# Esercizio 1

## Testo

List firstname, surname and occupation of customers in Burnaby with a name starting with "M" and finishing with "y".

## Interpretazione

## Mostrare nome, cognome e occupazione dei clienti in Burnaby con nome che inizi con “M” e finisca con “y”.

Tutte le informazioni cercate sono nella tabella ‘customer’, dunque basterà proiettare i campi ‘fname’, ‘lname’ e ‘occupation’ filtrando la città come ‘Burnaby’ e il nome come una qualsiasi stringa che abbia come primo carattere ‘M’ ed ultimo carattere ‘y’.

La tabella ‘customer’ presenta in ogni riga un diverso cliente, dunque se ci fossero due righe con stesso nome, cognome e occupazione queste farebbero riferimento a due clienti distinti: per questo nella query non è stata inserita la ‘distinct’.

## Query

**SELECT** fname, lname,occupation **from** customer

**where** city ='Burnaby' **and** fname **LIKE** 'M%y'

## Screen

Immagine che contiene testo, elettronica, schermata, schermo

Descrizione generata automaticamente

# Esercizio 2

## Testo

List the products bought by only woman customers with a store cost > 2.00

## Interpretazione

Mostare I prodotti comprati solamente da donne con un costo maggiore di 2.00. Le informazioni relative alle vendite sono nella tabella ‘sales\_fact’, in particolare nella tabella solo censiti tutti gli acquisti effettuati presso i diversi negozi. Le informazioni sui clienti sono nella tabella ‘customer’, in particolare il genere del cliente è uno dei campi presenti nella tabella.

Nella query seguente abbiamo considerato tutti gli acquisti fatti da donne, ad un prezzo maggiore di 2.00, per farlo abbiamo considerato una join tra le tabelle ‘sales\_fact’ e ‘customer’ sul campo ‘customer\_id’, abbiamo filtrato per genere uguale a ‘F’ e prezzo maggiore di 2.00, ed infine abbiamo proiettato la selezione sul campo product\_id che rappresenta il prodotto. Da questa selezione abbiamo escluso tutti gli acquisti effettuati dagli uomini, per farlo abbiamo considerato una join tra le tabelle ‘sales\_fact’ e ‘customer’ sul campo ‘customer\_id’, abbiamo filtrato per genere uguale a ‘M’, infine abbiamo proiettato la selezione sul campo product\_id. L’esclusione è stata eseguita utilizzando il comando ‘EXCEPT’, questo automaticamente escluderà tutti i duplicati quindi non abbiamo dovuto inserire il costrutto ‘DISTINCT’ nelle selezioni al fine di eliminare i duplicati.

Si noti che con la query seguente vengono considerati tutti i prodotti da donne sopra un certo prezzo che non siano stati comprati anche da uomini. In particolare uno stesso prodotto comprato da una donna a 3$ e da un uomo a 1$ verrà escluso.

## Query

**select** sf.product\_id **from** sales\_fact sf

**join** customer c **on** sf.customer\_id =c.customer\_id

**where** c.gender ='F' **and** sf.store\_cost > 2

**EXCEPT** (

**select** sf.product\_id **from** sales\_fact sf

**join** customer c **on** sf.customer\_id =c.customer\_id

**where** c.gender ='M'

)

## ScreenImmagine che contiene testo, schermata, software, Pagina Web Descrizione generata automaticamente

## Interpretazione alternativa

Alternativamente a quanto fatto precedentemente è possibile decidere di considerare tutti i prodotti comprati da donne a un costo maggiore di 2$, escludendo da questi tutti i prodotti da uomini ad un costo maggiore di 2$. Per farlo bisogna aggiungere una ulteriore condizione sul prezzo dei prodotti acquistati da uomini.

