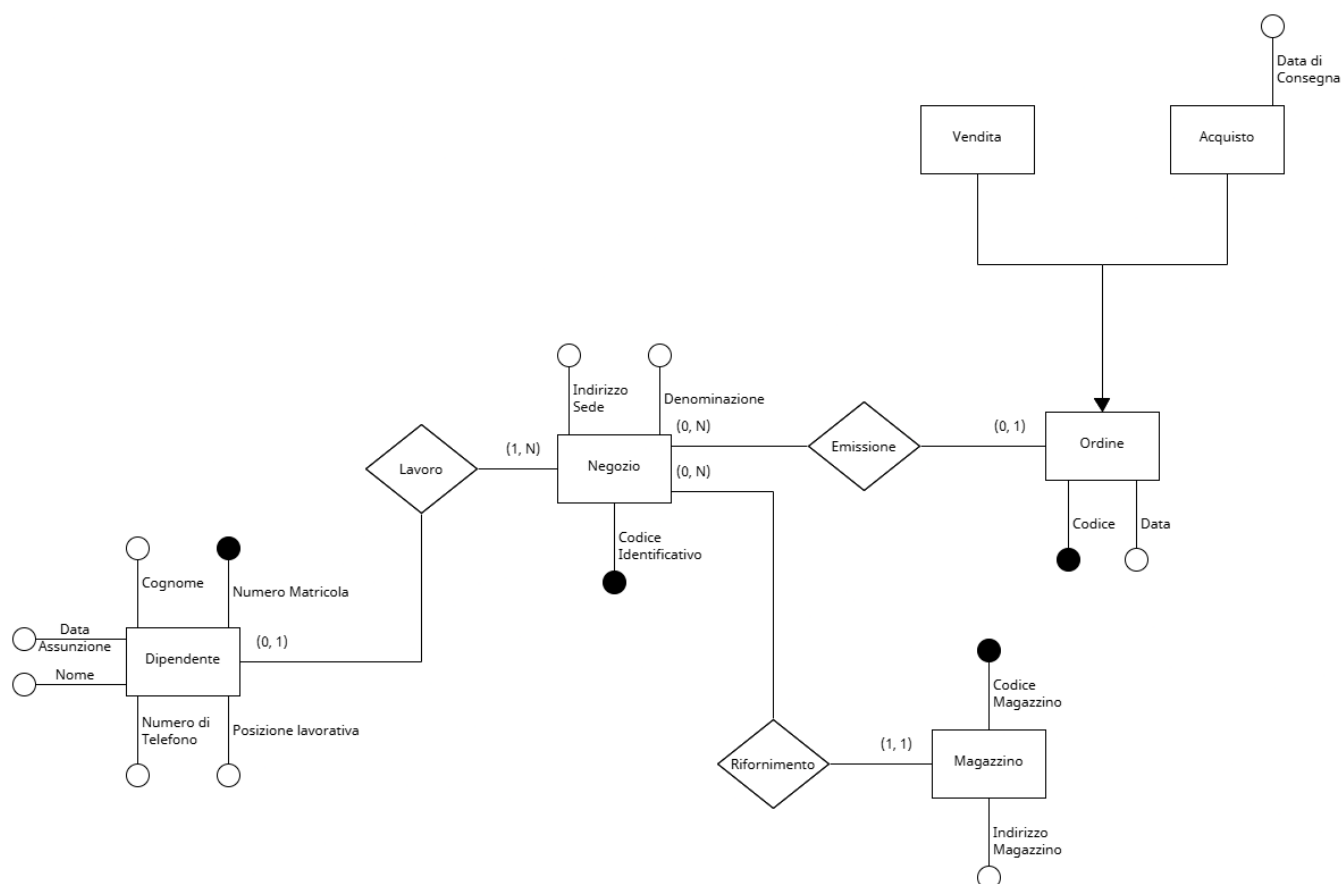


Figura 2.11: Diagramma negozio Top-Down

2.3.3 Identificazione Relazioni ordine (Top-Down)

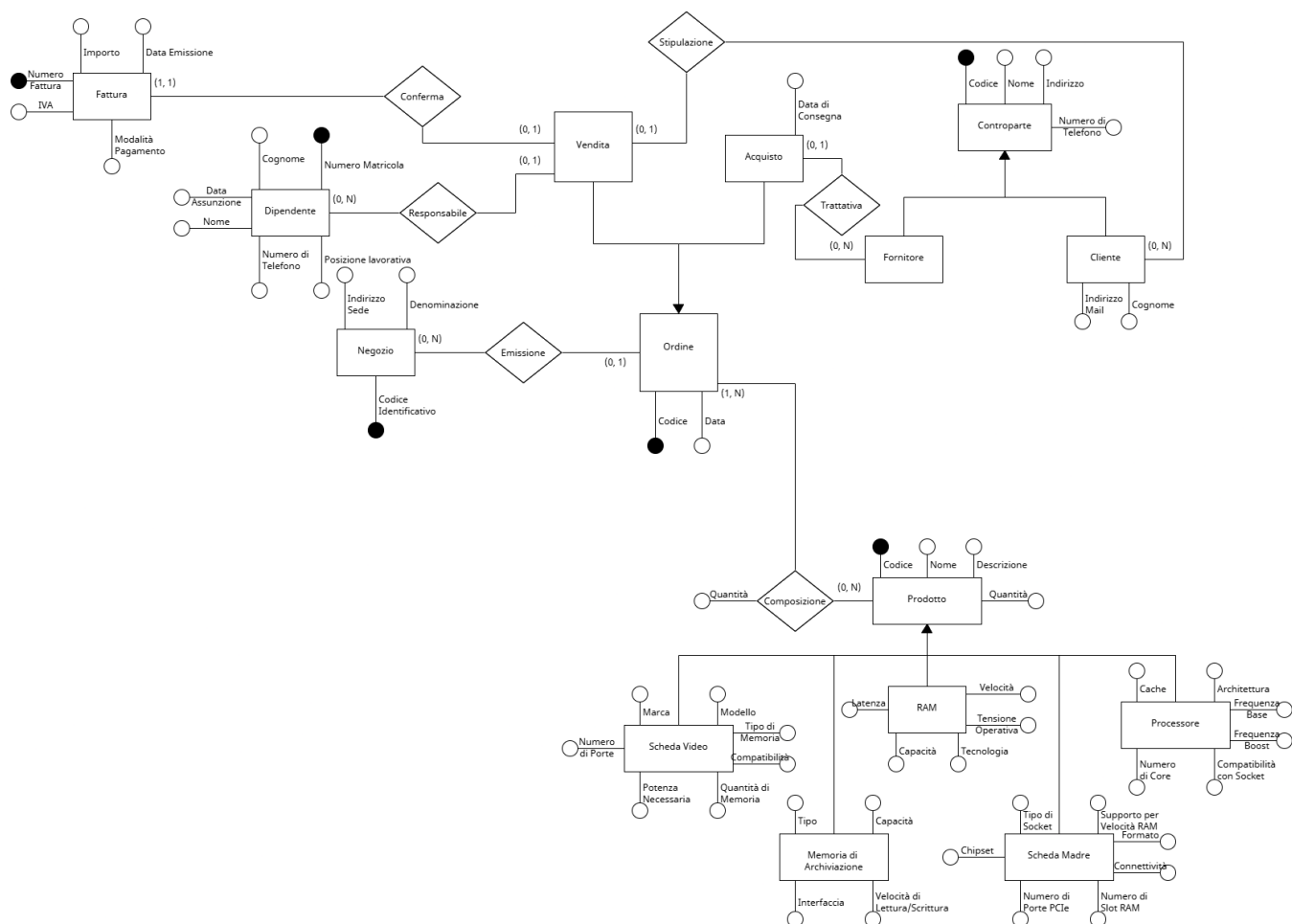


Figura 2.12: Diagramma ordine Top-Down

2.3.4 Identificazione Relazioni prodotto (Top-Down)

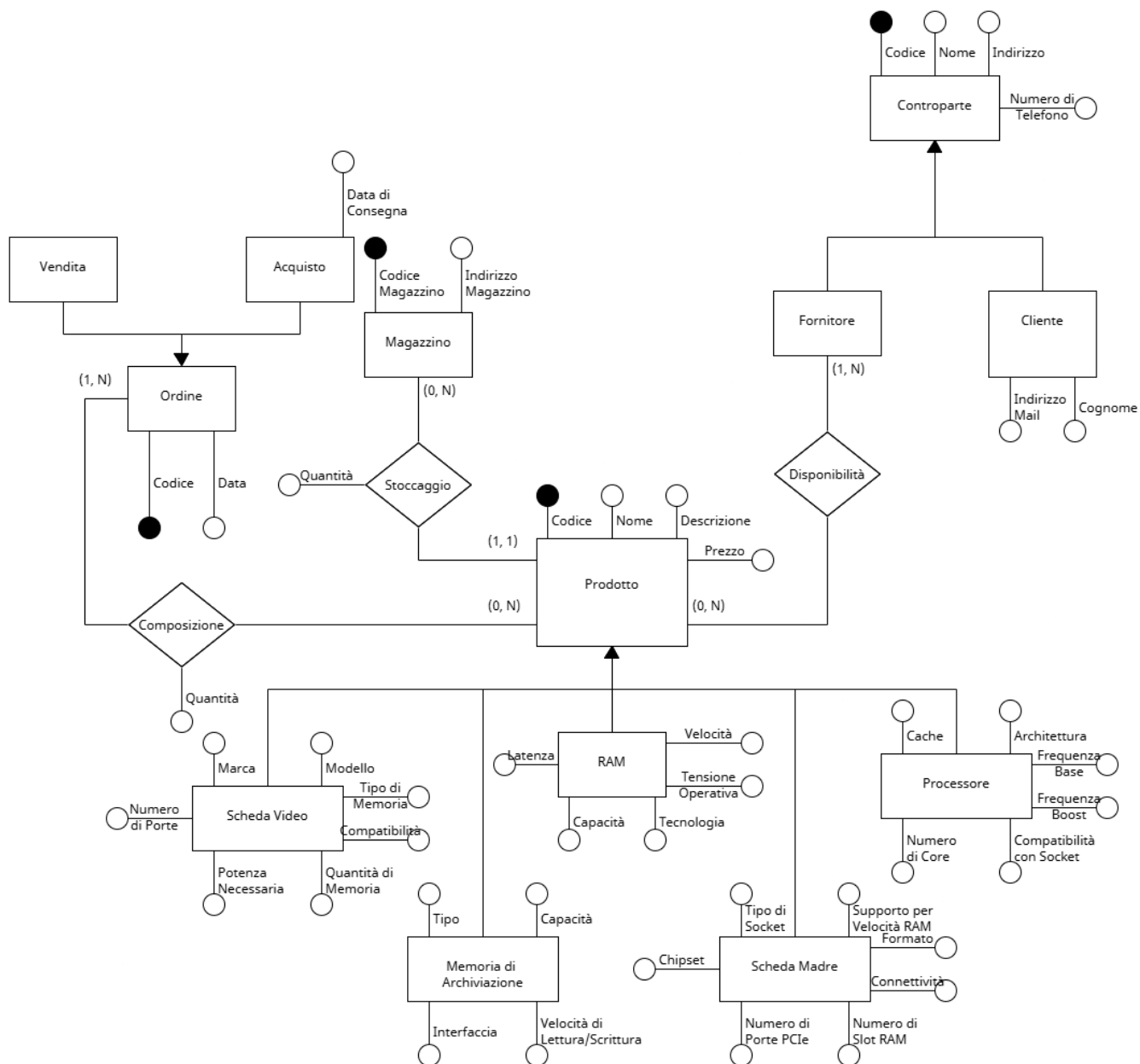


Figura 2.13: Diagramma prodotto Top-Down

2.3.5 Identificazione Relazioni fattura (Top-Down)

