

Progettazione

Introduzione

Si vuole realizzare una base di dati che serva da supporto ad una determinata banca per la gestione delle informazioni relative all'attività di consulenza. L'obiettivo del database è di controllare tutti i movimenti finanziari effettuati dai clienti e la produttività dei propri consulenti.

Requisiti della base di dati

Del consulente si è interessati a conoscere il nominativo, il portafoglio che gestisce e il numero di telefono. Inoltre, ogni consulente è identificato da un codice univoco che la banca gli associa non appena viene assunto.

Il consulente lavora presso un ufficio. Dell'ufficio si vuole sapere la provincia, città, CAP e via.

Ciascun consulente segue un determinato numero di clienti della banca, ai quali fa firmare un contratto di apertura all'inizio della collaborazione e un contratto di chiusura nel caso in cui i clienti decidano di affidarsi ad altri consulenti.

Dei clienti si vuol tener nota dei nominativi, portafoglio che detengono, numeri di telefono, provincia, città, CAP e via dove risiedono. Anche ai clienti viene assegnato un codice identificativo.

Inoltre, un cliente può essere in possesso di uno o più conti correnti.

I conti correnti possono essere sia cointestati che non, e di ognuno si vuole conoscere il saldo contabile presente, la data di apertura, la tipologia e il numero identificativo.

I consulenti si occupano di organizzare appuntamenti con i propri clienti. Degli appuntamenti si vuole tenere traccia della data nei quali sono avvenuti, il contatto (se via telefono o colloquio fisico, per esempio) e dell'esito (successo o insuccesso).

Ciascun consulente, infine, ha il compito di gestire le transazioni che vengono eseguite dai propri clienti. Ad ogni transazione è associato un codice e di queste si è interessati a sapere l'importo investito e la data in cui sono avvenute. Degli appuntamenti e delle transazioni si è interessati a conoscere lo storico dell'ultimo anno.

Le transazioni coinvolgono dei prodotti. I prodotti si suddividono in prodotti finanziari, polizze o finanziamenti, hanno ognuno un nome specifico e possono avere una durata.

Glossario dei termini

Termine	Definizione	Sinonimi	Collegamento
Cliente	Colui che acquista servizi e prodotti		Consulente, Conto corrente, Appuntamento, Transazione
Consulente	Colui che offre assistenza e consulenza riguardo le decisioni di investimento	Promotore finanziario	Cliente, Appuntamento, Transazione,
Portafoglio cliente	Insieme degli investimenti di un cliente		Cliente
Portafoglio consulente	Insieme dei portafogli dei propri clienti		Consulente
Transazione	Operazione finanziaria	Operazione	Consulente, Cliente, Prodotto
Conto corrente	Strumento finanziario per la gestione del denaro, regola le entrate e le uscite finanziarie. Può essere cointestato.		Cliente
Saldo contabile	Somma presente sul conto corrente		Conto corrente
Prodotto	Rappresenta il prodotto coinvolto nella transazione		Transazione
Prodotto finanziario	Strumento attraverso il quale si impiega il denaro in varie forme di investimento		Transazione
Polizza	Rappresentano i contratti di assicurazioni		Transazione
Finanziamento	Prestiti e mutui		Transazione
Durata	Periodo di investimento di un prodotto		Prodotto
Appuntamento	Incontro, fisico e non, tra un cliente ed un consulente		Cliente, Consulente
Contatto	Tipo di interazione tra cliente e consulente		Appuntamento

Suddivisione del testo in frasi omogenee

- Frase sui consulenti:

“Del consulente si è interessati a conoscere il nominativo, il portafoglio che gestisce e il numero di telefono. Inoltre, ogni consulente è identificato da un codice univoco che la banca gli associa non appena viene assunto.

Il consulente lavora presso un ufficio.”

“Ciascun consulente segue un determinato numero di clienti della banca, ai quali fa firmare un contratto di apertura all’inizio della collaborazione e un contratto di chiusura nel caso in cui i clienti decidano di affidarsi ad altri consulenti.”

“I consulenti si occupano di organizzare appuntamenti con i propri clienti.”

“Ciascun consulente, infine, ha il compito di gestire le transazioni che vengono eseguite dai propri clienti.”

- Frase sui clienti:

“Dei clienti si vuol tener nota dei nominativi, portafogli che detengono, numeri di telefono, provincia, città, CAP e via dove risiedono. Anche ai clienti viene assegnato un codice identificativo.”

“Inoltre, un cliente può essere in possesso di uno o più conti corrente.”

- Frase sugli uffici:

“Dell’ufficio si vuole sapere la provincia, città, CAP e via.”

