Domanda 1

Risposta non data

Punteggio max.: 1,50

Given the following data warehouse schema:

Items(itemID, name, type, category)
Time(timeID, date, month, semester, year)
Transactions(itemID, timeID, value)

where a type of item belongs to a specific category,
the following query:

```
SELECT 100*SUM(value)/SUM(SUM(value))
OVER (PARTITION BY category, month)
FROM Transactions TS, Time T, Items I
WHERE TS.itemID=I.itemID
and TS.timeID=T.timeID
GROUP BY type, category, month, year
```

- (a) computes the yearly percentage of value of each type with respect to the overall yearly value of the category
- (b) none of the other answers is correct
- (c) computes the monthly percentage of value of each category with respect to the overall monthly value of the type
- (d) computes the monthly percentage of value of each type with respect to the overall monthly value of the category
- (e) computes the yearly percentage of value of each category with respect to the overall yearly value of the type

Risposta errata.

La risposta corretta è: computes the monthly percentage of value of each type with respect to the overall monthly value of the category

Domanda 2

Risposta non data

Punteggio max.: 1,00

The NoSQL design pattern named “schema versioning” has the advantage of:

- (a) no downtime needed, control of schema migration, and reduced future technical debt
- (b) control of schema migration
- (c) reduced future technical debt
- (d) no downtime needed
- (e) none of the other answers is correct

Risposta errata.

La risposta corretta è: no downtime needed, control of schema migration, and reduced future technical debt

Domanda 3

Risposta non data

Punteggio max.: 1,00

The command

```
db.members.createIndex( { lastname: 1, firstname: 1 })
```

creates:

- (a) a single field index. The index contains references to documents sorted first by the values of the **lastname** field and, within each value of the **lastname** field, sorted by values of the **firstname** field.
- (b) a compound index. The index contains references to documents sorted first by the values of the **firstname** field and, within each value of the **firstname** field, sorted by values of the **lastname** field.
- (c) none of the other answers is correct
- (d) a compound index. The index contains references to documents sorted first by the values of the **lastname** field and, within each value of the **lastname** field, sorted by values of the **firstname** field.
- (e) a single field index. The index will contain references to documents sorted first by the values of the **firstname** field and, within each value of the **firstname** field, sorted by values of the **lastname** field.

Risposta errata.

La risposta corretta è: a compound index. The index contains references to documents sorted first by the values of the **lastname** field and, within each value of the **lastname** field, sorted by values of the **firstname** field.

Domanda 4

Risposta non data

Punteggio max.: 1,00

Which of the following answers is **NOT** a solution to the overplotting problem?

- (a) Use of small shapes
- (b) Use of transparent shapes
- (c) Use of jittering
- (d) Use of multiple shapes
- (e) Use of outlined shapes

Risposta errata.

La risposta corretta è: Use of multiple shapes

Domanda 5

Risposta non data

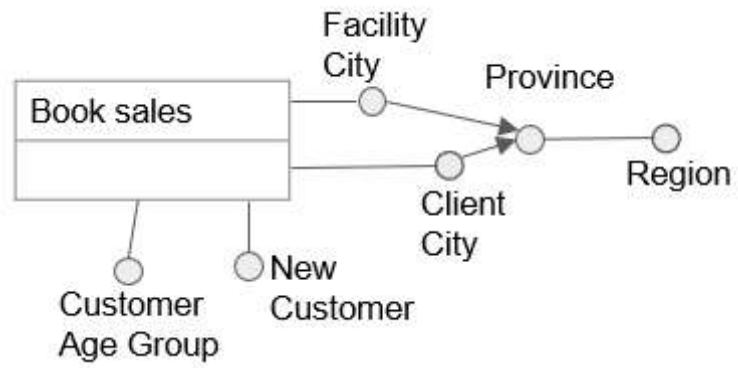
Punteggio max.: 0,50

An editorial group, operating an e-commerce website selling books, is interested in analizing the statistics of their revenue for each book sold.
The data analysts of the company would like to perform the analyses based on the following features.