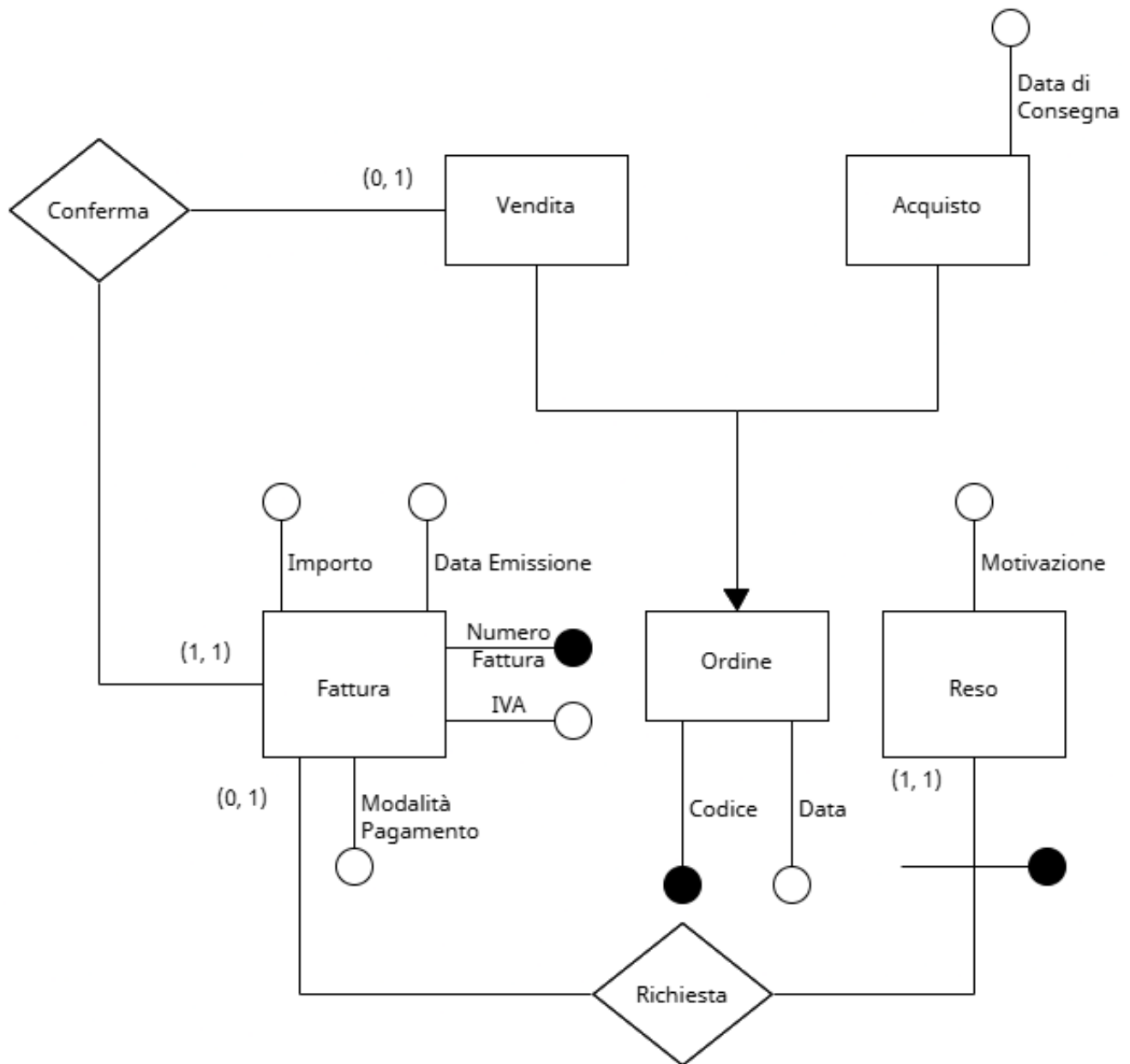


Figura 2.14: Diagramma fattura Top-Down

2.3.6 Identificazione Relazioni reso (Top-Down)

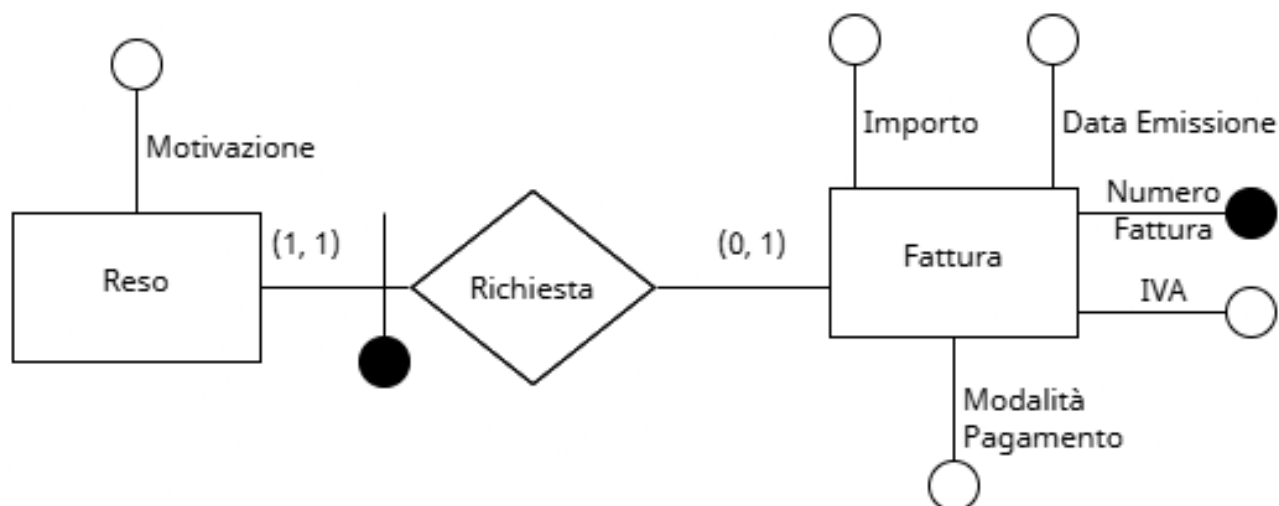


Figura 2.15: Diagramma reso Top-Down

2.3.7 Identificazione Relazioni controparte (Top-Down)

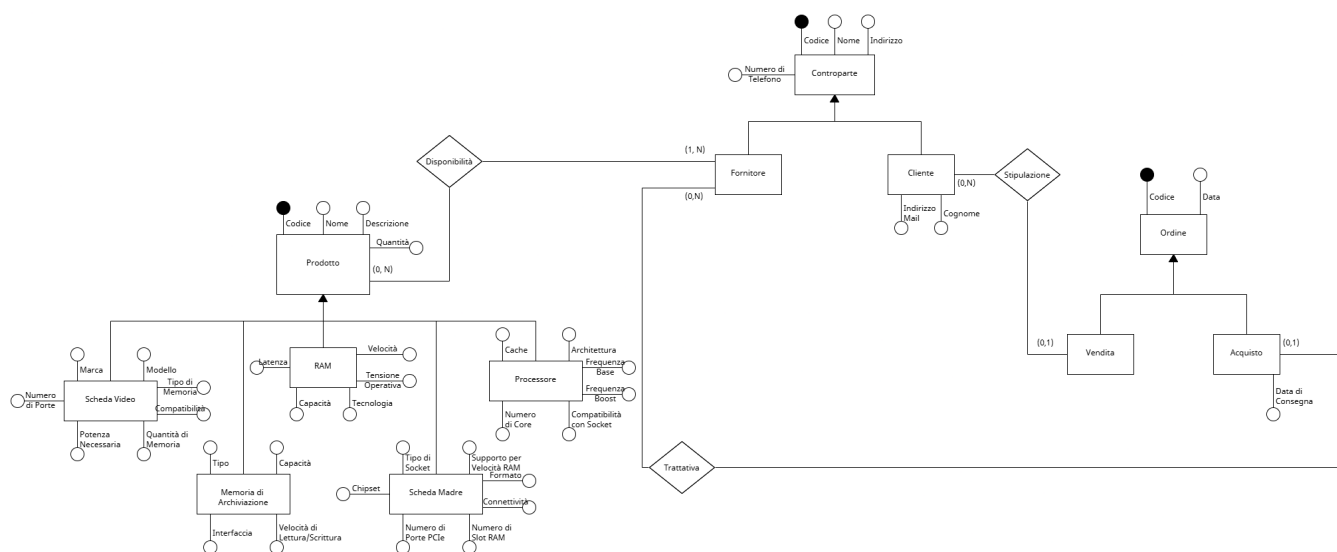


Figura 2.16: Diagramma controparte Top-Down

2.4 Schema ER finale

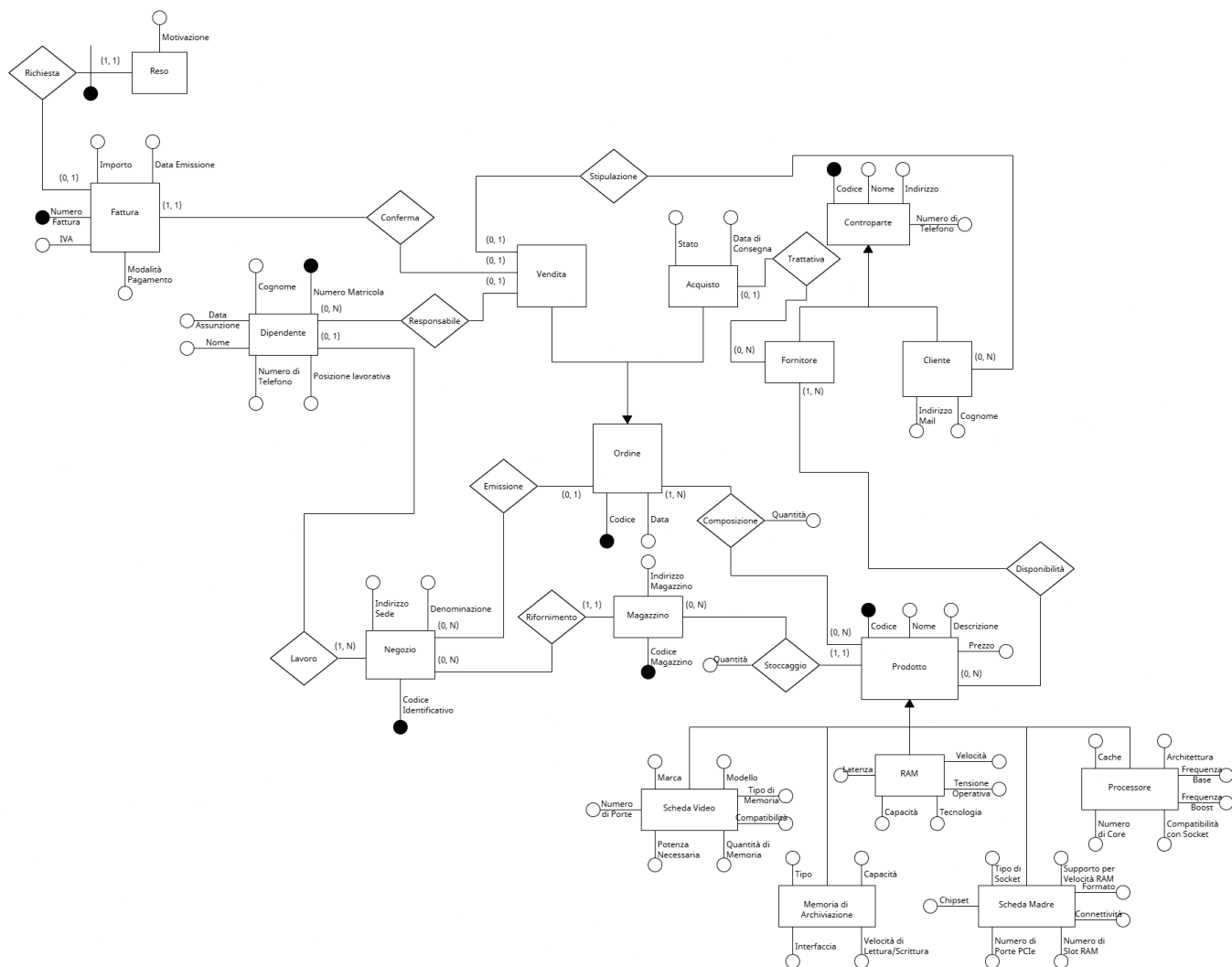


Figura 2.17: Schema ER finale

Capitolo 3

Progettazione Logica

3.1 Ristrutturazione dello schema concettuale

3.1.1 Rimozione delle gerarchie

Le gerarchie di ereditarietà vanno ricostruite come vincoli di integrità referenziale, dove il figlio ha un riferimento al codice del padre, che è anche la chiave primaria del figlio, mentre il padre non ha conoscenza del suo possibile figlio

