

### Domanda 1

Risposta corretta

Punteggio ottenuto 1,50 su 1,50

#### 1.5 punti (15% di penalità per una risposta sbagliata)

##### Notazione:

$rN(V)$ : read dell'oggetto  $V$  da parte della transazione  $N$

$wN(V)$ : write dell'oggetto  $V$  da parte della transazione  $N$

Viene dato il seguente schedule di 3 transazioni:

$S = W2(z) W0(x) W2(z) W1(z) R2(y) R1(y) R2(x) R1(y) R0(z) W0(y)$

$S$  è conflict serializable perchè è conflict equivalent allo schedule seriale:

- (a)  $W0(x), R0(z), W0(y), W1(z), R1(y), R1(y), W2(z), W2(z), R2(y), R2(x)$
- (b)  $W2(z), W2(z), R2(y), R2(x), W1(z), R1(y), R1(y), W0(x), R0(z), W0(y)$
- (c)  $W0(x), R0(z), W0(y), W2(z), W2(z), R2(y), R2(x), W1(z), R1(y), R1(y)$
- (d)  $W1(z), R1(y), R1(y), W2(z), W2(z), R2(y), R2(x), W0(x), R0(z), W0(y)$
- (e)  $S$  non è conflict serializable ✓
- (f)  $W2(z), W2(z), R2(y), R2(x), W0(x), R0(z), W0(y), W1(z), R1(y), R1(y)$
- (g)  $W1(z), R1(y), R1(y), W0(x), R0(z), W0(y), W2(z), W2(z), R2(y), R2(x)$

Risposta corretta.

La risposta corretta è:  $S$  non è conflict serializable

## Domanda 2

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

### 1 punto (15% di penalità per una risposta sbagliata)

È data la seguente sequenza di operazioni all'interno di un file di log:

CK(), B(T1), C(T1), B(T3), U3(0), B(T0), C(T0), B(T2), D2(2), CK(T2,T3), U2(2), A(T2), I3(0), C(T3), F  
AILURE

Notazione:

- $T_n$ : Id della transazione  $n$
- $B(T_n)$ : Begin della transazione  $T_n$
- $CK(T_a, T_b, \dots)$ : checkpoint con transazioni  $T_a, T_b, \dots$  non concluse
- $C(T_n)$ : commit della transazione  $T_n$
- $A(T_n)$ : abort (rollback) della transazione  $T_n$
- $U_n(x)$ : update eseguito dalla transazione  $T_n$  sull'oggetto  $x$
- $I_n(x)$ : insert eseguito dalla transazione  $T_n$  sull'oggetto  $x$
- $D_n(x)$ : delete eseguito dalla transazione  $T_n$  sull'oggetto  $x$

Quali sono gli insiemi di UNDO e REDO finali per il warm restart?

- 
- (a) redo = {3, 0}, undo = {2}
  - (b) redo = {0}, undo = { }
  - (c) redo = {2, 3}, undo = { }
  - (d) redo = {2}, undo = {3}
  - (e) redo = {3}, undo = {2} ✓
  - (f) redo = {3}, undo = {0, 2}
  - (g) redo = { }, undo = {2, 3}
  - (h) Nessuna delle precedenti

Risposta corretta.

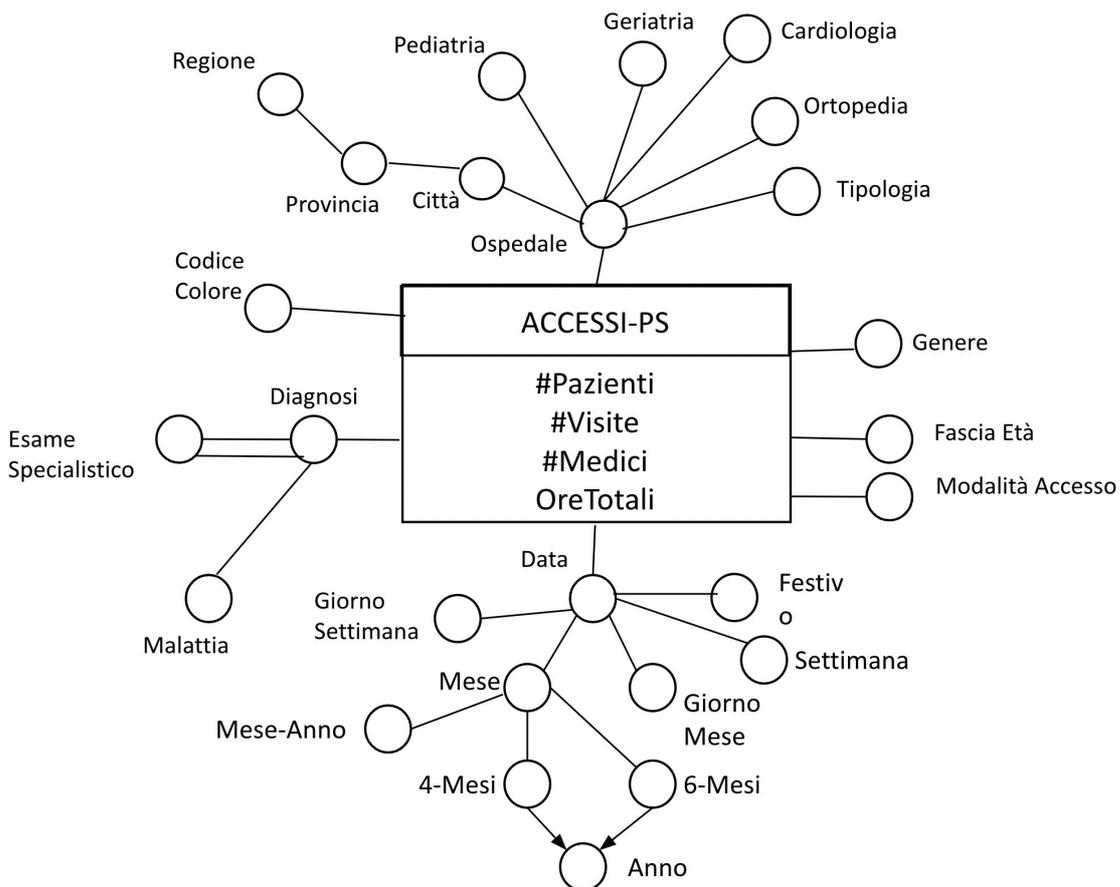
La risposta corretta è: redo = {3}, undo = {2}

## Domanda 3

**11 punti totali (nessuna penalità per risposta sbagliata)**

Il seguente data warehouse memorizza le informazioni relative agli accessi effettuati da pazienti (caratterizzati da fascia di età e genere) al pronto soccorso degli ospedali siti in Italia. Ogni ospedale appartiene ad una singola tipologia (pubblico o privato) e vi possono essere presenti uno o più reparti tra pediatria, geriatria, ortopedia, e cardiologia (questi attributi sono booleani e rappresentano una configurazione). Al triage viene assegnato un codice colore tra (bianco, verde, azzurro, giallo e rosso) e si tiene traccia della modalità di accesso al pronto soccorso (in ambulanza o autonomia). All'accesso al pronto soccorso viene assegnato una diagnosi caratterizzata da uno o più esami specialistici.

Il data warehouse è caratterizzato dal seguente schema concettuale e dal corrispondente schema logico. Ogni evento è caratterizzato da quattro misure: numero pazienti, numero visite, numero medici che hanno effettuato le visite e ore di permanenza al pronto soccorso (OreTotali).



CARATTERISTICHE\_PAZIENTE(**CodCP**, FasciaEtà, Genere)  
TRIAGE(**CodTriage**, ModalitàArrivo, CodColore)  
OSPEDALE(**CodO**, Ospedale, Tipologia, Città, Provincia, Regione, Pediatria, Geriatria, Ortopedia, Cardiologia)  
DIAGNOSI(**CodD**, Diagnosi, Malattia)  
DIAGNOSI-ESAME(**CodD**, EsameSpecialistico)  
TEMPO(**CodT**, Data, GiornoSettimana, Settimana, GiornoMese, Festivo, Mese, MeseAnno, 4-Mesi, 6-Mesi, Anno)  
ACCESSI-PRONTO-SOCCORSO(**CodCP, CodTriage, CodO, CodD, CodT**, #Pazienti, #Visite, #Medici, OreTotali)

Dato lo schema logico precedente, scrivere nel box sottostante in SQL esteso le seguenti interrogazioni, separate da uno spazio.

### Query 1 (3 punti)

Considerando i pazienti di genere femminile nella fascia di età 24-34, separatamente per trimestre (3-Mesi) e Provincia, calcolare:

- Il numero medio di visite per medico, il numero medio di ore di permanenza al pronto soccorso per paziente,
- La percentuale del numero di ore di permanenza al pronto soccorso rispetto al numero totale per regione
- Il numero complessivo delle visite al trascorrere dei trimestri (3-Mesi), separatamente per anno.

Si effettui l'analisi separatamente per modalità di arrivo al pronto soccorso.

### Query 2 (4 punti)

Considerando gli accessi al pronto soccorso con codice (CodColore) rosso effettuati da pazienti di genere femminile, separatamente per diagnosi, provincia e quadrimestre (4-Mesi), calcolare:

- Il numero medio mensile di visite
- il numero medio di pazienti per medici
- il numero complessivo di pazienti indipendentemente dalla diagnosi e provincia
- Le ore totali di attesa indipendentemente dal quadrimestre.

Si effettui l'analisi separatamente per ospedali pubblici e privati.

### Query 3 (punti 4)

Separatamente per mese e città in cui è sita un ospedale, visualizzare:

- Il numero medio giornaliero di ore di permanenza
- La percentuale di pazienti rispetto al complessivo del semestre
- Il numero complessivo di pazienti separatamente per provincia in cui è sito l'ospedale

Assegnare ad ogni record:

- Un rank separatamente per provincia. La posizione 1 va assegnata al record con il più alto

numero medio di visite per medico

- Un rank separatamente per semestre. La posizione 1 va assegnata al record con il più basso numero di pazienti)

QUERY 1)

```
SELECT 4-Mesi, Provincia, ModalitaArrivo
SUM(#Visite)/SUM(#Medici),
SUM(OreTotali)/SUM(#Pazienti),
100*SUM(OreTotali)/SUM(SUM(OreTotali)) OVER (PARTITION BY Regione, ModalitaArrivo),
SUM(SUM(#Visite)) OVER (PARTITION BY Anno,ModalitaArrivo ORDER BY 4-Mesi ROWS
UNBOUNDED PRECEDINGS)
FROM OSPEDALE O, ACCESSI-PRONTO-SOCCORSO APS, TEMPO T, caratteristiche-
paziente cp
WHERE APS.CodO=O.CodD
AND APS.CodT=T.CodT
AND APS.CodCp=CP.CodCP
AND Genere="Femminile" AND FasciaEta="24-34"
```

GROUP BY 4-Mesi, Provincia, Regione, Anno, ModalitaArrivo

QUERY 2)

```
SELECT Diagnosi, Provincia, 4-Mesi, Tipologia,
SUM(#Visite)/COUNT(DISTINCT Mese),
SUM(#Pazienti)/SUM(#Medici),
SUM(SUM(#Pazienti)) OVER (PARTITION BY Tipologia, 4-Mesi),
SUM(SUM(OreTotali)) OVER (PARTITION BY Tipologia, Diagnosi, Provincia)
FROM ACCESSO-PRONTO-SOCCORSO APS, TEMPO T, DIAGNOSI D, OSPEDALE O,
TRIAGE Tr
WHERE APS.CodO=O.CodO
AND APS.CodT=T.CodT
AND APS.CodD=D.CodD
AND APS.CodTriage=Tr.CodTriage
AND CodColore="Rosso" AND Genere="Femminile"
GROUP BY Diagnosi, Provincia, 4-Mesi, Tipologia
```