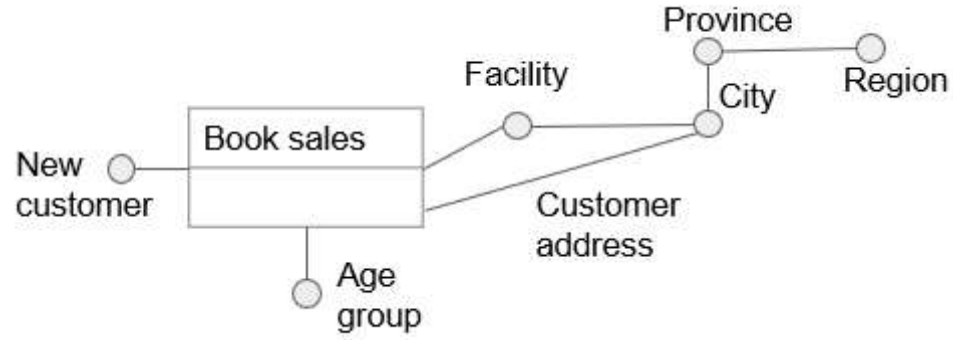
- The facilities from where the book is physically shipped, their region, province and city.
- The information about the kind of shipment: it can be either private home delivery or in-store pickup.
- The age group of the customer (<18, 18-35, 35-60, 60+), the status of the customer (it can be new customer or not new), and the customer address in terms of region, province and city.
- The book title (e.g., *The Hobbit*), the authors (e.g., *J. R. R. Tolkien*), the book genre, and the year of first publication.
  - Each book is associated to a genre, e.g., crime story, fantasy/sci-fi, historical, ecc.
  - There are 7 genres overall.
  - Suppose that the book title is unique.
- The date of the book sale, the month, bimester, trimester, semester, year, and if the day is a holiday or not.
- The data warehouse must be designed to efficiently analyze the average revenue per book.

Select, among the following dimensions, those that meet the requirements described in the problem specification (at most one answer is correct).

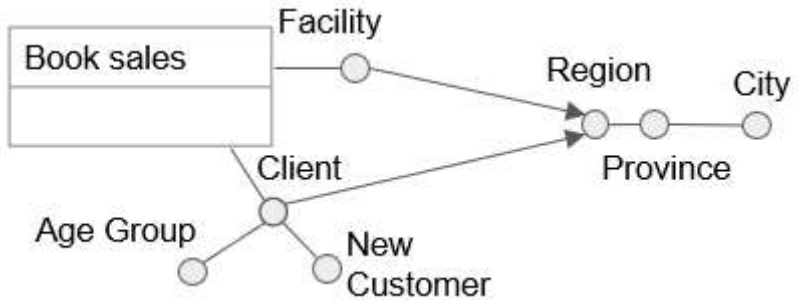
(a)



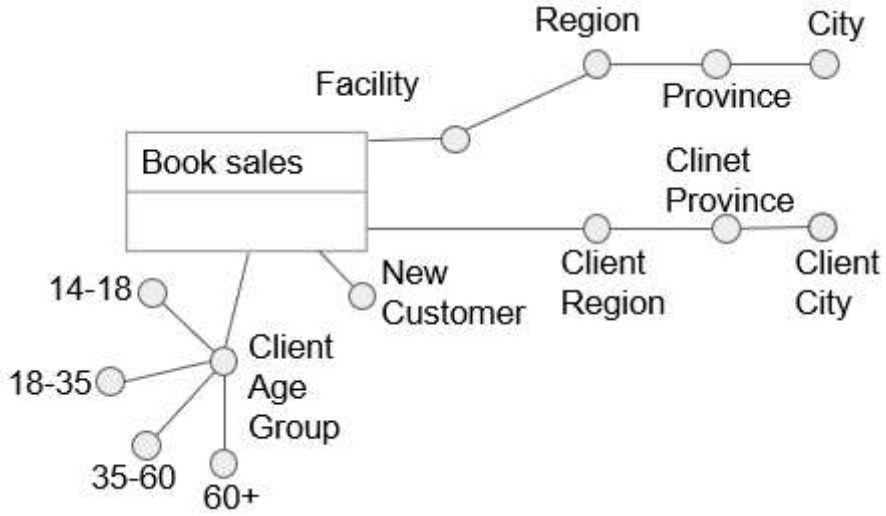
(b)



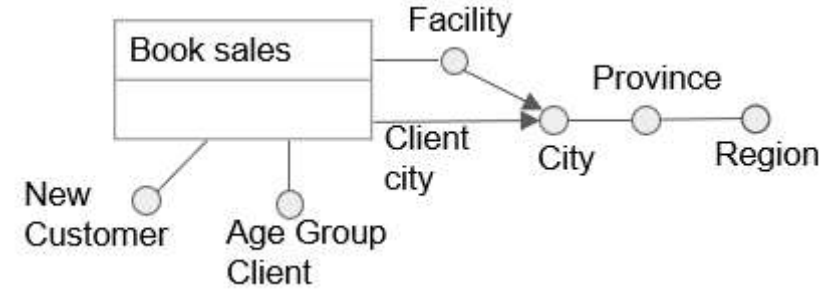
(c)