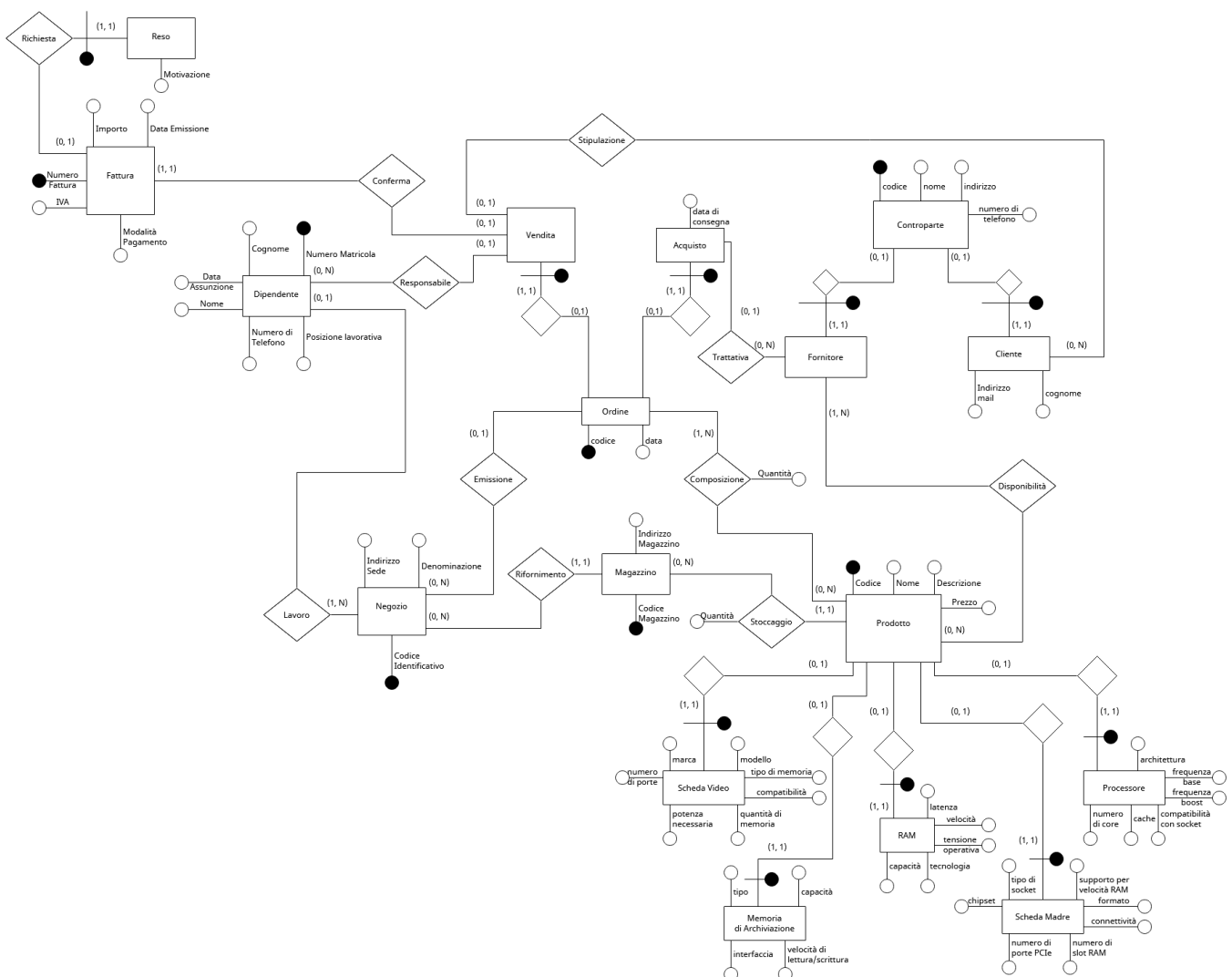


Figura 3.1: Schema ER Finale senza ereditarietà

3.1.2 Rimozione della ridondanza

Fortunatamente il database è già privo di ridondanze

3.2 Business Rules

- Un Cliente può effettuare un reso SSE ha fatto almeno un acquisto (numero resi < numero acquisti);
- Nel caso il cliente decida di effettuare il reso, deve farlo su TUTTI i prodotti che rientrano nell'acquisto (non solo di alcuni);
- I prodotti coinvolti in un reso vengono rimessi nel primo magazzino disponibile legato al negozio dal quale sono stati acquistati e rimessi in vendita come se non fossero mai stati venduti;
- Un cliente, anche se non ha mai fatto un acquisto, può comunque essere presente nel DB;
- Il numero di prodotti in un magazzino è maggiore o uguale a 0;
- Un dipendente lavora al massimo per 1 negozio;
- Si può vendere solamente i prodotti che sono ancora nel magazzino (quindi che sono vendibili);
- I volumi vengono determinati assumendo di monitorare una giornata di attività all'interno di una catena di negozi;
- Per rimuovere un magazzino è necessario che il suddetto magazzino non abbia prodotti con quantità > 0;
- Un cliente può decidere di effettuare un reso in qualsiasi momento senza vincoli temporali;
- Un prodotto ha lo stesso prezzo sia per i clienti che comprano dal negozio che per questi ultimi che si riforniscono dai Fornitori;
- Il prezzo di un prodotto NON è modificabile in alcun modo (non è possibile applicare sconti, abbassare/alzare il prezzo, etc...);
- La fattura non è generata in automatico ma deve essere creata manualmente da una persona;
- Per effettuare un reso, per ragioni aziendali, è necessario prima richiedere una fattura
- Il prezzo dei prodotti NON considera la quota di IVA;
- È necessario tenere traccia dell'IVA in fattura, poichè lo stato potrebbe decidere di modificare la percentuale e dunque le scorse fatture devono comunque contare quella valida al momento della loro emissione;
- Un reso è sempre accettato, basta solo che il cliente lo richieda e ne fornisca una motivazione.

3.3 Identificazione delle chiavi

Durante la fase di progettazione concettuale, avevamo già previsto l'inserimento di opportuni attributi "Codice" nei casi in cui non fossero disponibili altri candidati a chiavi primarie "adeguati". Questa soluzione dovrebbe essere considerata un semplice espediente da utilizzare con moderazione.

3.4 Parsing del modello relazionale e dizionario dei dati

Entità	Traduzione	Descrizione
Prodotto	Prodotto(<u>Codice</u> , Nome, Descrizione, Prezzo)	Prodotto in vendita, non esiste mai nella forma generale nel database senza un'istanza di una sua forma più specifica
Scheda Video	SchedaVideo(Codice , Marca, Modello, TipoMemoria, QuantitàMemoria, NumeroPorte, PotenzaNecessaria, Compatibilità)	Scheda video in vendita, Prodotto specifico; Codice → Prodotto.Codice
RAM	RAM(<u>Codice</u> , Tecnologia, Capacità, Velocità, TensioneOperativa, Latenza)	RAM in vendita, Prodotto specifico; Codice → Prodotto.Codice
Processore	Processore(Codice , Architettura, NumeroCore, FrequenzaBase, FrequenzaBoost, Cache, CompatibilitàSocket)	Processore in vendita, Prodotto specifico; Codice → Prodotto.Codice
Scheda Madre	SchedaMadre(Codice , TipoSocket, Chipset, Formato, NumeroSlotRAM, NumeroPortePCIe, SupportoVelocitàRAM, Connettività)	Scheda Madre in vendita, Prodotto specifico; Codice → Prodotto.Codice
Memoria di Archiviazione	MemoriaArchiviazione(Codice , Tipo, Capacità, VelocitàRW, Interfaccia)	Memoria di Archiviazione in vendita, Prodotto specifico; Codice → Prodotto.Codice
Controparte	Controparte(<u>Codice</u> , Nome, Indirizzo, NumeroTelefono)	Una persona coinvolta in acquisti o vendite, è sempre istanziato insieme a un fornitore o cliente che sia la sua forma più specifica

Continua nella pagina successiva...

Entità	Traduzione	Descrizione
Fornitore	Fornitore(<u>Codice</u> , Nome, Indirizzo, NumeroTelefono)	Persona che rappresenta un fornitore con cui facciamo acquisti; Codice → Controparte.Codice
Cliente	Cliente(<u>Codice</u> , Nome, Cognome, Indirizzo, IndirizzoMail, NumeroTelefono)	Cliente a cui vendiamo prodotti; Codice → Controparte.Codice
Fattura	Fattura(<u>Numero</u> , Importo, Iva, ModalitàPagamento, Data, Vendita , Reso)	Fattura relativa a una vendita o un reso fatti con un cliente; Vendita → Vendita.Codice, Reso → Reso.Codice
Dipendente	Dipendente(<u>Numero</u> , Nome, Cognome, DataAssunzione, PosizioneLavorativa, NumeroDiTelefono, Negozio)	Dipendente dell'azienda; Negozio → Negozio.Codice
Negozio	Negozio(<u>Codice</u> , IndirizzoSede, Denominazione)	Negozio fisico dove lavorano dipendenti e i clienti possono fare acquisti di persona
Reso	Reso(<u>Codice</u> , Motivazione)	Restituzione o sostituzione di un prodotto; Codice → Fattura.Codice
Ordine	Ordine(<u>Codice</u> , Data, Negozio)	Ordine di un prodotto da acquistare o da vendere; Negozio → Negozio.Codice
Acquisto	Acquisto(Codice , DataDiConsegna, Fornitore)	Acquisto di prodotti da parte nostra verso il fornitore; Codice → Ordine.Codice, Fornitore → Fornitore.Codice

Continua nella pagina successiva...