- Frase sugli appuntamenti:

“Degli appuntamenti si vuole tenere traccia della data nei quali sono avvenuti, il contatto (se via telefono o colloquio fisico, per esempio) e dell’esito (successo o insuccesso).”

“Degli appuntamenti e delle transazioni si è interessati a conoscere lo storico dell’ultimo anno.”

- Frase sulle transazioni:

“ Ad ogni transazione è associato un codice e di queste si è interessati a sapere l’importo investito e la data in cui sono avvenute.

Le transazioni coinvolgono dei prodotti.”

“Degli appuntamenti e delle transazioni si è interessati a conoscere lo storico dell’ultimo anno.”

- Frasi sui prodotti:

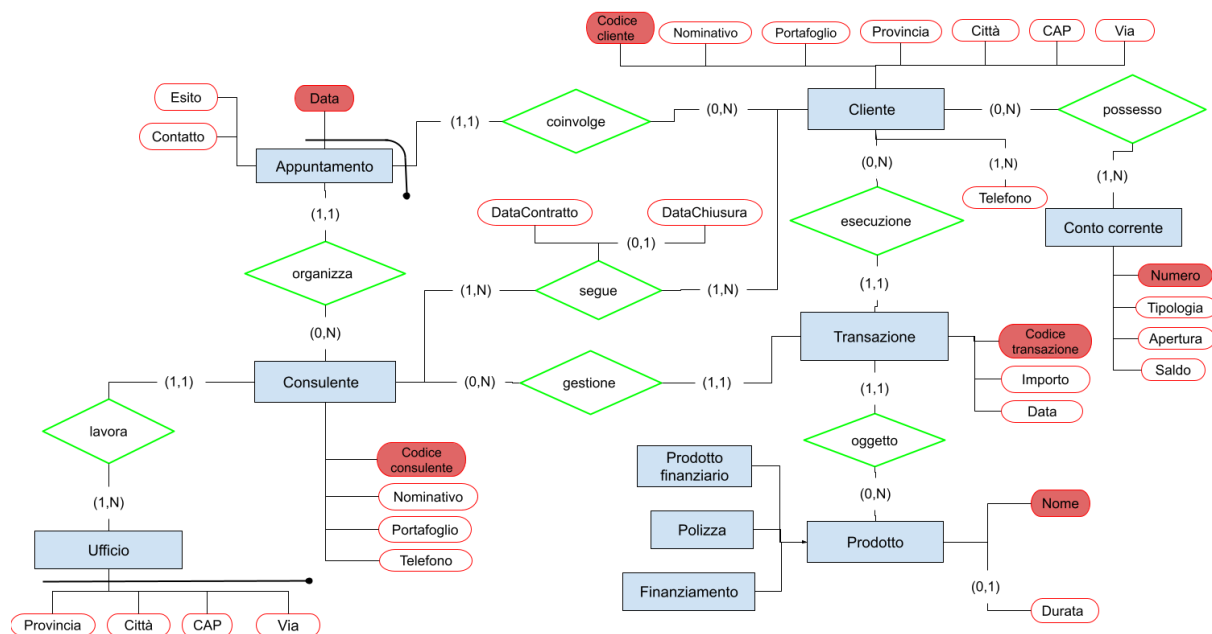
“I prodotti si suddividono in prodotti finanziari, polizze o finanziamenti, hanno ognuno un nome specifico e possono avere una durata.”

- Frasi sui conti correnti:

“Inoltre, un cliente può essere in possesso di uno o più conti correnti.

I conti correnti possono essere sia cointestati che non, e di ognuno si vuole conoscere il saldo contabile presente, la data di apertura, la tipologia e il numero identificativo.”

Diagramma Entity-Relationship



Dizionario dei dati (Entità)

Entità	Descrizione	Attributi	Identificatore
Consulente	Consulente finanziario della banca	Codice Consulente, Nominativo, Portafoglio, Telefono	Codice Consulente
Cliente	Cliente della banca	Codice Cliente, Nominativo, Portafoglio, Telefono, Provincia, Città, CAP, Via	Codice Cliente

Ufficio	Ufficio del consulente	Provincia, Città, CAP, Via	Provincia, Città, CAP, Via
Appuntamento	Appuntamento tra un cliente e un consulente	Data, Esito, Contatto	Data, Coinvolge (R)
Transazione	Transazione eseguita da un cliente e gestita dal suo consulente	Codice Transazione, Data, Importo	Codice Transazione
Prodotto	Il prodotto è oggetto di una transazione	Nome, Durata	Nome
Prodotto finanziario	I prodotto finanziario è oggetto di una transazione	Nome, Durata	Nome
Polizza	Le polizze sono oggetto di una transazione	Nome, Durata	Nome
Finanziamento	I finanziamenti sono oggetto di una transazione	Nome, Durata	Nome
Conto corrente	Conto corrente di un cliente	Numero, Saldo, Tipologia, Apertura	Numero