QUERY 3)

```
SELECT Mese, Citta,
SUM(OreTotale)/COUNT(DISTINCT Data),
100*SUM(#Pazienti)/SUM(SUM(#Pazienti)) OVER (PARTITION BY 6-Mesi),
SUM(SUM(#Pazienti)) OVER (PARTITION BY Provincia)),
RANK() OVER (PARTITION BY Provincia ORDER BY SUM(#Visite)/SUM(#Medici) DESC),
RANK() OVER (PARTITION BY 6-Mesi ORDER BY SUM(#Pazienti) ASC)
FROM OSPEDALE O, TEMPO T, ACCESSI-PRONTO-SOCCORSO APS
WHERE APS.CodO=O.CodO
AND APS.CodT=APS.CodT
GROUP BY Mese, Citta, 6-Mesi, Provincia
```

### Query 1

```
SELECT 3-Mesi, Provincia, ModalitàArrivo
SUM (#Visite)/SUM(#Medici), SUM(OreTotali)/SUM(#Pazienti),
SUM(OreTotali)/ SUM(SUM(OreTotali)) OVER (PARTITION BY 3-Mesi, ModalitàArrivo, Regione),
SUM(SUM(#Visite)) OVER (PARTITION BY Provincia, ModalitàArrivo, Anno ORDER BY 3-Mesi
ROWS UNBOUNDED PRECEDINGS)
FROM CARATTERISTICHE_PAZIENTE CP, ACCESSI-PRONTO-SOCCORSO APS,
OSPEDALE O, TEMPO TE, TRIAGE T
WHERE CP.CodCP= APS.CodCP AND O.CodO=APS.CodO AND TE.CodT=APS.CodT AND
T.CodTriage=APS.CodTriage AND Genere ='F' and Fascia-Età ='24-34'
GROUP BY 3-Mesi, Provincia, ModalitàArrivo, Regione, Anno
```

### Query 2

```
SELECT Diagnosi, Provincia, 4-Mesi, Tipologia
SUM (#Visite)/COUNT(DISTICT Mese),
SUM(#Pazienti)/SUM(#Medici),
SUM(SUM(#Pazienti)) OVER (PARTITION BY 4-Mesi, Tipologia),
SUM(SUM(OreTotali)) OVER (PARTITION BY Diagnosi, Provincia, Tipologia)
FROM CARATTERISTICHE_PAZIENTE CP, ACCESSI-PRONTO-SOCCORSO APS,
OSPEDALE O, TEMPO TE, DIAGNOSI D, TRIAGE T
WHERE CP.CodCP= APS.CodCP AND O.CodO=APS.CodO AND TE.CodT=APS.CodT AND
APS.CodTriage=T.CodTriage D.CodD=APS.CodD AND Genere ='F' AND CodColore ='Rosso'
GROUP BY Diagnosi, Provincia, 4-Mesi, Tipologia
```

### Query 3

```
SELECT Mese, Città,
SUM(OreTotali)/COUNT(DISTINCT Data),
100*SUM(#Pazienti)/SUM(SUM(#Pazienti)) OVER (PARTITION BY 6-Mesi, Città),
SUM(SUM(#Pazienti)) OVER (PARTITION BY Mese, Provincia)
RANK() OVER (PARTITION BY Provincia ORDER BY SUM(#Visite)/SUM(#Medici) DESC)
RANK() OVER (PARTITION BY 6-Mesi ORDER BY SUM(#Pazienti))
FROM ACCESSI-PRONTO-SOCCORSO APS, OSPEDALE O, TEMPO TE
WHERE O.CodO=APS.CodO AND TE.CodT=APS.CodT
GROUP BY Mese, 6-Mesi, Città, Semestre, Provincia
```

Commento:

Query 1 - Punti 2,75

1 errore nella prima partition e 1 nella seconda

Query 2 - Punti 4

Query 3 - Punti 3,5

Manca un attributo in ogni partition

#### Domanda 4

Risposta corretta

Punteggio ottenuto 1,50 su 1,50

#### 1.5 punti (-15% di penalità per ogni risposta sbagliata)

Si crei un FP-Tree per l'estrazione degli itemset frequenti con  $\text{MinSup} > 3$  (un itemset è frequente se compare in più di 3 transazioni) per la seguente lista di transazioni. Quale dei seguenti è un percorso di nodi presente nel FP-Tree?

Transazioni:

- {a,b,c,d}
- {a,c,d}
- {a,b,d,f}
- {b,d,e}
- {a,c,e}
- {c,d,e}
- {a,b,c,d,e}
- {a,d,f}
- {a,b,c,d}

- 
- (a)  $d:8 \rightarrow a:6 \rightarrow f:1$
  - (b)  $d:8 \rightarrow b:2 \rightarrow e:1$
  - (c)  $d:8 \rightarrow a:6 \rightarrow b:1 \rightarrow f:1$
  - (d)  $a:2 \rightarrow c:1 \rightarrow e:1$
  - (e)  $d:8 \rightarrow c:2 \rightarrow e:2$
  - (f)  $d:8 \rightarrow a:6 \rightarrow f:1$
  - (g)  $d:8 \rightarrow c:1 \rightarrow e:1$  ✓
  - (h) Nessuna risposta è corretta

Risposta corretta.

La risposta corretta è:  $d:8 \rightarrow c:1 \rightarrow e:1$

#### Domanda 5

Risposta corretta

Punteggio ottenuto 5,00 su 5,00

## 5 punti totali (penalità -15% per ogni risposta sbagliata)

Sono date le seguenti tabelle:

DIPENDENTE(DID, Nome, Cognome, Dipartimento, Stipendio)  
PROGETTO(PID, NomeProgetto, TipoProgetto, Budget, DimensioneTeam)  
DIPENDENTE-LAVORA-SU(DID, PID, DataInizioAssegnazione)  
CLIENTE(CID, NomeCliente, Azienda)  
CONTRATTO(PID, CID, ImportoContratto, DataContratto)

Si assumano le seguenti cardinalità:

- $\text{card}(\text{DIPENDENTE}) = 10^5$  tuple
  - Valori distinti di Dipartimento = 10
- $\text{card}(\text{PROGETTO}) = 100$  tuple
  - Valori distinti di DimensioneTeam = 5
- $\text{card}(\text{DIPENDENTE-LAVORA-SU}) = 6 \cdot 10^6$  tuple
  - $\text{MIN}(\text{DataInizioAssegnazione}) = 1/1/2004$ ,  $\text{MAX}(\text{DataInizioAssegnazione}) = 31/12/2023$
- $\text{card}(\text{CLIENTE}) = 5 \cdot 10^4$  tuple
  - Valori distinti di Azienda = 50
- $\text{card}(\text{CONTRATTO}) = 2 \cdot 10^5$  tuple

Inoltre, sono dati i seguenti fattori di riduzione per le clausole having:

Having COUNT(DISTINCT TipoProgetto) > 1 = 1/3  
Having SUM(ImportoContratto) > 10k = 1/2

Si consideri la seguente query:

```

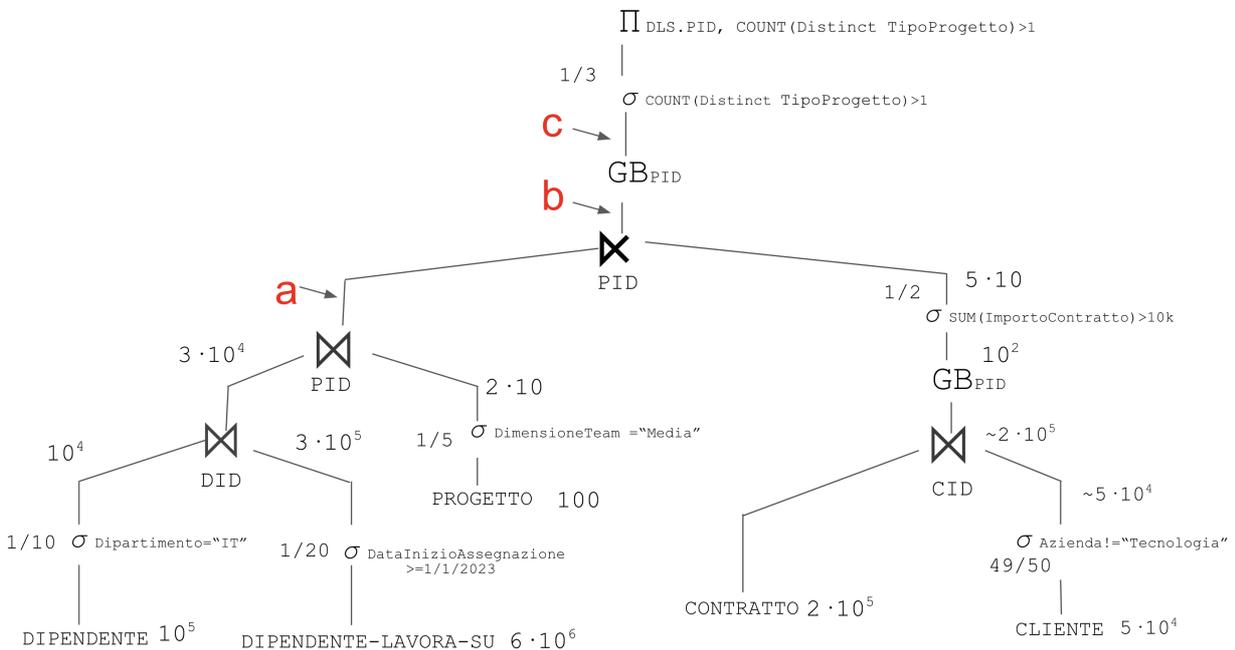
SELECT DLS.PID, COUNT(DISTINCT P.TipoProgetto)
FROM DIPENDENDTE D, PROGETTO P, DIPENDENTE-LAVORA-SU DLS
WHERE D.DID = DLS.DID
  AND P.PID = DLS.PID
  AND P.DimensioneTeam = 'Media'
  AND DLS.DataInizioAssegnazione >= 1/1/2023
  AND D.Dipartimento = 'IT'
  AND DLS.PID IN (SELECT C1.PID
                  FROM CONTRATTO C1, CLIENTE CL
                  WHERE C1.CID = CL.CID
                    AND CL.Azienda != 'Tecnologia'
                  GROUP BY C1.PID
                  HAVING SUM(ImportoContratto) > 10k)
GROUP BY DLS.PID
HAVING COUNT(DISTINCT P.TipoProgetto) > 1;

```

## Cardinalità

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

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



Seleziona la risposta corretta per la cardinalita' di (a):

- $6 \cdot 10^3$ 
  $6 \cdot 10^4$ 
  $6 \cdot 10^2$ 
  $6 \cdot 10^5$

Punteggio ottenuto 5,00 su 5,00

La risposta corretta è:  $6 \cdot 10^3$

Seleziona la risposta corretta per la cardinalita' di **(b)**:

- $\sim 3 \cdot 10^2$         $\sim 5 \cdot 10$         $\sim 3 \cdot 10^3$  ✓        $\sim 5 \cdot 10^3$