Entità	Traduzione	Descrizione
Vendita	Vendita(<u>Codice</u> , Fattura , Cliente , Responsabile)	Vendita di prodotti ad un Cliente; Codice → Or- dine.Codice, Fattura → Fat- tura.Numero, Cliente → Cliente.Codice, Responsabile → Dipenden- te.Numero
Magazzino	Magazzino(<u>CodiceMagazzino</u> , In- dirizzo)	Magazzino che contiene prodotti

Relazione	Traduzione	Descrizione
Stoccaggio	Stoccaggio(<u>Magazzino</u> , <u>Prodotto</u> , Quantità)	Magazzino → Magaz- zino.Codice, Prodotto → Prodotto.Codice
Rifornimento	Rifornimento(<u>Magazzino</u> , <u>Negozio</u>)	Magazzino → Magaz- zino.Codice, Negozio → Negozio.Codice
Composizione	Composizione(<u>Ordine</u> , <u>Prodotto</u> , Quantità)	Ordine → Ordine.Codice, Pro- dotto → Prodotto.Codice
Disponibilità	Disponibilità(<u>Fornitore</u> , <u>Prodotto</u>)	Fornitore → Fornito- re.Codice, Prodotto → Prodotto.Codice

3.5 Tavole dei volumi e delle operazioni

I volumi vengono determinati assumendo di monitorare una giornata di attività all'interno di una catena di negozi.

3.5.1 Tavola dei volumi

Concetto	Tipo	Volume
Fornitore	Entità	50
Cliente	Entità	500
Dipendente	Entità	30
Fattura	Entità	1000
Magazzino	Entità	10
Negozio	Entità	20
Acquisto	Entità	2000
Vendita	Entità	5000
Scheda Video	Entità	200
Memoria di Archiviazione	Entità	300
RAM	Entità	400
Scheda Madre	Entità	150
Processore	Entità	250
Reso	Entità	100

Tabella 3.3: Tabella dei concetti con tipo e volume

3.5.2 Tavola delle operazioni

Operazione	Frequenza
Inserire un ordine d'acquisto	2000
Inserire un ordine di vendita	5000
Inserire un membro dello staff di un negozio	10
Rimuovere un membro dello staff di un negozio	8
Inserire un nuovo magazzino	1
Rimuovere un magazzino	1
Modificare la quantità di un prodotto in un magazzino	4
Inserimento di Controparti	550
Inserire una fattura	1000
Inserire un cliente	200
Inserire un fornitore	20
Inserire una RAM	200
Inserire una Scheda Video	100
Inserire un Processore	125
Inserire una Scheda Madre	75
Inserire una Memoria d'archiviazione	150
Inserire un rimborso	100
Mostrare la lista dei dipendenti che lavorano in un negozio specifico	10
Mostrare i clienti che hanno acquistato in un negozio specifico	50
Mostrare i prodotti disponibili in un magazzino di un negozio specifico	4
Mostrare gli acquisti di un cliente presso un determinato negozio	50
Mostrare tutti gli acquisti di tutti i clienti presso un determinato negozio	100
Mostrare gli ordini evasi da un dipendente specifico	6
Mostrare il costo totale d'acquisto di un tipo di prodotto	5
Mostrare l'importo totale di vendita di un tipo di prodotto	5
Mostrare tutti i dettagli di un Magazzino	10
Mostare tutti i magazzini	5
Mostrare tutti i negozi	10
Mostrare i dettagli di un Negozio	20
Mostrare fattura associata a un ordine specifico	500

Tabella 3.4: Tabella dei concetti con tipo e volume

Capitolo 4

Codifica SQL

Per semplicità, è stato scelto di adottare SQLite come DBMS. Tuttavia, questa soluzione presenta alcune limitazioni, come la gestione semplificata dei tipi. Un esempio di questa semplificazione è l'assenza di un tipo di dato `DATE` nativo, il che significa che le date vengono trattate come testi, numeri o blob. Inoltre, alcune funzionalità avanzate come le procedure memorizzate, le viste materializzate e i permessi granulari sugli utenti non sono supportate.

I tipi di dato principali supportati da SQLite sono:

- `NULL`: nessun valore
- `INTEGER`: numeri interi
- `REAL`: numeri in virgola mobile
- `TEXT`: testo (stringhe)
- `BLOB`: dati binari

4.1 Creazione delle tabelle

```
BEGIN TRANSACTION;
CREATE DATABASE IF NOT EXISTS shopdb;
USE shopdb;

CREATE TABLE IF NOT EXISTS Negozio (
    Codice INTEGER PRIMARY KEY AUTOINCREMENT,
    IndirizzoSede TEXT NOT NULL,
    Denominazione TEXT NOT NULL
);

CREATE TABLE IF NOT EXISTS Prodotto (
    Codice INTEGER PRIMARY KEY AUTOINCREMENT,
    Nome TEXT NOT NULL,
    Descrizione TEXT NOT NULL,
    Prezzo REAL NOT NULL
);

CREATE TABLE IF NOT EXISTS SchedaVideo (
    Codice INTEGER PRIMARY KEY,
    Marca TEXT NOT NULL,
    Modello TEXT NOT NULL,
    TipoDiMemoria TEXT NOT NULL,
    QuantitàDiMemoria INTEGER NOT NULL,
```

```

        NumeroDiPorte INTEGER NOT NULL,
        PotenzaNecessaria INTEGER NOT NULL,
        Compatibilità TEXT NOT NULL,
        FOREIGN KEY (Codice) REFERENCES Prodotto(Codice) ON DELETE CASCADE
    );

CREATE TABLE IF NOT EXISTS RAM (
    Codice INTEGER PRIMARY KEY,
    Tecnologia TEXT NOT NULL,
    Capacità INTEGER NOT NULL,
    Velocità INTEGER NOT NULL,
    TensioneOperativa REAL NOT NULL,
    Latenza REAL NOT NULL,
    FOREIGN KEY (Codice) REFERENCES Prodotto(Codice) ON DELETE CASCADE
);

CREATE TABLE IF NOT EXISTS Processore (
    Codice INTEGER PRIMARY KEY,
    Architettura TEXT NOT NULL,
    NumeroDiCore INTEGER NOT NULL,
    FrequenzaBase REAL NOT NULL,
    FrequenzaBoost REAL NOT NULL,
    Cache INTEGER NOT NULL,
    CompatibilitàConSocket TEXT NOT NULL,
    FOREIGN KEY (Codice) REFERENCES Prodotto(Codice) ON DELETE CASCADE
);

CREATE TABLE IF NOT EXISTS SchedaMadre (
    Codice INTEGER PRIMARY KEY,
    TipoDiSocket TEXT NOT NULL,
    Chipset TEXT NOT NULL,
    Formato TEXT NOT NULL,
    NumeroDiSlotRAM INTEGER NOT NULL,
    NumeroDiPortePCIe INTEGER NOT NULL,
    SupportoPerVelocitàRAM INTEGER NOT NULL,
    Connettività TEXT NOT NULL,
    FOREIGN KEY (Codice) REFERENCES Prodotto(Codice) ON DELETE CASCADE
);

CREATE TABLE IF NOT EXISTS MemoriaArchiviazione (
    Codice INTEGER PRIMARY KEY,
    Tipo TEXT NOT NULL,
    Capacità INTEGER NOT NULL,
    VelocitàRW REAL NOT NULL,
    Interfaccia TEXT NOT NULL,
    FOREIGN KEY (Codice) REFERENCES Prodotto(Codice) ON DELETE CASCADE
);

CREATE TABLE IF NOT EXISTS Fattura (
    Numero INTEGER PRIMARY KEY AUTOINCREMENT,
    Importo REAL NOT NULL,
    Iva REAL NOT NULL,
    DataEmissione TEXT NOT NULL,
    Vendita INTEGER NOT NULL,
    Reso INTEGER DEFAULT NULL,