Dizionario dei dati (Relationships)

Entità	Descrizione	Componenti	Attributi
Gestione	Gestione di una transazione da parte di un consulente	Consulente, Transazione	
Segue	Ogni consulente segue un determinato numero di clienti	Consulente, Cliente	Data Contratto, Data Chiusura
Organizza	Un appuntamento è organizzato da un consulente	Consulente, Appuntamento	
Coinvolge	Un appuntamento coinvolge un determinato cliente	Appuntamento, Cliente	
Lavora	Ciascun consulente lavora presso un ufficio	Consulente, Ufficio	

Oggetto	I prodotti sono oggetto delle transazioni eseguite da un cliente	Transazione, Prodotto	
Esecuzione	Esecuzione di una transazione da parte di un cliente	Cliente, Transazione	
Possesso	Un cliente può possedere uno o più conti corrente	Cliente, Conto Corrente	

Vincoli non esprimibili graficamente

Il portafoglio di un cliente non può essere maggiore del portafoglio del suo consulente attuale.

La data di chiusura di un contratto non può essere precedente alla data di apertura.

Considerazioni generali

Si osserva che l'attributo Telefono relativo all'entità Consulente potrebbe essere un attributo multivalore. Si decide però di non considerarlo tale poiché si è interessati a conoscere solamente i singoli numeri di telefono di lavoro di ciascun consulente e non anche quelli secondari e personali.

Al contrario, per l'entità Clienti si è interessati a tenere traccia di tutti i recapiti telefonici che vengono forniti al momento della firma del contratto col proprio consulente.

Si osserva che la relazione "Segue" presenta una cardinalità (1,N) all'estremità dell'entità Cliente. Ciò è stato fatto al fine di tenere traccia anche della consulenza passata alla quale il cliente si è affidato all'interno della banca.

La generalizzazione con entità padre l'entità Prodotto è di tipo totale ed esclusiva, siccome non esistono transazioni che coinvolgono più prodotti o prodotti che non siano prodotti finanziari, polizze o finanziamenti.

Infine, si vuole precisare che l'attributo CAP, relativo alle entità Ufficio e Cliente, è riferito al codice postale generico delle città e province, e non è correlato alla via di residenza.

Tavola dei volumi

Suppongo che la banca abbia all'incirca 2000 consulenti finanziari e 150 uffici.

Si stima una media di 200 clienti per ciascun consulente della banca.

Si stima una media di 20 appuntamenti settimanali per consulente e 48 settimane lavorative complete per ciascuno.

Si considerano circa 20000 tipi di prodotti finanziari, 15 tipi di polizze e 10 tipi di finanziamento.

Si stima una media di 5 transazioni all'anno per ciascun cliente.

Si ipotizza che ogni cliente abbia in media un solo conto corrente.

Si ipotizza che ciascun cliente in media abbia avuto un solo consulente all'interno della banca per tutto il periodo in cui è stato assistito.

Concetto	Tipo	Volume
Consulente	E	2000
Cliente	E	400000 (=2000x200)
Transazione	E	2000000 (=5x400000)
Prodotto Finanziario	E	20000
Polizza	E	15
Finanziamento	E	10
Prodotti	E	20025 (=20000+15+10)
Conto Corrente	E	400000
Ufficio	E	150
Appuntamento	E	1920000 (=20x48x2000)
Segue	R	400000
Gestione	R	2000000
Coinvolge	R	1920000
Organizza	R	1920000
Lavora	R	2000
Esecuzione	R	2000000
Oggetto	R	2000000
Possesso	R	400000

Operazioni di interesse

Operazione	Tipo	Frequenza
Ricerca dei consulenti che hanno raggiunto una determinata dimensione di portafoglio	Interattiva	2/mese

Ordinamento decrescente dei consulenti per ammontare di valore nelle transazioni gestite nell'ultimo anno	Interattiva	2/mese
Ricerca delle transazioni che coinvolgono finanziamenti avvenute nell'ultima settimana, ordinate per dimensione dell'importo finanziato	Interattiva	1/settimana
Ricerca del saldo contabile presente su una determinata tipologia di conto corrente di un determinato cliente	Interattiva	20/giorno
Aggiornamento informazioni clienti	Batch	1/giorno

Analisi delle ridondanze

Si osserva che è presente una ridondanza dovuta a ciclo che coinvolge le entità Transazione, Cliente e Consulente e le relazioni "Esecuzione", "Segue" e "Gestione".

