

Data Science e Tecnologie per le Basi di Dati

Iniziato sabato, 9 settembre 2023, 08:56

Stato Completato

Terminato sabato, 9 settembre 2023, 08:56

Tempo impiegato 12 secondi

Valutazione 0,00 su un massimo di 31,00 (0%)

Domanda 1

Risposta non data

Punteggio max.:
1,00

Un albero decisionale viene addestrato su di un training set contenente 1000 record. I dati presentano 20 attributi, mentre il numero di classi da riconoscere è pari a 3. Il test set ha dimensione 100 record. Quale delle seguenti asserzioni è corretta?

- (a) Nessuna delle asserzioni è corretta.
- (b) Il numero massimo di foglie dell'albero per ottenere un'accuratezza pari al 100% sul training set è 3.
- (c) L'algoritmo di Hunt prevede che l'albero abbia al massimo 3 foglie per poter evitare un fenomeno di overfitting sul test set proposto.
- (d) Il numero minimo di rami dell'albero per ottenere un'accuratezza pari al 100% sul training set è 20.
- (e) L'algoritmo di Hunt prevede che l'albero generato abbia almeno 3 foglie per poter ottenere la massima accuratezza sul test set proposto.
- (f) Se l'algoritmo di training ottenessse il 100% di accuratezza sul training set, questa configurazione garantirebbe la massima accuratezza ottenibile sul test set, seppur minore di quella sul training set.

Risposta errata.

La risposta corretta è: Nessuna delle asserzioni è corretta.

Domanda 2

Risposta non data

Punteggio max.: 1,00

L'algoritmo di clustering KMeans viene applicato ad un dataset con 30 attributi e 1000 record, utilizzando la distanza euclidea. Il valore di K scelto per l'algoritmo è 4. Quale delle seguenti asserzioni è corretta?

- (a) L'algoritmo calcola 1000×1000 distanze ad ogni iterazione.
- (b) L'algoritmo calcola 30×1000 distanze rispetto ad ogni centroide ad ogni iterazione.
- (c) Nessuna delle asserzioni è corretta.
- (d) L'algoritmo calcola 4×1000 distanze ad ogni iterazione.
- (e) Il numero massimo di iterazioni dell'algoritmo è pari a 1000×4 .
- (f) Il numero massimo di iterazioni dell'algoritmo è pari a 1000.

Risposta errata.

La risposta corretta è: L'algoritmo calcola 4×1000 distanze ad ogni iterazione.

Domanda 3

Risposta non data

Punteggio max.: 1,00

Un DBMS gestisce tre risorse A, B, C mediante locking gerarchico. B, C sono nodi figli di A. Il nodo A si trova in IXL, B si trova in XL, mentre C è libero. Quale delle seguenti affermazioni è corretta?

- (a) E' possibile richiedere una ISL su A, per poi richiedere una SL su B.
- (b) Nessuna delle affermazioni è corretta.
- (c) E' possibile richiedere una XL su A, per poi richiedere una XL su C.
- (d) E' possibile richiedere una IXL su A, per poi richiedere una IXL su C.
- (e) E' possibile richiedere una SL su A, per poi richiedere una SL su C.
- (f) E' possibile richiedere una ISL su A, per poi richiedere una IXL su C.

Risposta errata.

La risposta corretta è: E' possibile richiedere una IXL su A, per poi richiedere una IXL su C.

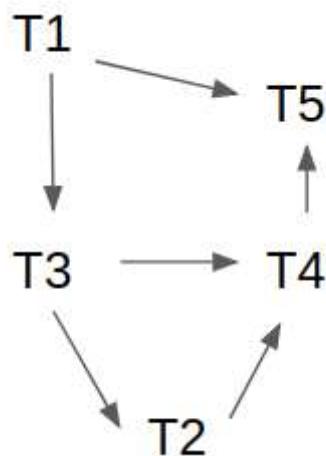
Domanda 4

Risposta non data

Punteggio max.:

1,00

Il sistema di scheduling di un DBMS genera il grafo dei conflitti in figura.

**Notazione:** T_n = Id della transazione n $w_n(x)$ = write eseguita dalla transazione n sull'oggetto x $r_n(x)$ = read eseguita dalla transazione n sull'oggetto x $\dots w_1(x) r_2(y) r_1(x) \dots$ = frammento di schedule in cui si susseguono w_1 , r_2 , r_1 sugli oggetti x ed y

Quale delle seguenti affermazioni è corretta?

- (a) Lo scheduling è CSR; uno schedule contenente il frammento ... $w_1(x) w_3(x) r_2(x) \dots$ può aver generato il grafo in figura.
- (b) Lo scheduling è VSR; uno schedule contenente il frammento ... $w_5(x) w_4(x) r_2(x) \dots$ può aver generato il grafo in figura.
- (c) Lo scheduling non è CSR; uno schedule contenente il frammento ... $r_1(x) r_3(x) w_2(x) \dots$ può aver generato il grafo in figura.
- (d) Lo scheduling è CSR; uno schedule contenente il frammento ... $r_1(x) w_2(y) r_3(y) w_5(x) \dots$ può aver generato il grafo in figura.
- (e) Lo scheduling è CSR; uno schedule contenente il frammento ... $r_3(x) w_2(x) w_4(x) \dots$ non può aver generato il grafo in figura.
- (f) Nessuna delle affermazioni è corretta.

Risposta errata.

La risposta corretta è: Lo scheduling è CSR; uno schedule contenente il frammento ... $w_1(x) w_3(x) r_2(x) \dots$ può aver generato il grafo in figura.

Domanda 5

Risposta non data

Punteggio max.:

2,00

Cardinalità (2 punti, penalità -15% per risposta sbagliata)

ALLOGGIO(CodA, NumeroDiOspiti, Indirizzo, Citta', Regione)
 SERVIZIO(CodS, NomeServizio, TipoServizio)
 ALLOGGIO-HA-SERVIZI(CodA, CodS)
 UTENTE(CodU, Nome, Cognome, BusinessAccount, DataNascita, Indirizzo, Citta', Regione)
 PRENOTAZIONE(CodA, DataInizio, CodU, DataFine)

Assumere le seguenti cardinalità:

- $\text{card}(\text{ALLOGGIO}) = 10^5$ tuple,
 Valori distinti di Regione = 20
- $\text{card}(\text{SERVIZIO}) = 10^2$ tuple,
 Valori distinti di TipoServizio = 20
- $\text{card}(\text{ALLOGGIO-HA-SERVIZIO}) = 10^6$ tuple,
 Valori distinti di Regione = 20
- $\text{card}(\text{Utente}) = 10^4$ tuples,
 MIN(DATE(DataNascita)) = 1/1/1930, MAX(DATE(DataNascita)) = 31/12/2009
 Valori distinti di Regione = 20
 Valori distinti di BusinessAccount = 2 ("True", "False")
- $\text{card}(\text{Prenotazione}) = 10^7$ tuple,
 MIN(DataInizio) = 1/9/2017, MAX(DataInizio)) = 31/08/2020

Inoltre, assumere il seguente fattore di riduzione per la condizione di group by:

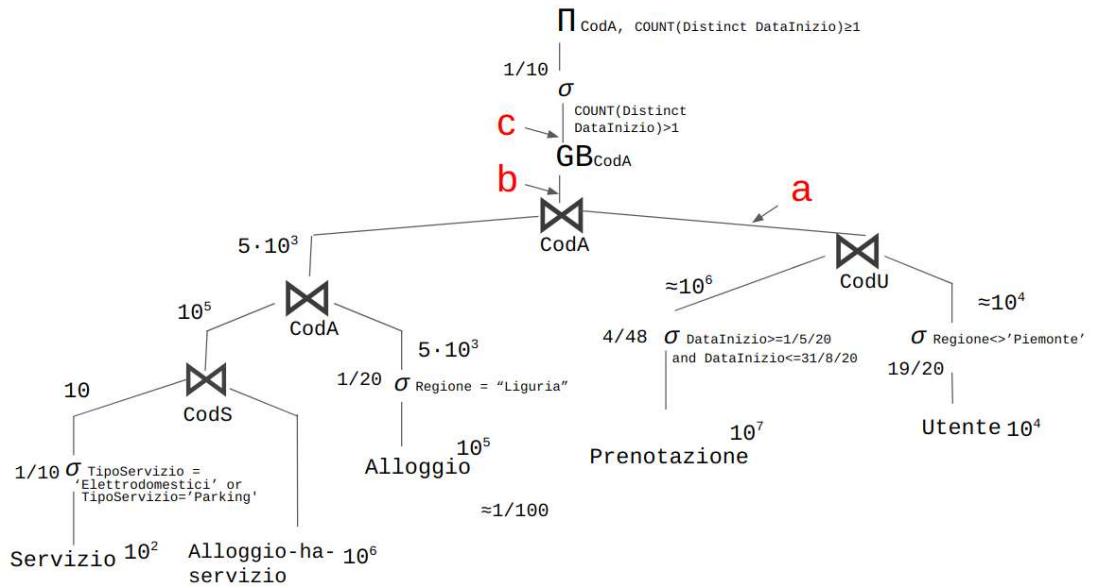
- Having COUNT(Distinct DataInizio)>1 ≈ 1/10

Si consideri la seguente query:

```
select A.CodA, count(Distinct DataInizio)
from SERVIZIO S, ALLOGGIO-HA-SERVIZI AHS, ALLOGGIO A, PRENOTAZIONE P, UTENTE U
where S.CodS=AHS.CodS and A.CodA=AHS.CodA and U.CodU=P.CodU and P.CodA=A.CodA
and (S.TipoServizio="Parcheggio" or S.TipoServizio="Elettrodomestici")
and A.Regione="Liguria" and B.DataInizio>=1/5/20 and B.DataInizio<=31/8/20
and U.Regione<>"Piemonte"
group by CodA
Having COUNT(Distinct DataInizio)>1
```

La figura sottostante rappresenta il query tree per la query precedente.

Specificare la cardinalità di ogni ramo indicato dai caratteri in rosso (**a**, **b**, **c**) nella figura sottostante.



- (a) a: $<10^6$ b: $<5 \cdot 10^4$ c: $\approx 5 \cdot 10^3$
- (b) a: $<10^2$ b: $<5 \cdot 10^2$ c: $<10^6$
- (c) a: $<10^6$ b: $<5 \cdot 10^4$ c: $\approx 10^6$
- (d) a: $<10^2$ b: $<5 \cdot 10^2$ c: $<5 \cdot 10$
- (e) a: $<10^6$ b: $<10^3$ c: $\approx 5 \cdot 10^3$
- (f) a: $<10^5$ b: $<5 \cdot 10^4$ c: $\approx 10^6$
- (g) a: $<10^3$ b: $<5 \cdot 10^4$ c: $\approx 5 \cdot 10^3$
- (h) a: $<10^6$ b: $<2 \cdot 10^4$ c: $\approx 2 \cdot 10^3$

Risposta errata.

La risposta corretta è: a: $<10^6$ b: $<5 \cdot 10^4$ c: $\approx 5 \cdot 10^3$

Domanda 6

Risposta non data

Punteggio max.:

2,00

Indici (2 punti, penalità -15% per ogni risposta sbagliata)

ALLOGGIO (CodA, NumeroDiOspiti, Indirizzo, Citta', Regione)
 SERVIZIO (CodS, NomeServizio, TipoServizio)
 ALLOGGIO-HA-SERVIZI (CodA, CodS)
 UTENTE (CodU, Nome, Cognome, BusinessAccount, DataNascita, Indirizzo, Citta', Regione)
 PRENOTAZIONE (CodA, DataInizio, CodU, DataFine)

Assumere le seguenti cardinalità:

- $\text{card}(\text{ALLOGGIO}) = 10^5$ tuple,
Valori distinti di Regione = 20
- $\text{card}(\text{SERVIZIO}) = 10^2$ tuple,
Valori distinti di TipoServizio = 20
- $\text{card}(\text{ALLOGGIO-HA-SERVIZIO}) = 10^6$ tuple,
 $\text{card}(\text{Utente}) = 10^4$ tuples,
MIN(DATE(DataNascita)) = 1/1/1930, MAX(DATE(DataNascita)) = 31/12/2009
Valori distinti di Regione = 20
Valori distinti di BusinessAccount = 2 ("True", "False")
- $\text{card}(\text{Prenotazione}) = 10^7$ tuple,
MIN(DataInizio) = 1/9/2017, MAX(DataInizio)) = 31/08/2020

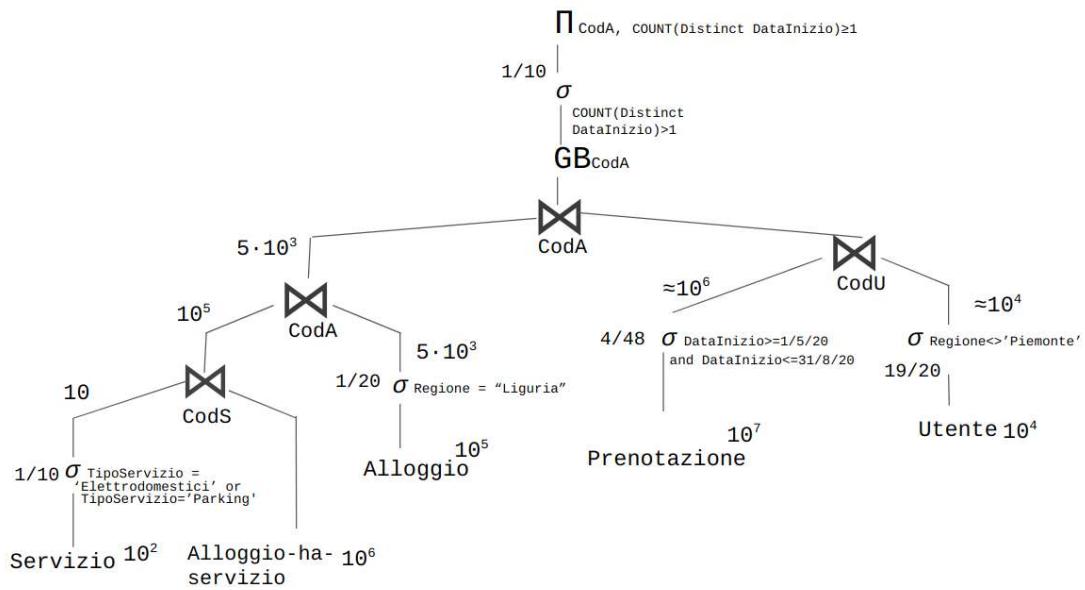
Inoltre, assumere il seguente fattore di riduzione per la condizione di group by:

- Having COUNT(Distinct DataInizio)>1 ≈ 1/10

Si consideri la seguente query:

```
select A.CodA, count(Distinct DataInizio)
from SERVIZIO S, ALLOGGIO-HA-SERVIZI AHS, ALLOGGIO A, PRENOTAZIONE P, UTENTE U
where S.CodS=AHS.CodS and A.CodA=AHS.CodA and U.CodU=P.CodU and P.CodA=A.CodA
and (S.TipoServizio="Parcheggio" or S.TipoServizio="Elettrodomestici")
and A.Regione="Liguria" and B.DataInizio>=1/5/20 and B.DataInizio<=31/8/20
and U.Regione<>"Piemonte"
group by CodA
Having COUNT(Distinct DataInizio)>1
```

La figura sottostante rappresenta il query tree per la query precedente.



Si selezionino le strutture fisiche accessorie per migliorare le prestazioni dell'interrogazione (se possibile).

Scegli una o più alternative:

- (a) CREATE INDEX IndexF ON Servizio(TipoServizio) - B+-Tree
- (b) CREATE INDEX IndexG ON Alloggio(Regione) - HASH
- (c)
CREATE INDEX IndexB ON Utente(Regione) - B+-Tree
- (d) Nessuna - le strutture fisiche accessorie non migliorerebbero le prestazioni dell'interrogazione
- (e) CREATE INDEX IndexA ON Utente(Regione) - HASH
- (f) CREATE INDEX IndexH ON Alloggio(Regione) - B+-Tree
- (g) CREATE INDEX IndexD ON Prenotazione(DataInizio) - HASH
- (h) CREATE INDEX IndexE ON Servizio(TipoServizio) - HASH
- (i) CREATE INDEX IndexC ON Prenotazione(DataInizio) - B+-Tree

Risposta errata.

La risposta corretta è: CREATE INDEX IndexC ON Prenotazione(DataInizio) - B+-Tree, CREATE INDEX IndexG ON Alloggio(Regione) - HASH

Domanda 7

Risposta non data

Punteggio max.:
2,00**Anticipo Group By (2 punti, penalità -15% per risposta sbagliata)**

ALLOGGIO (CodA, NumeroDiOspiti, Indirizzo, Citta', Regione)
 SERVIZIO (CodS, NomeServizio, TipoServizio)
 ALLOGGIO-HA-SERVIZI (CodA, CodS)
 UTENTE (CodU, Nome, Cognome, BusinessAccount, DataNascita, Indirizzo, Citta', Regione)
 PRENOTAZIONE (CodA, DataInizio, CodU, DataFine)

Assumere le seguenti cardinalità:

- $\text{card}(\text{ALLOGGIO}) = 10^5$ tuple,
 Valori distinti di Regione = 20
- $\text{card}(\text{SERVIZIO}) = 10^2$ tuple,
 Valori distinti di TipoServizio = 20
- $\text{card}(\text{ALLOGGIO-HA-SERVIZIO}) = 10^6$ tuple,
 Valori distinti di Regione = 20
- $\text{card}(\text{Utente}) = 10^4$ tuples,
 MIN(DATE(DataNascita)) = 1/1/1930, MAX(DATE(DataNascita)) = 31/12/2009
 Valori distinti di Regione = 20
 Valori distinti di BusinessAccount = 2 ("True", "False")
- $\text{card}(\text{Prenotazione}) = 10^7$ tuple,
 MIN(DataInizio) = 1/9/2017, MAX(DataInizio)) = 31/08/2020

Inoltre, assumere il seguente fattore di riduzione per la condizione di group by:

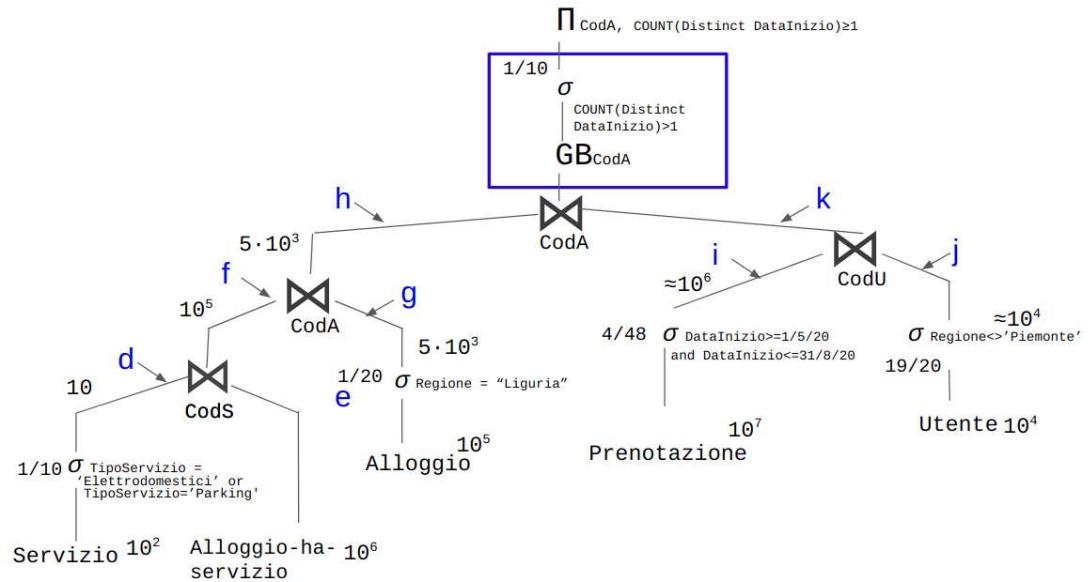
- Having COUNT(Distinct DataInizio)>1 ≈ 1/10

Si consideri la seguente query:

```
select A.CodA, count(Distinct DataInizio)
from SERVIZIO S, ALLOGGIO-HA-SERVIZI AHS, ALLOGGIO A, PRENOTAZIONE P, UTENTE U
where S.CodS=AHS.CodS and A.CodA=AHS.CodA and U.CodU=P.CodU and P.CodA=A.CodA
and (S.TipoServizio="Parcheggio" or S.TipoServizio="Elettrodomestici")
and A.Regione="Liguria" and B.DataInizio>=1/5/20 and B.DataInizio<=31/8/20
and U.Regione<>"Piemonte"
group by CodA
Having COUNT(Distinct DataInizio)>1
```

La figura sottostante rappresenta il query tree per la query precedente.

Analizzare l'anticipazione della **Group By**.



- (a) E' possibile anticiparla nel ramo e
- (b) E' possibile anticiparla nel ramo j
- (c) E' possibile anticiparla nel ramo d
- (d) E' possibile anticiparla nel ramo g
- (e) Non e' possibile anticipare la Group By
- (f) E' possibile anticiparla nel ramo f
- (g) E' possibile anticiparla nel ramo i
- (h) E' possibile anticiparla nel ramo h
- (i) E' possibile anticiparla nel ramo k

Risposta errata.

La risposta corretta è: E' possibile anticiparla nel ramo k

Domanda 8

Risposta non data

Punteggio max.:

1,00

Schema concettuale 1 (1 punto, penalità -15% per ogni risposta sbagliata)

Gli amministratori di una piattaforma per il tracciamento di attività sportive vogliono costruire un data warehouse per analizzare le varie sessioni di attività. In particolare, la piattaforma consente ai propri utenti di registrare le proprie sessioni di attività sportive effettuate all'aperto.

Una **sessione di attività sportiva** può appartenere a una di due **categorie di sport**: a piedi o in bici. Nello specifico, queste categorie includono i seguenti sport:

- **a piedi**: corsa, camminata
- **in bici**: bici da corsa, mountain bike

Una sessione di attività sportiva si svolge in una determinata **data**. Si assume che le sessioni non possano avvenire a cavallo di più giorni. Si vuole anche tenere traccia del **mese dell'anno** in cui si svolge l'attività e della **stagione meteorologica**. Le stagioni meteorologiche vengono definite come segue:

- **Inverno**: inizio 1 Dicembre, fine 28/29 Febbraio
- **Primavera**: inizio 1 Marzo, fine 31 Maggio
- **Estate**: inizio 1 Giugno, fine 31 Agosto
- **Autunno**: inizio 1 Settembre, fine 30 Novembre

La sessione di attività sportiva è inoltre caratterizzata da un punto di **partenza** e uno di **arrivo**, determinati da una **cella geografica**. Queste celle geografiche sono coprenti e non sovrapposte (ossia, ciascun punto dei territori interessati appartiene a una e una sola cella geografica). Si vogliono effettuare analisi anche a livello di **comune**, **regione** e **stato**. Si consideri che una cella geografica appartiene a un solo comune, mentre un comune può contenere più celle geografiche.

Per ogni sessione di attività sono inoltre note le **condizioni meteorologiche** che la hanno caratterizzata: sole, nuvoloso, pioggia, neve, o una combinazione di queste (per esempio, una sessione di attività potrebbe essere caratterizzata da pioggia, nuvoloso, sole).

L'attività viene svolta da un atleta, per cui è nota l'**età** e la **fascia di età** in cui rientra (< 14 anni, 14-25 anni, 26-39 anni, 40-59 anni, > 60 anni), il **peso** e la **fascia di peso** (<45 kg, 45-64 kg, 65-85 kg, > 85 kg).

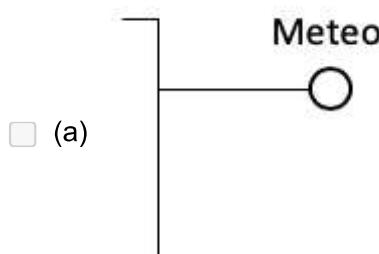
Sulla base delle precedenti variabili, si vuole monitorare:

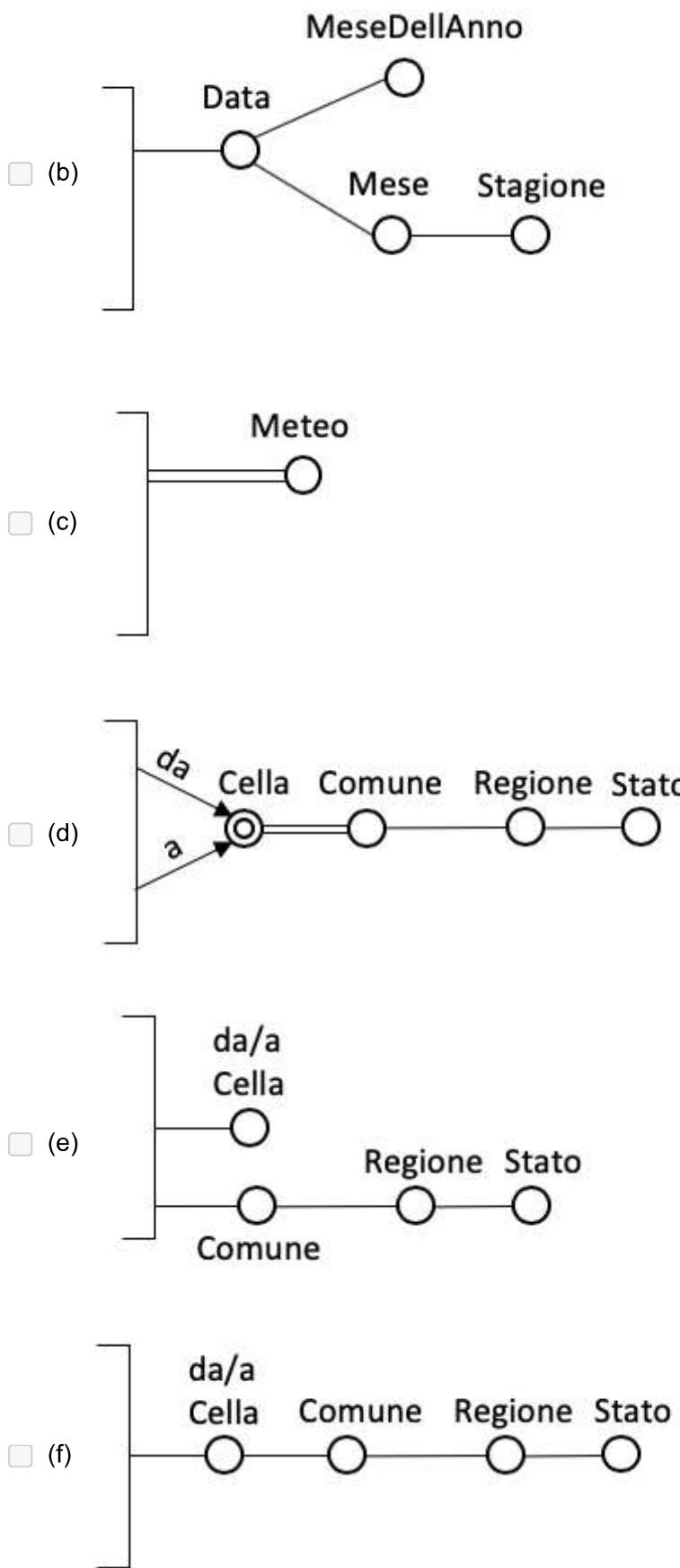
- il totale dei chilometri percorsi
- il numero di chilometri medio per sessione di attività

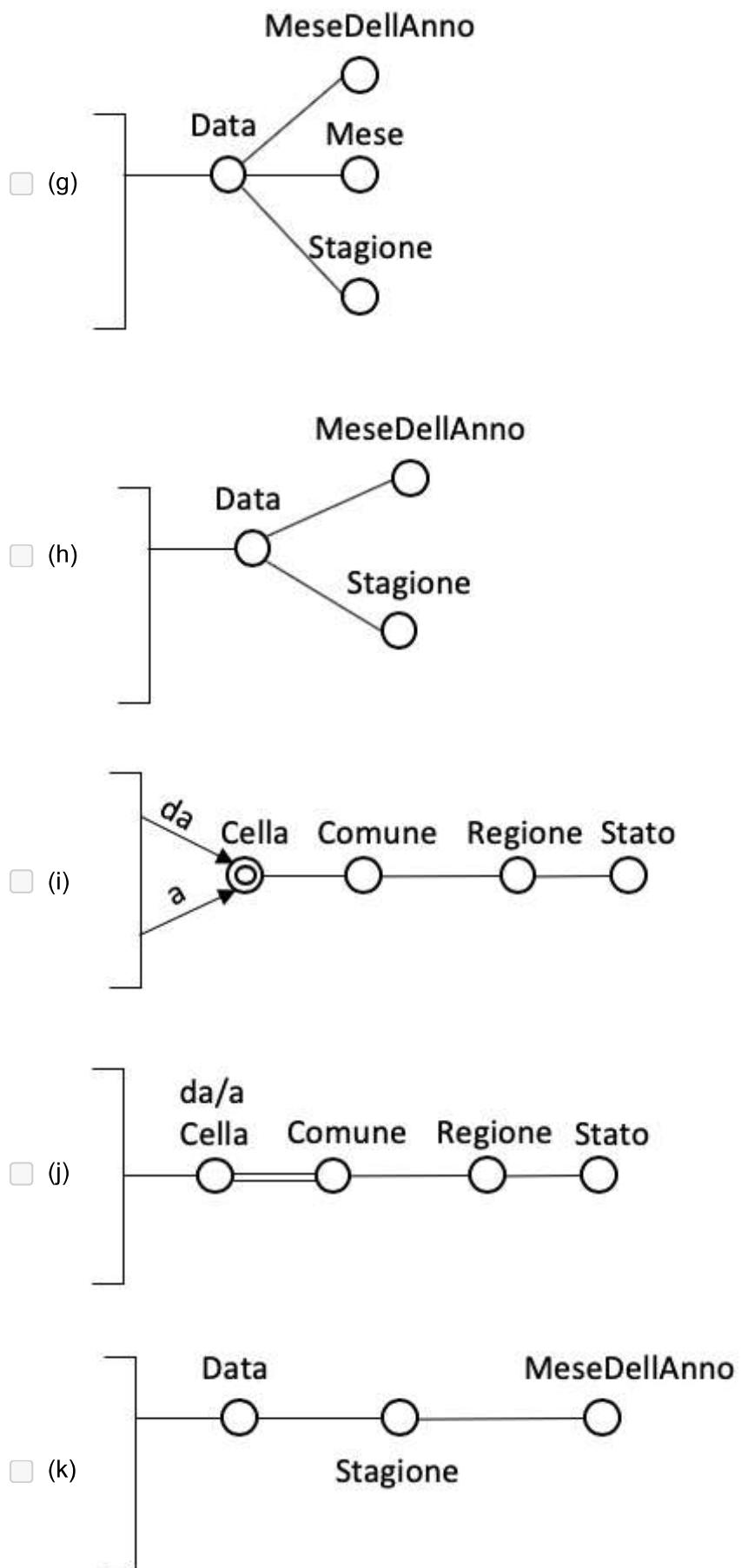
Selezionare, tra le dimensioni proposte temporale, spaziale e meteo, quelle che soddisfano le richieste descritte nelle specifiche del problema nel modo più efficiente.

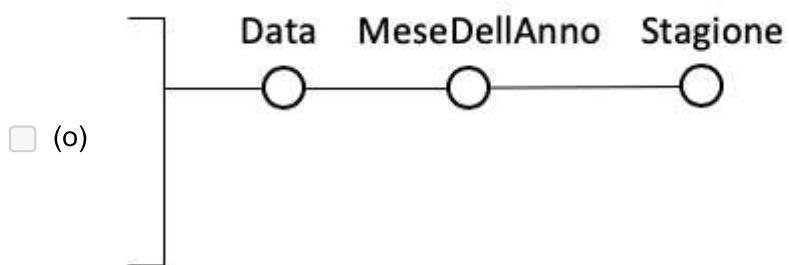
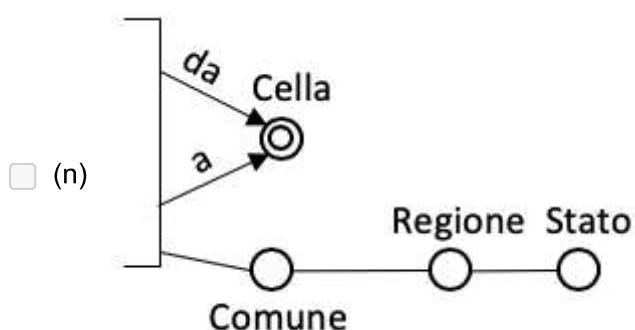
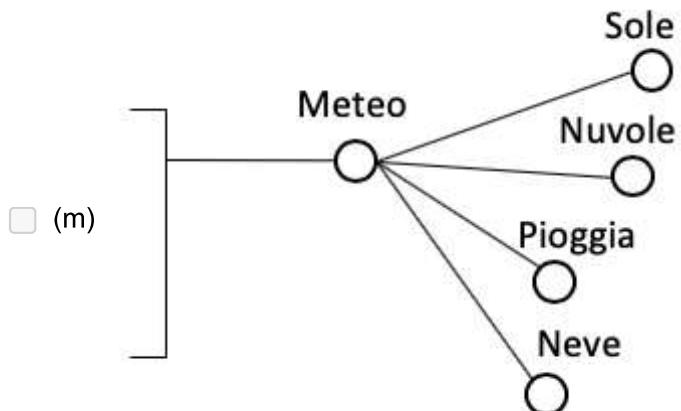
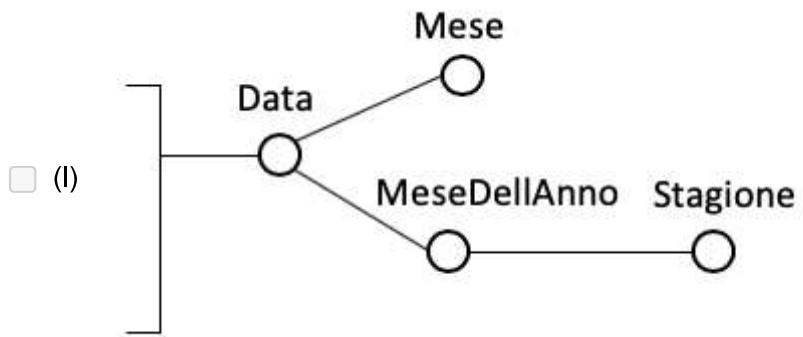
NB: selezionare tutte e solo le risposte corrette; più risposte sono corrette (in particolare, una risposta giusta per ogni dimensione).

Scegli una o più alternative:





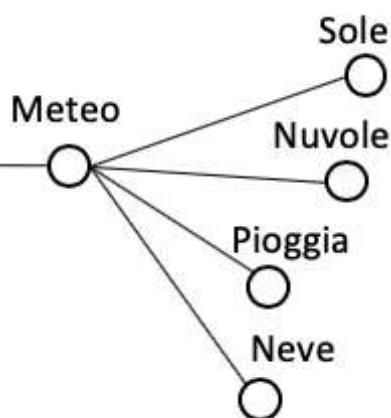
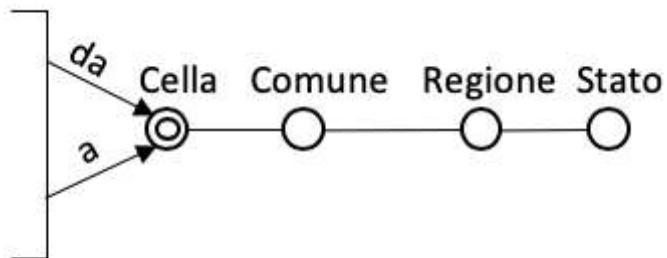




Risposta errata.

Data MeseDellAnno Stagione

La risposta corretta è:



Domanda 9

Risposta non data

Punteggio max.:

1,00

Schema concettuale 2 (1 punto, penalità -15% per ogni risposta sbagliata)

Gli amministratori di una piattaforma per il tracciamento di attività sportive vogliono costruire un data warehouse per analizzare le varie sessioni di attività. In particolare, la piattaforma consente ai propri utenti di registrare le proprie sessioni di attività sportive effettuate all'aperto.

Una **sessione di attività sportiva** può appartenere a una di due **categorie di sport**: a piedi o in bici. Nello specifico, queste categorie includono i seguenti sport:

- **a piedi**: corsa, camminata
- **in bici**: bici da corsa, mountain bike

Una sessione di attività sportiva si svolge in una determinata **data**. Si assume che le sessioni non possano avvenire a cavallo di più giorni. Si vuole anche tenere traccia del **mese dell'anno** in cui si svolge l'attività e della **stagione meteorologica**. Le stagioni meteorologiche vengono definite come segue:

- **Inverno**: inizio 1 Dicembre, fine 28/29 Febbraio
- **Primavera**: inizio 1 Marzo, fine 31 Maggio
- **Estate**: inizio 1 Giugno, fine 31 Agosto
- **Autunno**: inizio 1 Settembre, fine 30 Novembre

La sessione di attività sportiva è inoltre caratterizzata da un punto di **partenza** e uno di **arrivo**, determinati da una **cella geografica**. Queste celle geografiche sono coprenti e non sovrapposte (ossia, ciascun punto dei territori interessati appartiene a una e una sola cella geografica). Si vogliono effettuare analisi anche a livello di **comune**, **regione** e **stato**. Si consideri che una cella geografica appartiene a un solo comune, mentre un comune può contenere più celle geografiche.

Per ogni sessione di attività sono inoltre note le **condizioni meteorologiche** che la hanno caratterizzata: sole, nuvoloso, pioggia, neve, o una combinazione di queste (per esempio, una sessione di attività potrebbe essere caratterizzata da pioggia, nuvoloso, sole).

L'attività viene svolta da un atleta, per cui è nota l'**età** e la **fascia di età** in cui rientra (< 14 anni, 14-25 anni, 26-39 anni, 40-59 anni, > 60 anni), il **peso** e la **fascia di peso** (<45 kg, 45-64 kg, 65-85 kg, > 85 kg).

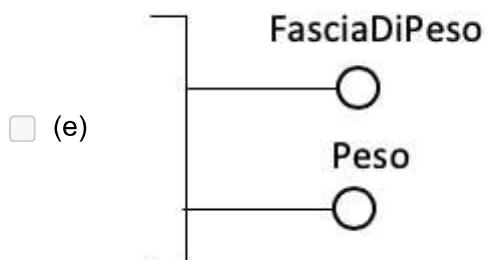
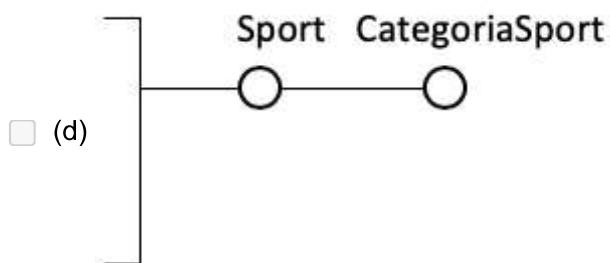
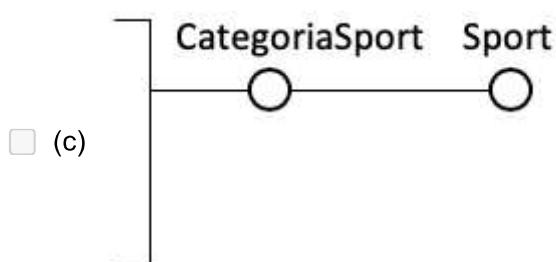
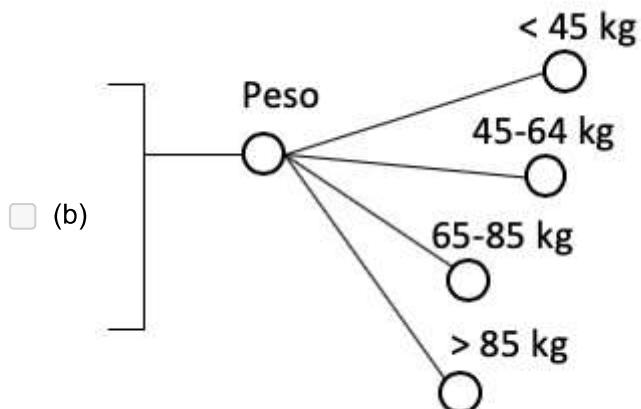
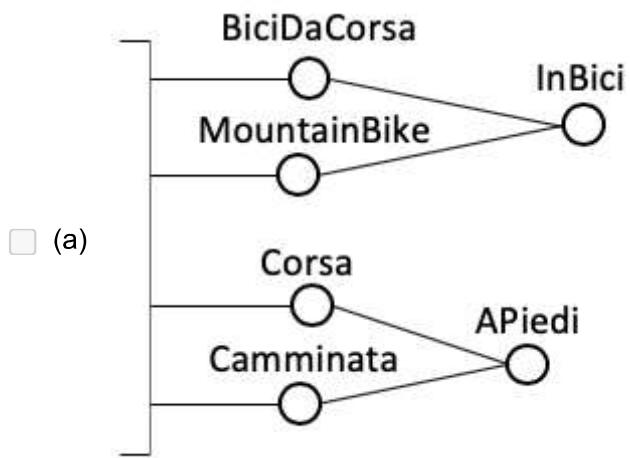
Sulla base delle precedenti variabili, si vuole monitorare:

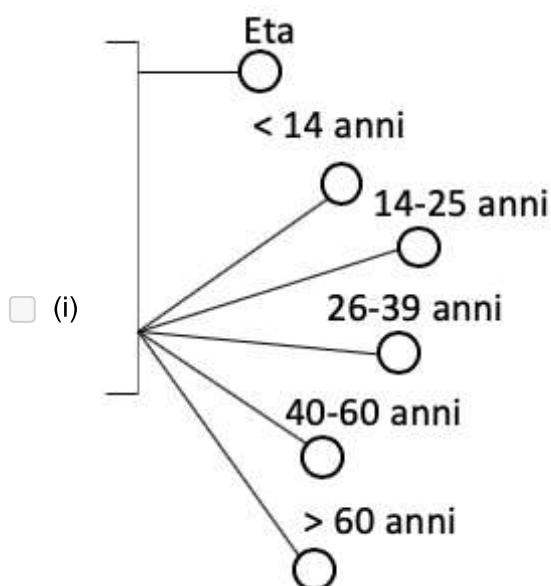
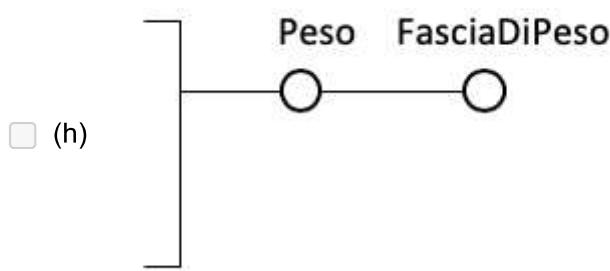
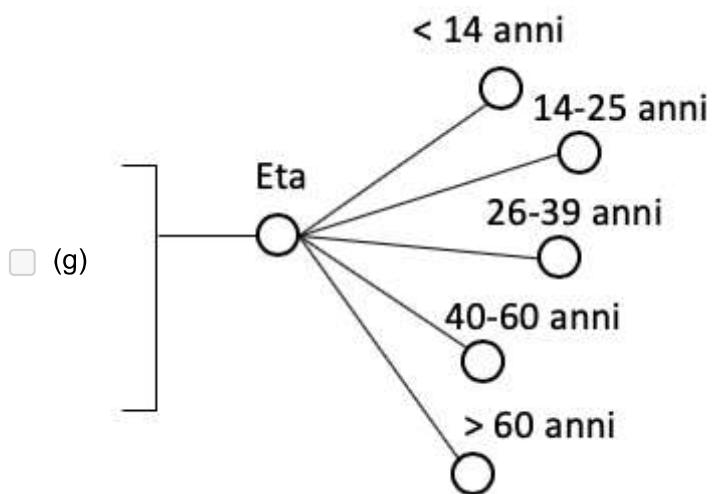
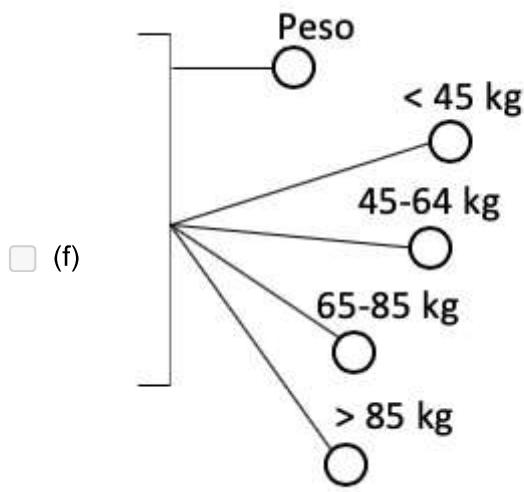
- il totale dei chilometri percorsi
- il numero di chilometri medio per sessione di attività

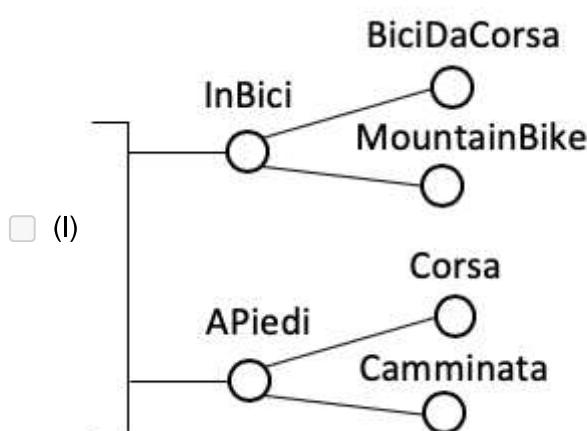
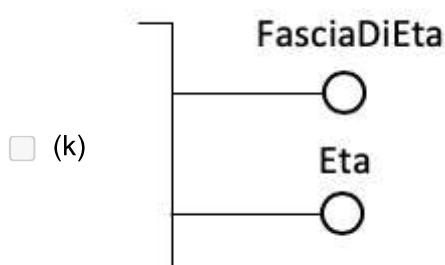
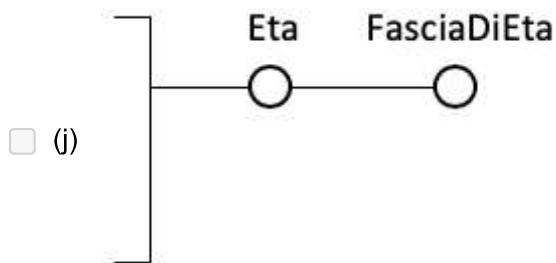
Selezionare, tra le dimensioni proposte di seguito (sportive, di peso e di età), quelle che soddisfano le richieste descritte nelle specifiche del problema nel modo più efficiente.

NB: selezionare tutte e solo le risposte corrette; più risposte sono corrette (in particolare, una risposta giusta per ogni dimensione).

Scegli una o più alternative:

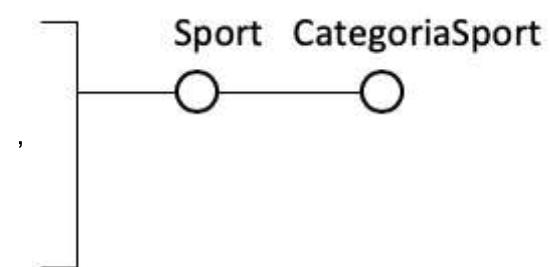
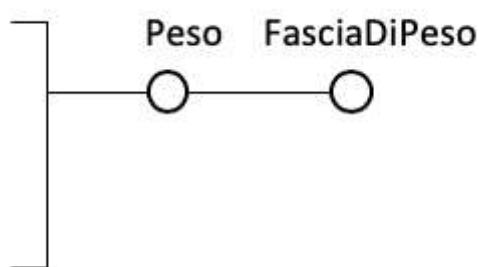
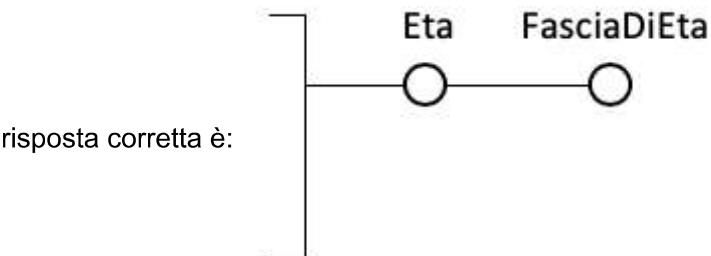






Risposta errata.

La risposta corretta è:



Domanda 10

Risposta non data

Punteggio max.:
1,00**Misure (1 punto, penalità -15% per ogni risposta sbagliata)**

Gli amministratori di una piattaforma per il tracciamento di attività sportive vogliono costruire un data warehouse per analizzare le varie sessioni di attività. In particolare, la piattaforma consente ai propri utenti di registrare le proprie sessioni di attività sportive effettuate all'aperto.

Una **sessione di attività sportiva** può appartenere a una di due **categorie di sport**: a piedi o in bici. Nello specifico, queste categorie includono i seguenti sport:

- **a piedi**: corsa, camminata
- **in bici**: bici da corsa, mountain bike

Una sessione di attività sportiva si svolge in una determinata **data**. Si assume che le sessioni non possano avvenire a cavallo di più giorni. Si vuole anche tenere traccia del **mese dell'anno** in cui si svolge l'attività e della **stagione meteorologica**. Le stagioni meteorologiche vengono definite come segue:

- **Inverno**: inizio 1 Dicembre, fine 28/29 Febbraio
- **Primavera**: inizio 1 Marzo, fine 31 Maggio
- **Estate**: inizio 1 Giugno, fine 31 Agosto
- **Autunno**: inizio 1 Settembre, fine 30 Novembre

La sessione di attività sportiva è inoltre caratterizzata da un punto di **partenza** e uno di **arrivo**, determinati da una **cella geografica**. Queste celle geografiche sono coprenti e non sovrapposte (ossia, ciascun punto dei territori interessati appartiene a una e una sola cella geografica). Si vogliono effettuare analisi anche a livello di **comune, regione e stato**. Si consideri che una cella geografica appartiene a un solo comune, mentre un comune può contenere più celle geografiche.

Per ogni sessione di attività sono inoltre note le **condizioni meteorologiche** che la hanno caratterizzata: sole, nuvoloso, pioggia, neve, o una combinazione di queste (per esempio, una sessione di attività potrebbe essere caratterizzata da pioggia, nuvoloso, sole).

L'attività viene svolta da un atleta, per cui è nota l'**età** e la **fascia di età** in cui rientra (< 14 anni, 14-25 anni, 26-39 anni, 40-59 anni, > 60 anni), il **peso** e la **fascia di peso** (<45 kg, 45-64 kg, 65-85 kg, > 85 kg).

Sulla base delle precedenti variabili, si vuole monitorare:

- il totale dei chilometri percorsi
- il numero di chilometri medio per sessione di attività

Selezionare dall'elenco tutti e solo gli attributi necessari per modellare correttamente le metriche richieste dalle specifiche nella tabella dei fatti (sono possibili più risposte corrette).

Scegli una o più alternative:

- (a) Quantità media di km percorsi per sessione
- (b) Quantità totale di km percorsi
- (c) Numero totale di atleti
- (d) Durata media delle sessioni di attività
- (e) Numero giornaliero di atleti
- (f) Numero totale di sessioni di attività
- (g) Durata totale delle sessioni di attività

Risposta errata.

La risposta corretta è: Quantità totale di km percorsi, Numero totale di sessioni di attività

Domanda 11

Risposta non data

Punteggio max.:

4,00

Query SQL esteso 1 (4 punti)

Il seguente schema relazionale serve per gestire una catena di piscine dislocate in più stati. Un atleta può accedere a diverse piscine. La tabella *INGRESSO* registra, per ciascuna piscina, atleta e data, il numero di ingressi effettuati e la durata di tali ingressi.

PISCINA (CodP, NomePiscina, Comune, Regione, Stato)

ATLETA (CodA, Nome, Cognome)

DATA (CodD, Data, Mese, Anno)

INGRESSO (CodP, CodA, CodD, totaleOre, numeroIngressi)

Per le piscine in Piemonte, separatamente per mese e comune, calcolare:

- La durata media degli ingressi
- La frazione degli ingressi di ciascun comune rispetto a tutti gli ingressi in Piemonte, per ogni mese
- Il rank del mese per numero decrescente di ingressi (una classifica separata per ogni comune)

```
SELECT Mese, Comune,
       SUM(totaleOre)/SUM(numeroIngressi),
       SUM(numeroIngressi)/SUM(SUM(numeroIngressi)) OVER (PARTITION BY Mese),
       RANK() OVER (PARTITION BY Comune ORDER BY SUM(numeroIngressi) DESC)
FROM PISCINA p, DATA d, INGRESSO i
```

WHERE
i.CodP = p.CodP AND i.CodD = d.CodD
AND Regione = 'Piemonte'
GROUP BY Mese, Comune

Domanda 12

Risposta non data

Punteggio max.:

4,00

Query SQL esteso 2 (4 punti)

Il seguente schema relazionale serve per gestire una catena di piscine dislocate in più stati. Un atleta può accedere a diverse piscine. La tabella *INGRESSO* registra, per ciascuna piscina, atleta e data, il numero di ingressi effettuati e la durata di tali ingressi.

PISCINA (CodP, NomePiscina, Comune, Regione, Stato)

ATLETA (CodA, Nome, Cognome)

DATA (CodD, Data, Mese, Anno)

INGRESSO (CodP, CodA, CodD, totaleOre, numeroIngressi)

Per l'anno 2020, separatamente per mese e piscina, calcolare:

- Il numero totale di ingressi,
- Il mensile cumulativo di ingressi da inizio anno,
- La frazione di ingressi di ciascuna piscina rispetto al numero di ingressi della piscina dello stesso comune con il maggior numero di ingressi, per ogni mese

```
SELECT Mese, NomePiscina,
       SUM(numeroIngressi),
       SUM(SUM(numeroIngressi)) OVER (PARTITION BY NomePiscina ORDER BY Mese
ROWS UNBOUNDED PRECEDING),
       SUM(numeroIngressi)/MAX(SUM(numeroIngressi)) OVER (PARTITION BY Comune,
Mese)
FROM PISCINA p, DATA d, INGRESSO i
WHERE
```

p.CodP = i.CodP and i.CodD = d.CodD
AND Anno = 2020
GROUP BY Mese, NomePiscina, Comune

Domanda 13

Risposta non data

Punteggio max.:

8,00

Trigger 1 (8 punti)

Sono date le relazioni seguenti (le chiavi primarie sono sottolineate)

CENTRO(CodCentro, Nome, Indirizzo, Città, GeoCoordCentro)

PRESTAZIONI_OFFERTE(CodPrestazione, CodCentro, CostoPerSeduta)

CONVENZIONI_ASSICURATIVE(CodCentro, CodAssicurazione, CodPrestazione, PercSconto)

PAZIENTE (CodFiscale, Cognome, Indirizzo, Città, GeoCoordPaziente, CodAssicurazione)

RICHIESTA_INFORMAZIONI(DataRichiesta, CodFiscale, CodPrestazione, NumeroDiSedute)

NOTIFICA_INFORMAZIONI(DataRichiesta, CodFiscale, CodPrestazione, CodCentro, CostoTotale)

Si vogliono gestire in modo automatico le richieste di informazioni per un consorzio di centri fisioterapici della regione Piemonte. Si scriva il trigger per gestire la seguente attività.

Richiesta informazioni sulle prestazioni mediche disponibili. Un paziente richiede informazioni su una prestazione medica di cui desidera usufruire (inserimento di un record nella tabella RICHIESTA_INFORMAZIONI). Si devono svolgere le seguenti attività.

Si deve trovare il centro fisioterapico più vicino al paziente in cui sia offerta la prestazione medica richiesta dal paziente. La tabella PRESTAZIONI_OFFERTE memorizza le prestazioni mediche offerte da ciascun centro e il relativo costo per seduta. Per calcolare la distanza geografica tra un centro e il paziente utilizzare la funzione dist(GeoCoord_PuntoA, GeoCoord_PuntoB), che restituisce la distanza tra due punti a partire dalle rispettive coordinate geografiche. Le tabelle CENTRO e PAZIENTE memorizzano le informazioni su centri medici e pazienti, incluse le rispettive coordinate geografiche (attributi GeoCoordCentro e GeoCoordPaziente). Si ipotizzi che, per il paziente che ha inoltrato la richiesta di informazioni, sia presente il record corrispondente nella tabella PAZIENTE.

Se nessun centro fornisce la prestazione richiesta dal paziente, il trigger termina con errore. Altrimenti, si ipotizzi che esista al più un centro che soddisfa il requisito di distanza minima. In questo caso, si deve calcolare il costo totale della prestazione tenendo conto del costo per seduta (attributo CostoPerSeduta), del numero di sedute richieste dal paziente (attributo NumeroDiSedute) e dell'eventuale percentuale di sconto in caso il paziente disponga di una assicurazione medica convenzionata (attributo PercSconto). La tabella CONVENZIONI_ASSICURATIVE memorizza le convenzioni assicurative tra centri e assicurazioni e la percentuale di sconto applicata per ciascuna prestazione. Il trigger deve quindi notificare il centro identificato e il costo totale della prestazione (inserimento di un record nella tabella NOTIFICA_INFORMAZIONI).

```

create or replace trigger CALCOLO_COSTO_TOTALE_PRESTAZIONE
after insert on RICHIESTA_INFORMAZIONI
for each row
declare
myCoordPaz, mydist, mycentro, sconto number;

begin
---trovo le coordinate del paziente
select GeoCoordPaziente into myCoordPaz
from PAZIENTE
where CodFiscale = :new.CodFiscale;

---trovo il centro più vicino (min distanza) che fornisce la prestazione richiesta
SELECT MIN (dist(myCoordPaz, GeoCoordCentro)) INTO mydist
FROM CENTRO
WHERE CodCentro IN (Select CodCentro
                     FROM PRESTAZIONI_OFFERTE
                     WHERE CodPrestazione = :new.CodPrestazione);

if (mydist is NULL) then
--- nessun centro offre la prestazione
    raise_application_error(...);
end if;

--- leggo codice centro e costo per seduta
select CodCentro, CostoPerSeduta into mycentro, mycosto
from CENTRO C, PRESTAZIONI_OFFERTE P
where C.CodCentro = P.CodCentro
and dist(myCoordPaz, GeoCoordCentro) = mydist and CodPrestazione =
:new.CodPrestazione;

----verifico se è disponibile uno sconto
select max(PercSconto) into sconto
from CONVENZIONI_ASSICURATIVE CA, PAZIENTE P
where CodCentro= mycentro and CA.CodAssicurazione = P.CodAssicurazione
and SSB = :NEW.SSN and CodPrestazione = :NEW.CodPrestazione;

if (sconto is null) then
--- non esiste sconto
    sconto :=0;
end if;

costo_totale := (mycosto*:new.NumeroSedute)*(1-sconto);

insert into NOTIFICA_INFORMAZIONI(DataRichiesta, CodFiscale, CodPrestazione,
CodCentro, CostoTotale) values (:new.DataRichiesta, :new.CodFiscale,
:new.CodPrestazione, mycentro, costo_totale);

end;

```

Domanda 14

Risposta non data

Punteggio max.:

2,00

Trigger 2 (2 punti)

Sono date le relazioni seguenti (le chiavi primarie sono sottolineate)

CENTRO(CodCentro, Nome, Indirizzo, Città, GeoCoordCentro)

PRESTAZIONI_OFFERTE(CodPrestazione, CodCentro, CostoPerSeduta)

CONVENZIONI_ASSICURATIVE(CodCentro, CodAssicurazione, CodPrestazione,
PercSconto)

PAZIENTE (CodFiscale, Cognome, Indirizzo, Città, GeoCoordPaziente,
CodAssicurazione)

RICHIESTA_INFORMAZIONI(DataRichiesta, CodFiscale, CodPrestazione,
NumeroDiSedute)

NOTIFICA_INFORMAZIONI(DataRichiesta, CodFiscale, CodPrestazione,
CodCentro, CostoTotale)

Si vogliono gestire in modo automatico le richieste di informazioni per un consorzio di centri fisioterapici della regione Piemonte. Si scriva il trigger per gestire la seguente attività.

Vincolo di integrità sulla percentuale di sconto per prestazione. La percentuale di sconto fornita dall'assicurazione per una prestazione medica con codice prestazione 'AAT' oppure 'TTB' oppure 'EEF' non deve mai essere inferiore a 10%. In caso di operazioni di modifica della tabella CONVENZIONI_ASSICURATIVE che causino la violazione di tale vincolo si deve assegnare all'attributo PercSconto il valore minimo consentito, pari a 10%. Valutare attentamente tutti gli eventi innescanti sulla tabella CONVENZIONI_ASSICURATIVE.

```
create or replace trigger VERIFICA_SCONT
after insert or update of PercSconto, CodPrestazione on
CONVENZIONI_ASSICURATIVE
for each row
when (new.PercSconto < 10) and (new.CodPrestazione = 'AAT' or
new.CodPrestazione = 'TTB' or new.CodPrestazione = 'EEF')
begin

:new.PercSconto := 10;

end;
```

Domanda 15

Risposta non data

Non valutata

Questa non è una domanda dell'esame

Puoi utilizzare l'area di testo sottostante per appunti o bozze (per esempio, per scrivere i passi intermedi di un esercizio).

Il testo inserito in questo esercizio non verrà considerato in fase di correzione dell'esame.