


M3 – My SQL

Progettazione di DB



A partire da un caso di studio di suo interesse (es. ecommerce di prodotti cosmetici; dipendenti di una multinazionale farmaceutica, personale amministrativo, docente e studenti di un liceo; utenti, pagine, gruppi di un social network), si esegua la progettazione di una base dati (almeno 5 relazioni dello schema concettuale). La progettazione consiste di un documento con:

1. Descrizione caso di studio scelto
2. Modello ER
3. Modello logico
4. Progettazione fisica (codice DDL e DML per la creazione della base di dati, delle tabelle, delle chiavi primarie ed esterne, degli attributi)
5. Almeno 10 interrogazioni (tutte con almeno due tra raggruppamenti e ordinamenti, join e date)

Attività di acquisto di prodotti cosmetici in store

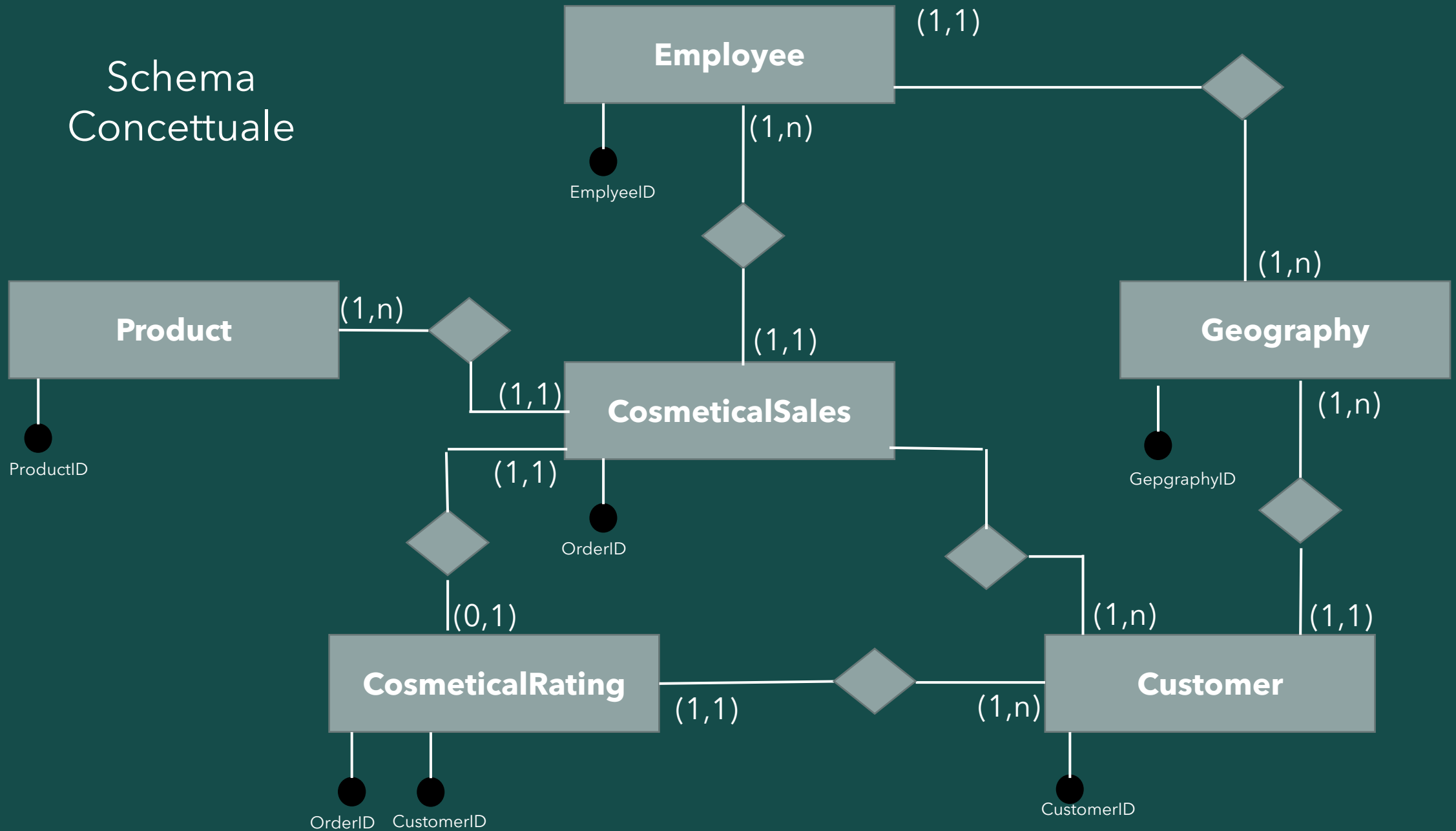
- La base di dati contiene le informazioni relative a tutte le persone impiegate presso lo store. Per ogni impiegato sono noti il Codice identificativo, il nome, il cognome, l'area geografica e il ruolo. Il codice identificativo permette di identificare univocamente l'impiegato.
- I prodotti disponibili presso la catena sono identificati dal codice prodotto, e sono noti il nome del prodotto, la categoria a cui appartengono, il prezzo e il costo.
- Ogni cliente è registrato attraverso un codice cliente univoco e sono noti anche il nome, il cognome e l'area geografica in cui sono residenti
- Prendiamo come riferimento le vendite dei prodotti nel periodo gennaio-marzo 2019 alle quali corrispondono un codice ordine univoco e sono noti l'area geografica del cliente, il cliente, l'impiegato, il codice prodotto, la quantità, la data ordine, il totale della vendita, il costo della vendita e il profitto.
- Infine abbiamo il rating di gradimento di ciascun cliente per ogni ordine effettuato ed è identificato dalla combinazione univoca di codice ordine e cliente, e sono noti il tipo di pagamento e il Rating di gradimento dell'ordine.