Book sales
Facility
Region
City
Province
Client
Age Group
New
Customer

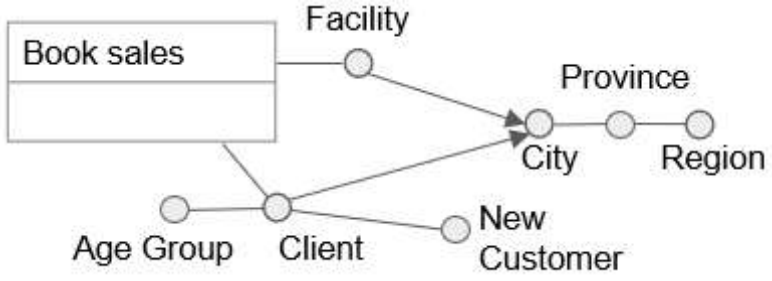
(d)


Region
City
Facility
Province
Book sales
Clinet
Province
Client
Region
Client
City
New
Customer
14-18
18-35
35-60
60+
Client
Age
Group

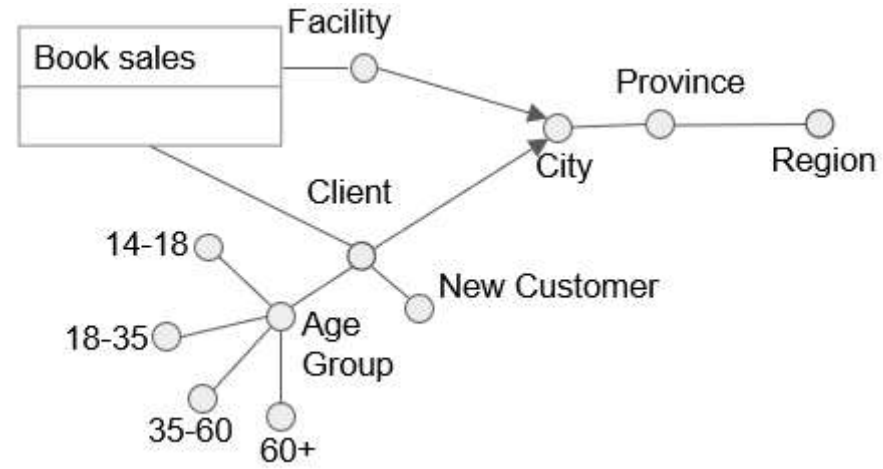
(e)


Book sales
Facility
Province
Client
city
City
Region
New
Customer
Age Group
Client

(f)

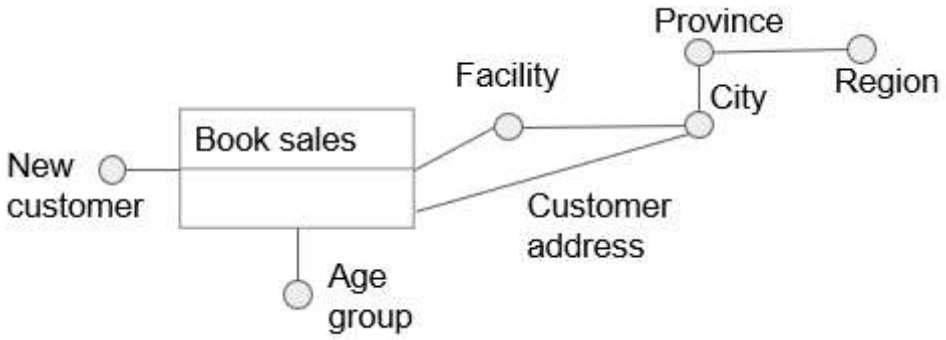

Facility
Book sales
Province
City
Region
Age Group
Client
New
Customer

(g)


Facility
Book sales
Province
City
Region
Client
14-18
New Customer
18-35
Age
Group
35-60
60+

Risposta errata.

La risposta corretta è:



Domanda 6

Risposta non data

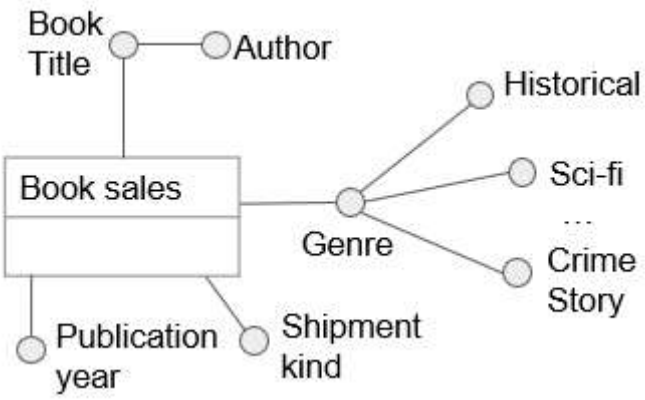
Punteggio max.: 0,50

An editorial group, operating an e-commerce website selling books, is interested in analizing the statistics of their revenue for each book sold.
The data analysts of the company would like to perform the analyses based on the following features.