## Query alternativa

**select** sf.product\_id **from** sales\_fact sf

**join** customer c **on** sf.customer\_id =c.customer\_id

**where** c.gender ='F' **and** sf.store\_cost > 2

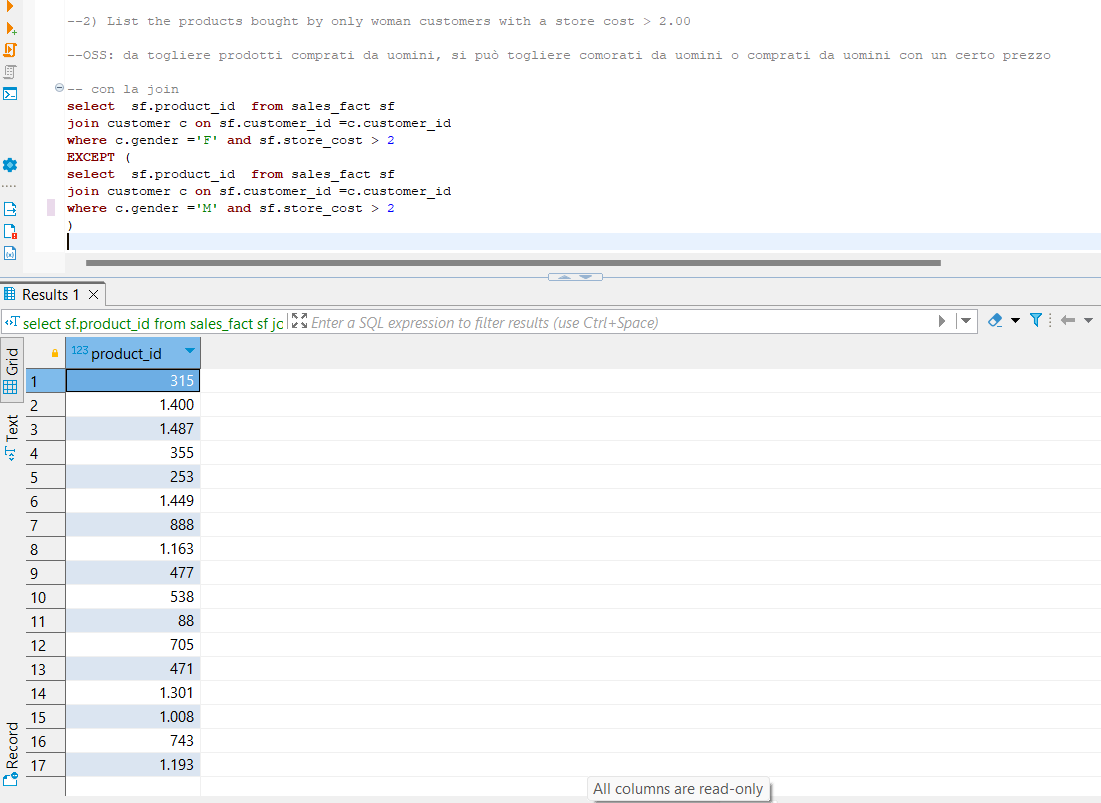
**EXCEPT** (

**select** sf.product\_id **from** sales\_fact sf

**join** customer c **on** sf.customer\_id =c.customer\_id

**where** c.gender ='M’ **and** sf.store\_cost > 2)

Screen



Esercizio 3

## Testo

List of products (ID and name of the product) bought in 1998 and belonging to the brand "Washington" or "Bravo".

## Interpretazione

Mostare identificativo e nome dei prodotti acquistati nel 1998 di brand ‘Washington’ o ‘Bravo’.

Per farlo bisogna considerare le vendite effettuate nel 1998, questa informazione è deducibile considerando la join tra le tabelle ‘time\_by\_day’ e ‘sales\_fact’, a parità di ‘time\_id’, e filtrando le vendite con ‘the\_year’=1998. Inoltre, per selezionare il brand del prodotto venduto bisogna fare una ulteriore join con la tabella ‘product’ a parità di ‘product\_id’ ed aggiungere alla condizione di filtro la condizione sul nome del brand.

Lo stesso prodotto piò essere venduto più volte nel corso dello stesso anno ed utilizzando la tabella ‘sales\_fact’, in cui sono censite tutte le vendite, è possibile avere delle righe duplicate, per questo nella proiezione sui campi ‘product\_name’ e ‘product\_name’ viene richiesta la ‘DISTINCT’.

