Data management and visualization

Iniziato	venerdì, 4 febbraio 2022, 19:41
Stato	Completato
Terminato	venerdì, 4 febbraio 2022, 19:41
Tempo impiegato	21 secondi
Valutazione	**0,00** su un massimo di 31,00 (**0**%)

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)
Sales(itemID, timeID, amount)

the following query:

```
SELECT RANK() OVER (PARTITION BY month ORDER BY SUM(amount)
DESC)
FROM Sales S, Time T, Items I
WHERE S.itemID=I.itemID
and S.timeID=T.timeID
GROUP BY Type, Month
```

- (a) none of the other answers is correct
- (b) is ranking the types of items
- (c) is ranking the items individually
- (d) is ranking the dates
- (e) is ranking the months

Risposta errata.

La risposta corretta è: is ranking the types of items

Domanda 2

Risposta non data

Punteggio max.: 1,00

In a replica set configuration the write acknowledgment is returned when

- (a) none of the other answers is correct
- (b) all the secondary and the primary nodes apply the write if w:”majority”
- (c) all the secondary and the primary nodes apply the write if w:1
- (d) the primary locally applies the write if w:1

Risposta errata.

La risposta corretta è: the primary locally applies the write if w:1

Domanda 3

Risposta non data

Punteggio max.: 1,00

The query

```
db.student.find( { "tests": { $elemMatch: { “score”: { $gt: 18, $lte: 24 } }   } } )
```

returns:

- (a) **all the documents** where the “tests” array has, as first element, a document that contains the field “score” greater than 18 and less than 24
- (b) **all the documents** where the “tests” array has at least one embedded document that contains the field “score” greater than 18 and less than or equal to 24
- (c) **one document** where the “tests” array has at least one embedded document that contains the field “score” greater than 18 and less than 24
- (d) none of the other answers is correct
- (e) **all the documents** where at least one embedded document in the “tests” array has the “score” field greater than 18 and at least one embedded document (but not necessarily the same one) in the “tests” array has the “score” field less than or equal to 24

Risposta errata.

La risposta corretta è: **all the documents** where the “tests” array has at least one embedded document that contains the field “score” greater than 18 and less than or equal to 24

Domanda 4

Risposta non data

Punteggio max.: 1,00

Which of the following is a valid reason to use graphs instead of tables?

- (a) To use very precise values
- (b) To look up individual values
- (c) To compare values over time
- (d) To reveal relationships among multiple values
- (e) To report more than one unit of measure

Risposta errata.

La risposta corretta è: To reveal relationships among multiple values

Domanda 5

Risposta non data

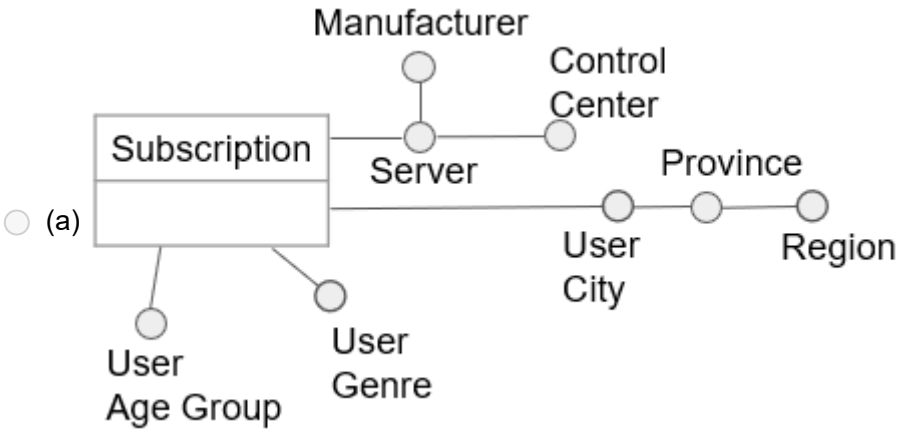
Punteggio max.: 0,50

A software company that rents servers to store data is interested in analyzing statistics about their revenue and user subscription.
The data analysts of the company would like to perform the analyses based on the following features.

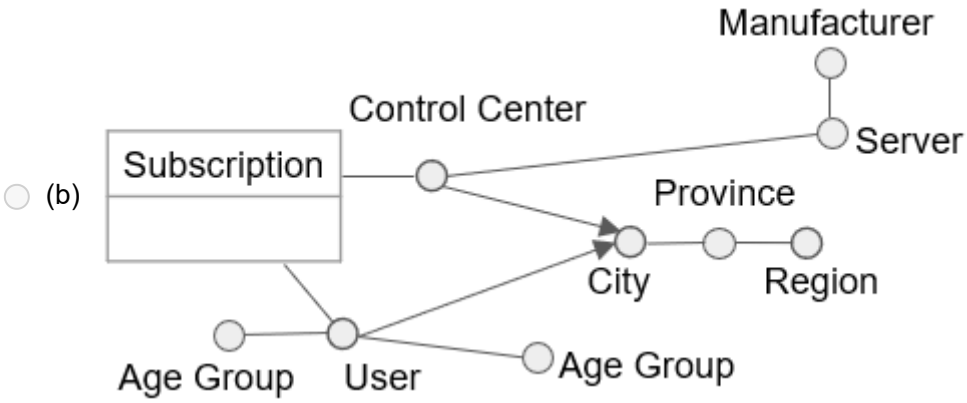
- The **server** in which the company stores users' data, the server manufacturer, and the server control center. Each server is associated with a specific control center. More servers can be associated with the same control center. Each server has only one manufacturer.
- The age group, the genre, and the billing address for each **user**.
  - The age groups are <18, 18-25, 25-35, 35-50, >50 years.
  - The information of the city, province, and region of the billing address of the user.
- The type and duration of the **subscribed plan** associated to each payment performed by a user.
  - The type of plan can be either basic, premium, or enterprise.
  - The duration of any plan can be: trial period, 1 month, 12 months, lifetime.
- Each subscription plan can be associated with different **additional services**, which span from assistance h24 to dedicated specific computational resources. The number of possible additional services is large and growing, hence the full list is not known in advance.
- The analysis must be carried out considering the date, month, 2 months, 4 months, semester and year of the first subscription, also considering the day of the week.
- The data warehouse must be designed to efficiently analyze:
  - The total revenue per subscription
  - The average data (in GB) stored per subscription