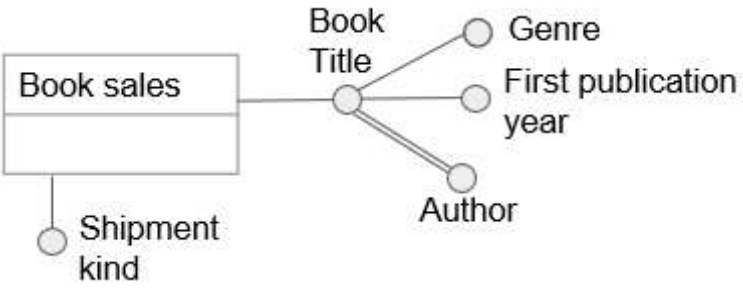
- The facilities from where the book is physically shipped, their region, province and city.
- The information about the kind of shipment: it can be either private home delivery or in-store pickup.
- The age group of the customer (<18, 18-35, 35-60, 60+), the status of the customer (it can be new customer or not new), and the customer address in terms of region, province and city.
- The book title (e.g., *The Hobbit*), the authors (e.g., *J. R. R. Tolkien*), the book genre, and the year of first publication.
  - Each book is associated to a genre, e.g., crime story, fantasy/sci-fi, historical, ecc.
  - There are 7 genres overall.
  - Suppose that the book title is unique.
- The date of the book sale, the month, bimester, trimester, semester, year, and if the day is a holiday or not.
- The data warehouse must be designed to efficiently analyze the average revenue per book.

Select, among the following dimensions, those that meet the requirements described in the problem specification (at most one answer is correct).

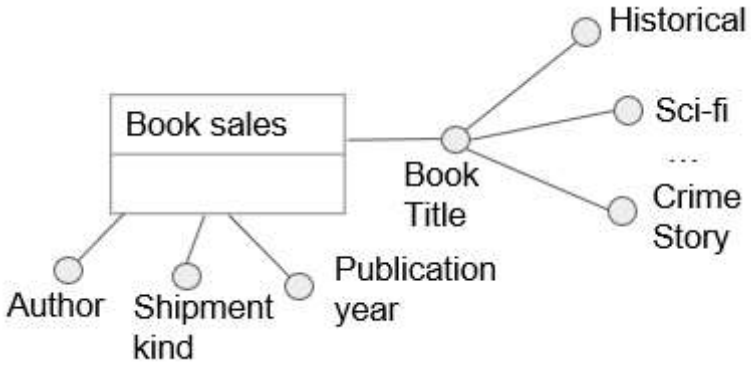
(a)



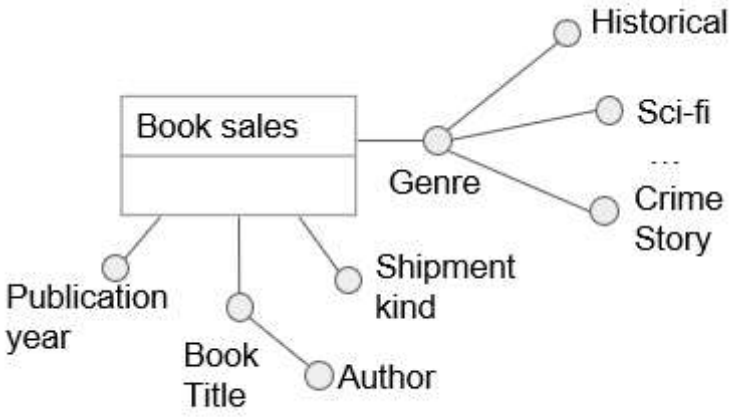
(b)



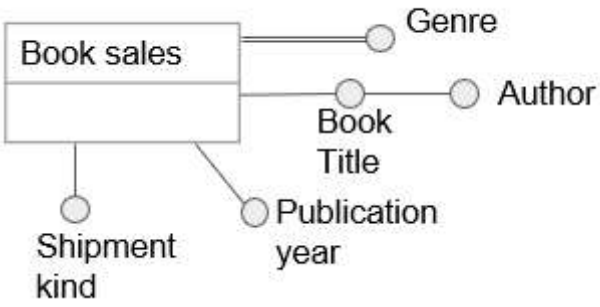
(c)


Book sales
Book Title
Historical
Sci-fi
...
Crime Story
Author
Shipment kind
Publication year