## Query

**select** **DISTINCT** p.product\_id, p.product\_name **from** sales\_fact sf

**join** time\_by\_day tbd **on** sf.time\_id = tbd.time\_id

**join** product p **on** sf.product\_id =p.product\_id

**where** tbd.the\_year =1998 **and** (p.brand\_name= 'Washington' **or** p.brand\_name= 'Bravo')

## Screen

Immagine che contiene testo, schermata, software, Icona del computer

Descrizione generata automaticamente

Link al csv completo:

<https://docs.google.com/spreadsheets/d/1TvWPJgnVOQ5LcUDlaqmOLoOC5fB_F3lRL47XR5GuTQw/edit?usp=sharing>

# Esercizio 4

## Testo

List the products bought only in 1998

## Interpretazione

Mostrare I prodotti venduti nel solo anno 1998. Per farlo abbiamo considerato tutti i prodotti venduti nel 1998 e a questi abbiamo escluso i prodotti venduti in anni diversi dal 1998.

Se un prodotto è stato venduto sarà censito nella tabella ‘sales\_fact’, in cui però non è presente l’informazione sull’anno della vendita e nome del prodotto. Per selezionare i prodotti venduti nel 1998 è stata considerata la join tra le tabelle ‘sales\_fact’ e ‘time\_by\_day’ a parità di ‘time\_id’ e filtrato per ‘the\_year’=1998, per avere l’informazione sul nome del prodotto è stata fatta una ulteriore join con la tabella ‘product’ a parità di ‘product\_id’. Infine, è stata considerata la proiezione sui campi ‘product\_id’ e ‘product\_name’. Per escludere i prodotti venduti in anni diversi dal 1998 abbiamo selezionato analogamente i prodotti desiderati ed usato il costrutto ‘EXCEPT’.

## Query

**select** p.product\_id, p.product\_name **from** sales\_fact sf

**join** time\_by\_day tbd **on** sf.time\_id = tbd.time\_id

**join** product p **on** sf.product\_id =p.product\_id

**where** tbd.the\_year =1998

**except**

(

**select** p.product\_id, p.product\_name **from** sales\_fact sf

**join** time\_by\_day tbd **on** sf.time\_id = tbd.time\_id

**join** product p **on** sf.product\_id =p.product\_id

**where** tbd.the\_year <>1998)

## Screen

Immagine che contiene testo, elettronica, schermata, software

Descrizione generata automaticamente

# Esercizio 5

## Testo

List the products (indicating the code and the name) bought with the promotion "Price Winners" and that in 1997 have been bought at least once with store sales > 15.00, while in 1998 with store sales > 10.00.

## Interpretazione

Mostrare i prodotti comprati con la promozione ‘Prince Winner’, di questi selezionare solo quelli che sono stati venduti almeno una volta con store\_sales>15 nel 1997 e nel 1998 con store\_sales>10.

Nella costruzione della query abbiamo inizialmente selezionato tutti i prodotti acquistati nel 1997 con la promozione desiderata con store\_sales>15, tra questi abbiamo filtrato i product\_id venduti nel 1998 con la stessa promozione con store\_sales>10. Per filtrare i product\_id abbiamo utilizzato il costrutto ‘IN’. Infine, abbiamo selezionato la proiezione sui campi ‘product\_id’ e ‘product\_name’.

Si osservi che i prodotti con le caratteristiche richieste possono essere stati venduti diverse volte nel 1997 e 1998, dunque abbiamo aggiunto il costrutto ‘DISTINCT’ nella selezione al fine di eliminare possibili duplicati.