Punteggio ottenuto 5,00 su 5,00

La risposta corretta è:  $\sim 3 \cdot 10^3$

Seleziona la risposta corretta per la cardinalita' di **(c)**:

- $3 \cdot 10^2$         $3 \cdot 10^3$         $<10^2$         $<2 \cdot 10$  ✓

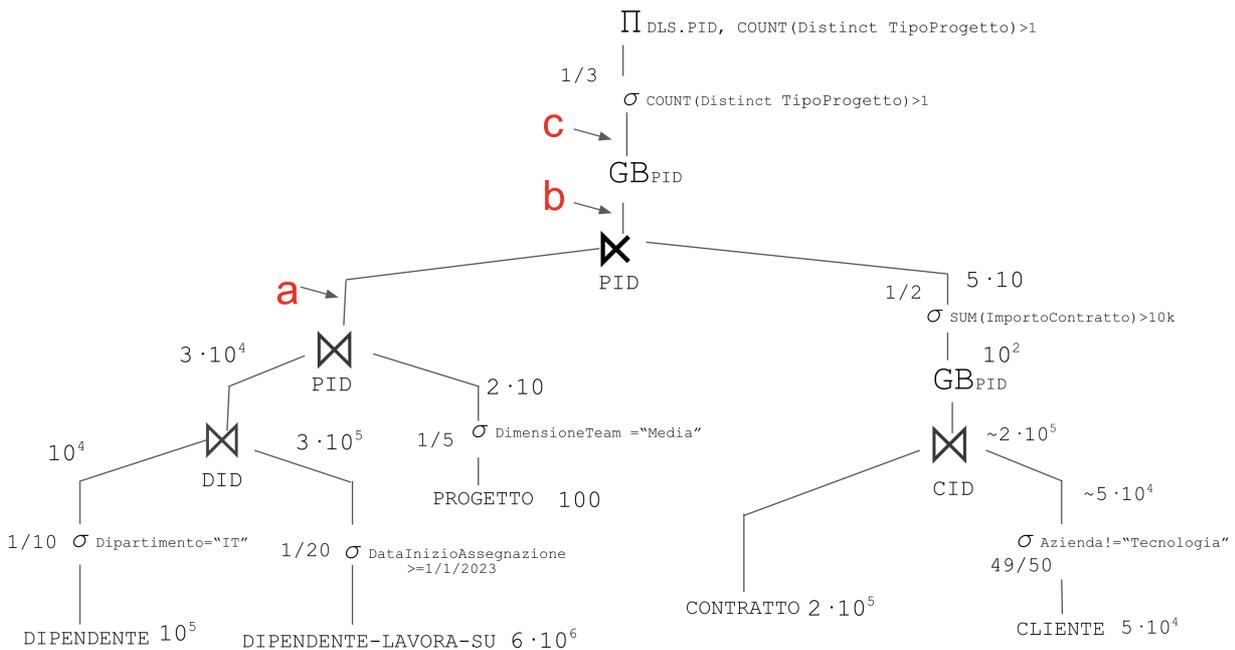
Punteggio ottenuto 5,00 su 5,00

La risposta corretta è:  $<2 \cdot 10$

## Indici

**(1.5 punti, penalità -15% per ogni risposta sbagliata)**

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



Selezionare, per ogni tabella, le strutture fisiche accessorie per migliorare le prestazioni dell'interrogazione (se possibile) tra le opzioni di seguito.

### Tabella DIPENDENTE

- CREATE INDEX IndexB ON DIPENDENTE(Dipartimento) - B+-Tree
- CREATE INDEX IndexA ON DIPENDENTE(Dipartimento) - HASH ✓
- Nessuna strutture fisica accessoria su questa tabella migliorerebbe le prestazioni della interrogazione

Punteggio ottenuto 3,00 su 3,00

La risposta corretta è: CREATE INDEX IndexA ON DIPENDENTE(Dipartimento) - HASH

### Tabella DIPENDENTE-LAVORA-SU

- CREATE INDEX IndexD ON DIPENDENTE-LAVORA-SU(DataInizioAssegnazione) - B+-Tree ✓
- CREATE INDEX IndexC ON DIPENDENTE-LAVORA-SU(DataInizioAssegnazione) - HASH
- Nessuna strutture fisica accessoria su questa tabella migliorerebbe le prestazioni della interrogazione

Punteggio ottenuto 3,00 su 3,00

La risposta corretta è: CREATE INDEX IndexD ON DIPENDENTE-LAVORA-SU(DataInizioAssegnazione) - B+-Tree

### Tabella PROGETTO

- CREATE INDEX IndexF ON PROGETTO(TipoProgetto) - B+-Tree
- CREATE INDEX IndexE ON PROGETTO(TipoProgetto) - HASH
- Nessuna strutture fisica accessoria su questa tabella migliorerebbe le prestazioni della interrogazione ✓

Punteggio ottenuto 3,00 su 3,00

La risposta corretta è: Nessuna strutture fisica accessoria su questa tabella migliorerebbe le prestazioni della interrogazione

### Tabella CLIENTE

- CREATE INDEX IndexH ON CLIENTE(Azienda) - B+-Tree
- CREATE INDEX IndexG ON CLIENTE(Azienda) - HASH
- Nessuna strutture fisica accessoria su questa tabella migliorerebbe le prestazioni della interrogazione ✓

Punteggio ottenuto 3,00 su 3,00

La risposta corretta è: Nessuna strutture fisica accessoria su questa tabella migliorerebbe le prestazioni della interrogazione

### Tabella PROGETTO

- CREATE INDEX IndexJ ON PROGETTO(DimensioneTeam) - B+-Tree
- Nessuna strutture fisica accessoria su questa tabella migliorerebbe le prestazioni della interrogazione ✓
- CREATE INDEX IndexI ON PROGETTO(DimensioneTeam) - HASH

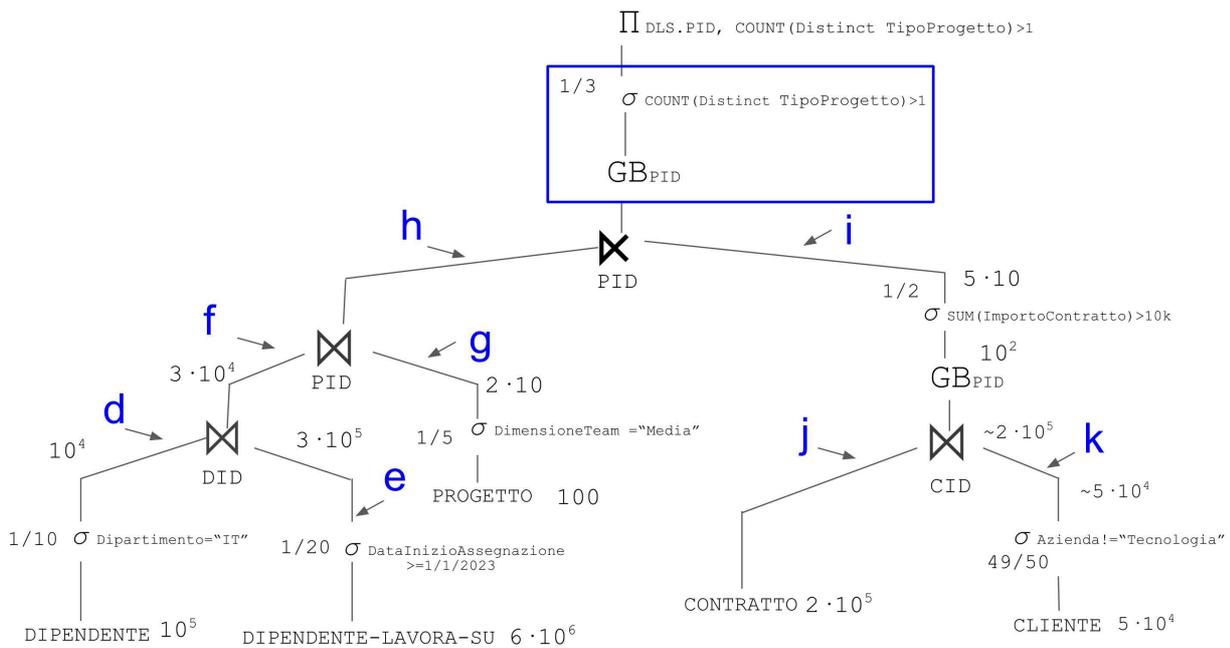
Punteggio ottenuto 3,00 su 3,00

La risposta corretta è: Nessuna strutture fisica accessoria su questa tabella migliorerebbe le prestazioni della interrogazione

## Anticipo Group By

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

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



Analizzare l'antico della GROUP BY **GROUP BY DLS.PID HAVING COUNT(DISTINCT P.TipoProgetto) > 1** rappresentata nel riquadro. Selezionare la soluzione che **consente la massima efficienza** nell'esecuzione della query (se esiste).

- Non e' possibile anticipare la Group BY GROUP BY DLS.PID HAVING COUNT(DISTINCT P.TipoProgetto) > 1
- E' possibile anticiparla nel ramo f
- E' possibile anticiparla nel ramo h ✓
- E' possibile anticiparla nel ramo k
- E' possibile anticiparla nel ramo i
- E' possibile anticiparla nel ramo e
- E' possibile anticiparla nel ramo d
- E' possibile anticiparla nel ramo g
- E' possibile anticiparla nel ramo j

Punteggio ottenuto 20,00 su 20,00

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

- 1) La risposta corretta è :  $6 \cdot 10^3$
- 2) La risposta corretta è :  $\sim 3 \cdot 10^3$
- 3) La risposta corretta è :  $< 2 \cdot 10$

- 4) La risposta corretta è : CREATE INDEX IndexA ON DIPENDENTE(Dipartimento) - HASH
- 5) La risposta corretta è : CREATE INDEX IndexD ON DIPENDENTE-LAVORA-SU(DataInizioAssegnazione) - B+-Tree
- 6) La risposta corretta è : Nessuna strutture fisica accessoria su questa tabella migliorerebbe le prestazioni della interrogazione
- 7) La risposta corretta è : Nessuna strutture fisica accessoria su questa tabella migliorerebbe le prestazioni della interrogazione
- 8) La risposta corretta è : Nessuna strutture fisica accessoria su questa tabella migliorerebbe le prestazioni della interrogazione
- 9) La risposta corretta è : E' possibile anticiparla nel ramo h

### Domanda 6

Risposta corretta

Punteggio ottenuto 1,50 su 1,50

### 1.5 punto (-15% di penalità per ogni risposta sbagliata)

La policy di MAX Linkage prevede che la distanza tra due cluster sia calcolata come:

$$\text{dist}(X, Y) = \max(\text{dist}(x, y))$$

dove  $x \in X, y \in Y$  e  $\text{dist}(x, y)$  è funzione che calcola la distanza tra due punti.

Data la seguente matrice delle distanza tra 5 punti:

	a	b	c	d	e
a	0	8	6	9	12
b	8	0	5	15	3
c	6	5	0	7	5
d	9	15	7	0	16
e	12	3	5	16	0

Come vengono aggregati i punti per ottenere 2 cluster se utizziamo la precedente policy in un clustering gerarchico agglomerativo?

- (a) Nessuna risposta è corretta
- (b) { a, b, c, d } {e}

- (c) { a, e, c } {b, d}
- (d) {b, e, c, d} {a}
- (e) { b, e, d } {c, a}
- (f) { b, e, c } {d, a} ✓
- (g) { a, d, b } {c, e}
- (h) { d, b, c } {a, e}

Risposta corretta.