Si decide di analizzare se risulta essere conveniente o meno mantenere la relazione "Gestione", studiando il numero di accessi in presenza e in assenza di tale ridondanza per la seguente operazione:

"Ordinamento decrescente dei consulenti per ammontare di valore nelle transazioni gestite nell'ultimo anno"

- Assenza di ridondanza:

Concetto	Costrutto	Accessi	Tipo
Transazione	E	N	L
Cliente	E	M	L
Segue	R	M	L
Consulente	E	P	L

N = numero di transazioni avvenute nell'ultimo anno = 2000000

M = numero clienti che hanno eseguito transazioni nell'ultimo anno = 400000

P = numero consulenti che hanno gestito le transazioni nell'ultimo anno = 2000

- Presenza di ridondanza:

Concetto	Costrutto	Accessi	Tipo
Transazione	E	N	L
Consulente	E	P	L

Si può dedurre che l'assenza di ridondanza mi porterebbe ad avere $M \cdot 2$ accessi in lettura in più per un'operazione come quella appena esaminata.

Malgrado la frequenza di tale operazione sia solo di due volte al mese, si osserva che il numero di accessi verrebbe ridotto significativamente in presenza di ridondanza. Si decide, quindi, di mantenere la ridondanza in quanto andrebbe a semplificare notevolmente la ricerca.

Si osserva che l'attributo "Portafoglio" dell'entità Consulente sia anch'esso ridondante, poichè ricavabile dall'attributo "Portafoglio" appartenente all'entità Cliente.

Si decide di analizzare tale ridondanza studiando il numero di accessi che la seguente operazione richiederebbe:

"Ricerca dei consulenti che hanno raggiunto una determinata dimensione di portafoglio."

- Assenza di ridondanza:

Concetto	Costrutto	Accessi	Tipo
Cliente	E	X	L
Segue	R	X	L
Consulente	E	Y	L

X = numero clienti della banca = 400000

Y = numero consulenti della banca = 2000

- Presenza di ridondanza:

Concetto	Costrutto	Accessi	Tipo
Consulente	E	Y	L

Si osserva che, in assenza di ridondanza, l'operazione richiederebbe $X+X+Y$ accessi in lettura, mentre, in presenza di ridondanza, questa prevederebbe Y accessi in lettura.

Nonostante la frequenza di tale operazione sia solo di due volte al mese, si osserva che il numero di accessi, in presenza di tale ridondanza, è notevolmente inferiore, considerando che ciascun consulente presenta un numero medio di 200 clienti. Di conseguenza, si decide di mantenere la ridondanza poiché porterebbe ad avere un significativo vantaggio in termini di

calcolo computazionale e poichè si ritiene opportuno permettere un accesso diretto ai portafogli dei propri consulenti, anche se ciò comporterebbe un appesantimento degli aggiornamenti.

Si può notare che è presente un'ulteriore ridondanza dovuta a ciclo nello schema. Tale ridondanza è data dalle entità Consulente, Cliente e Appuntamento e le relative relazioni "Segue", "Coinvolge" e "Organizza". Osserviamo, però, che nessuna delle operazioni indicate coinvolge la ridondanza e perciò non è possibile procedere con uno studio analogo a quelli appena effettuati per determinarne l'utilità.

Si ritiene, perciò, lecito eliminare la relazione "Organizza" al fine di guadagnare più memoria e non appesantire ulteriormente il database.

Eliminazione delle generalizzazioni

Si ha una sola generalizzazione con entità padre Prodotti ed entità figlie rispettivamente Prodotto finanziario, Polizza e Finanziamento.

Possiamo osservare che nessuna delle entità figlie presenta attributi diversi o relazioni separate. Inoltre, si ritiene che non si avrà interesse ad accedere a tali entità in maniera distinta e specifica.

Si decide, quindi, di eseguire un accorpamento nell'entità padre, andando ad aggiungere un attributo "Tipo" nell'entità Prodotto.

Partizionamento di entità

Si osserva che nessuna entità presenta degli attributi che si riferiscono a concetti diversi o che necessitano un accesso separato, perciò non ci sembra opportuno fare alcun tipo di partizionamento di entità.

Eliminazione attributi multivalore

L'attributo "Telefono" relativo alla entità Cliente presenta una cardinalità (1,N), il che implica che questo possa assumere più valori. Di conseguenza si decide di eliminare tale attributo e creare una nuova entità Telefono Clienti. Questa entità sarà collegata tramite una relazione "Recapito" all'entità Cliente e presenterà un attributo "Numero".

Inoltre, si osserva che l'attributo "Durata" relativo all'entità Prodotto presenta una cardinalità specificata (0,1), ciò implica che tale attributo potrà ammettere un valore nullo.