```

```

        FOREIGN KEY (Vendita) REFERENCES Vendita(Codice) ON DELETE CASCADE,
        FOREIGN KEY (Reso) REFERENCES Reso(Codice) ON DELETE SET NULL
    );

CREATE TABLE IF NOT EXISTS Reso (
    Codice INTEGER PRIMARY KEY AUTOINCREMENT,
    Motivazione TEXT NOT NULL,
    FOREIGN KEY(Codice) REFERENCES Fattura(Numero) ON DELETE CASCADE
);

CREATE TABLE IF NOT EXISTS Dipendente (
    Numero INTEGER PRIMARY KEY AUTOINCREMENT,
    Cognome TEXT NOT NULL,
    DataAssunzione TEXT NOT NULL,
    Nome TEXT NOT NULL,
    PosizioneLavorativa TEXT NOT NULL,
    NumeroDiTelefono TEXT NOT NULL,
    Negozio INTEGER,
    FOREIGN KEY (Negozio) REFERENCES Negozio(Codice) ON DELETE SET NULL
);

CREATE TABLE IF NOT EXISTS Magazzino (
    Codice INTEGER PRIMARY KEY AUTOINCREMENT,
    Indirizzo TEXT NOT NULL
);

CREATE TABLE IF NOT EXISTS Controparte (
    Codice INTEGER PRIMARY KEY AUTOINCREMENT,
    Nome TEXT NOT NULL,
    Indirizzo TEXT NOT NULL,
    NumeroDiTelefono TEXT NOT NULL
);

CREATE TABLE IF NOT EXISTS Fornitore (
    Codice INTEGER PRIMARY KEY,
    FOREIGN KEY (Codice) REFERENCES Controparte(Codice) ON DELETE CASCADE
);

CREATE TABLE IF NOT EXISTS Cliente (
    Codice INTEGER PRIMARY KEY,
    IndirizzoMail TEXT NOT NULL,
    Cognome TEXT NOT NULL,
    FOREIGN KEY (Codice) REFERENCES Controparte(Codice) ON DELETE CASCADE
);

CREATE TABLE IF NOT EXISTS Ordine (
    Codice INTEGER PRIMARY KEY AUTOINCREMENT,
    Data TEXT NOT NULL,
    Negozio INTEGER,
    FOREIGN KEY (Negozio) REFERENCES Negozio(Codice) ON DELETE SET NULL
);

CREATE TABLE IF NOT EXISTS Acquisto (
    Codice INTEGER PRIMARY KEY,
    DataDiConsegna TEXT NOT NULL,

```

```

    Fornitore INTEGER,
    FOREIGN KEY (Codice) REFERENCES Ordine(Codice) ON DELETE CASCADE,
    FOREIGN KEY (Fornitore) REFERENCES Fornitore(Codice) ON DELETE SET NULL
);

CREATE TABLE IF NOT EXISTS Vendita (
    Codice INTEGER PRIMARY KEY,
    Responsabile INTEGER,
    Cliente INTEGER,
    Fattura INTEGER DEFAULT NULL,
    FOREIGN KEY (Codice) REFERENCES Ordine(Codice) ON DELETE CASCADE,
    FOREIGN KEY (Fattura) REFERENCES Fattura(Numero) ON DELETE SET NULL,
    FOREIGN KEY (Cliente) REFERENCES Cliente(Codice) ON DELETE SET NULL,
    FOREIGN KEY (Responsabile) REFERENCES Dipendente(Codice) ON DELETE SET NULL
);

-- Tabelle per le relazioni
CREATE TABLE IF NOT EXISTS Stoccaggio (
    Magazzino INTEGER,
    Prodotto INTEGER,
    Quantità INTEGER NOT NULL,
    PRIMARY KEY (Magazzino, Prodotto),
    FOREIGN KEY (Magazzino) REFERENCES Magazzino(Codice) ON DELETE CASCADE,
    FOREIGN KEY (Prodotto) REFERENCES Prodotto(Codice) ON DELETE CASCADE
);

CREATE TABLE IF NOT EXISTS Rifornamento (
    Magazzino INTEGER,
    Negozio INTEGER,
    PRIMARY KEY (Magazzino, Negozio),
    FOREIGN KEY (Magazzino) REFERENCES Magazzino(Codice) ON DELETE CASCADE,
    FOREIGN KEY (Negozio) REFERENCES Negozio(Codice) ON DELETE CASCADE
);

CREATE TABLE IF NOT EXISTS Composizione (
    Ordine INTEGER,
    Prodotto INTEGER,
    Quantità INTEGER NOT NULL,
    PRIMARY KEY (Ordine, Prodotto),
    FOREIGN KEY (Ordine) REFERENCES Ordine(Codice) ON DELETE CASCADE,
    FOREIGN KEY (Prodotto) REFERENCES Prodotto(Codice) ON DELETE CASCADE
);

CREATE TABLE IF NOT EXISTS Disponibilità (
    Fornitore INTEGER,
    Prodotto INTEGER,
    PRIMARY KEY (Fornitore, Prodotto),
    FOREIGN KEY (Fornitore) REFERENCES Fornitore(Codice) ON DELETE CASCADE,
    FOREIGN KEY (Prodotto) REFERENCES Prodotto(Codice) ON DELETE CASCADE
);

COMMIT;