La risposta corretta è: { b, e, c } {d, a}

### Domanda 7

Risposta corretta

Punteggio ottenuto 3,00 su 3,00

### 3 punti totali (penalità 15% per ogni risposta sbagliata)

Una società di coworking vuole analizzare le informazioni relative all'affitto degli spazi nelle sue diverse sedi presenti nelle diverse nazioni.

Ciascuna sede dispone di diverse tipologie di spazi, quali ad esempio ufficio singolo, sala riunioni, sala convegni. Ciascuno spazio messo a disposizione può essere fornito di uno o più servizi quali la connessione WIFI, una stampante, o un proiettore. Ciascuno spazio può essere utilizzato per diverse attività quali lavoro, riunione, eventi. Per ciascuna sede sono note le aziende con cui la sede ha siglato una convenzione; tali aziende possono usufruire degli spazi a condizioni agevolate.

I clienti possono prenotare spazi presso una sede mediante una apposita applicazione (app), oppure inviando una mail o contattando telefonicamente la sede. Per poter prenotare degli spazi i clienti devono registrarsi presso la società. Sono previsti diversi tipi di registrazioni, quali cliente «occasionale», «frequente» e «assiduo», in base alla frequenza con cui il cliente prenota gli spazi disponibili.

I gestori della società vogliono analizzare l'importo medio speso per ciascuna prenotazione e la durata media espressa in ore di una prenotazione di uno spazio in base alle seguenti informazioni:

- sede e sua collocazione geografica in termini di circoscrizione della città, città, provincia, regione e nazione. Per ciascuna sede è nota la lista di tutte le aziende con cui la sede ha stipulato una convenzione.
- data, mese, giorno del mese, trimestre, quadrimestre, semestre, anno per cui il cliente ha

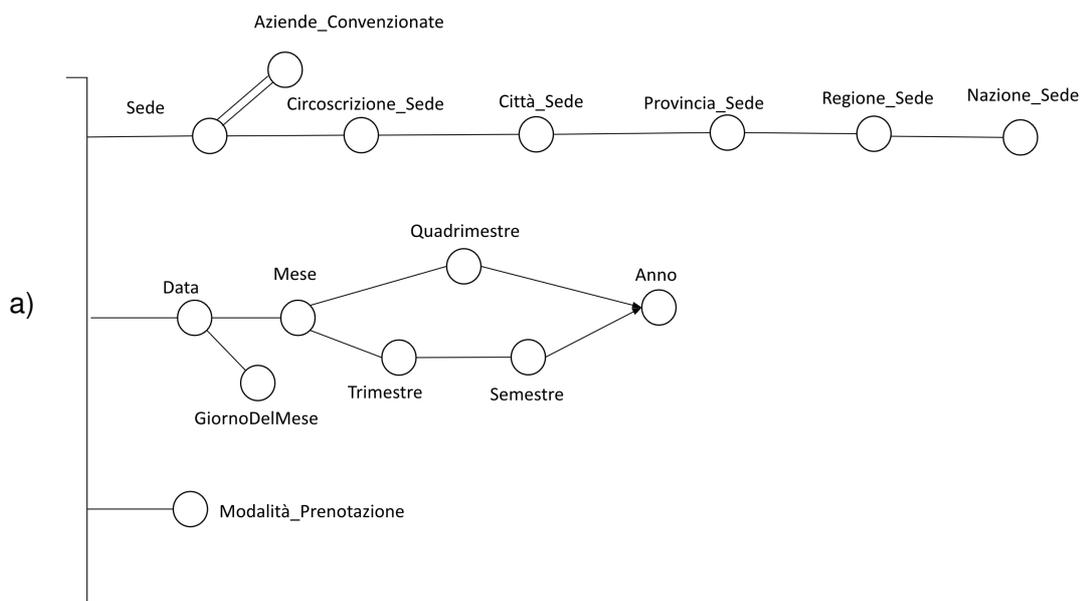
prenotato almeno uno spazio presso la sede

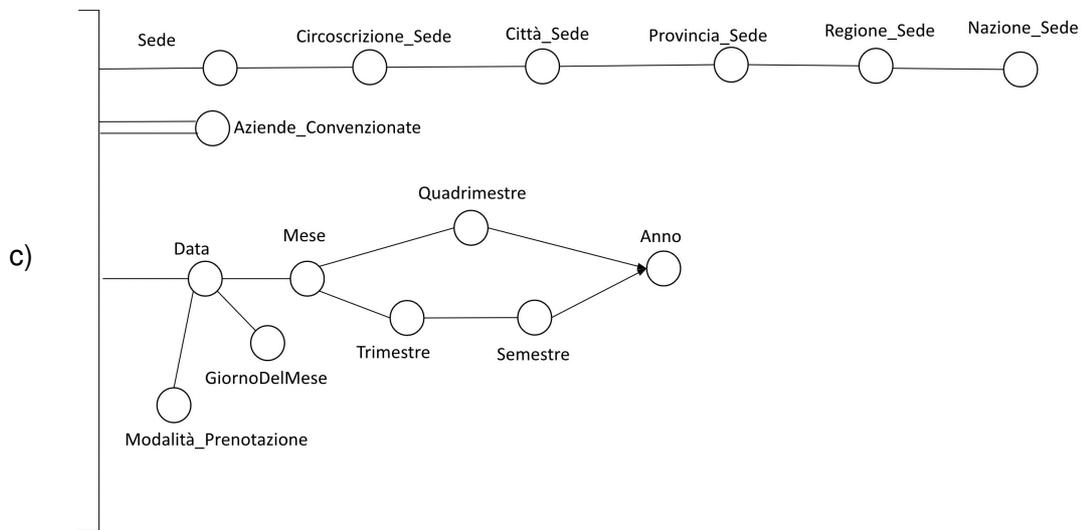
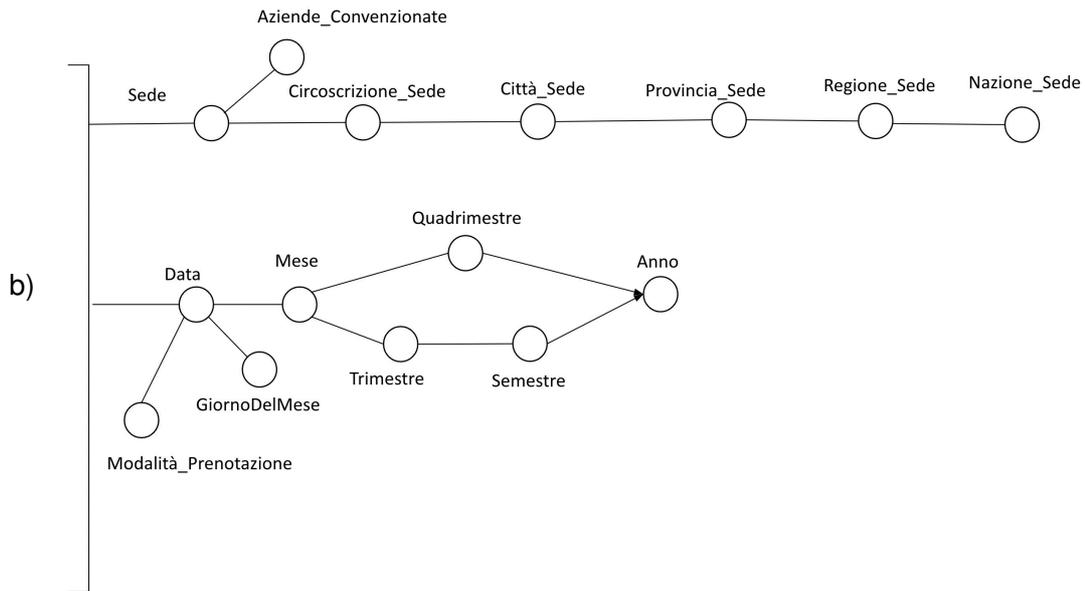
- modalità di prenotazione (un valore tra «app», «mail», telefonicamente»)
- tipo di registrazione del cliente (un valore tra «occasionale», «frequente» e «assiduo»)
- città, provincia, regione e nazione del cliente che ha prenotato lo spazio
- tipo di spazio prenotato (un valore tra «ufficio singolo», «postazione in spazio condiviso», «sala riunioni»). Ogni tipo di spazio è caratterizzato dai servizi disponibili (uno o più valori tra WIFI, stampante, proiettore) e dalla dimensione (un valore tra «piccolo», «medio», «grande»)
- tipo di pagamento (un valore tra «contanti», «bancomat», «carta di credito»)
- attività prevista (un valore tra «lavoro», «riunione», «evento»)

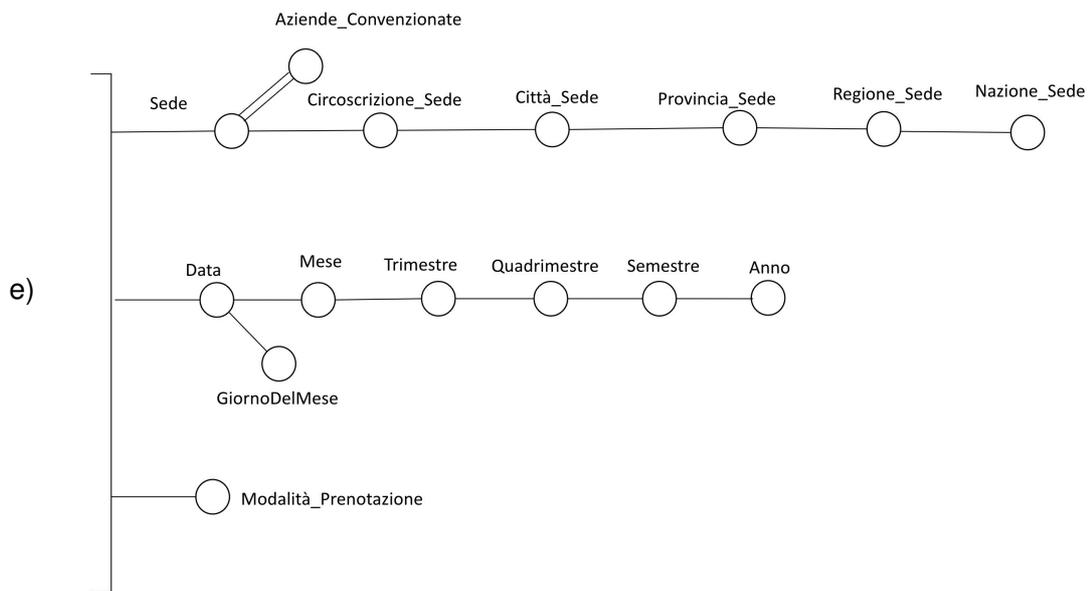
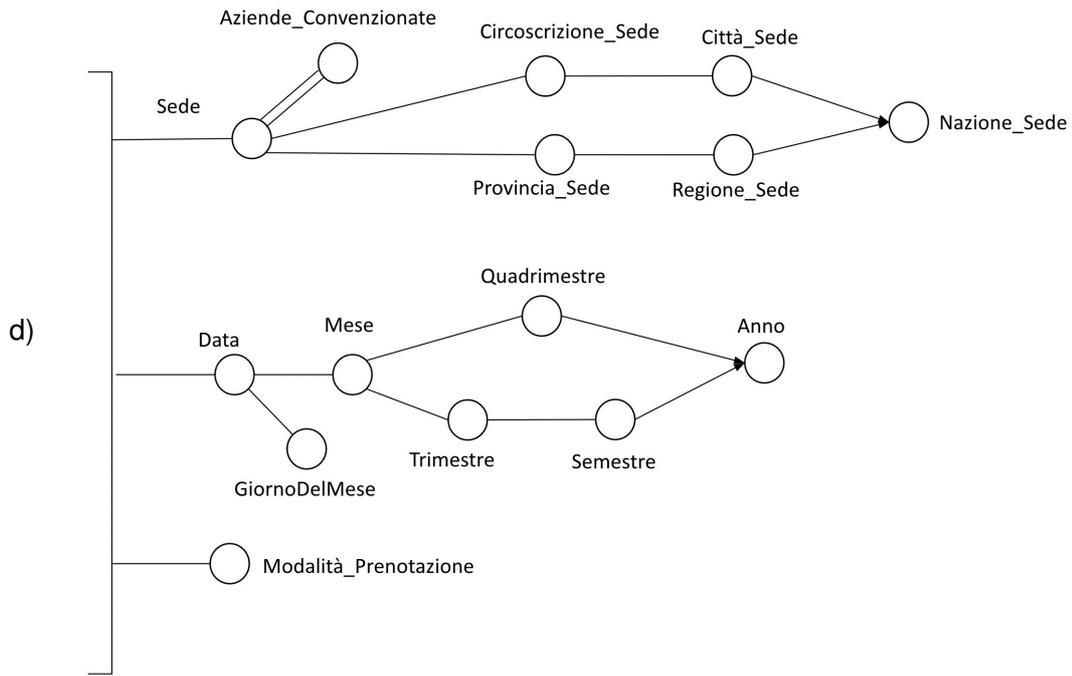
## SCHEMA CONCETTUALE 1