Partizionamento di relationship

Osservo che la relazione "Segue" che collega le entità Consulente e Cliente è una relazione molti a molti. La cardinalità (1,N) da parte del Cliente è giustificata dal fatto che il database vuole tenere traccia dei consulenti attuali e passati di ogni cliente all'interno della banca. Si decide quindi di eseguire un partizionamento della relazione "Segue" in due nuove relazioni: "Consulenza attuale" la quale richiederà una cardinalità (1,N) da parte del consulente e (1,1)

da parte del cliente, e “Consulenza passata” la quale, a sua volta, richiederà cardinalità (1,N) da ambedue le estremità della relazione.

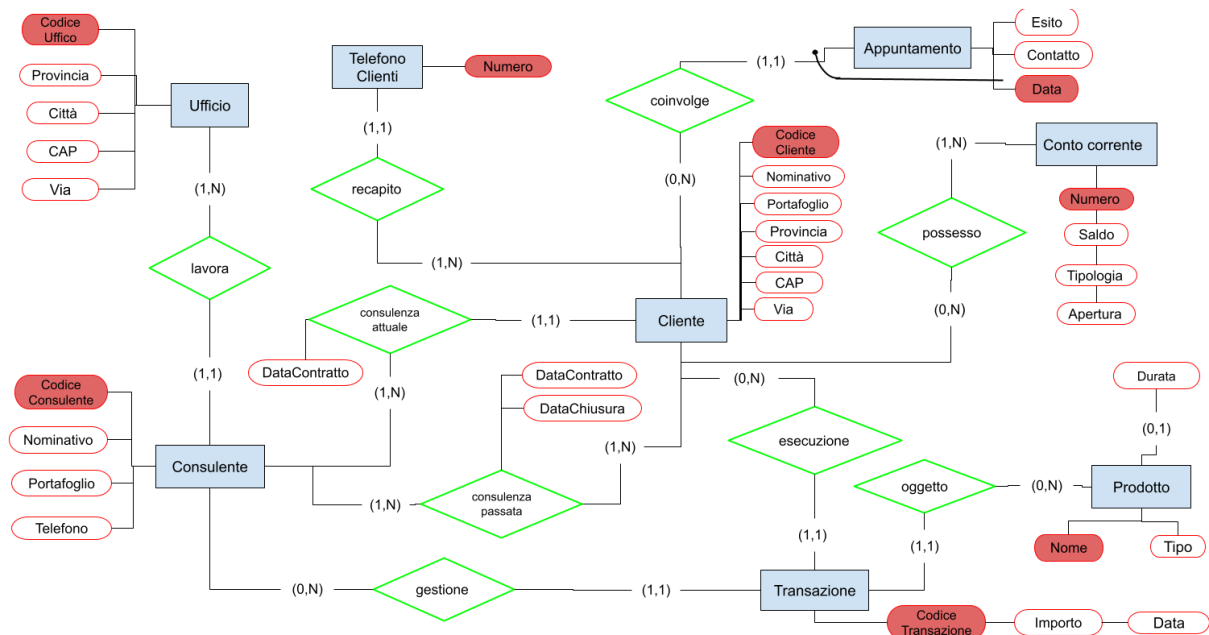
Accorpamento di entità o relationship

Non è necessario accorpare nessuna entità o relationship.

Scelta degli identificatori

- Consulente : si decide di utilizzare come identificatore l'attributo “Codice Consulente”;
- Cliente : si decide di utilizzare come identificatore l'attributo “Codice Cliente”;
- Transazione : si decide di utilizzare come identificatore l'attributo “Codice Transazione”;
- Conto corrente : si decide di utilizzare come identificatore l'attributo “Numero”;
- Ufficio : si decide di utilizzare come identificatore un nuovo attributo “Codice Ufficio”;
- Appuntamento : si decide di utilizzare come identificatori gli attributi “Data” e “Cliente” (identificatore esterno);
- Prodotto: si decide di utilizzare come identificativo l'attributo “Nome”;
- Telefono Clienti: si decide di utilizzare come identificatore l'attributo “Numero”;