Implementazione concettuale

Creazione di un DB con le tabelle e relativi attributi con vincoli di chiave primaria:

- CUSTOMER:(CustomerID, GeographyID, FirstName, LastName)
- PRODUCT:(ProductID, Category, ProductName, UnitPrice, UnitCost)
- EMPLOYEE: (PersonID, GeographyID, FirstName, LastName, JobRole)
- GEOGRAPHY:(GeographyID, Region, City, Street)
- COSMETICSales: (OrderID, GeographyID, CustomerID, PersonID, ProductID, Quantity, SalesDate, SalesAmount, ProductCost, Profit)
- COSMETICRating: (OrderID, CustomerID, PaymentType, Rating)

Schema Concettuale



Schema Logico

ProductID	Category	ProductName	UnitPrice	UnitCost
P-03	sun care	Doposole	9,05	4,07
P-06	body care and perfumes	Eau de Toilette A	27,07	12,18
P-13	body care and perfumes	Eau de Toilette B	66,47	29,91
P-14	body care and perfumes	Eau de Toilette C	74,69	33,61
P-16	body care and perfumes	Eau de Toilette D	80,47	36,21
P-17	body care and perfumes	Eau de Toilette E	81,23	36,55

PERSON ID	GeographyID	NOME	COGNOME	Role
E01	GEO-33617	Abraham	Bennet	Manager
E02	GEO-33616	Reginald	Blotchett-Halls	SalesPerson
E03	GEO-33615	Cheryl	Carson	Director
E04	GEO-33614	Michel	DeFrance	Manager
E05	GEO-33613	Innes	del Castillo	manager
E06	GEO-33612	Ann	Dull	SalesPerson

OrderID	Area GeograficalID	Customer ID	ProductID	PersonID	UnitPrice	Quantity	SalesDate	TotalAmount	CostProduct
687-15-1097	GEO-33713	C-03	P-03	E02	9,05	2	43468	18,1	4,0725
343-87-0864	GEO-33714	C-02	P-06	E02	27,07	1	43468	27,07	12,1815
552-44-5977	GEO-33715	C-01	P-13	E02	66,47	8	43468	531,76	29,9115
132-23-6451	GEO-35439	C-153	P-14	E06	74,69	5	43469	373,45	33,6105
750-67-8428	GEO-35440	C-152	P-16	E06	80,47	7	43470	563,29	36,2115
729-06-2010	GEO-35441	C-151	P-17	E07	81,23	9	43471	731,07	36,5535

GeographyID	Regione	Città	Via
GEO-33617	lombardia	Varese	Masnago
GEO-33616	lombardia	Varese	Masnago
GEO-33615	lombardia	Varese	Masnago
GEO-33614	lombardia	Varese	Lissago - Calcinato
GEO-33613	lombardia	Varese	Lissago - Calcinato
GEO-33612	lombardia	Varese	Giubiano - San Carlo

