

# Infomatica (SGI) - Esercizi SQL – Esercizio 4 (Foodmart)

Link al foglio appunti condiviso con il docente: [goo.gl/8M88ao](https://goo.gl/8M88ao) (l'ultima è una o minuscola)

**1 Parte preliminare.** Avviate MySQL Workbench e connettetevi al server

## 2 Analisi del db Foodmart

Foodmart è una catena di supermercati (fittizia), che opera negli Stati Uniti, Canada e Messico, è composta da punti vendita la cui tipologia spazia dai piccoli negozi a supermercati di ampie dimensioni. Il database contiene i dati delle vendite, delle spese dei clienti, dei costi e delle promozioni ... e tante altre informazioni, riferiti ad un arco temporale di alcuni anni. Nel corso delle esercitazioni, analizzeremo prevalentemente la tabella **sales\_fact\_1998** (e le tabelle collegate) la quale contiene i dati delle vendite effettuate dai negozi della catena nell'anno 1998.

Ogni record della tabella sales\_fact\_1998 riporta i dati di un prodotto acquistato da una persona.

La tabella **sales\_fact\_1998** è composta dai seguenti attributi:

- **product\_id**: chiave esterna che identifica il prodotto venduto
- **time\_id**: chiave esterna che identifica il giorno in cui è avvenuta la vendita
- **customer\_id**: chiave esterna che identifica il cliente
- **promotion\_id**: chiave esterna che identifica l'eventuale promozione applicata
- **store\_id**: chiave esterna che identifica il negozio in cui è avvenuta la vendita
- **store\_sales**: prezzo unitario di vendita applicato
- **store\_cost**: stima del costo unitario che il supermercato ha sostenuto per acquistare il prodotto, trasportarlo, ...
- **unit\_sales**: quantità di prodotto venduta

Del database foodmart fanno parte anche le tabelle elencate qui di seguito (tra parentesi le traduzioni in italiano):

- **product** (prodotti)
- **product\_class** (classi di prodotti)
- **store** (negozi)
- **time\_by\_day** (informazioni sui giorni dell'anno)
- **promotion** (promozioni)
- **customer** (clienti)
- **region** (regione)

queste ultime tabelle contengono informazioni extra sulle vendite effettuate, per esempio le tipologie di prodotti venduti, il negozio che ha effettuato la vendita, i dati del cliente che ha effettuato l'acquisto, le date e così via. Le informazioni delle varie tabelle sono collegate alla tabella sales\_fact\_1998 e tra di loro per mezzo di chiavi. es.: product\_id, time\_id, ...

Svolgete le seguenti operazioni preliminari:

- Effettuate una ricognizione delle tabelle del database foodmart
  - Ricordatevi di indicare il database sul quale vorrete lavorare con il comando *use scrivi\_qui\_il\_nome\_del\_database*;
  - Per ispezionare una tabella, potete utilizzare la query generica: *SELECT \* FROM scrivi\_qui\_il\_nome\_della\_tabella*;
- Scrivete su un foglio di carta uno schema in cui appaia il nome delle tabelle principali, per ogni tabella individuate (se presenti):
  - la chiave primaria (o le chiavi primarie se la chiave è formata da più attributi)
  - i collegamenti con le altre tabelle (individuate cioè le chiavi secondarie). Suggerimento: rappresentate i collegamenti con delle linee.

*NB: lo scrivere le tabelle e le chiavi su carta, servirà a voi solo come riferimento per svolgere gli esercizi indicati di seguito, non impiegate più di 5 minuti per eseguire quest'ultimo punto*

Per riconfigurare la connessione al server (accessibile solo dai pc dei lab.)  
Nome connessione: statlettura  
Host: 10.107.8.90  
Port: 6033  
Username: statlettura  
Password: 123.stat

Dopo le operazioni preliminari, scrivete le query SQL necessarie per estrarre le informazioni richieste nei punti seguenti.