Diagramma E-R Ristrutturato

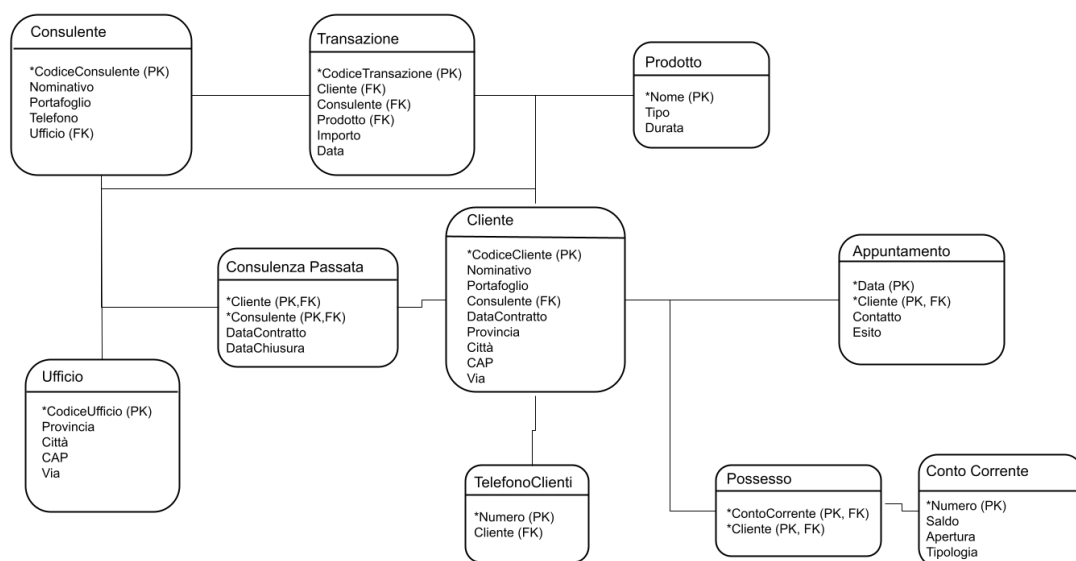


Passaggio al modello razione

Si individuano le seguenti relazioni, identificatori e attributi per lo schema logico con il modello relazionale:

- Consulente (CodiceConsulente, Nominativo, Portafoglio, Telefono, Ufficio);
- Cliente (CodiceCliente, Nominativo, Portafoglio, Consulente, DataContratto, Provincia, Città, CAP, Via);
- Consulenza passata (Cliente, Consulente, DataContratto, DataChiusura);
- Ufficio (CodiceUfficio, Provincia, Città, CAP, Via);
- Telefono Clienti (Numero, Cliente);
- Conto corrente (Numero, Saldo, Apertura, Tipologia);
- Possesso (ContoCorrente, Cliente);
- Appuntamento (Data, Cliente, Contatto, Esito);
- Transazione (CodiceTransazione, Cliente, Consulente, Prodotto, Importo, Data);
- Prodotti (Nome, Tipo, Durata);

Schema Logico



Normalizzazione

Si osserva che nelle tabelle Ufficio e Cliente non è rispettata la terza forma normale poiché la colonna CAP risulta dipendere dalle colonne Provincia e Città.

La soluzione a questo problema richiederebbe l'aggiunta di una nuova tabella chiamata "Località" composta dalle colonne CAP, Città e Provincia (con primary key composta dalla sola colonna CAP), in modo da mantenere il solo campo CAP nelle due tabelle inizialmente specificate. Tuttavia, nonostante l'opzione appena citata sia logicamente accettabile, si decide di non utilizzarla poiché andrebbe a complicare ulteriormente il database e non porterebbe alcun vantaggio. Per esempio, ipotizzando una ricerca di un determinato ufficio, si ritiene opportuno avere accesso diretto alle informazioni relative l'indirizzo e la località essendo queste quelle più di interesse. Si osserva, inoltre, che la colonna Via, relativa alle tabelle Ufficio e Cliente, non dipende da nessun altro campo poiché si ritiene che tale informazione sia specifica per ogni ufficio e personale di ogni cliente.

Si può notare che nella tabella Transazione la terza forma normale non è nuovamente rispettata poiché la colonna Consulente non dipende solamente dalla PK ma anche dalle colonne Cliente e Data. Affinché questa possa essere garantita, sarebbe necessario creare una nuova tabella detta "Consulenza" formata dalle colonne Cliente, Data (le quali diventerebbero primary key della nuova tabella) e Consulente. Si ritiene, però, che questa soluzione non sia vantaggiosa poiché tali informazioni sono già registrate all'interno della tabella Clienti, dalla colonna "Consulenza", e nella tabella Consulenza passata. Si decide, quindi, di non apportare alcuna modifica.

Infine, andando ad analizzare la tabella Cliente si può osservare che anche in questo caso la terza, e di conseguenza anche la seconda, forma normale è rispettata poiché la colonna "DataContratto" si riferirà sempre e solo al contratto stipulato dal consulente attuale, indipendentemente da chi esso sia.

Sarà la tabella Consulenza passata, infatti, a tenere traccia dei contratti firmati e poi chiusi con i consulenti precedenti.

Progettazione fisica

Si può osservare che le ricerche effettuate sulla tabella Conto Corrente avverranno spesso tramite l'attributo Tipologia. Siccome tali ricerche presentano una frequenza molto elevata, risulta conveniente aggiungere un indice secondario sull'attributo Tipologia della tabella Conto Corrente.