```

4.2 Inserimenti in tabelle

```
BEGIN TRANSACTION
-- Inserimento in Negozio
INSERT INTO Negozio (Codice, IndirizzoSede, Denominazione) VALUES
    ('Via Roma 123, Milano', 'Negozio Centrale'), -- Codice 1
    ('Corso Italia 45, Torino', 'Emporio Piemonte'), -- Codice 2
    ('Piazza Duomo 1, Firenze', 'Fiorentino Tech'), -- Codice 3
    ('Via Napoli 12, Roma', 'Capitol Store'), -- Codice 4
    ('Via Verdi 9, Bologna', 'Emilia Tech'), -- Codice 5
    ('Corso Vittorio 56, Napoli', 'Partenope Shop'), -- Codice 6
    ('Via Dante 89, Palermo', 'Sicilia Elettronica'), -- Codice 7
    ('Piazza San Marco 10, Venezia', 'Laguna Store'), -- Codice 8
    ('Via Garibaldi 33, Genova', 'Super Genova Store'), -- Codice 9
    ('Via Leopardi 18, Bari', 'Adriatico Shop'); -- Codice 10

-- Inserimento di Controparti
INSERT INTO Controparte (Nome, Indirizzo, NumeroDiTelefono)
VALUES
    ('Matteo', 'Via Padania 10, Milano', '3451234567'), -- Codice 1
    ('Giorgia', 'Via Tricolore 15, Roma', '3462345678'), -- Codice 2
    ('Enrico', 'Piazza Unità 8, Firenze', '3473456789'), -- Codice 3
    ('Giuseppe', 'Via Costituzione 22, Napoli', '3484567890'), -- Codice 4
    ('Silvio', 'Corso Italia 33, Milano', '3495678901'), -- Codice 5
    ('Elly', 'Via Progressista 18, Bologna', '3506789012'), -- Codice 6
    ('Luigi', 'Via Movimento 5, Napoli', '3517890123'), -- Codice 7
    ('Pier Luigi', 'Via Sinistra 44, Piacenza', '3528901234'), -- Codice 8
    ('Renato', 'Viale Riforme 9, Venezia', '3539012345'), -- Codice 9
    ('Emma', 'Via Europa 7, Torino', '3540123456'), -- Codice 10
    ('Antonio Tajani', 'Via Diplomazia 12, Roma', '3551234567'), -- Codice 11
    ('Maria Elena Boschi', 'Via Riforme 23, Firenze', '3562345678'), -- Codice
    ↪ 12
    ('Nicola Zingaretti', 'Via Democratica 34, Roma', '3573456789'), -- Codice
    ↪ 13
    ('Roberto Speranza', 'Via Salute 45, Potenza', '3584567890'), -- Codice 14
    ('Carlo Calenda', 'Corso Economia 56, Milano', '3595678901'); -- Codice 15

-- Inserimento in Cliente (derivato da Controparte)
INSERT INTO Cliente (Codice, IndirizzoMail, Cognome)
VALUES
    (1, 'matteo.salvini@example.com', 'Salvini'),
    (2, 'giorgia.meloni@example.com', 'Meloni'),
    (3, 'enrico.letta@example.com', 'Letta'),
    (4, 'giuseppe.conte@example.com', 'Conte'),
    (5, 'silvio.berlusconi@example.com', 'Berlusconi'),
    (6, 'elly.schlein@example.com', 'Schlein'),
    (7, 'luigi.dimaio@example.com', 'Di Maio'),
    (8, 'pierluigi.bersani@example.com', 'Bersani'),
    (9, 'renato.brunetta@example.com', 'Brunetta'),
    (10, 'emma.bonino@example.com', 'Bonino');

-- Inserimento in Fornitore (derivata da Controparte)
INSERT INTO Fornitore (Codice)
VALUES
    (11), -- Antonio Tajani
```

```

(12), -- Maria Elena Boschi
(13), -- Nicola Zingaretti
(14), -- Roberto Speranza
(15); -- Carlo Calenda

-- Inserimento in Prodotto (padre di SchedaVideo, RAM, Processore,
→ MemoriaArchiviazione, SchedaMadre)
INSERT INTO Prodotto (Nome, Descrizione, Prezzo)
VALUES
('RTX 3080', 'Scheda video ad alte prestazioni', 799.99), -- Codice 1
('Intel i9-12900K', 'Processore di ultima generazione', 599.99), -- Codice 2
('ASRock X870E Taichi', 'Scheda madre ad alte prestazioni', 557.03) -- Codice
→ 3
('Corsair Vengeance', 'RAM ad alte prestazioni', 149.99), -- Codice 4
('Samsung 970 EVO', 'SSD NVMe veloce', 129.99); -- Codice 5

-- Inserimento in SchedaVideo (derivata da Prodotto)
INSERT INTO SchedaVideo (Codice, Marca, Modello, TipoDiMemoria,
→ QuantitàDiMemoria, NumeroDiPorte, PotenzaNecessaria, Compatibilità)
VALUES
(1, 'NVIDIA', 'RTX 3080', 'GDDR6X', 10, 4, 750, 'PCIe 4.0');

-- Inserimento in RAM (derivata da Prodotto)
INSERT INTO RAM (Codice, Tecnologia, Capacità, Velocità, TensioneOperativa,
→ Latenza)
VALUES
(4, 'DDR4', 16, 3200, 1.35, 16);

-- Inserimento in Processore (derivata da Prodotto)
INSERT INTO Processore (Codice, Architettura, NumeroDiCore, FrequenzaBase,
→ FrequenzaBoost, Cache, CompatibilitàConSocket)
VALUES
(2, 'x86-64', 16, 3.2, 5.2, 30, 'LGA1700');

-- Inserimento in SchedaMadre (derivata da Prodotto)
INSERT INTO SchedaMadre (Codice, TipoDiSocket, Chipset, Formato, NumeroDiSlotRAM,
→ NumeroDiPortePCIe, SupportoPerVelocitàRAM, Connettività)
VALUES
(3, 'AM5', 'AMD X870E', 'ATX', 4, 2, 256, 'WIFI 7');

-- Inserimento in MemoriaArchiviazione (derivata da Prodotto)
INSERT INTO MemoriaArchiviazione (Codice, Tipo, Capacità, VelocitàRW,
→ Interfaccia)
VALUES
(5, 'SSD NVMe', 1024, 3500, 'PCIe 3.0');

-- Inserimento in Dipendente
INSERT INTO Dipendente (Cognome, DataAssunzione, Nome, PosizioneLavorativa,
→ NumeroDiTelefono, Negozio) VALUES
-- Dipendenti per Negozio Centrale
('Smith', '2018-05-14', 'John', 'Manager', '3331112222', 1), -- Codice 1
('Doe', '2020-03-11', 'Jane', 'Cassiere', '3331113333', 1), -- Codice 2

```



```

('Wick', '2021-07-23', 'Jonathan', 'Magazziniere', '3331114444', 1), --
→ Codice 3

-- Dipendenti per Emporio Piemonte
('Wayne', '2019-01-10', 'Bruce', 'Manager', '3332225555', 2), -- Codice 4
('Gordon', '2020-11-02', 'James', 'Cassiere', '3332226666', 2), -- Codice 5
('Dent', '2022-08-15', 'Harvey', 'Magazziniere', '3332227777', 2), -- Codice
→ 6

-- Dipendenti per Fiorentino Tech
('Stark', '2017-12-05', 'Tony', 'Manager', '3333338888', 3), -- Codice 7
('Potts', '2021-06-19', 'Pepper', 'Cassiere', '3333339999', 3), -- Codice 8
('Rogers', '2020-04-03', 'Steve', 'Magazziniere', '3333330000', 3), -- Codice
→ 9

-- Dipendenti per Capitol Store
('Skywalker', '2019-09-12', 'Luke', 'Manager', '3334441111', 4), -- Codice
→ 10
('Organa', '2021-01-22', 'Leia', 'Cassiere', '3334442222', 4), -- Codice 11
('Solo', '2022-10-30', 'Han', 'Magazziniere', '3334443333', 4), -- Codice 12

-- Dipendenti per Emilia Tech
('Anderson', '2020-05-18', 'Neo', 'Manager', '3335554444', 5), -- Codice 13
('Trinity', '2021-11-10', 'Trinity', 'Cassiere', '3335555555', 5), -- Codice
→ 14
('Morpheus', '2018-08-07', 'Morpheus', 'Magazziniere', '3335556666', 5), --
→ Codice 15

-- Dipendenti per Partenope Shop
('Potter', '2019-07-16', 'Harry', 'Manager', '3336667777', 6), -- Codice 16
('Granger', '2020-02-25', 'Hermione', 'Cassiere', '3336668888', 6), -- Codice
→ 17
('Weasley', '2021-09-14', 'Ron', 'Magazziniere', '3336669999', 6), -- Codice
→ 18

-- Dipendenti per Sicilia Elettronica
('Targaryen', '2018-03-09', 'Daenerys', 'Manager', '3337770000', 7), --
→ Codice 19
('Snow', '2020-07-21', 'Jon', 'Cassiere', '3337771111', 7), -- Codice 20
('Lannister', '2021-12-03', 'Tyrion', 'Magazziniere', '3337772222', 7), --
→ Codice 21

-- Dipendenti per Laguna Store
('Parker', '2019-04-17', 'Peter', 'Manager', '3338883333', 8), -- Codice 22
('Watson', '2020-10-10', 'Mary Jane', 'Cassiere', '3338884444', 8), -- Codice
→ 23
('Octavius', '2021-05-29', 'Otto', 'Magazziniere', '3338885555', 8), --
→ Codice 24

-- Dipendenti per Super Genova Store
('Baggins', '2018-06-30', 'Frodo', 'Manager', '3339996666', 9), -- Codice 25
('Gamgee', '2019-02-11', 'Samwise', 'Cassiere', '3339997777', 9), -- Codice
→ 26
('Legolas', '2021-08-25', 'Legolas', 'Magazziniere', '3339998888', 9), --
→ Codice 27

```

```

-- Dipendenti per Adriatico Shop
('Kirk', '2020-01-15', 'James T.', 'Manager', '3330009999', 10), -- Codice
→ 28
('Spock', '2021-07-12', 'Spock', 'Cassiere', '3330001111', 10), -- Codice 29
('Scott', '2022-03-20', 'Montgomery', 'Magazziniere', '3330002222', 10); --
→ Codice 30

-- Inserimento in Ordine (padre di Acquisto e Vendita)
INSERT INTO Ordine (Data, Negozio)
VALUES
    ('2025-01-10', 3), -- Codice 1
    ('2025-01-11', 8), -- Codice 2
    ('2025-01-11', 8), -- Codice 3
    ('2025-01-12', 1); -- Codice 4

-- Inserimento in Acquisto (derivata da Ordine)
INSERT INTO Acquisto (Codice, DataDiConsegna, Fornitore)
VALUES
    (1, '2025-01-15', 12),
    (2, '2025-01-16', 15);

-- Inserimento in Vendita (derivata da Ordine)
INSERT INTO Vendita (Codice, Responsabile, Cliente)
VALUES
    (3, 26, 5),
    (4, 23, 1);

-- Inserimento in Fattura
INSERT INTO Fattura (Importo, Iva, DataEmissione, Vendita)
VALUES
    (979.99, 22, '2025-01-13', 3); -- Codice 1
UPDATE Vendita V
SET V.Fattura = 1
WHERE V.Codice = 3;

INSERT INTO Fattura (Importo, Iva, DataEmissione, Vendita)
VALUES
    (699.99, 22, 'PayPal', '2025-01-12', 4); -- Codice 2
UPDATE Vendita V
SET V.Fattura = 2
WHERE V.Codice = 4;

-- Inserimento in Reso
INSERT INTO Reso (Codice, Motivazione, Stato, Modalità)
VALUES
    (1, 'Prodotto danneggiato', 'In lavorazione', 'Sostituzione'),
    (2, 'Non conforme alla descrizione', 'Completato', 'Rimborso');

-- Inserimento in Magazzino
INSERT INTO Magazzino (Indirizzo)
VALUES
    ('Via Roma, 10, Milano'), -- Codice 1