## Query

**select** **DISTINCT** p2.product\_id, p2.product\_name **from** sales\_fact sf

**join** promotion p **on** sf.promotion\_id=p.promotion\_id

**join** time\_by\_day tbd **on** sf.time\_id =tbd.time\_id

**join** product p2 **on** sf.product\_id =p2.product\_id

**where** sf.store\_sales >15 **and** p.promotion\_name ='Price Winners' **and** tbd.the\_year =1997

**and**

p2.product\_id **in** (

**select** sf.product\_id **from** sales\_fact sf

**join** time\_by\_day tbd **on** sf.time\_id =tbd.time\_id

**join** promotion p **on** p.promotion\_id =sf.promotion\_id

**where** tbd.the\_year =1998 **and** sf.store\_sales >10 **and** p.promotion\_name ='Price Winners')

## Screen

Immagine che contiene testo, schermata, software, Pagina Web

Descrizione generata automaticamente

# Esercizio 6

## Testo

List customers (indicating the firstname, surname, and number of children) who bought products of the category "Fruit" in January 1997 or "Seafood" January 1998.

## Interpretazione

Mostrare nome, cognome e numero di figli dei clienti che hanno comprato almeno un prodotto della categoria ‘Fruit’ nel gennaio del 1997 o ‘Seafood’ nel gennaio del 1998.

Le diverse informazioni richieste richiedono l’interrogazione di diverse tabelle, in particolare abbiamo bisogno di partire dalla tabella ‘sales\_fact’ dove sono censiti tutti gli acquisti, dobbiamo considerare la tabella ‘customer’ per avere le informazioni richieste sul cliente, la tabella ‘time\_by\_day’ per selezionare mese e anno della vendita, infine è necessario considerare la tabella ‘product\_class’ per selezionare la categoria del prodotto acquistato.

In particolare, per selezionare la categoria del prodotto venduto bisognerà selezionare il product\_id della tabella ‘sales\_fact’ in modo da avere l’identificativo del prodotto venduto, fare una join con la tabella ‘product’, tramite ‘product\_id’ e fare una ulteriore join tra questa e la tabella ‘product\_class’ attraverso il campo ‘product\_class\_id’, selezionando il campo ‘product\_category’.

## Query

**select** **DISTINCT** c.fname, c.lname, c.total\_children

**from** sales\_fact sf **join** customer c

**on** sf.customer\_id =c.customer\_id

**join** time\_by\_day tbd **on** tbd.time\_id =sf.time\_id

**join** product p **on** sf.product\_id =p.product\_id

**join** product\_class pc **on** p.product\_class\_id =pc.product\_class\_id

**where** (tbd.the\_month='January' **and** tbd.the\_year=1997 **and** pc.product\_category='Fruit')

**or** (tbd.the\_month='January' **and** tbd.the\_year=1998 **and** pc.product\_category='Seafood')

## Screen

Immagine che contiene testo, schermata, software, Icona del computer

Descrizione generata automaticamente

# Esercizio 7

## Testo

List store cities with at least 100 active customers in September 1998.

Interpretazione

Mostrare le città in cui sono allocati i negozi che nel settembre del 1998 hanno avuto almeno 100 clienti attivi. Senza avere ulteriori informazioni possiamo interpretare un cliente come ‘attivo’ se ha effettuato almeno un acquisto in un mese. In particolare, nella query considereremo con almeno 100 clienti che abbiano effettuato almeno un acquisto nel settembre del 1998.

Per farlo abbiamo inizialmente costruito la vista ‘view\_active\_customer’ dove abbiamo considerato le coppie distinte di cliente e negozio presenti nella tabella ‘sales\_fact’ nel settembre del 1998, dunque hanno effettuato almeno un acquisto. Successivamente abbiamo eseguito una join tra la vista e la tabella store in modo da poter selezionare la città. Infine, abbiamo fatto una GROUP BY a parità di ‘store\_city’, considerando solo quelli con almeno 100 clienti attivi, in modo da effettuare il conteggio distinto dei clienti.

Query

**with** view\_active\_customers **as**(

**select** **DISTINCT** sf.customer\_id, sf.store\_id

**from** sales\_fact sf

**join** time\_by\_day tbd

**on** tbd.time\_id = sf.time\_id

**where** tbd.the\_month='September'**and** tbd.the\_year=1998)

**select** s.store\_city, **count**(**DISTINCT** view\_active\_customers.customer\_id) **as** total\_active\_customers

**from** view\_active\_customers

**join** store s

**on** s.store\_id=view\_active\_customers.store\_id

**group** **by** s.store\_city

**having** **count**(**DISTINCT** view\_active\_customers.customer\_id)>= 100

Screen

Immagine che contiene testo, schermata, software, Pagina Web

Descrizione generata automaticamente

# Esercizio 8

## Testo

List for each store country the number of female customers and the number of male customers. Order the result with respect to the store country.