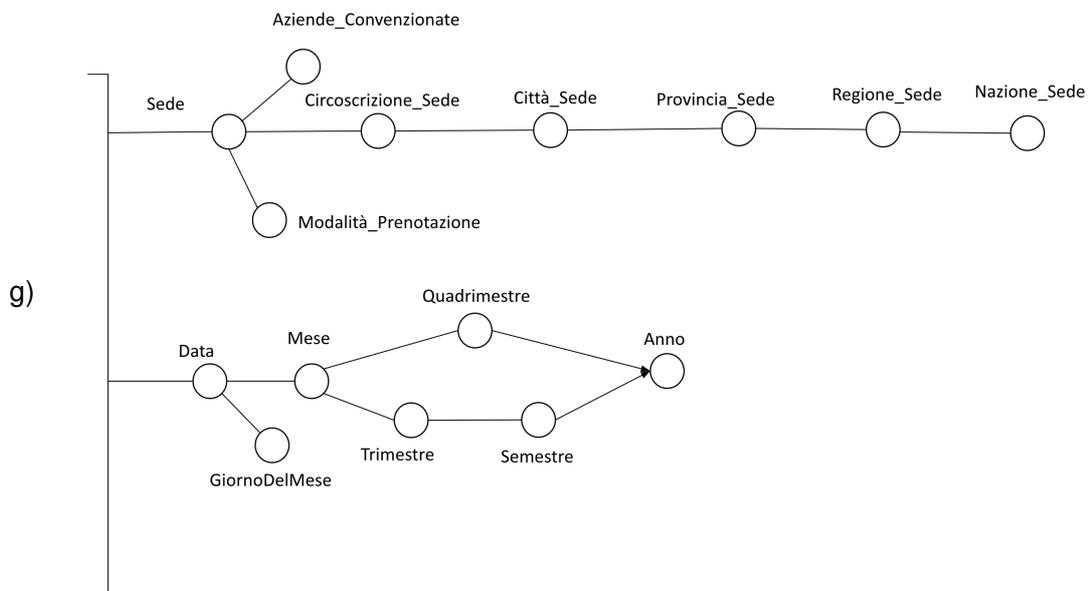
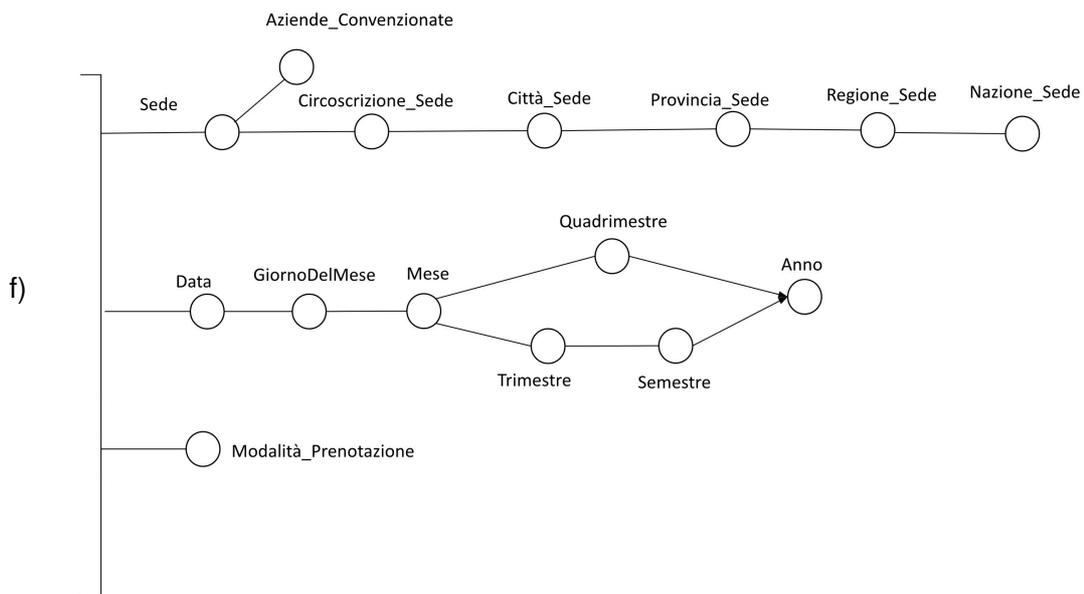
### 1 punto (penalità 15% per risposta sbagliata)

Selezionare, tra le dimensioni proposte di seguito, quelle che soddisfano le richieste descritte nelle specifiche del problema.









- d
  e
  f
  b
  c
  g
  a ✓

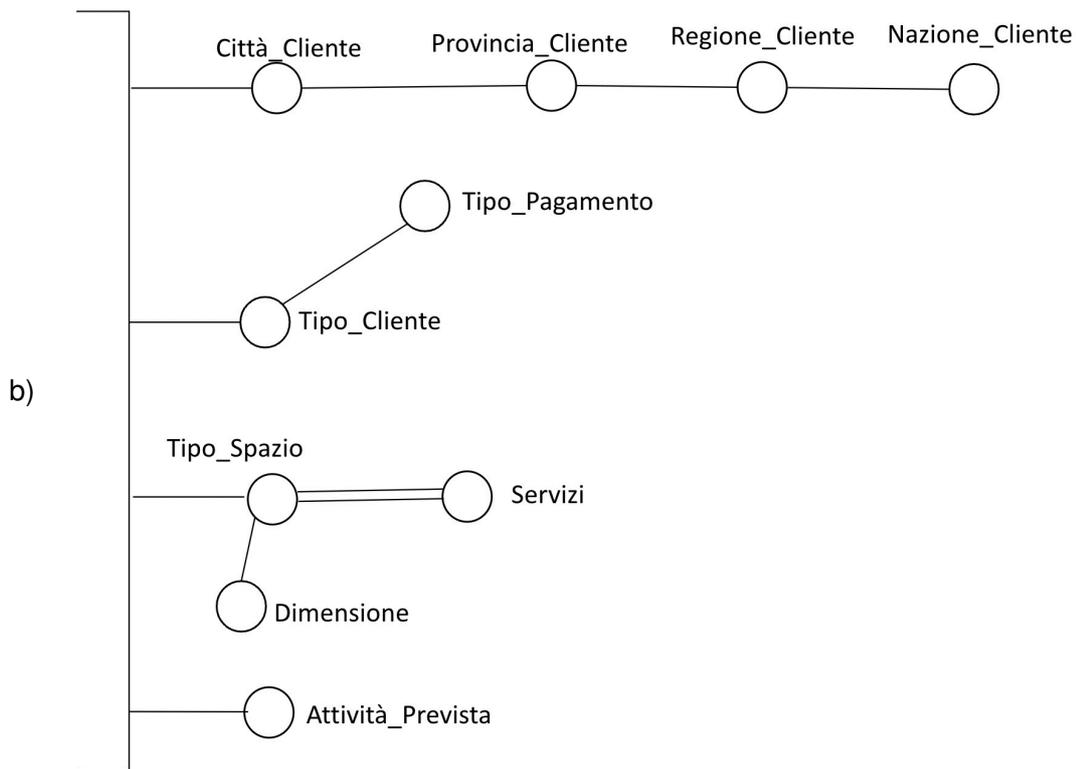
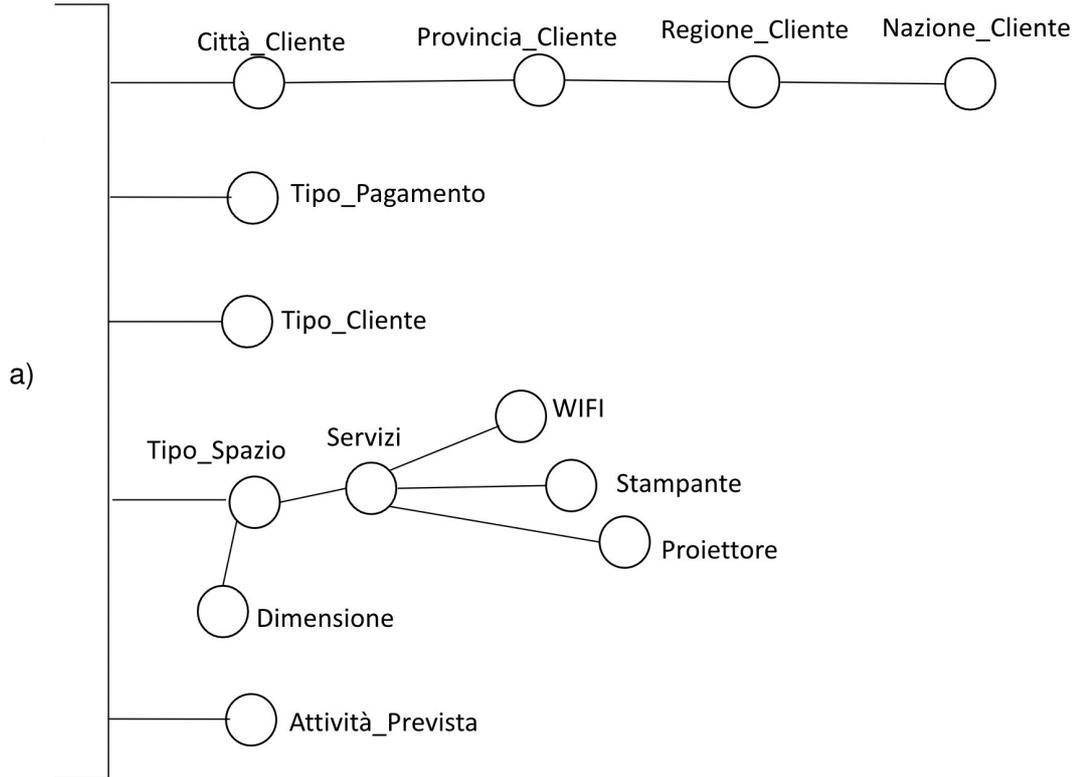
Punteggio ottenuto 1,00 su 1,00