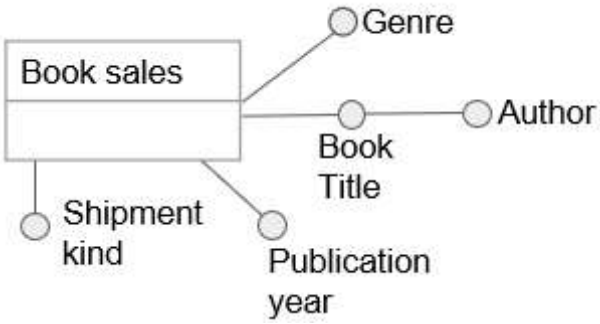
(d)


Book sales
Genre
Historical
Sci-fi
...
Crime Story
Publication year
Book Title
Author
Shipment kind

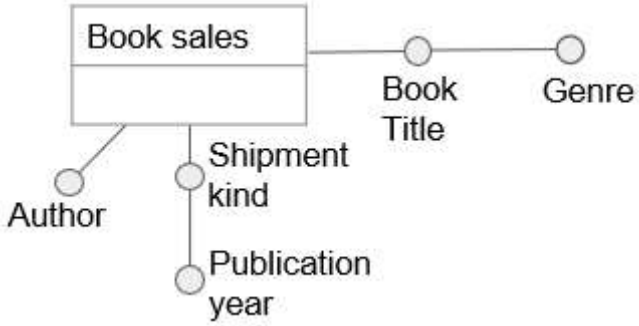
(e)


Book sales
Genre
Author
Book Title
Shipment kind
Publication year

(f)

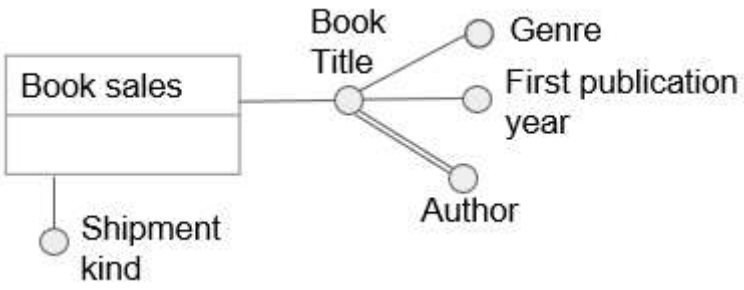

Book sales
Genre
Author
Book Title
Shipment kind
Publication year

(g)


Book sales
Book Title
Genre
Shipment kind
Author
Publication year

Risposta errata.

La risposta corretta è:



Domanda 7

Risposta non data

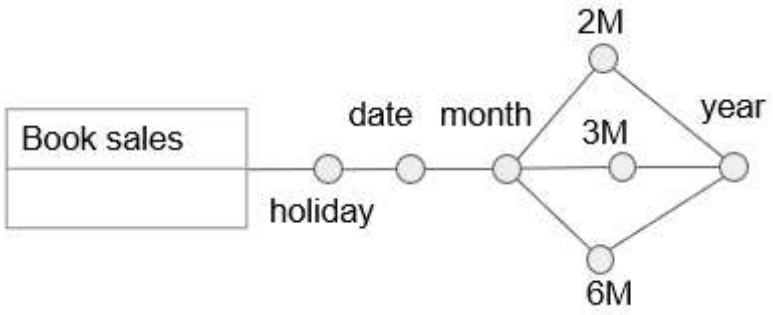
Punteggio max.: 0,50

An editorial group, operating an e-commerce website selling books, is interested in analizing the statistics of their revenue for each book sold.
The data analysts of the company would like to perform the analyses based on the following features.

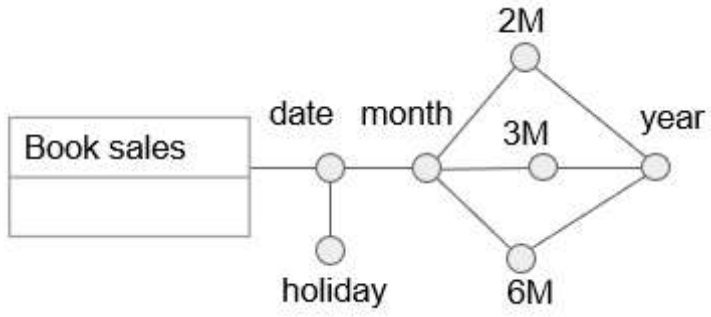
- The facilities from where the book is physically shipped, their region, province and city.
- The information about the kind of shipment: it can be either private home delivery or in-store pickup.
- The age group of the customer (<18, 18-35, 35-60, 60+), the status of the customer (it can be new customer or not new), and the customer address in terms of region, province and city.
- The book title (e.g., *The Hobbit*), the authors (e.g., *J. R. R. Tolkien*), the book genre, and the year of first publication.
  - Each book is associated to a genre, e.g., crime story, fantasy/sci-fi, historical, ecc.
  - There are 7 genres overall.
  - Suppose that the book title is unique.
- The date of the book sale, the month, bimester, trimester, semester, year, and if the day is a holiday or not.
- The data warehouse must be designed to efficiently analyze the average revenue per book.