## Interpretazione

Mostare il numero di clienti uomini e donne per ogni sore country. Per farlo è necessario costruire due viste, una che contenga le coppie distinte di cliente e negozio in cui ci siano solo donne, una analoga con solo clienti uomini e joinare queste viste con la tabella ‘store’, a partità di ‘store\_id’, selezionando il campo ‘store\_country’. A partire da questa join viene fatta una GROUP BY a parità di ‘store\_country’, in modo da poter contare il numero di clienti donne distinte e clienti uomini distinti in ogni paese.

Si osservi che viene effettuata una LEFT JOIN in modo da partire dalla tabella store e per ogni store\_id aggiungere le informazioni provenienti dalle viste, se esistenti. Inoltre, per calcolare il numero di clienti uomini e donne per ogni store\_country è stato utilizzato il costrutto DISTINCT per evitare ripetizioni proveniente da customer\_id relazionati a store\_id con stesso store\_country.

Infine, il risultato è ordinato per store\_country, per ordine lessicografico.

## Query

**with** female\_customers **as** (

**select** **DISTINCT** c.customer\_id, sf.store\_id

**from** customer c

**join** sales\_fact sf **on** c.customer\_id=sf.customer\_id

**where** c.gender='F'

),

male\_customers **as** (

**select** **DISTINCT** c.customer\_id, sf.store\_id

**from** customer c

**join** sales\_fact sf **on** c.customer\_id=sf.customer\_id

**where** c.gender='M'

)

**select** s.store\_country,**count**(**distinct** female\_customers.customer\_id) **as** total\_female,

**count**(**distinct** male\_customers.customer\_id) **as** total\_male

**from** store s

**left** **join** female\_customers **on** s.store\_id=female\_customers.store\_id

**left** **join** male\_customers **on** s.store\_id=male\_customers.store\_id

**group** **by** s.store\_country

**order** **by** s.store\_country

## Screen

Immagine che contiene testo, schermata, software, Pagina Web

Descrizione generata automaticamente

# Esercizio 9

## Testo

For each month provide the number of distinct customers who bought at least 10 distinct product categories

## Interpretazione

Mese per mese calcolare il numero di utenti distinti che abbiano comprato prodotti di almeno 10 categorie differenti. Per farlo è stata calcolata inizialmente la vista ‘view\_categorie\_distinte\_per\_mese’ facendo una GROUP BY per mese e customer\_id in modo da poter calcolare per ogni mese e per ogni utente quante categorie distinte ha acquistato, filtrando con il costrutto HAVING possiamo selezionare solo quei clienti che nello specifico mese hanno acquistato più di 10 categorie differenti. Per costruire la vista è necessario utilizzare delle join tra le tabelle sales\_fact, dove sono censite tutte le vendite effettuate e dunque i prodotti effettivamente venduti, la time\_by\_day, in cui è possibile selezionare il mese e la product\_class. Come già precedentemente osservato per poter selezionare le categorie associate ad un prodotto venduto è necessario partire dai product\_id presenti nella sales\_fact, fare una join con la tabella product a parità di product\_id, e attraverso questa fare una join con la tabella product\_class a parità di product\_class\_id e selezionare la categoria, *i.e.* il campo product\_category.

A partire dalla vista ‘view\_categorie\_distinte\_per\_mese’ viene effettuata una ulteriore GROUP BY a parità di mese per contare quanti clienti distinti sono associati ad ogni mese.

Si noti che nel conteggio dei ‘customer\_id’ della vista ‘view\_categorie\_distinte\_per\_mese’ non è stato inserito in DISTINCT in quanto si sta partendo da una vista costruita come GROUP BY a parità di mese e ‘customer\_id’, dunque non ci sono duplicati.