OrderID	Customer ID	Payment	RatingMarket
687-15-1097	C-03	Cash	97
343-87-0864	C-02	Credit card	71
552-44-5977	C-01	Credit card	62
132-23-6451	C-153	Cash	78
750-67-8428	C-152	Ewallet	91
729-06-2010	C-151	Cash	92

CustomerID	GeographyID	FirstName	LastName
C-02	GEO-33714	Adam	Phillips
C-01	GEO-33715	Abe	Nelson
C-153	GEO-35439	Fukiko	Martin
C-152	GEO-35440	Frederick	Robinson
C-151	GEO-35441	Fred	Deborde
C-413	GEO-42204	Robertson	Kelly

Implementazione fisica

```
CREATE DATABASE E_Commerce2019;
```

```
CREATE TABLE AGeography (  
  GeographyID varchar(25)  
  , Region varchar (25)  
  , City varchar (25)  
  , Address varchar(50)  
  CONSTRAINT PK_dimGeography_GeographyID primary key  
  (GeographyID))
```

```
CREATE TABLE Customer (  
  CustomerID varchar(25)  
  ,GeographyID varchar(25)  
  ,FirstName varchar(25)  
  ,LastName varchar(25)  
  ,CONSTRAINT PK_Costumer_CustomerID primary key  
  (CustomerID)  
  ,CONSTRAINT FK_Customer_dimGeography_GeographyID  
  foreign key (GeographyID)  
  references dimGeography (GeographyID))
```

```
CREATE TABLE Product (  
ProductID varchar(25)  
,Category varchar (25)  
,ProductName varchar (25)  
,UnitPrice money  
,UnitCost money  
,CONSTRAINT PK_Product_ProductID primary key (ProductID))
```

```
CREATE TABLE Employee (  
PersonID varchar (25)  
, GeographyID varchar (25)  
, FirstName varchar (25)  
, LastName varchar (25)  
, Role varchar(25)  
, CONSTRAINT PK__Employee_PersonID primary key (PersonID)  
, CONSTRAINT FK_Employee_dimGeography_GeographyID  
foreign key (GeographyID)  
references dimGeography (GeographyID))
```

```
CREATE TABLE CosmeticalRating (  
  OrderID varchar(25)  
, CustomerID varchar(25)  
, PaymentType varchar(25)  
, Rating int  
, CONSTRAINT PK_CosmeticalRating_OrderID primary key (OrderID)  
, CONSTRAINT FK_CosmeticalSales_Customer_CustomerID foreign key (CustomerID)  
  references Customer (CustomerID))
```



```
CREATE TABLE CosmeticalSales(  
    OrderID varchar(25)  
    , GeographyID varchar(25)  
    , CustomerID varchar(25)  
    , ProductID varchar(25)  
    , PersonID varchar(25)  
    , UnitPrice money  
    , Quantity int  
    , SalesDate date  
    , SalesAmount money  
    , ProductCost money  
    , Profit money  
    , CONSTRAINT PK_CosmeticalSales_OrderID primary key (OrderID)  
    , CONSTRAINT FK_CosmeticalSales_Customer_CustomerID foreign key (CustomerID)  
      references Customer (CustomerID)  
    , CONSTRAINT FK_CosmeticalSales_Product_ProductID foreign key (ProductID)  
      references Product (ProductID)  
    , CONSTRAINT FK_CosmeticalSales_Employee_PersonID foreign key (PersonID)  
      references Employee (PersonID))
```