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

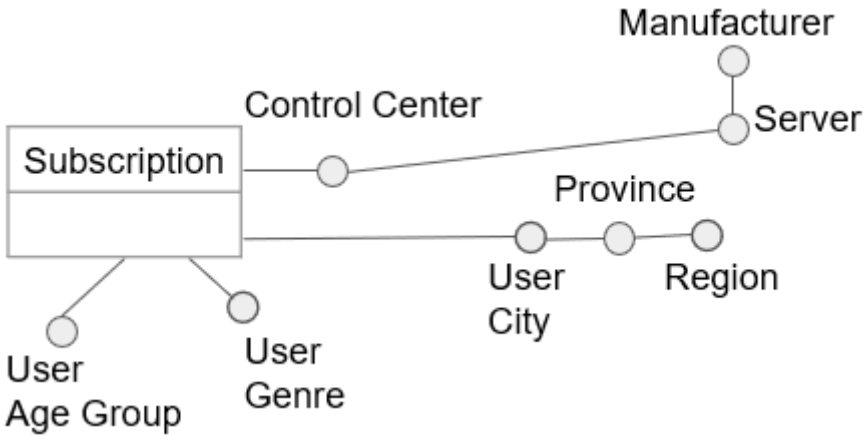
(a)



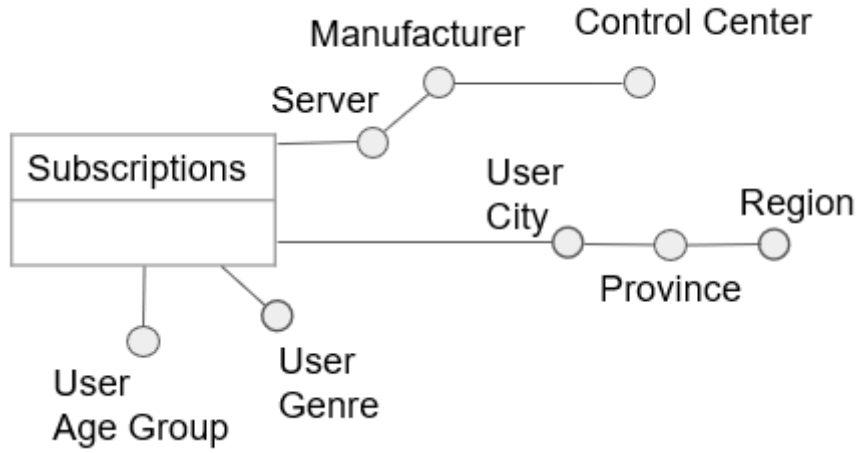
(b)



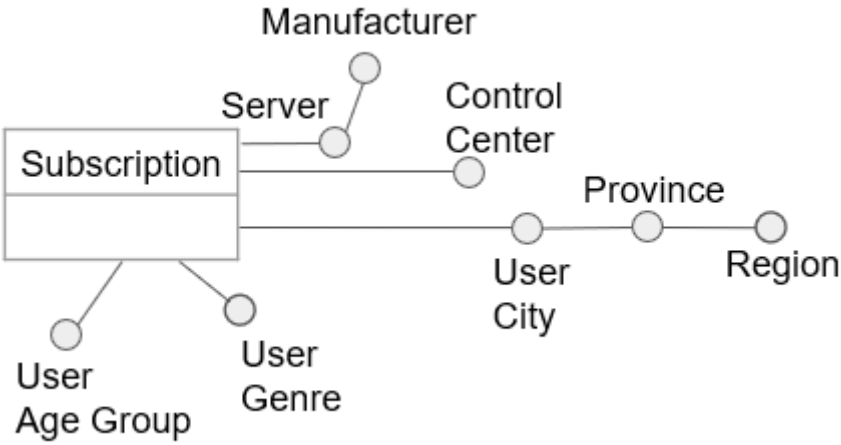
(c)


Manufacturer
Control Center
Server
Subscription
Province
User
City
Region
User
Age Group
User
Genre

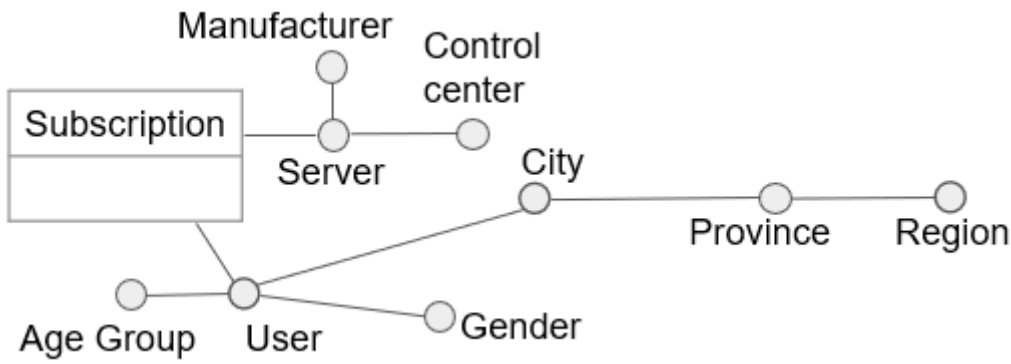
(d)


Manufacturer
Control Center
Server
Subscriptions
User
City
Region
Province
User
Age Group
User
Genre

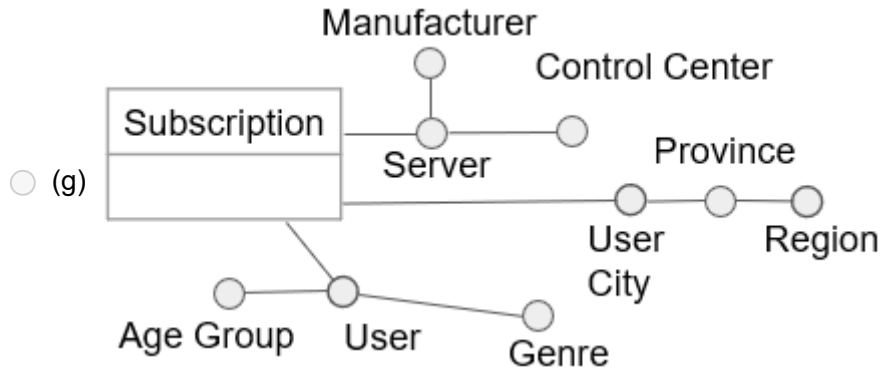
(e)


Manufacturer
Server
Control
Center
Subscription
Province
User
City
Region
User
Age Group
User
Genre

(f)

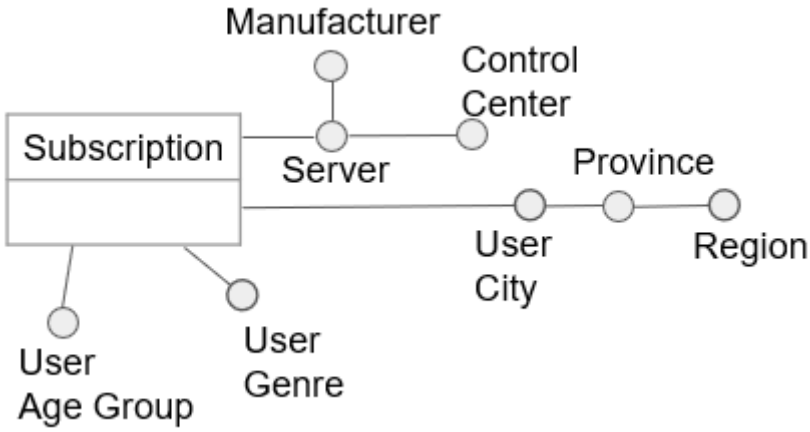

Manufacturer
Control
center
Subscription
Server
City
Province
Region
Age Group
User
Gender

(g)



Risposta errata.

La risposta corretta è:



Domanda 6

Risposta non data

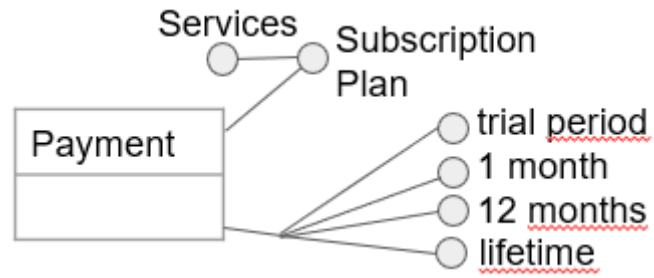
Punteggio max.: 0,50

A software company that rents servers to store data is interested in analyzing statistics about their revenue and user subscription.
The data analysts of the company would like to perform the analyses based on the following features.

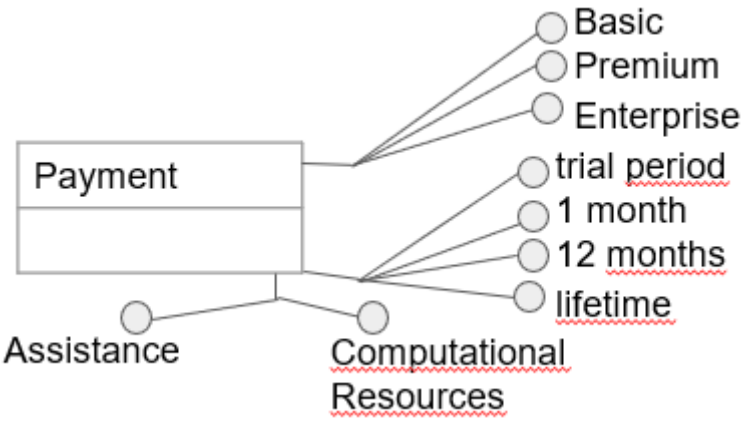
- The **server** in which the company stores users' data, the server manufacturer, and the server control center. Each server is associated with a specific control center. More servers can be associated with the same control center. Each server has only one manufacturer.
- The age group, the genre, and the billing address for each **user**.
  - The age groups are <18, 18-25, 25-35, 35-50, >50 years.
  - The information of the city, province, and region of the billing address of the user.
- The type and duration of the **subscribed plan** associated to each payment performed by a user.
  - The type of plan can be either basic, premium, or enterprise.
  - The duration of any plan can be: trial period, 1 month, 12 months, lifetime.
- Each subscription plan can be associated with different **additional services**, which span from assistance h24 to dedicated specific computational resources. The number of possible additional services is large and growing, hence the full list is not known in advance.
- The analysis must be carried out considering the date, month, 2 months, 4 months, semester and year of the first subscription, also considering the day of the week.
- The data warehouse must be designed to efficiently analyze:
  - The total revenue per subscription
  - The average data (in GB) stored per subscription

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

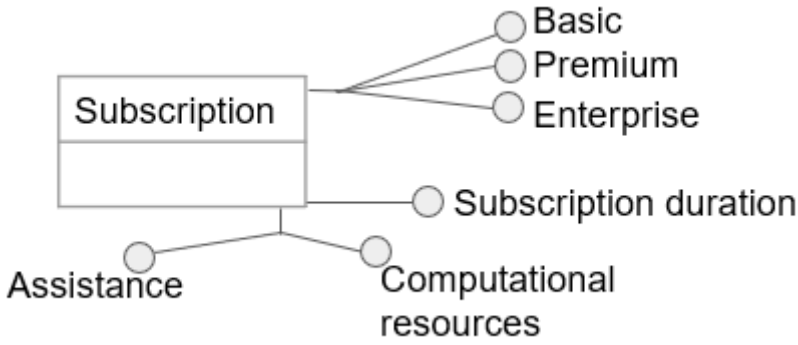
(a)



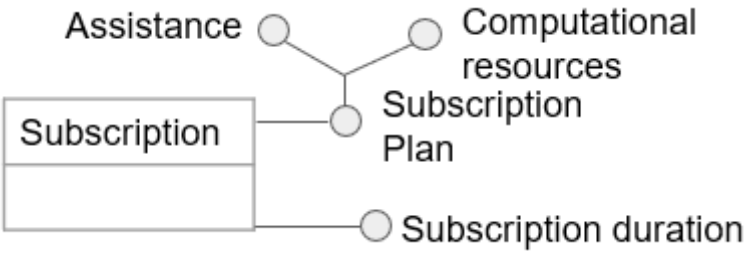
(b)



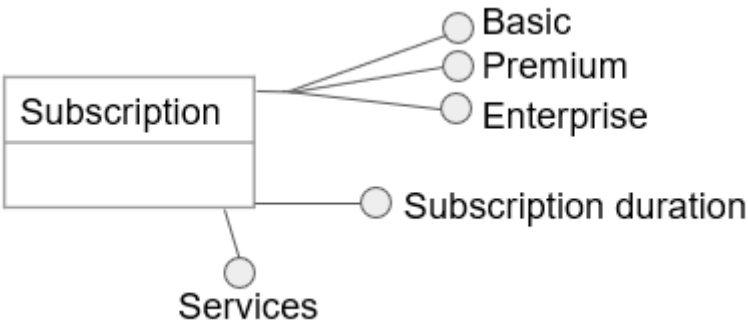
(c)


Basic
Premium
Enterprise
Subscription
Subscription duration
Assistance
Computational resources

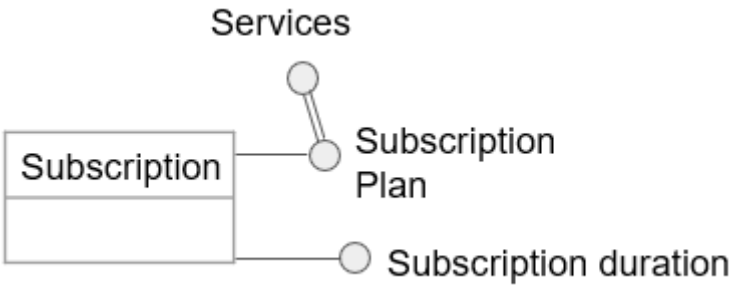
(d)


Assistance
Computational resources
Subscription Plan
Subscription
Subscription duration

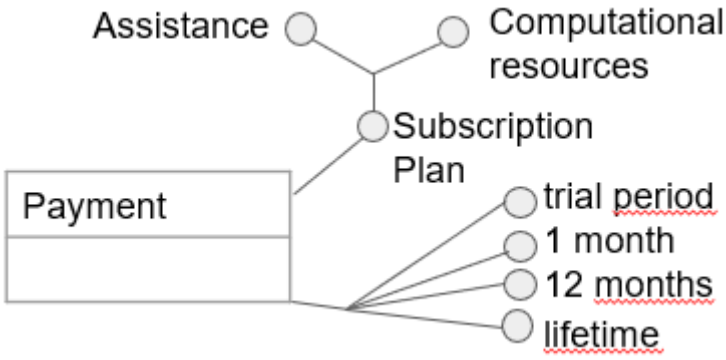
(e)


Basic
Premium
Enterprise
Subscription
Subscription duration
Services

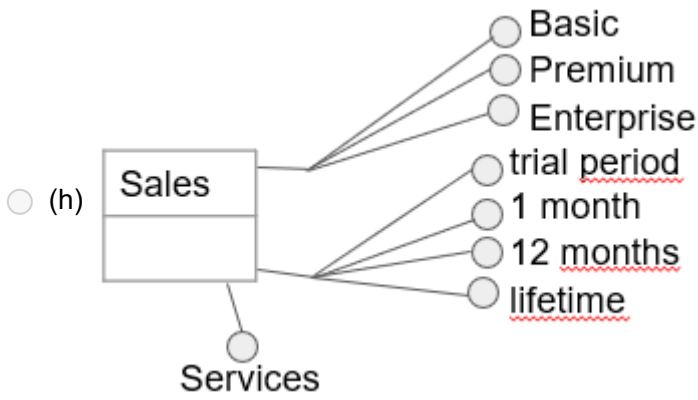
(f)


Services
Subscription Plan
Subscription
Subscription duration

(g)

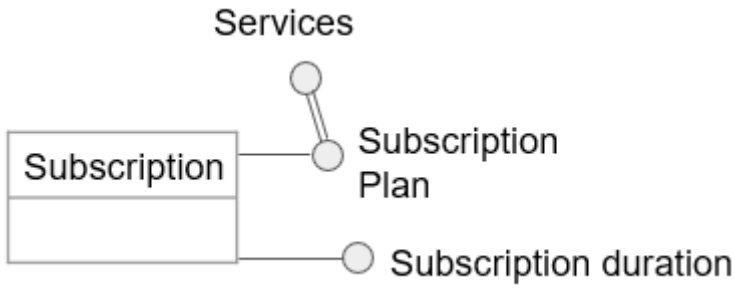

Assistance
Computational resources
Subscription Plan
Payment
trial period
1 month
12 months
lifetime

(h)



Risposta errata.

La risposta corretta è:



Domanda 7

Risposta non data

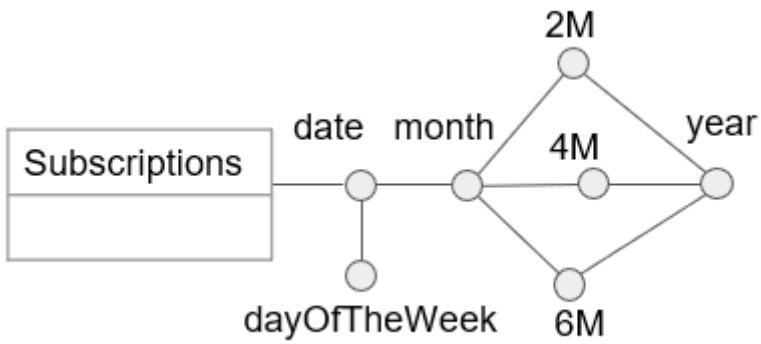
Punteggio max.: 0,50

A software company that rents servers to store data is interested in analyzing statistics about their revenue and user subscription.
The data analysts of the company would like to perform the analyses based on the following features.

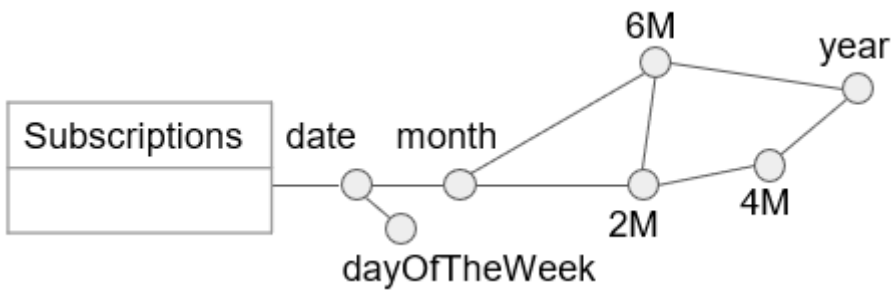
- The **server** in which the company stores users' data, the server manufacturer, and the server control center. Each server is associated with a specific control center. More servers can be associated with the same control center. Each server has only one manufacturer.
- The age group, the genre, and the billing address for each **user**.
  - The age groups are <18, 18-25, 25-35, 35-50, >50 years.
  - The information of the city, province, and region of the billing address of the user.
- The type and duration of the **subscribed plan** associated to each payment performed by a user.
  - The type of plan can be either basic, premium, or enterprise.
  - The duration of any plan can be: trial period, 1 month, 12 months, lifetime.
- Each subscription plan can be associated with different **additional services**, which span from assistance h24 to dedicated specific computational resources. The number of possible additional services is large and growing, hence the full list is not known in advance.
- The analysis must be carried out considering the date, month, 2 months, 4 months, semester and year of the first subscription, also considering the day of the week.
- The data warehouse must be designed to efficiently analyze:
  - The total revenue per subscription
  - The average data (in GB) stored per subscription

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

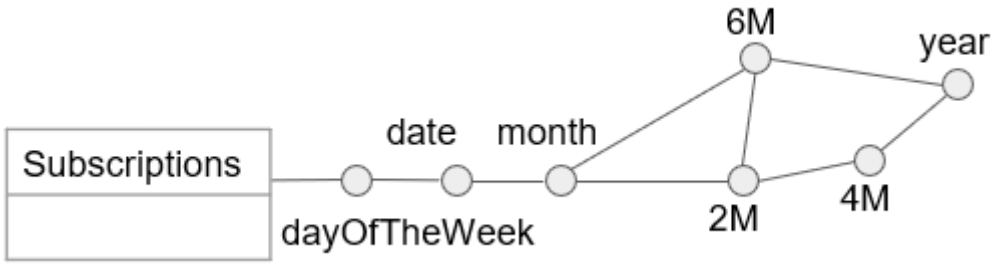
(a)



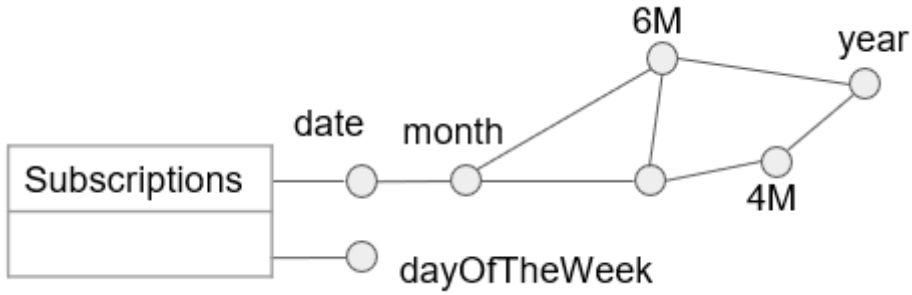
(b)



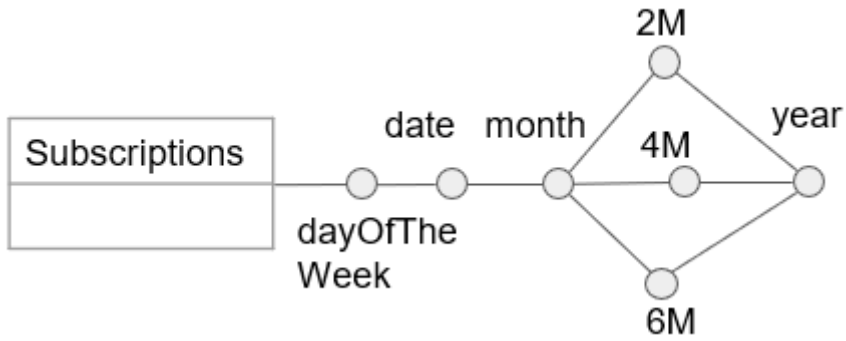
(c)



(d)

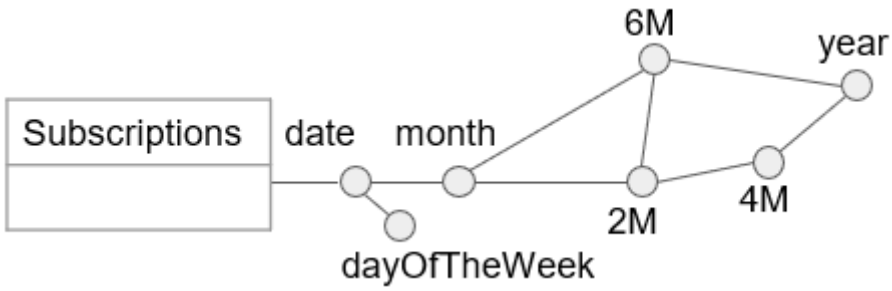


(e)



Risposta errata.

La risposta corretta è:



Domanda 8

Risposta non data

Punteggio max.: 1,50

A software company that rents servers to store data is interested in analyzing statistics about their revenue and user subscription.
The data analysts of the company would like to perform the analyses based on the following features.

- The **server** in which the company stores users' data, the server manufacturer, and the server control center. Each server is associated to a specific control center. More servers can be associated with the same control center. Each server has only one manufacturer.
- The age group, the genre, and the billing address for each **user**.
  - The age groups are <18, 18-25, 25-35, 35-50, >50 years.
  - The information of the city, province, and region of the billing address of the user.
- The type and duration of the **subscribed plan** associated to each payment performed by a user.
  - The type of plan can be either basic, premium, or enterprise.
  - The duration of any plan can be: trial period, 1 month, 12 months, lifetime.
- Each subscription plan can be associated with different **additional services**, which span from assistance h24 to dedicated specific computational resources. The number of possible additional services is large and growing, hence the full list is not known in advance.
- The data warehouse must be designed to efficiently analyze:
  - The total revenue per subscription
  - The average data (in GB) stored per subscription

Select all and only the required measures of the fact table in the conceptual schema design among the following (multiple choice question). Hint: do consider the dimensions defined by the previous answers.

Scegli una o più alternative:

- [ ] (a)
  Total number of services (count)

- [ ] (b)
  Total number of users per server (count)

- [ ] (c)
  Total stored data of all subscriptions (GB)

- [ ] (d)
  Total revenue of all subscriptions (euros)

- [ ] (e)
  Average number of users per server (count)

- [ ] (f)
  Average number of servers per control centers (count)

- [ ] (g)
  Average monthly subscription bill per user (euros)

- [ ] (h)
  Average monthly stored data per subscription (GB)

- [ ] (i)
  Total number of servers (count)

- [ ] (j)
  Total revenue per server (euros)

- [ ] (k)
  Total number of subscriptions (count)

- [ ] (l)
  Average number of users per subscription plan (count)

- [ ] (m)
  Average number of users per server (count)

- [ ] (n)
  Average revenue per server (euros)

Risposta errata.

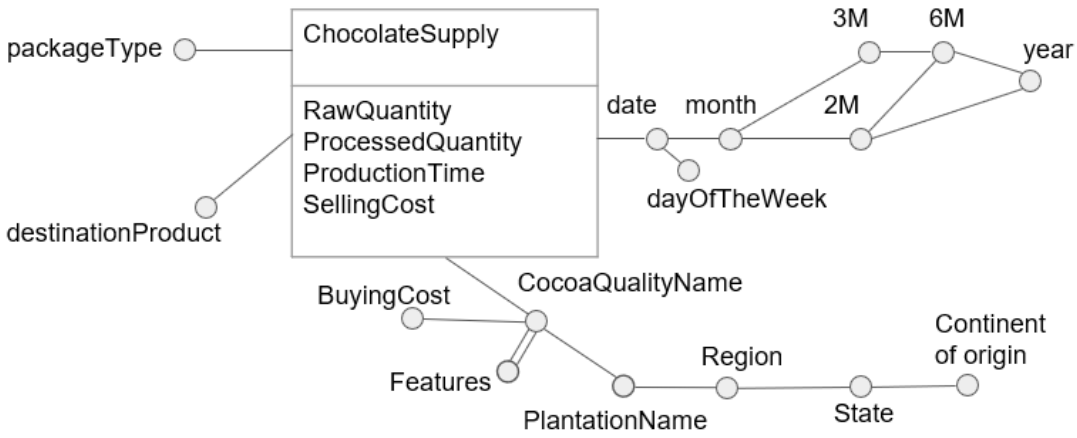
La risposta corretta è:
Total number of subscriptions (count)
,
Total revenue of all subscriptions (euros)
,
Total stored data of all subscriptions (GB)

Domanda 9

Risposta non data

Punteggio max.: 1,50

Given the following conceptual schema:



- Each cocoa quality has a unique name and can have one or more features.
- The system stores the final type of product in the destinationProduct field (i.e. dark chocolate, cinnamon-flavoured, etc.)
- PackageType has a cardinality of 2. The system stores with the integer 0 the bar types and with 1 the truffle package type.

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.

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

Domanda 10

Risposta non data

Punteggio max.: 4,00

```
Bike(CodM, BikeModel, Producer, ChildrenBike)
RentingPoint(CodP, Name, City, Region)
Time(CodT, Date, dayOfTheWeek, holiday, month, 2M, 3M, 4M, year)
Fact(CodM, CodP, CodT, total_rentings, total_rented_bikes,
total_income)
```

The name of the renting point is unique. The ChildrenBike is True if the bike is for children, False otherwise.
Separately for each Renting Point and trimester, compute:

- the average number of rented bikes per renting
- the cumulative total number of rentings since the beginning of the year
- for each region, assign a rank to the renting points based on the total rentings (rank 1st the highest number), separately for each trimester

Write the requested SQL query.

```
SELECT 3M, Name
      SUM(total_rented_bikes) / SUM(total_rentings),
      SUM(SUM(total_rentings)) OVER (PARTITION BY R.Name, year ORDER
BY 3M ROWS UNBOUNDED PRECEDING),
      RANK() OVER (PARTITION BY region, 3M ORDER BY
SUM(total_rentings) DESC),
FROM Fact F, Time T, Bike B, RentingPoint R
WHERE T.CodT=F.CodT and B.CodB=F.CodB and R.CodP=F.CodP
GROUP BY 3M, R.Name, R.Region, year
```

Alternative solution:

```
[…]
GROUP BY 3M, R.CodP, R.Region, year
```

Domanda 11

Risposta non data

Punteggio max.: 4,00

```
Bike(CodM, BikeModel, Producer, ChildrenBike)
RentingPoint(CodP, Name, City, Region)
Time(CodT, Date, dayOfTheWeek, holiday, month, 2M, 3M, 4M, year)
Fact(CodM, CodP, CodT, total_rentings, total_rented_bikes,
total_income)
```

The name of the renting point is unique. The ChildrenBike is True if the bike is for children, False otherwise.
Separately for each city, producer, and month, compute:

- the percentage of incomes in each month and city of each producer, with respect to the producer yearly total for the city
- the average income per rented bike
- for each city, assign a rank to the producer based on the income (rank 1st the highest income), separately for each month

Write the requested SQL query.

```
Select city, B.Producer, month,
     100 * SUM(total_income) / SUM(SUM(total_income)) OVER
(PARTITION BY producer, city, year),
     SUM(total_income)/SUM(total_rented_bikes),
     RANK() OVER (PARTITION BY city, month ORDER BY
SUM(total_income) DESC),
FROM Fact F, Time T, Bike B, RentingPoint R

WHERE T.CodT=F.CodT and B.CodB=F.CodB and R.CodP=F.CodP

GROUP BY month, year, City, B.Producer
```

Domanda 12

Risposta non data

Punteggio max.: 2,00

The following document structure represents a document of a set of events measured by a sensor. Each document collects all the measurements of a sensor for a given date.

```
{"_id": ObjectId(“xyz”),
"sensor":{
        “id”: 1,
        “location”:{
            “building”: “A”
            “floor”: 1,
            “type”: “bedroom”
        }
    },
"start”: Date("2022-01-15T00:00:00.000Z"),
“measures : [
        {“ts”: Date("2022-01-15T01:59:59.000Z"), “temperature”: 21},
        {“ts”: Date("2022-01-15T23:59:59.000Z"), “temperature”: 21.5}
    ],
“n”: 2,
“sum_temp”: 42.5
}
```

Write a MongoDB query to insert a new measure for the sensor 5 acquired on 2021-12-01 at 08:00 with a temperature equal to 19.

The document to be updated is related to the sensor 5 and to the set of measures of the day 2021-12-01 (start attribute).

Increase also the statistics (i.e., attributes “n” and “sum_temp”) stored in the document, which describe the number of measurements and the sum of the temperatures.

N.B. Use the syntax new Date (string) to manage date attributes, e.g., "attribute": new Date("2022-01-28")

```
db.timeserie.updateOne (
{'sensor.id': 5,
'start': new Date(2021-12-01T00:00:00.000Z) },
{'$push': {'measures':{ts:  new Date("2021-12-01T08:00:00.000Z"), “temperature”: 19 } },
 '$inc':{'n':1, ‘sum_temp’: 19},
})
```

Domanda 13

Risposta non data

Punteggio max.: 3,00

The following document structure represents a document of a set of events measured by a sensor. Each document collects all the measurements of a sensor for a given date.

```
{"_id": ObjectId(“xyz”),
"sensor":{
        “id”: 1,
        “location”:{
            “building”: “A”
            “floor”: 1,
            “type”: “bedroom”
        }
    },
"start”: Date("2022-01-15T00:00:00.000Z"),
“measures : [
        {“ts”: Date("2022-01-15T01:59:59.000Z"), “temperature”: 21},
        {“ts”: Date("2022-01-15T23:59:59.000Z"), “temperature”: 21.5}
    ],
“n”: 2,
“sum_temp”: 42.5
}
```

Considering only measurements acquired in June 2021 by sensors located at floor 2, for each building, select the average and the maximum temperature.

N.B. Use the syntax new Date (string) to manage date attributes, e.g., "attribute": new Date("2022-01-28")

```
db.collection_name.aggregate([
{$match: {
    “sensor.location.floor”: 2,
    “start”: {
        $gte: new Date(‘2021-06-01’),
        $lte: new Date('2021-06-30')}
    }
},
{ $unwind: ‘$measures’},
{$group: {
    '_id': ‘$sensor.location.building’,
    'avg_temp':  {'$avg': '$measures.temperature'},
    'max_temp':  {'$max': '$measures.temperature'}
    }
}
])
```

Domanda 14

Risposta non data

Punteggio max.: 4,00

Design a MongoDB database to store offers of merchants for a mobile app according to the following requirements.

Data to display for each merchant includes the merchant's name, type of merchant (e.g., restaurant, hotel, supermarket), a textual description, the venue, the average rating given by customers, and the contact information. The venue consists of the full address, the city, ZIP code, and country.

Contact information includes the phone number and email address. The official website address and the Facebook page might be included in the contact information.

Each offer consists of a title, a textual description, a list of categories (e.g., wellness, food, wine), a price in euros, and the validity period (i.e., start and end dates). Each offer is related to a specific merchant. A merchant can have many offers.

Given a merchant, the database must be designed to efficiently provide all the data describing the merchant, the number of available offers, and their price range (i.e., min and max prices). Instead, given an offer, the database must efficiently provide the merchant name, the merchant type and its venue information.

Write a sample document for each collection of the database.

Important: besides the sample documents, explicitly indicate the design patterns used.

MERCHANT

```
{
   _id: ObjectId(),
   name: <string>,
   type: <string>,
   description: <string>,
   rating: <number>,
   venue: {
      address: <url>,
      city: <string>
      zipcode: <number>,
      country: <string>
   },
   contact: {
      phone: <string>,
      email: <string>,
      website: <url>
      facebook: <string>
   },
   available_offers: <number>,
   price_range:{
      min: <number>,
      max: <number>
   }
}
```

OFFER

```
{   _id: ObjectId(),
    title: <string>,
    description: <string>,
    category: [<string>],
    price: <number>,
    start: <date>,
    end: <date>,
    merchant: {
        _id: ObjectId(),
        name: <string>,
        type: <string>,
        venue: {
            address: <url>,
            city: <string>,
            zipcode: <number>,
            country: <string>
        }
    }
}
```

Patterns used:

- Polymorphic pattern to track the contact information in the merchant collection (due to the optional website and facebook info).
- Computed pattern for the total number of offers and the price range available for each merchant.
- Extended reference for the offers collection to show the merchant info.

Domanda 15

Risposta non data

Punteggio max.: 0,25

Question

Is there a clearly defined question addressed by the visualization? Write it down.

Domanda 16

Risposta non data

Punteggio max.: 1,25

Data

Is the data quality appropriate? Identify the inadequate characteristics and explain.

Domanda 17

Risposta non data

Punteggio max.: 0,75

Visual Proportionality

Are the values encoded in a uniformly proportional way?

Domanda 18

Risposta non data

Punteggio max.: 0,75

Visual Utility

All the elements in the graph convey useful information?

Domanda 19

Risposta non data

Punteggio max.: 0,50

Visual Clarity

Are the data in the graph clearly identifiable and understandable (properly described)?

Domanda 20

Risposta non data

Punteggio max.: 0,25

Design data

Design the visualization based on the following data structure (to be completed).

Domanda 21

Risposta non data

Punteggio max.: 1,25

Design schema & Sketch

Fill in the required schema elements; formulas can be used if required. Then describe in words the design proposal.

Domanda 22

Risposta non data

Non valutata

This is a blank question to be used as your personal notepad during the exam.

Anything written here will NOT be evaluated.