Select, among the following dimensions, those that meet the requirements described in the problem specification (at most one answer is correct).

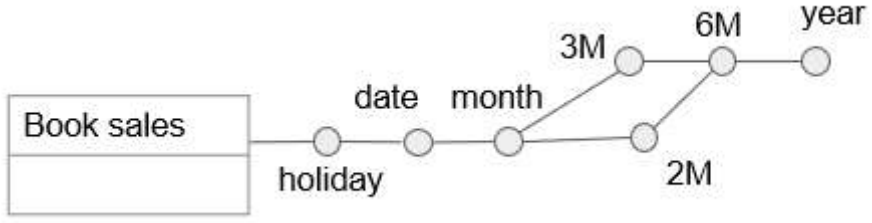
(a)



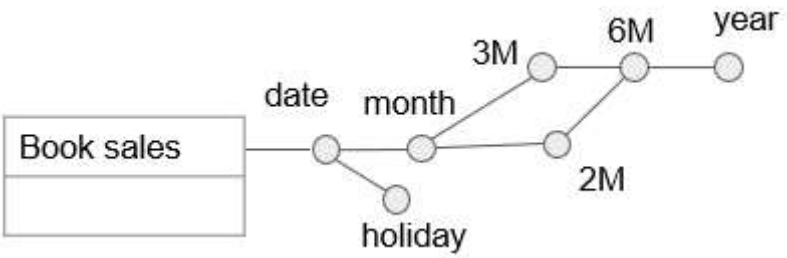
(b)



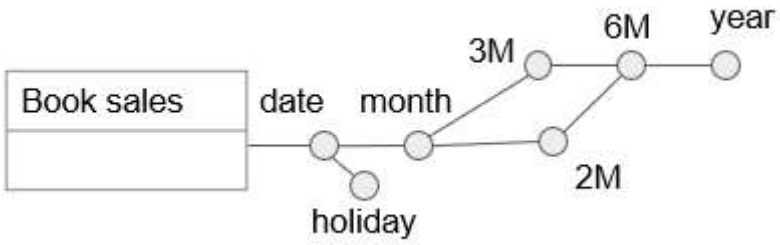
(c)



(d)

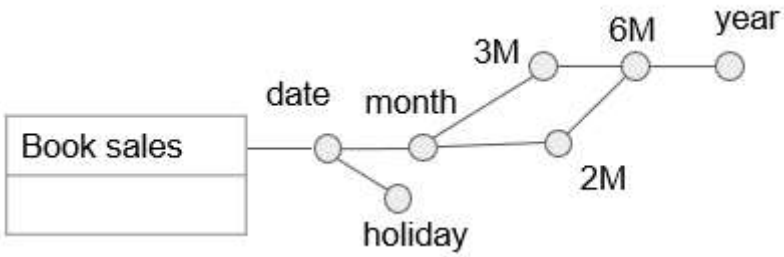


(e)



Risposta errata.

La risposta corretta è:



Domanda 8

Risposta non data

Punteggio max.: 1,50

An editorial group, operating an e-commerce website selling books, is interested in analizing the statistics of their revenue for each book sold.
The data analysts of the company would like to perform the analyses based on the following features.

- The facilities from where the book is physically shipped, their region, province and city.
- The information about the kind of shipment: it can be either private home delivery or in-store pickup.
- The age group of the customer (<18, 18-35, 35-60, 60+), the status of the customer (it can be new customer or not new), and the customer address in terms of region, province and city.
- The book title (e.g., *The Hobbit*), the authors (e.g., *J. R. R. Tolkien*), the book genre, and the year of first publication.
  - Each book is associated to a genre, e.g., crime story, fantasy/sci-fi, historical, ecc.
  - There are 7 genres overall.
  - Suppose that the book title is unique.
- The date of the book sale, the month, bimester, trimester, semester, year, and if the day is a holiday or not.
- The data warehouse must be designed to efficiently analyze the average revenue per book.

Scegli una o più alternative:

- [ ] (a) Average revenue per book (euros)
- [ ] (b) Total revenue per region (euros)
- [ ] (c) Total number of new Customers (count)
- [ ] (d) Total revenue of book sales (euros)
- [ ] (e) Total number of books sold (count)
- [ ] (f) Average monthly shipment cost per Customer (euros)
- [ ] (g) Total number of Customers (count)
- [ ] (h) Average monthly revenue per book title (euros)
- [ ] (i) Average number of shipments per establishment (count)

- [ ] (j)
  Total number of establishments per state (count)

- [ ] (k)
  Total number of books per author (count)

- [ ] (l)
  Average number of Customers per region (count)

- [ ] (m)
  Total number of shipments (count)

Risposta errata.