Query

-- 1 Vogliamo visualizzare nome e cognome dei clienti per fatturato in ordine decrescente

```
SELECT*  
FROM CosmeticalSales
```

```
SELECT  
  c.FirstName as Nome  
,c.LastName as Cognome  
,SUM (cSales.SalesAmount) as Fatturato  
FROM CosmeticalSales cSales  
INNER JOIN Customer c  
ON cSales.CustomerID=c.CustomerID  
WHERE cSales.SalesAmount =(SELECT
```

```
SUM (cSales.SalesAmount) as Fatturato  
FROM CosmeticalSales cSales  
INNER JOIN Customer c  
ON cSales.CustomerID=c.CustomerID)
```

```
GROUP BY c.FirstName, c.LastName  
ORDER BY Fatturato DESC
```

```
-- 2 visualizza solo il cliente con il maggior fatturato
```

```
SELECT List1.Nome, List1.Cognome, List1.CustomerID, List1.Fatturato
FROM (
    SELECT c.FirstName AS Nome
        ,c.LastName AS Cognome
        , c.CustomerID
        , SUM(cSales.SalesAmount) AS Fatturato
    FROM CosmeticalSales cSales
    INNER JOIN Customer c
    ON cSales.CustomerID = c.CustomerID
GROUP BY c.FirstName, c.LastName, c.CustomerID) AS List1
WHERE List1.Fatturato = (
    SELECT MAX(Fatturato)
    FROM (
        SELECT SUM(cSales.SalesAmount) AS Fatturato
        FROM CosmeticalSales cSales
        INNER JOIN Customer c ON cSales.CustomerID = c.CustomerID
        GROUP BY c.CustomerID) AS MaxFatturato)
```

```
-- 3 Vogliamo sapere quanti ordini sono stati effettuati con pagamento
attraverso CreditCard
```

```
SELECT
COUNT(*) PaymentCC
FROM CosmeticalRating r
WHERE r.PaymentType = 'Credit card'
GROUP BY r.PaymentType
```

```
-- 4 i codici prodotto con media Rating maggiore di 30
```

```
SELECT  
,Med.ProductID  
,Med.RatingAVG  
FROM (  
    SELECT  
        ,s.ProductID  
        ,AVG (r.Rating) as RatingAVG  
    FROM CosmeticalSales s  
    INNER JOIN CosmeticalRating r  
    ON s.OrderID=r.OrderID  
    GROUP BY s.ProductID) as Med  
WHERE RatingAVG>30
```

```
-- 5 metodo 2 Ordinato per RATING DECRESCENTE
```

```
SELECT  
,s.ProductID  
,AVG (s.Rating) as RatingAVG  
FROM CosmeticalSales s  
INNER JOIN CosmeticalRating r  
ON s.OrderID=r.OrderID  
WHERE r.Rating > 30  
GROUP BY s.ProductID  
ORDER BY RatingAVG DESC
```

```
-- 6 Ottenere l'elenco dei prodotti con fatturato totale > 100 nel mese di marzo.  
--Ordinare l'elenco dei prodotti per fatturato decrescente.  
--Esporre nel result set la quantità totale e la media del prezzo di vendita
```

```
SELECT  
s.ProductID  
, SUM(s.SalesAmount)AS TotalSales  
, SUM(Quantity)AS TotalQty  
, AVG(UnitPrice)AS AverageUnitPrice  
FROM  CosmeticalSales s  
WHERE MONTH (s.SalesDate) = 3  
GROUP BY s.ProductID  
HAVING SUM(s.CosmeticalSales) > 100  
ORDER BY s.ProductID DESC
```

```
-- 7 determinare l'ultima transazione di vendita per ciascun prodotto
```

```
SELECT DISTINCT s1.ProducID, s1.SalesDate  
FROM CosmeticalSales s1  
WHERE s1.SalesDate =(  
                SELECT MAX(s2.SalesDate)  
                FROM CosmeticalSales s2  
                WHERE s1.ProductID=s2.ProductID)
```

```
-- 8 elenco delle transazioni riferite a clienti che vivono in Lombardia
```

```
SELECT
OrderID as NumOrder
,ProductID as codiceProdotto
,CustomerID
FROM CosmeticalSales
WHERE CustomerID in (
    SELECT CustomerID
    FROM Customer
    WHERE GeographyID IN (
        SELECT GeographyID
        FROM DimGeography
        WHERE Region = 'Lombardia'))
```

```
-- 9 crea lista per ottenere join di tabella clienti e area geografica
```

```
CREATE VIEW JDM_GeoCostumer AS (
    SELECT
    c.CustomerID
    ,C.GeographyID
    ,Concat (FirstName, ' ', LastName) AS Cliente
    ,g.Region
    ,g.City
    ,g.Street
    FROM Customer c
    INNER JOIN dimGeography g
    ON c.GeographyID=g.GeographyID)
```

-- 10 crea lista per ottenere join di tabella dipendenti e area geografica

```
CREATE VIEW JDM_GeoEmployee AS (  
    SELECT  
        e.PersonID  
        ,e.GeographyID  
        ,Concat (FirstName, ' ', LastName) AS Agente  
        ,g.Region  
        ,g.City  
        ,g.Street  
    FROM Employee e  
    INNER JOIN dimGeography g  
    ON e.GeographyID=g.GeographyID)
```