1. Visualizzate `product_id`, nome prodotto e la data delle vendite contenute nella tabella `sales_fact_1998`. La data deve essere visualizzata in un formato nel quale si riesca a distinguere giorno, mese e anno; Il nome del prodotto e la data vanno recuperate da tabelle diverse da `sales_fact_1998`. I risultati devono essere ordinati per data e per nome prodotto. Nota: la prima riga del risultato dovrebbe essere: 1006; American Foot-Long Hot Dogs; 1998-01-01 00:00:00. Se utilizzate MySQL Workbench, vengono visualizzate solo le prime 1000 righe del risultato.
2. Visualizzate (per ogni singola vendita) il `time_id`, il `product_id` e il guadagno conseguito sulla base dei dati contenuti nella tabella `sales_fact_1998`. Ordinate il risultato in base al `time_id` e al `product_id`. Vi ricordiamo che:  $\text{guadagno} = (\text{ricavo\_unitario} - \text{costo\_unitario}) * \text{quantità}$ . Si ricorda che nella tabella `sales_fact_1998` gli attributi **`store_sales`**, **`store_cost`** e **`unit_sales`** rappresentano rispettivamente: **`ricavo_unitario`**, **`costo_unitario`** e **`quantità_vendute`**. Nota: la prima riga di risultato dovrebbe essere: 732; 4; 18.01800000
3. Visualizzate l'elenco dei prodotti venduti nel corso dell'anno. Nella tabella risultato deve apparire il nome per esteso del prodotto e l'ID. Ogni prodotto anche se venduto più volte deve apparire una sola volta nel risultato. (dovete usare la clausola GROUP BY). I prodotti devono essere mostrati in ordine crescente rispetto al nome. Nota, la prima riga di risultato dovrebbe essere: 351; ADJ Rosy Sunglasses
4. Per ogni prodotto, calcolate il totale dei costi di acquisto di tutte le vendite (ogni prodotto anche se venduto più volte, deve apparire una sola volta nella tabella risultato);  $\text{costo di una singola vendita} = \text{costo\_unitario} * \text{qt\_venduta}$ . Nel risultato va visualizzato il nome del prodotto. Nota, nel risultato ci sarà: 1; Washington Berry Juice; 519.75450000
5. Per ogni prodotto calcolate il totale dei guadagni (ogni prodotto anche se venduto più volte, deve apparire una sola volta). Nel risultato va visualizzato il nome del prodotto. Il guadagno associato ad una singola vendita può essere calcolato con la formula  $(\text{ricavo\_unitario} - \text{costo\_unitario}) * \text{qt\_venduta}$ . Nota, il prodotto con `product_id` 12 ha questi dati: Jeffers Oatmeal; 1017.78600000
6. Ordinate l'elenco del punto precedente in ordine decrescente in base al guadagno. Nota, i dati del prodotto che ha fatto guadagnare di più sono: 274; Great Pumpnickel Bread; 3166.32140000
7. Individuate i supermercati che hanno venduto più quantità di prodotti (create una lista in ordine decrescente). Nota, il primo della lista è: Store 12 con qt pari a 37680.0000
8. Individuate i supermercati che hanno venduto la minor quantità di prodotti. Visualizzate una lista in ordine crescente. Nota, il primo della lista è: Store 2 con qt pari a 1984.0000
9. Individuate le tipologie di prodotto che hanno fatto guadagnare di più (per le tipologie fate riferimento all'attributo `product_category` della tabella `product_class`). Visualizzate i risultati in ordine decrescente. Nota, la prima categoria è: Vegetables; 267508.50670000
10. Individuate i prodotti più venduti (in termini di quantità), nella regione del 'Central West' (per la regione, fate riferimento all'attributo `sales_region` della tabella `region`). Nota, il più venduto è: (product\_id) 391; Urban Large Eggs; 10.0000
11. Individuate i prodotti più venduti (in termini di quantità), negli USA (per la nazione fate riferimento all'attributo `sales_country` della tabella `region`). Nota, il primo della lista è: (product\_id) 876; Thresher Mint Chocolate Bar; (qt) 268.0000
12. Calcolate il guadagno totale di tutti i prodotti venduti nei mesi di aprile e maggio 1998. Nota: risultato 377409.68240000
13. Individuate i clienti che hanno speso di meno in tutto il 1998 (è sufficiente visualizzare i clienti in ordine crescente rispetto alle spese effettuate). Nota, cliente che ha speso meno: (customer\_id) 2383
14. Individuate i clienti che hanno speso di più in tutto il 1998 (è sufficiente visualizzare i clienti in ordine decrescente rispetto alle spese effettuate). Nota, cliente che ha speso di più: (customer\_id) 5466
15. Individuate la città (attributo `sales_city` della tabella `region`) nella quale sono stati incassati più soldi nel 1998 (l'incasso è l'equivalente del ricavo). Effettuate il group by sull'attributo `region.sales_city`, non su `region.region_id`. Inoltre, tra gli attributi della tabella `region` visualizzate solamente l'attributo `sales_city` nel risultato finale (niente `region_id`, `sales_district`, ...). Nota: Hidalgo con 336305.46000000
16. Individuate le 5 `sales_region` in cui la catena di supermercati ha guadagnato di più. Nota: risultato: North West, Mexico Central, South West, Canada West, Mexico South
17. Svolgete delle query a piacere (cercate ad esempio di capire in che paese/città sarebbe opportuno aprire un nuovo negozio, oppure quali prodotti ritirare dalla vendita, ...).