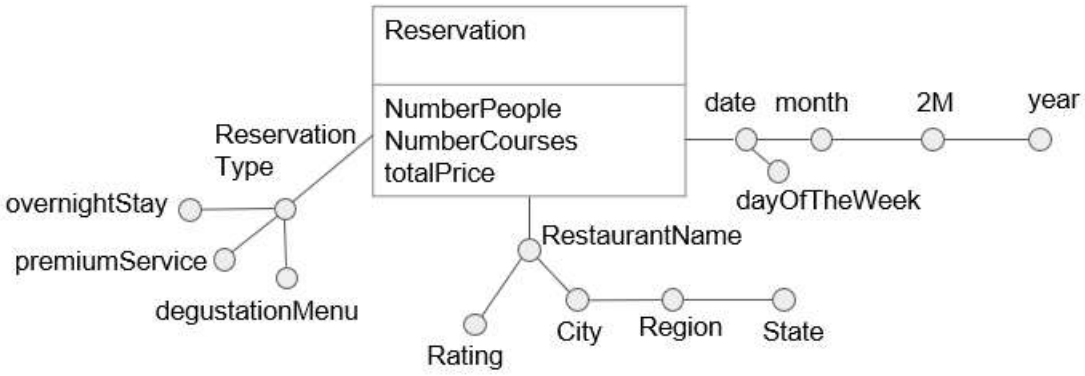
La risposta corretta è:
Total revenue of book sales (euros)
,
Total number of books sold (count)

Domanda 9

Risposta non data

Punteggio max.: 1,50

Given the following conceptual schema:



- Each restaurant has a unique name.
- Reservation types can have any number of additional services among the available 3
- Restaurant rating is an integer going from 1 to 5

Write the logical design of the conceptual DW schema indicated in the picture.
Write each table on a new line.
Use the **bold** or the underline for identifying primary-key attributes.

```
Reservations (TimeId, RestaurantId, ReservationId, NumberPeople,
NumberCourses, totalPrice)
Time(TimeId, date, month, 2M, year, dayOfTheWeek)
Restaurants(RestaurantId, RestaurantName, Rating, City, Region,
State)
Reservation(ReservationId,
ReservationType, overnightStay, premiumService, degustationMenu)
```

Domanda 10

Risposta non data

Punteggio max.: 4,00

ChocolateSupply (TimeId, QualityId, PackageType, RawQuantity, ProcessedQuantity, ProductionTime, SellingPrice)
Time(TimeId, date, month, 2M, 3M, 6M, year, dayOfTheWeek)
CocoaQuality(QualityId, QualityName, BuyingCost, Continent, State, Region, PlantationName)

The name of the Cocoa quality (QualityName) is unique.
Each plantation is within a specific continent, state and region.
Separately for each plantation and semester, compute the following metrics:

- the total raw quantity
- assign a rank to the plantation based on the total production time (rank 1st the fastest), separately for each semester and state
- assign a rank to the plantation based on the total processed quantity (rank 1st the largest), separately for each semester and state

Write the requested SQL query.

```
SELECT 6M, Plantation,
      SUM(RawQuantity),
      RANK() OVER (PARTITION BY State, 6M ORDER BY
SUM(ProductionTime)),
      RANK() OVER (PARTITION BY State, 6M ORDER BY
SUM(ProcessedQuantity) DESC),
FROM ChocolateSupply C, Time T, CocoaQuality Q
WHERE T.TimeId=C.TimeId and Q.QualityId=C.QualityId
GROUP BY 6M, Plantation, State
```

Domanda 11

Risposta non data

Punteggio max.: 4,00

ChocolateSupply (TimeId, QualityId, PackageType, RawQuantity, ProcessedQuantity, ProductionTime, SellingPrice)
Time(TimeId, date, month, 2M, 3M, 6M, year, dayOfTheWeek)
CocoaQuality(QualityId, QualityName, BuyingCost, Continent, State, Region, PlantationName)

The name of the Cocoa quality is unique.
PackageType has a cardinality of 2, then the system stores either 0 or 1 for the different kinds.
Considering only chocolate supplies of PackageType 0, separately for each region, and month, compute the following metrics:

- the percentage of Processed Quantity with respect to the original Raw Quantity
- the percentage of the monthly Processed Quantity for each region, with respect to the monthly Raw Quantity of the state
- the cumulative Raw Quantity since the beginning of the trimester (3M)