La risposta corretta è: a

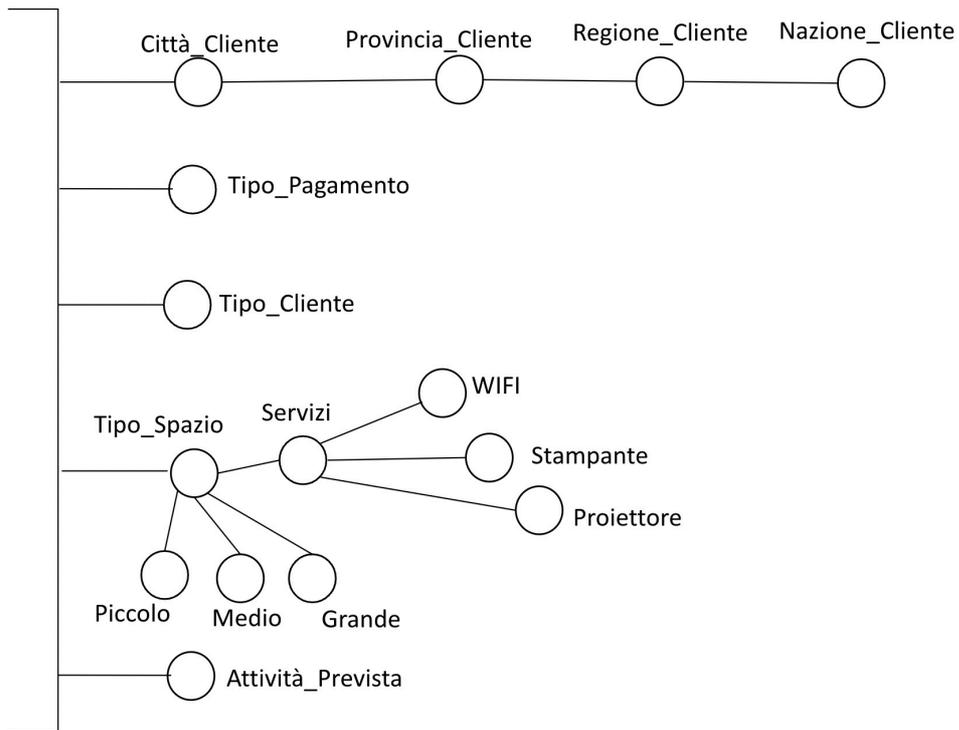
## SCHEMA CONCETTUALE 2

1 punto (penalità 15% per risposta sbagliata)

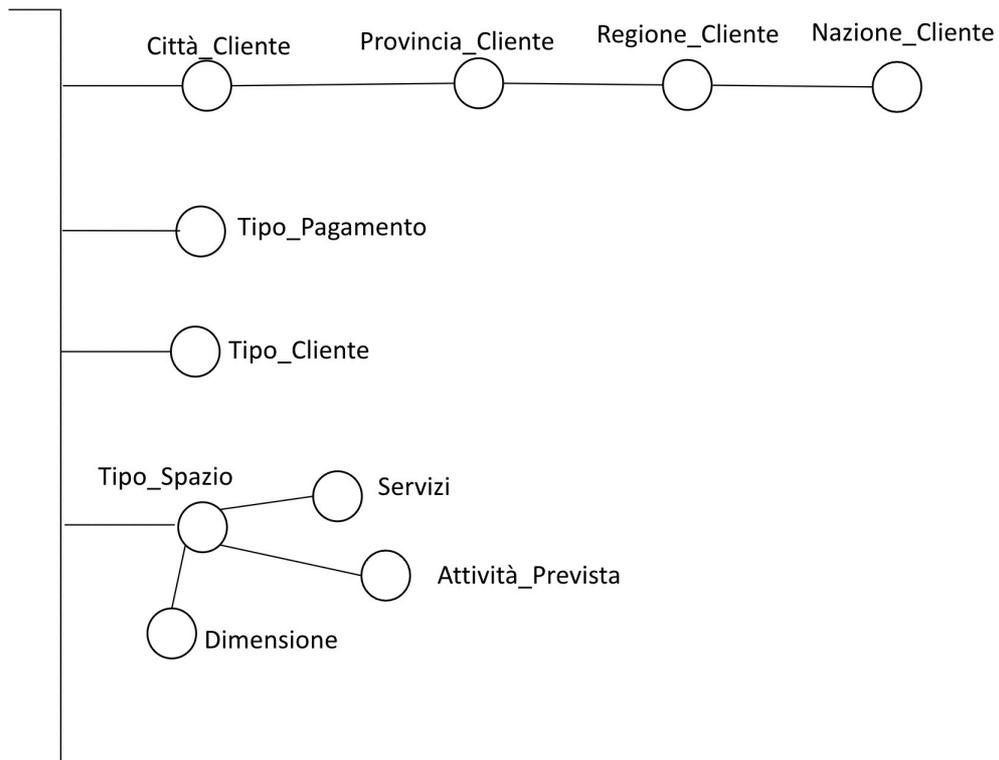
Selezionare, tra le dimensioni proposte di seguito, quelle che soddisfano le richieste descritte nelle specifiche del problema.

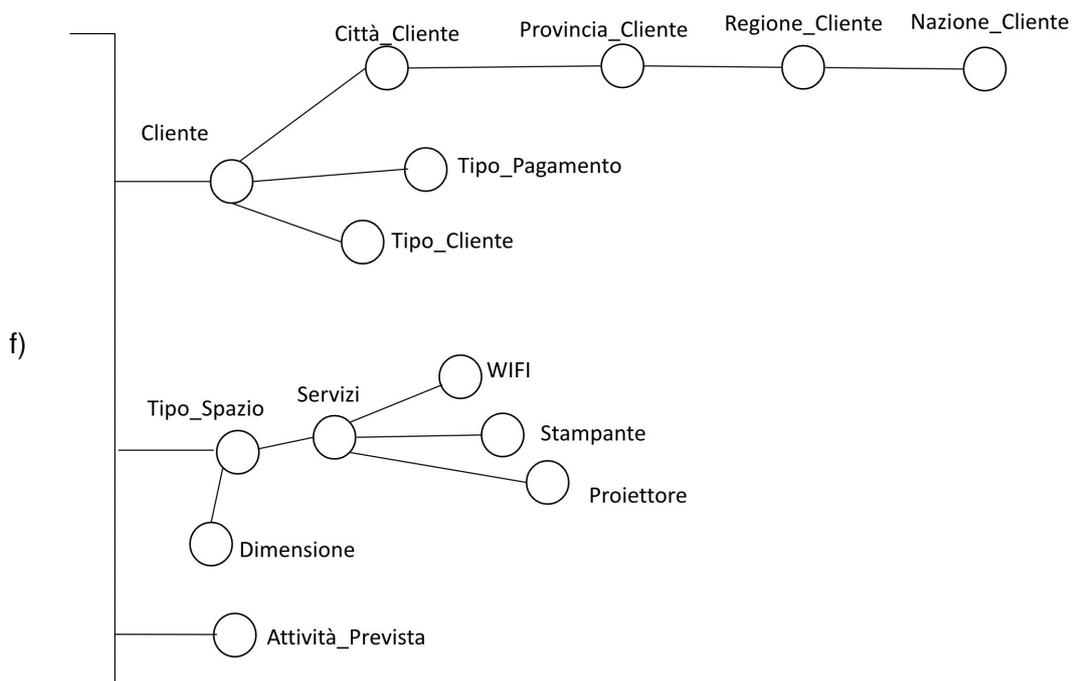
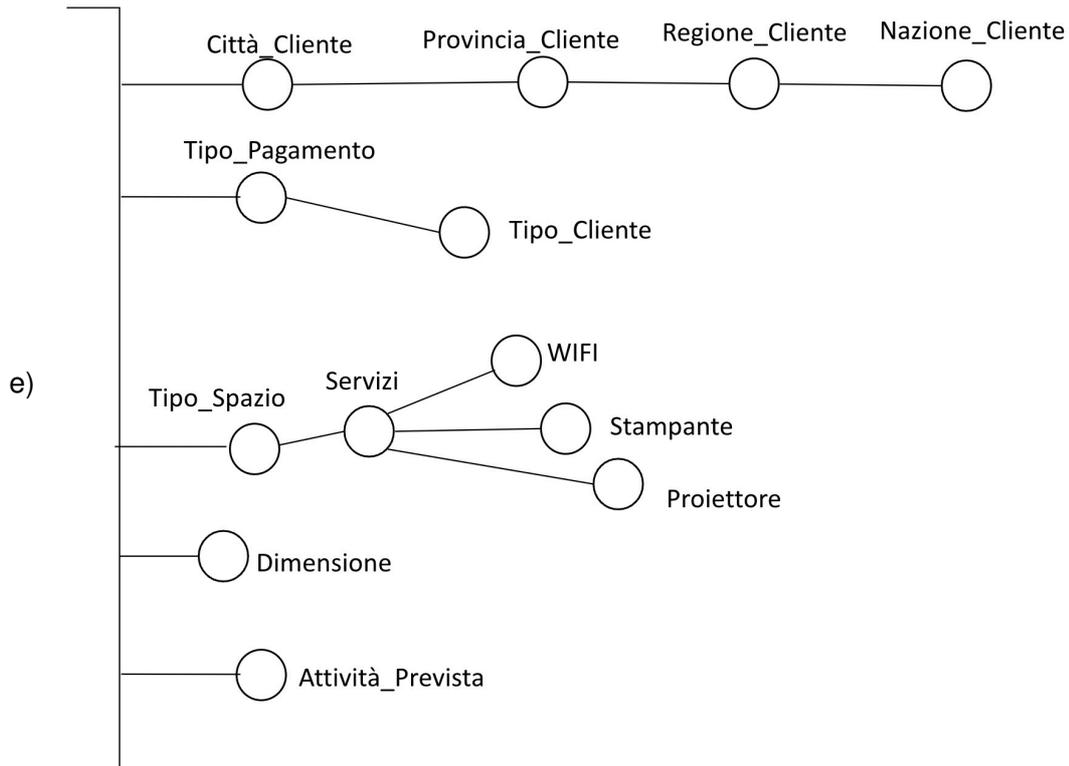