## Query

**with** view\_categorie\_distinte\_per\_mese **as** (

**select** **count**(**distinct** pc.product\_category) **as** categorie\_distinte\_per\_mese, tbd.the\_month, sf.customer\_id

**from** sales\_fact sf **join** product p

**on** p.product\_id =sf.product\_id

**join** product\_class pc **on** pc.product\_class\_id =p.product\_class\_id

**join** time\_by\_day tbd **on** tbd.time\_id=sf.time\_id

**group** **by** tbd.the\_month, sf.customer\_id

**having** **count**(**distinct** pc.product\_category)>=10)

**select** view\_categorie\_distinte\_per\_mese.the\_month, **count**( view\_categorie\_distinte\_per\_mese.customer\_id) **as** total\_customer\_at\_least\_10\_category

**from** view\_categorie\_distinte\_per\_mese

**group** **by** view\_categorie\_distinte\_per\_mese.the\_month

## Screen

Immagine che contiene testo, schermata, software, Pagina Web

Descrizione generata automaticamente

# Esercizio 10

## Testo

Given the year 1998, provide for each store and month the average gain with respect to the number of customers and the ration of that value with respect to yearly gain of that store. Thus, assuming that the average gain with respect to the number of customers of a store is the number K and that the yearly gain of that store is T, then the ratio is K/T.

## Interpretazione

Considerando l’anno 1998 calcolare per ogni mese e store il guadagno rispetto al numero di clienti e il ratio, K, di questo valore rispetto al guadagno annuo dello store, K/T.

Per avere le informazioni desiderate abbiamo inizialmente costruito la vista ‘sales\_98’ in cui sono presenti customer\_id, store\_id, mese, vendite e costi relativi agli acquisti censiti nel 1998.

Sulla base di questa vista sono state definite altre due viste: la ‘avg\_per\_store\_month’ in cui per ogni mese e per ogni store viene calcolato il guadagno mensile dello store, campo K, come store\_sales – store\_cost diviso per il numero di clienti distinti; la ‘avg\_per\_store’ in cui per ogni store viene calcolato il guadagno annuale, campo T, come store\_sales – store\_cost diviso per il numero di clienti distinti. In entrambi i casi viene utilizzato il costrutto GROUP BY.

Infine, le due viste ‘avg\_per\_store\_month’ e ‘avg\_per\_store’ vengono joinate a parità di mese, il risultato viene proiettato sui campi mese, store, T, K e il campo calcolato ‘ratio’, *i.e.* K/T.

## Query

**create** **view** sales\_98 **as**

**select** sf.customer\_id, sf.store\_id, tbd.the\_month, sf.store\_sales, sf.store\_cost

**from** sales\_fact sf

**join** time\_by\_day tbd **on** tbd.time\_id =sf.time\_id

**where** tbd.the\_year =1998

**with** avg\_per\_store\_month **as**(

---media rispetto ai clienti, per mese e store: totale guadagno/numero dei clienti

**select** sales\_98.the\_month, sales\_98.store\_id,(**sum**( sales\_98.store\_sales - sales\_98.store\_cost))/(**count**(**DISTINCT** sales\_98.customer\_id )) **as** K

**from** sales\_98

**group** **by** sales\_98.the\_month, sales\_98.store\_id),

avg\_per\_store **as** (

---media rispetto ai clienti, per store: totale guadagno/numero dei clienti

**select** sales\_98.store\_id, (**sum**( sales\_98.store\_sales-sales\_98.store\_cost))/(**count**(**DISTINCT** sales\_98.customer\_id )) **as** T

**from** sales\_98

**group** **by** sales\_98.store\_id

)

**select** avg\_per\_store\_month.the\_month, avg\_per\_store\_month.store\_id, avg\_per\_store\_month.K, avg\_per\_store.T,

(avg\_per\_store\_month.K)/(avg\_per\_store.T) **as** ratio

**from** avg\_per\_store\_month

**join** avg\_per\_store **on** avg\_per\_store.store\_id=avg\_per\_store\_month.store\_id

## Screen

Immagine che contiene testo, schermata, software, numero

Descrizione generata automaticamente