Write the requested SQL query.

```
Select Region, month,
     100 * SUM(ProcessedQuantity) / SUM(RawQuantity),
     100 * SUM(ProcessedQuantity) / SUM(SUM(ProcessedQuantity)) OVER
(PARTITION BY state, month),
     SUM(SUM(RawQuantity)) OVER (PARTITION BY 3M, region ORDER BY month
ROWS UNBOUNDED PRECEDING)
FROM ChocolateSupply C, Time T, CocoaQuality Q
WHERE T.TimeId=C.TimeId and Q.QualityId=C.QualityId and PackageType=0
GROUP BY region, month, state, 3M
```

Domanda 12

Risposta non data

Punteggio max.: 2,00

The following document structure represents a document of a set of monthly invoices of a telephone company. Each document collects some information about the customer and the monthly usage of the different communication services.

```
{
   "_id" : ObjectId("61fa5b8f6f631bb5339dc4b7"),
   "customer" : {
       "id" : ObjectId("32af5b7a6f133ca5133dc4c8"),
       "city" : "Turin",
       "region" : “Piedmont
   },
   "month" : 6,
   "year" : 2021,
   "total_cost_EUR" : 6.6,
   "usage" : [
       {service: "call_minutes", qty: 552},
       {service: "internet_GB", qty: 2.96},
       {service: "sms", qty: 15}
   ]
}
```

Write a MongoDB query to find all invoices of customers in the city of Turin with a usage of the call service less than or equal to 200 minutes.

```
db.invoice.find (
{"customer.city":"Turin", consumption: {$elemMatch: { 'service': "call_minutes", qty:{$lte: 20
0}}} }
})
```

Domanda 13

Risposta non data

Punteggio max.: 3,00

The following document structure represents a document of a set of monthly invoices of a telephone company. Each document collects some information about the customer and the monthly usage of the different communication services.

```
{
   "_id" : ObjectId("61fa5b8f6f631bb5339dc4b7"),
   "customer" : {
       "id" : ObjectId("32af5b7a6f133ca5133dc4c8"),
       "city" : "Turin",
       "region" : “Piedmont
   },
   "month" : 6,
   "year" : 2021,
   "total_cost_EUR" : 6.6,
   "usage" : [
       {service: "call_minutes", qty: 552},
       {service: "internet_GB", qty: 2.96},
       {service: "sms", qty: 15}
   ]
}
```

Considering only invoices earlier than year 2021 (excluding 2021), separately for each service and each customer city, select the maximum usage amount (“qty”). Within the same service, sort the results in descending order according to the maximum usage amount.

```
db.collection_name.aggregate([
        {$match: {
                “year”: {$lt: 2021}
        },
        { $unwind: ‘$usage’},
        { $group: {
                '_id': {
                        service: ‘$usage.service’,
                        city: ‘$customer.city},
                'max_usage':  {
                        '$max: '$usage.qty’}
                }
        },
        { $sort: { ‘_id.service’: 1, 'max_usage': -1} }
])
```

Domanda 14

Risposta non data

Punteggio max.: 4,00

Design a MongoDB database to store the daily exchange of stocks for a mobile app according to the following requirements.

Data to display for each stock include the symbol (e.g., ‘ABC’), the market (e.g., “NYSE”), a category and the currency of exchanges.

The exchanges are related to a stock and are characterized by a timestamp, the type (i.e., purchase or sell), the quantity, the price, and the identifier of the trader.

The application should efficiently retrieve all the daily exchanges of a given stock (identified by its symbol and its market).

Given a specific day, besides the list of all exchanges of the stock, some statistics should be readily presented. In particular, the total count of exchanges, the average price, and the average quantity, separately for sold and purchased stocks.

Given a stock option, the database must be designed to efficiently provide the last price.

Write a sample document for each collection of the database.
Important: besides the sample documents, explicitly indicate the design patterns used.

EXCHANGE

```
{
_id: ObjectId(),
stock: {
          _id: ObjectId(),
          symbol: <string>,
          market: <string>
}
day: <datetime>,
sold: {
       count: <number>,
          avg_qty: <number>,
          avg_price: <number>
          },
purchased: {
       count: <number>,
          avg_qty: <number>,
          avg_price: <number>
          },
events: [
          { ts: <datetime>,
            type: <string>,
            qty: <string>,
            price: <number>,
            trader: ObjectId()
       }
]
}
```

STOCK

```
{_id: ObjectId(),
symbol: <string>,
market: <string>,
category: <string>,
currency: <string>
last_price" : <number>,
}
}
```

Patterns used:

- **Bucket** pattern to track events
- **Computed** pattern for the sold and purchased stats in events collection, and stats attributes in stock collection.
- **Extended reference** for the events collection to show the stock information.