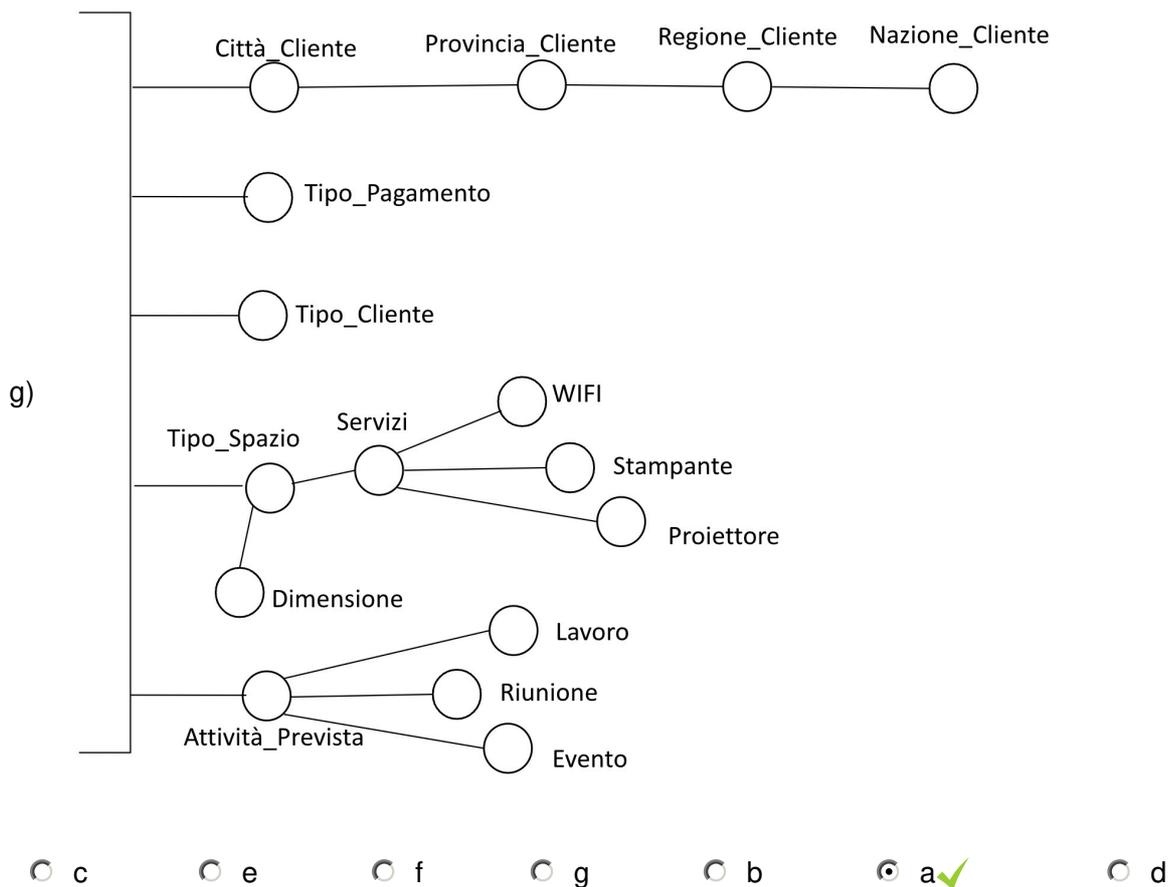
c)



d)







Punteggio ottenuto 1,00 su 1,00

La risposta corretta è: a

## MISURE

### 1 punto (penalità 15% per risposta sbagliata)

Selezionare l'insieme di attributi necessari per modellare correttamente le misure nella tabella dei fatti richieste dalle specifiche.

- Tempo medio espresso in ore di prenotazione di uno spazio, Numero totale di clienti, Importo massimo speso
- Tempo totale espresso in ore di prenotazione di uno spazio, Numero complessivo di prenotazioni, Importo medio speso
- Tempo totale espresso in ore di prenotazione di uno spazio, Numero complessivo di prenotazioni, Importo massimo speso
- Tempo totale espresso in ore di prenotazione di uno spazio, Numero complessivo di prenotazioni, Importo complessivo ✓
- Tempo totale espresso in ore di prenotazione di uno spazio, Numero totale di clienti, Importo

massimo speso

- Tempo medio espresso in ore di prenotazione di uno spazio, Numero complessivo di prenotazioni, Importo complessivo
- Tempo medio espresso in ore di prenotazione di uno spazio, Numero totale di clienti, Importo complessivo
- Tempo medio espresso in ore di prenotazione di uno spazio, Numero totale di clienti, Importo medio speso

Punteggio ottenuto 1,00 su 1,00

La risposta corretta è: Tempo totale espresso in ore di prenotazione di uno spazio, Numero complessivo di prenotazioni, Importo complessivo

- 1) La risposta corretta è : a
- 2) La risposta corretta è : a
- 3) La risposta corretta è : Tempo totale espresso in ore di prenotazione di uno spazio, Numero complessivo di prenotazioni, Importo complessivo

### Domanda 8

Risposta corretta

Punteggio ottenuto 1,50 su 1,50

#### 1.5 punti (-15% di penalità per ogni risposta sbagliata)

##### Definizioni

Precisione(C): frazione degli elementi correttamente classificati nella classe C in tutti i fold, sugli elementi assegnati a C in tutti i fold

Recall(C): frazione degli elementi correttamente classificati nella classe C in tutti i fold, sugli elementi appartenenti a C in tutti i fold

Abbiamo a disposizione un dataset composto da 6000 campioni di cui 2400 appartenenti alla classe positiva (1) e 3600 alla classe negativa (0). Stiamo valutando un classificatore tramite stratified 3-fold cross-validation (mantenendo la distribuzione delle classi).

Il classificatore ottiene i seguenti risultati per ogni fold:

1. 600 predizioni corrette per la classe 1, 1000 predizioni corrette per la classe 0
2. 500 predizioni corrette per la classe 1, 900 predizioni corrette per la classe 0
3. 600 predizioni corrette per la classe 1, 900 predizioni corrette per la classe 0

Quale delle seguenti affermazioni riguardante la **classe 1** è corretta?

- (a) La precisione in almeno un fold supera 0.8
- (b) La precisione è più alta della recall
- (c) Nessuna risposta è corretta
- (d) La recall in almeno un fold è inferiore a 0.6
- (e) La recall è più bassa di 0.6
- (f) La precisione in almeno un fold è inferiore a 0.5
- (g) La precisione è maggiore di 0.9
- (h) La precisione è più bassa della recall ✓

Risposta corretta.

La risposta corretta è: La precisione è più bassa della recall

### Domanda 9

Completo

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.**

---

QUERY 1, c'è scritto solo separatamente per anno, per questo motivo non c'è aggiunta regione-provincia etc., stessa cosa nel sep per regione.

### Domanda 10

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

#### 1 punto (-15% di penalità per ogni risposta sbagliata)

Il seguente documento è preso da un collection MongoDB chiamata "inventario", che contiene le informazioni sui prodotti in vendita in un negozio di alimentari online.

```
{
  "productId": "45956",
  "name": "French Bread",
  "origin": "France",
  "price": 29.99,
  "discount": 0.0,
  "category": "Bread",
  "stock": 100
}
```

Identifica le categorie di prodotti con un prezzo medio più alto di 30 e uno stock massimo maggiore di 100 unità.

---

○ (a)

```
db.collection.aggregate([
  {
    $match: {
      price: { $gt: 30 },
      stock: { $gt: 100 },
    }
  },
  {
    $group: {
      _id: "$category",
      avg_price: { $avg: "$price" },
      max_stock: { $max: "$stock" }
    }
  },
  {
    $project: {
      _id: true,
    }
  },
])
```

⦿ (b)

```
db.collection.aggregate([
  {
    $group: {
      _id: "$category",
      avg_price: { $avg: "$price" },
      max_stock: { $max: "$stock" }
    }
  },
  {
    $match: {
      avg_price: { $gt: 30 },
      max_stock: { $gt: 100 },
    }
  },
  {
    $project: {
      _id: true,
    }
  },
])
```



(c)

```
db.collection.aggregate([
  {
    $group: {
      _id: "$category",
      avg_price: { $avg: "$price" }, { $gt: 30 },
      max_stock: { $max: "$stock" }, { $gt: 100 }
    }
  },
  {
    $project: {
      _id: true,
    }
  },
])
```

(d) Nessuna soluzione è corretta

(e)

```
db.collection.aggregate([
  {
    $group: {
      _id: "$category",
      avg_price: { $avg: "$price" },
      max_stock: { $max: { $sum: "$stock" } }
    }
  },
  {
    $match: {
      avg_price: { $gt: 30 },
      max_stock: { $gt: 100 },
    }
  },
  {
    $project: {
      _id: true,
    }
  },
])
```

(f)

```
db.collection.aggregate([
  {
    $match: {
      price: { $gt: 30 },
      stock: { $gt: 100 },
    }
  },
  {
    $group: {
      _id: "$category",
      avg_price: { $avg: "$price" },
      max_stock: { $max: { $sum: "$stock" } }
    }
  },
  {
    $project: {
      _id: true,
    }
  },
])
```

Risposta corretta.

La risposta corretta è:

```
db.collection.aggregate([
  {
    $group: {
      _id: "$category",
      avg_price: { $avg: "$price" },
      max_stock: { $max: "$stock" }
    }
  },
  {
    $match: {
      avg_price: { $gt: 30 },
      max_stock: { $gt: 100 },
    }
  },
  {
    $project: {
      _id: true,
    }
  },
])
```

### Domanda 11

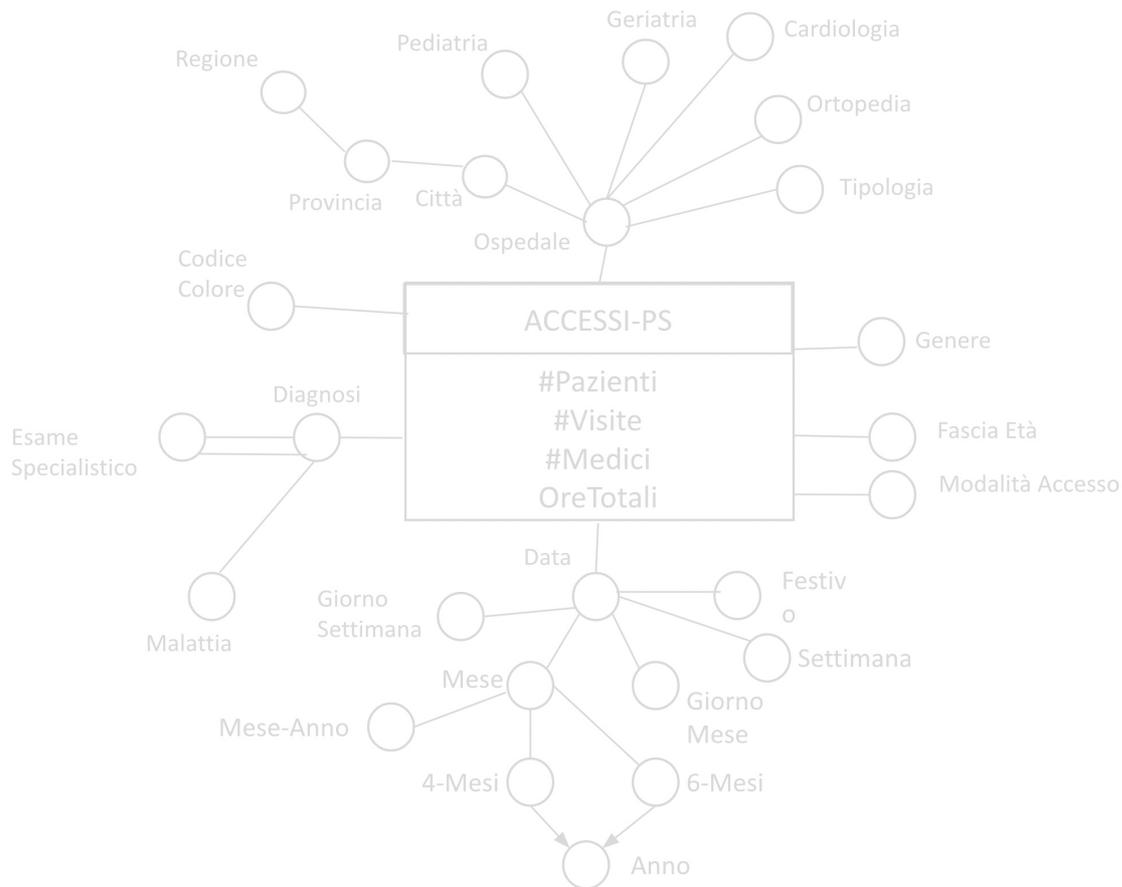
Completo

Punteggio ottenuto 5,00 su 5,00

#### 5 punti (nessuna penalità per risposta sbagliata)

Il seguente data warehouse memorizza le informazioni relative agli accessi effettuati da pazienti (caratterizzati da fascia di età e genere) al pronto soccorso degli ospedali siti in Italia. Ogni ospedale appartiene ad una singola tipologia (pubblico o privato) e vi possono essere presenti uno o più reparti tra pediatria, geriatria, ortopedia, e cardiologia (questi attributi sono booleani e rappresentano una configurazione). Al triage viene assegnato un codice colore tra (bianco, verde, azzurro, giallo e rosso) e si tiene traccia della modalità di accesso al pronto soccorso (in ambulanza o autonomia). All'accesso al pronto soccorso viene assegnato una diagnosi caratterizzata da uno o più esami specialistici.