```

```

('Corso Vittorio Emanuele II, 45, Torino'), -- Codice 2
('Piazza della Repubblica, 22, Firenze'), -- Codice 3
('Via Garibaldi, 5, Bologna'), -- Codice 4
('Viale Mazzini, 12, Napoli'), -- Codice 5
('Via Dante Alighieri, 34, Palermo'), -- Codice 6
('Corso Umberto I, 18, Bari'), -- Codice 7
('Via Verdi, 28, Genova'), -- Codice 8
('Piazza San Marco, 14, Venezia'), -- Codice 9
('Viale Manzoni, 50, Catania'); -- Codice 10

-- Inserimenti nelle tabelle delle relazioni
INSERT INTO Stoccaggio (Magazzino, Prodotto, Quantità)
VALUES
    (1, 1, 50), -- 50 RTX 3080 nel magazzino 1
    (1, 2, 30), -- 30 Intel i9-12900K nel magazzino 1
    (2, 3, 100), -- 100 RAM Corsair nel magazzino 2
    (2, 4, 75); -- 75 SSD Samsung nel magazzino 2

INSERT INTO Rifornamento (Magazzino, Negozio)
VALUES
    (1, 1),
    (2, 2);

INSERT INTO Composizione (Ordine, Prodotto, Quantità)
VALUES
    (1, 1, 1), -- Un RTX 3080 nell'ordine 1
    (2, 2, 1); -- Un Intel i9-12900K nell'ordine 2
    (3, 1, 1), -- Un RTX 3080 nell'ordine 3
    (4, 2, 1); -- Un Intel i9-12900K nell'ordine 4

INSERT INTO Disponibilità (Fornitore, Prodotto)
VALUES
    (1, 1),
    (1, 2),
    (2, 3),
    (2, 4);

COMMIT;

```

4.3 Mostrare le tabelle

```

-- Mostrare i Dipendenti che lavorano in un Negozio
SELECT *
FROM Dipendente D
WHERE D.Negozio = 5;

-- Mostrare tutti i clienti che hanno utilizzato un certo Negozio
SELECT DISTINCT C.Codice, CP.Nome, C.Cognome, C.IndirizzoMail,
    ↪ CP.NumeroDiTelefono
FROM Cliente C
JOIN Controparte CP ON C.Codice = CP.Codice
JOIN Vendita V ON C.Codice = V.Cliente
JOIN Ordine O ON V.Codice = O.Codice
WHERE O.Negozio = 2;

```

```

-- Mostrare tutti i dettagli di un Magazzino
SELECT p.Codice, p.Nome, p.Descrizione, s.Quantita
FROM Magazzino m
JOIN Stoccaggio s ON m.Codice = s.Magazzino
JOIN Prodotto p ON s.Prodotto = p.Codice
WHERE m.Codice = 4;

-- Mostare tutti i magazzini
SELECT *
FROM Magazzino;

-- Mostrare i Prodotti disponibili in un Negozio
SELECT DISTINCT P.Codice, P.Nome, P.Prezzo, M.Codice, M.Indirizzo, S.Quantita
FROM Prodotto P
JOIN Stoccaggio S ON P.Codice = S.Prodotto
JOIN Magazzino M ON S.Magazzino = M.Codice
JOIN Rifornimento R ON M.Codice = R.Magazzino
WHERE R.Negozio = 20;

-- Mostrare tutti i negozi
SELECT Codice, IndirizzoSede, Denominazione
FROM Negozio;

-- Mostrare i dettagli di un Negozio
SELECT *
FROM Negozio N
WHERE N.Codice = 4;

-- Mostrare gli acquisti di un cliente presso un determinato negozio
SELECT V.Codice, O.Data, O.Negozio
FROM Vendita V
JOIN Ordine O ON V.Codice = O.Codice
WHERE V.Cliente = 50 AND O.Negozio = 4;

-- Mostrare tutti gli acquisti di tutti i clienti presso un determinato negozio
SELECT C.Nome, C.Cognome, V.Codice, O.Data, O.Negozio
FROM Vendita V
JOIN Ordine O ON V.Codice = O.Codice
JOIN Cliente C ON V.Cliente = C.Codice
WHERE O.Negozio = 100;

-- Mostrare gli ordini evasi da un dipendente specifico
SELECT V.Codice, O.Data
FROM Vendita V
JOIN Ordine O ON V.Codice = O.Codice
WHERE V.Responsabile = 6;

-- Mostrare il costo totale d'acquisto di un tipo di prodotto
SELECT SUM(P.Prezzo * C.Quantità) AS CostoTotale
FROM Acquisto A
JOIN Composizione C ON A.Codice = C.Ordine
JOIN Prodotto P ON P.Codice = C.Prodotto
WHERE P.Nome = 'RTX 3080';

```

```

-- Mostrare l'importo totale di vendita di un tipo di prodotto
SELECT SUM(P.Prezzo * C.Quantità) AS VenditeTotali
FROM Prodotto P
JOIN Composizione C ON P.Codice = C.Prodotto
JOIN Vendita V ON C.Vendita = V.Codice
WHERE P.Nome = 'RTX 3080';

-- Mostrare fattura associata a un ordine specifico
SELECT F.Numero, F.Importo, F.Iva, F.DataEmissione, CP.Nome, C.Cognome,
       ↪ CP.Indirizzo
FROM Fattura F
JOIN Vendita V ON F.Vendita = V.Codice
JOIN Cliente C ON V.Cliente = C.Codice
JOIN Controparte CP ON C.Codice = CP.Codice
WHERE v.Codice = 10;

-- Mostrare stato di un ordine (Verifica se un ordine ha una fattura e se è stato
↪ effettuato un res)
SELECT
    CASE WHEN F.Numero IS NOT NULL THEN 1 ELSE 0 END AS Has_Invoice,
    CASE WHEN F.Reso IS NOT NULL THEN 1 ELSE 0 END AS Has_Return
FROM Vendita V
LEFT JOIN Fattura F ON V.Fattura = F.Numero
WHERE v.Codice = 6;

-- Mostrare i Fornitori
SELECT F.Codice, C.Nome, C.Indirizzo, C.NumeroDiTelefono
FROM Fornitore F
JOIN Controparte C ON F.Codice = C.Codice;

-- Mostrare i Prodotti di un Fornitore specifico
SELECT P.Codice, P.Nome, P.Descrizione, P.Prezzo
FROM Prodotto P
JOIN Disponibilità D ON P.Codice = D.Prodotto
WHERE D.Fornitore = 31;

```

4.4 Rimuovere dalle tabelle

```

-- Rimozione di un dipendente specifico
DELETE FROM Dipendente D
WHERE D.Numero = 1;

-- Rimozione di un Magazzino specifico
DELETE FROM Magazzino M
WHERE M.Codice = 1;

-- Questa query rimuove ciascuna istanza di Stoccaggio che ha Magazzino = 1, ma
↪ le Business Rules impongono che, affinché un magazzino sia rimovibile, non
↪ esistano istanze di stoccaggio che lo contengono con Quantità > 0, quindi non
↪ causa problemi.

-- Rimozione di un'istanza di Stoccaggio
DELETE FROM Stoccaggio S
WHERE S.Magazzino = 1 AND S.Prodotto = 1;

```

```
-- Rimozione di tutte le relazioni di disponibilità per un fornitore  
DELETE FROM Disponibilità D  
WHERE D.Fornitore = 1;
```

4.5 Aggiornare le tabelle

```
-- Modificare la quantità di un prodotto in un magazzino  
-- Supponiamo che il codice del prodotto sia 29394 e quello del magazzino sia 4  
UPDATE Stoccaggio S  
SET S.Quantità = S.Quantità + 10  
WHERE S.Prodotto = 29394 AND S.Magazzino = 4;
```

Capitolo 5

Testing

5.1 Link al sito che permette di testare il database

[Sito](#)