Allo stesso modo, si osserva che l'attributo Tipo relativo alla tabella Prodotto verrà utilizzato spesso nella ricerca dei prodotti coinvolti nelle transazioni. Si decide quindi di inserire un indice secondario sull'attributo Tipo.

Al contrario, non si ritiene opportuno introdurre un indice secondario sull'attributo Data nella tabella Transazione, anche se questo risulterà essere spesso oggetto di ricerche su tale tabella, poiché la frequenza di inserimento di nuove transazioni all'interno della tabella sarà molto maggiore di quella prevista dalle operazioni e ciò richiederebbe un aggiornamento continuo dell'indice.

Infine, andando ad analizzare la tabella Consulente, si può affermare che l'attributo Portafoglio sarà utilizzato nella ricerca di alcuni consulenti.

Siccome si assume che l'assunzione di nuovi consulenti, e, di conseguenza, l'inserimento di questi nella tabella Consulenti, sia poco frequente all'interno della banca, si ritiene conveniente aggiungere un indice secondario riferito all'attributo Portafoglio nella tabella Consulente.

Creazione del database

```
CREATE DATABASE ConsulenzaBanca;
```

```
USE ConsulenzaBanca;
```

```
CREATE TABLE Ufficio (  
CodiceUfficio INTEGER AUTO_INCREMENT,  
Provincia VARCHAR(2),  
Città VARCHAR(45),  
CAP MEDIUMINT,  
Via VARCHAR(60),  
PRIMARY KEY(CodiceUfficio)  
);
```

```
CREATE TABLE Consulente (  
CodiceConsulente INTEGER AUTO_INCREMENT  
Nominativo VARCHAR(45),  
Portafoglio DOUBLE,  
Telefono CHAR(20),  
CodiceUfficio INTEGER,  
PRIMARY KEY(CodiceConsulente),  
FOREIGN KEY (CodiceUfficio)  
REFERENCES Ufficio(CodiceUfficio)  
);
```

```
CREATE TABLE Cliente (  
CodiceCliente INTEGER AUTO_INCREMENT,  
Nominativo VARCHAR(45),  
Portafoglio DOUBLE NOT NULL,  
ConsulenteAttuale INTEGER,  
DataContratto DATE,  
Provincia VARCHAR(2),  
Città VARCHAR(45),  
CAP MEDIUMINT,  
Via VARCHAR(60),  
PRIMARY KEY(CodiceCliente),  
FOREIGN KEY (ConsulenteAttuale)  
REFERENCES Consulente(CodiceConsulente)  
);
```

```
CREATE TABLE ConsulenzaPassata (  
CodiceCliente INTEGER,  
CodiceConsulente INTEGER,  
DataContratto DATE,  
ChiusuraContratto DATE,  
PRIMARY KEY(CodiceCliente, CodiceConsulente),  
FOREIGN KEY (CodiceCliente)  
REFERENCES Cliente(CodiceCliente),  
FOREIGN KEY (CodiceConsulente)
```

```
REFERENCES Consulente(CodiceConsulente)
);
```

```
CREATE TABLE Appuntamento (
DataAppuntamento DATE,
CodiceCliente INTEGER,
Contatto VARCHAR(45),
Esito VARCHAR(45),
PRIMARY KEY(DataAppuntamento, CodiceCliente),
FOREIGN KEY (CodiceCliente)
REFERENCES Cliente(CodiceCliente)
);
```

```
CREATE TABLE TelefonoClienti (
Numero VARCHAR(45),
CodiceCliente INTEGER,
PRIMARY KEY(Numero),
FOREIGN KEY (CodiceCliente)
REFERENCES Cliente(CodiceCliente)
);
```

```
CREATE TABLE Prodotto (
NomeProdotto VARCHAR(45),
Tipo VARCHAR(45),
Durata VARCHAR(45),
PRIMARY KEY (NomeProdotto)
);
```

```
CREATE TABLE Transazione (
CodiceTransazione INTEGER AUTO_INCREMENT,
CodiceCliente INTEGER,
CodiceConsulente INTEGER,
Prodotto VARCHAR(45),
Importo DOUBLE,
DataTransazione DATE,
PRIMARY KEY (CodiceTransazione),
FOREIGN KEY (CodiceCliente)
REFERENCES Cliente(CodiceCliente),
FOREIGN KEY (CodiceConsulente)
REFERENCES Consulente(CodiceConsulente),
FOREIGN KEY (Prodotto)
REFERENCES Prodotto(NomeProdotto)
);
```

```
CREATE TABLE ContoCorrente (
Numero INTEGER AUTO_INCREMENT
Saldo DOUBLE,
DataApertura DATE,
Tipologia VARCHAR(45),
```



```
PRIMARY KEY (Numero)
);
```

```
CREATE TABLE PossessoCC(
NumeroContoCorrente INTEGER,
CodiceCliente INTEGER,
PRIMARY KEY (NumeroContoCorrente,CodiceCliente),
FOREIGN KEY (NumeroContoCorrente)
REFERENCES ContoCorrente(Numero),
FOREIGN KEY (CodiceCliente)
REFERENCES Cliente(CodiceCliente)
);
```

Realizzazione delle operazioni

Nonostante le prime due operazioni citate presentino delle frequenze più limitate rispetto quelle successive, si ritiene utile realizzarle tutte tramite delle stored procedure che rimarranno memorizzate all'interno del DBMS, in modo da poterle sfruttare anche in sessioni future.

- **Operazione 1:**

“Ricerca dei consulenti che hanno raggiunto una determinata dimensione di portafoglio.”

```
DELIMITER $$
CREATE PROCEDURE ConsulenzaBanca.consulentiPortafoglio(IN
portafoglioDesiderato DOUBLE)
BEGIN
    SELECT CodiceConsulente, Nominativo, Portafoglio FROM
    ConsulenzaBanca.Consulente
    WHERE Portafoglio>= portafoglioDesiderato
    ORDER BY Portafoglio DESC;
END $$
DELIMITER;
```

- **Operazione 2:**

“Ordinamento decrescente dei consulenti per ammontare di valore nelle transazioni gestite nell'ultimo anno”

```
DELIMITER $$
CREATE PROCEDURE ConsulenzaBanca.investimentiAnnuali()
BEGIN
    SELECT CodiceConsulente ,sum(Importo) as investimentoAnnuale
    FROM ConsulenzaBanca.Transazione
    WHERE DataTransazione> CURDATE()-365
    GROUP BY CodiceConsulente
    ORDER BY investimentoAnnuale DESC;
END $$
DELIMITER;
```

- **Operazione 3:**

“Ricerca delle transazioni che coinvolgono finanziamenti avvenute nell’ultima settimana, ordinate per dimensione dell’importo finanziato”:

```
DELIMITER $$
CREATE PROCEDURE ConsulenzaBanca.transazioniSett(IN tipoProdotto
VARCHAR(45))
BEGIN
    SELECT * FROM ConsulenzaBanca.Transazione INNER JOIN
    ConsulenzaBanca.Prodotto ON Prodotto.NomeProdotto=
    Transazione.Prodotto
    WHERE Prodotto.Tipo = tipoProdotto AND
    Transazione.DataTransazione > CURDATE()-7
    ORDER BY Importo DESC;
END $$
DELIMITER;
```

- **Operazione 4:**

“Ricerca del saldo contabile presente su una determinata tipologia di conto corrente di un determinato cliente”:

```
DELIMITER $$
CREATE PROCEDURE controlloSaldo(IN userNominativoCliente
VARCHAR(45), IN userTipoCC VARCHAR(45))
BEGIN
    SELECT Saldo FROM ConsulenzaBanca.ContoCorrente
    WHERE Tipologia= userTipoCC AND Numero = (SELECT
    NumeroContoCorrente FROM ConsulenzaBanca.PossessoCC INNER
    JOIN ConsulenzaBanca.Cliente ON PossessoCC.CodiceCliente=
    Cliente.CodiceCliente
    WHERE Cliente.Nominativo = userNominativoCliente);
END $$
DELIMITER;
```

Trigger

Per rispettare il primo vincolo non esprimibile graficamente indicato nella parte di progettazione, si decide di creare un trigger che vada ad incrementare il portafoglio di un consulente ogni volta che viene aggiunto un nuovo cliente all’interno della tabella Clienti.

```
DELIMITER $$
CREATE TRIGGER trg_aggiornaPortafoglio
AFTER INSERT ON Cliente FOR EACH ROW
BEGIN
    UPDATE ConsulenzaBanca.Consulente SET
    Portafoglio=Portafoglio+new.Portafoglio WHERE
    CodiceConsulente = new.ConsulenteAttuale;
END $$
DELIMITER;
```

Affinché venga rispettato anche il secondo vincolo non esprimibile graficamente, creo un trigger associato alla tabella Consulenza Passata:

```
DELIMITER $$
CREATE TRIGGER trg_controlloData
BEFORE INSERT
ON ConsulenzaPassata FOR EACH ROW
BEGIN
    IF new.ChiusuraContratto < new.DataContratto THEN
        SIGNAL SQLSTATE '45001' SET MESSAGE_TEXT='Le date non sono
        valide';
    END IF;
END $$
DELIMITER;
```