Il data warehouse è caratterizzato dal seguente schema concettuale e dal corrispondente schema logico. Ogni evento è caratterizzato da quattro misure: numero pazienti, numero visite, numero medici che hanno effettuato le visite e ore di permanenza al pronto soccorso (OreTotali).



CARATTERISTICHE\_PAZIENTE(**CodCP**, FasciaEtà, Genere)

TRIAGE(**CodTriage**, ModalitàArrivo, CodColore)

OSPEDALE(**CodO**, Ospedale, Tipologia, Città, Provincia, Regione, Pediatria, Geriatria, Ortopedia, Cardiologia)

DIAGNOSI(**CodD**, Diagnosi, Malattia)

DIAGNOSI-ESAME(**CodD**, EsameSpecialistico)

TEMPO(**CodT**, Data, GiornoSettimana, Settimana, GiornoMese, Festivo, Mese, MeseAnno, 4-Mesi, 6-Mesi, Anno)

ACCESSI-PRONTO-SOCCORSO(**CodCP**, **CodTriage**, **CodO**, **CodD**, **CodT**, #Pazienti, #Visite, #Medici, OreTotali)

Dato lo schema logico precedente, considerare le seguenti query di interesse:

1. Considerando i triage con colore giallo, separatamente per regione e anno, visualizzare il numero totale di visite, le ore medie di permanenza al pronto soccorso per paziente e il numero medio di pazienti per provincia.
2. Considerando la modalità di arrivo con ambulanza e i triage con colore rosso, visualizzare il valore cumulativo annuale del numero di pazienti al trascorrere dei mesi.
3. Considerando gli ospedali della regione Piemonte con reparto di ortopedia ma non di geriatria, separatamente per provincia e semestre (attributo 6-Mesi), visualizzare il numero complessivo di pazienti e il numero medio di visite per paziente.

Dato lo schema logico precedente, si svolgano le seguenti attività:

1. Definire una vista materializzata con CREATE MATERIALIZED VIEW, in modo da ridurre il tempo di risposta delle query di interesse da (1) a (3) sopra riportate. In particolare si specifichi la query SQL associata al Blocco A nella seguente istruzione:

```
CREATE MATERIALIZED VIEW ViewAccessi
BUILD IMMEDIATE
REFRESH FAST ON COMMIT
AS
Blocco A
```

2. Definire l'**insieme minimale** di attributi che permette di identificare le tuple appartenenti alla vista materializzata ViewAccessi.

3. Si ipotizzi che la gestione della vista materializzata (tabella derivata) sia svolta mediante trigger. Scrivere il trigger per propagare le modifiche alla vista materializzata ViewAccessi in caso di inserimento di un nuovo record nella tabella dei fatti ACCESSI\_PRONTO\_SOCCORSO.

---

1. SELECT SUM(#Visite), SUM(#Ore), SUM(#Pazienti),  
W CodColore="Giallo",  
GB Regione, Anno, Provincia

2. SELECT SUM(#Pazienti)  
W ModArrivo="Ambulanza", CODCOLORE  
GB Mesi, Anno

3. SELECT SUM(#PAZIENTI),  
W ort=1 ger=0 REG=PIEMONTE  
GB Provincia, 6-Mesi

-----  
BLOCCO A

```
SELECT Provincia, Regione, Mese, 6-Mesi, Anno, CodColore, ModArrivo, Ortopedia, Geriatria,
SUM(#Visite) AS TotVisite, SUM(#Pazienti) AS TotPazienti, SUM(OreTotali) AS TotOre
FROM OSPEDALE O, ACCESSI-PRONTO-SOCCORSO APS, TRIAGE TR, TEMPO T
WHERE APS.CodO=O.CodO AND APS.CodT=T.CodT AND APS.CodTriage=TR.CodTriage
GROUP BY Provincia, Regione, Mese, 6-mesi, Anno, CodColore, ModArrivo, ortopedia, Geriatria
-----
```

2.

Insieme minimale: Provincia, Mese, CodColore, ModArrivo, Ortopedia, Geriatria  
-----

```

CREATE OR REPLACE TRIGGER t1
AFTER INSERT ON ACCESSI_PRONTO_SOCCORSO
FOR EACH ROW

DECLARE
tprov, treg, tmese, t6, tanno, tcodcolore, tmodarrivo, tortopedia, tgeriatria VARCHAR(20);
n INT;

BEGIN

SELECT Mese, 6-Mesi, Anno INTO tmese, t6, tanno
FROM TEMPO T
WHERE T.CodT=:new.CodT;

SELECT Provincia, Regione, Ortopedia, Geriatria INTO tprov, treg, tortopedia, tgeriatria
FROM OSPEDALE O
WHERE O.CodO= :new.CodO;

SELECT ModalitaArrivo, CodColore INTO tmodarrivo, tcodcolore
FROM TRIAGE TR
WHERE TR.CodTriage= :new.CodTriage;

SELECT COUNT(*) INTO n
FROM ViewAccessi
WHERE Provincia=tprov AND Mese=tmese AND tcodcolore=CodColore AND
ModalitaArrivo=tModArrivo
AND Ortopedia=tortopedia AND Geriatria = tGeriatria;

IF n > 0 THEN
UPDATE ViewAccessi
SET TotVisite= TotVisite + :new.#Visite, TotOre= TotORe + :new.OreTotali, TotPazienti=
TotPazienti + :new.#Pazienti
WHERE Provincai=tprov AND Mese=tmese AND CodColore=tcodColore AND ModalitaArrivo=
tModArrivo AND Ortopedia=tortopedia AND Geriatria=tGeriatria;

ELSE
INSERT INTO ViewAccessi
VALUES (tprov, treg, tmese, t6, tanno, tcodcolore, tmodarrivo, tortopedia, tgeriatria, :new.#Visite,
:new.#pazienti, :new.Oretotali);

ENDIF;
END;

```

**Query di interesse:**

(a) Considerando i triage con colore giallo, separatamente per regione e anno, visualizzare il numero totale di visite, le ore medie di permanenza al pronto soccorso per paziente e il numero medio di pazienti per provincia.

```
Select Regione, Anno, SUM(#Visite), SUM(OreTotali)/SUM(#Pazienti), SUM(#Pazienti)/
COUNT(DISTINCT Provincia)
FROM TRIAGE, OSPEDALE, TEMPO, ACCESSI_PRONTO_SOCCORSO
WHERE joinAND CodColore = 'giallo'
Group by Regione, Anno
```

(b) Considerando la modalità di arrivo con ambulanza e i triage con colore rosso, visualizzare il valore cumulativo annuale del numero di pazienti al trascorrere dei mesi.

```
Select Mese, Anno, SUM(SUM(#Pazienti)) OVER (PARTITION BY Anno
ORDER BY Mese
ROWS UNBOUNDED PRECEDING)
FROM TRIAGE, TEMPO, ACCESSI_PRONTO_SOCCORSO
WHERE join AND ModalitàArrivo = 'ambulanza' AND CodColore = 'rosso'
Group by Mese, Anno
```

(c) Considerando gli ospedali della regione Piemonte con reparto di ortopedia ma non di geriatria, separatamente per provincia e semestre (attributo 6-Mesi), visualizzare il numero complessivo di pazienti e il numero medio di visite per paziente.

```
Select Provincia, 6-Mesi, SUM(#NumpPazienti), SUM(#Visite)/SUM(#NumpPazienti)
FROM TRIAGE, OSPEDALE, TEMPO, ACCESSI_PRONTO_SOCCORSO
WHERE join AND Regione = 'Piemonte' AND Ortopedia = 1 AND Geriatria =0
Group by Provincia, 6-Mesi
```

### 1. Blocco A - Query per vista materializzata

```
SELECT Mese, 6-Mesi, Anno, Provincia, Regione, Ortopedia, Gerliatria, ModalitàArrivo,
CodColore, SUM(#Visite) AS VisiteTot, SUM(OreTotali) AS OreTot, SUM(#Pazienti) AS
PazientiTot
FROM OSPEDALE O, TEMPO T, TRIAGE TR, ACCESSI_PRONTO_SOCCORSO A
WHERE O.CodO = A.CodO AND T.CodT = A.CodT AND TR.CodTriage = A.CodTriage
GROUP BY Mese,6-Mesi,Anno, Provincia, Regione, ModalitàArrivo, CodColore,
Ortopedia,Geriatria
```

### 2. Identificatore

Mese, Provincia, ModalitàArrivo, CodColore, Ortopedia, Geriatria

### 3. Trigger

```
CREATE OR REPLACE TRIGGER MaintenanceViewAccessi
AFTER INSERT ON ACCESSI_PRONTO_SOCCORSO
FOR EACH ROW
DECLARE
```

```

VarMese, VarAnno, Var6Mesi DATE;
VarProvincia, VarRegione, VarModArr, VarColore VARCHAR(10); VarOrto, VarGer BOOLEAN;
N INTEGER;
BEGIN
    SELECT Mese, 6-Mesi, Anno INTO VarMEse, Var6Mesi, VarAnno
    FROM TEMPO
    WHERE CodT = :NEW.CodT;

    SELECT ModalitàArrivo, CodColore INTO VarModaArr, VarColore
    FROM TRIAGE
    WHERE CodTriage = :NEW.CodTriage;

    SELECT Provincia, Regione, Ortopedia, Geriatria INTO VarProvincia, VarRegione, VarOrto,
VarGer
    FROM OSPEDALE
    WHERE CodO = :NEW.CodO;

    SELECT COUNT(*) INTO N
    FROM ViewAccessi
    WHERE Mese = VarMese AND Provincia = VarProvincia AND Ortopedia = VarOrto AND
Geriatria =VarGer AND ModalitàArrivo = VarMod AND CodColore = VarColore;

    IF N>0 THEN
        UPDATE ViewAccessi
        SET VisiteTot = VisiteTot +:NEW.#Visite, OreTot = OreTot + :NEW.OreTotali, PazientiTot =
PazientiTot + :NEW .#Pazienti
        WHERE Mese = VarMese AND Provincia = VarProvincia AND Ortopedia = VarOrto AND
Geriatria =VarGer AND ModalitàArrivo = VarMod AND CodColore = VarColore;

    ELSE

        INSERT INTO ViewAccessi(...) VALUES (VarMese, Var6Mesi, VarAnno, VarProvincia,
VarRegione, VarOrto, VarGer, VarModArr, VarColore, :NEW #Visite, :NEW.OreTotali, :NEW
.#Pazienti);

    END IF;
END;

```

Commento: