

COGNOME

NOME

MATRICOLA

Basi di Dati – 14 giugno 2021

Esercizio 1. (NORMALIZZAZIONE)

Considerare la relazione seguente:

CF	NomePaz	CT	Data	Esito	CodASL	ASL	SedeASL	IDReg	NomeReg
RSSMRI...	M. Rossi	1	12/12/20	Pos	1	RM 1	Via Po	LAZ	Lazio
RSSMRI...	M. Rossi	2	12/01/21	Neg	1	RM 1	Via Po	LAZ	Lazio
RSSMNL...	M. Rossi	3	15/12/20	Dub	1	RM 1	Via Po	LAZ	Lazio
RSSMNL...	M. Rossi	4	15/12/20	Neg	1	RM 1	Via Po	LAZ	Lazio
BRNLGU...	L. Bruni	5	12/01/21	Pos	3	LT	Via Italia	LAZ	Lazio
BNILCA ...	L. Bini	6	12/12/20	Neg	1	Ma 1	Via Roma	MAR	Marche
VRDPRI ...	P. Verdi	7	15/12/20	Pos	1	Mo 1	Corso ...	MOL	Molise

La relazione mostra (in forma non normalizzata) i dati relativi ad un insieme di tamponi eseguiti su pazienti, secondo le seguenti specifiche:

- ogni tampone viene eseguito su un paziente;
- ogni tampone ha un codice (CT), univoco, una data e un esito;
- ogni paziente ha un codice fiscale e un nome e appartiene ad una ASL;
- ogni ASL si trova in una regione, ha un codice (CodASL) che la identifica nell'ambito della regione e ha una sede;
- ogni regione ha un identificatore (IDReg) e un nome.

Con riferimento alle specifiche e ai dati forniti:

- a) mostrare le dipendenze funzionali rilevate (limitarsi a quelle che hanno a sinistra identificatori o codici);

$CT \rightarrow Data, Esito, CF$

$CF \rightarrow NomePaz, CodASL, IdReg$

~~$CodASL \rightarrow Sede, IdRegione$~~ $CodASL, IdReg \rightarrow ASL, SedeASL$

$IdReg \rightarrow NomeReg$

- b) individuare la chiave (o le chiavi) della relazione;

$\{CT\}^+ = Data, Esito, CT, CF, NomePaz, CodASL, Sede, IdRegione, NomeReg$
 = tutti gli attributi

CT è chiave



- c) spiegare perché essa non soddisfa la BCNF;

Perché non tutte le dip. funz. contengono o ss la chiave CT



COGNOME

NOME

MATRICOLA

- d) mostrare una decomposizione della relazione originaria che soddisfi la BCNF (mostrare le tabelle, con i dati, indicando le dipendenze associate a ciascuna tabella e la chiave);

Tempone (CT , Data, Esito, CF) $CT \rightarrow Data, Esito, CF$
 Personale (CF , Nome Pers, Cod ASL) $CF \rightarrow Nome Pers, Cod ASL$
 ASL (Cod ASL , Sede, Id Regione) $Cod ASL \rightarrow Sede, Id Regione$
 Regione (Id Reg , Nome Reg) $Id Reg \rightarrow Nome Reg$
 Almeno una relazione contiene CT ✓

[Tabelle queste, si sottolinea solo da alcune tabelle condensano tuple duplicate in una, vedi ASL]

e' in BCNF perché ogni dip contiene o s'è la chiave della relazione o cui si riferisce



- e) spiegare quali proprietà (decomposizione senza perdita, preservazione delle dipendenze) sono soddisfatte dalla decomposizione e perché.

La decomposizione ottenuta è in BCNF, quindi 3NF dunque soddisfa lossless join e preservazione delle dip. La prima perché almeno una relazione contiene la chiave; la seconda è garantita dall'algoritmo di decomposizione in quanto ad ogni tabella associa una dip. funz.

di conseguenza
 • → si osserva che il join delle relaz. decomp. genera l'otenzia iniziale



COGNOME	NOME	MATRICOLA
---------	------	-----------

Esercizio 2. (REVERSE ENGINEERING)

Dato il seguente schema logico relazionale

SEDE(IdSede, *NomeSede*, Indirizzo, CAP, Comune)

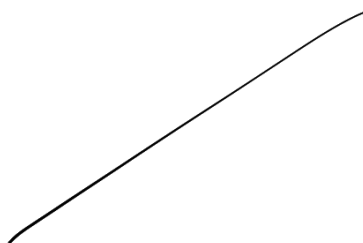
DIPENDENTE(Email, Nome, Cognome, Qualifica, Telefono, IdSede^{SEDE}, Dipartimento^{DIPARTIMENTO})

AULA(IdAula, IdSede^{SEDE}, NomeAula_o, TipoAula, Capienza, Accessibile, Dipartimento^{DIPARTIMENTO})

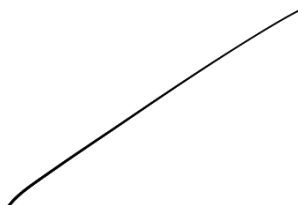
PRENOTAZIONE(IdAula^{AULA}, IdSede^{AULA}, Data, OraI, OraF, Descrizione, TipoAttività, PartecipantiAttesi, PrenotataDa^{DIPENDENTE}, ConfermataDa^{DIPENDENTE})

DIPARTIMENTO(Sigla, *Nome*, *Direttore*^{DIPENDENTE}) Nome e Direttore sono entrambe chiavi alternative (due chiavi, ciascuna costituita da un attributo)

- a) si proponga uno schema concettuale Entity Relationship la cui traduzione dia luogo a tale schema logico



- b) si modifichi lo schema per gestire il fatto che una singola prenotazione (che sarà identificata ora da un id), fatta da un dipendente e (eventualmente) confermata da un dipendente per una determinata aula relativamente a un'attività (di un certo tipo, con un certo numero di partecipanti attesi), possa coinvolgere più slot orari [attività che si ripete in diversi momenti di tempo]



COGNOME**NOME****MATRICOLA****Esercizio 3. (ALGEBRA RELAZIONALE)**

In riferimento al seguente schema:

SEDE(IdSede, NomeSede, Indirizzo, CAP, Comune)

DIPENDENTE(Email, Nome, Cognome, Qualifica, Telefono, IdSede^{SEDE})AULA(IdAula, IdSede^{SEDE}, NomeAula, TipoAula, Capienza, Accessibile)PRENOTAZIONE(IdAula^{AULA}, IdSede^{AULA}, Data, OraI, OraF, Descrizione, TipoAttività, PartecipantiAttesi, PrenotataDa^{DIPENDENTE}, ConfermataDa^{DIPENDENTE})**Formulare le seguenti interrogazioni in algebra relazionale**

- a) Determinare le aule di tipo laboratorio della sede (il cui nome è) Valletta Puggia in cui non sono mai state effettuate prenotazioni per attività di tipo esame

$$\pi_{IdAula, NomeAula} \left(\sigma_{TIPO = 'Laboratorio' \text{ AND } NomeSede = 'Valletta Puggia'} \right. \\ \left. (AULA \bowtie SEDE) \right)$$

$$- \pi_{IdAula, NomeAula} \left(\sigma_{\substack{TIPO = 'Laboratorio' \text{ AND } NomeSede = 'Valletta Puggia' \\ \text{AND } TIPOATTIVITA' = 'Esame'}} \right. \\ \left. ((AULA \bowtie SEDE) \bowtie PRENOTAZIONE) \right)$$

NON NECESSARIO

✓

- b) Determinare le aule della sede (il cui nome è) Valletta Puggia in cui sono state effettuate prenotazioni per attività di **tutti** i tipi

$$\pi_{IdAula, NomeAula, TipoAttività} \left(\sigma_{NomeSede = 'Valletta Puggia'} (SEDE) \bowtie PRENOTAZIONE \right)$$

↑
Altrimenti non c'è
predica su cui fare ÷

$$\div \pi_{TIPOATTIVITA} (PRENOTAZIONE)$$

COGNOME	NOME	MATRICOLA
---------	------	-----------

Esercizio 4. (SQL)

In riferimento al seguente schema:

SEDE(IdSede, NomeSede, Indirizzo, CAP, Comune)

DIPENDENTE(Email, Nome, Cognome, Qualifica, Telefono, IdSede^{SEDE})

AULA(IdAula, IdSede^{SEDE}, NomeAula, TipoAula, Capienza, Accessibile)

PRENOTAZIONE(IdAula^{AULA}, IdSede^{AULA}, Data, OraI, OraF, Descrizione, TipoAttività, PartecipantiAttesi, PrenotataDa^{DIPENDENTE}, ConfermataDa^{DIPENDENTE})

Formulare le seguenti interrogazioni in SQL

- a) Determinare per ogni sede il numero di prenotazioni effettuate, il numero medio di partecipanti attesi e quanti dipendenti diversi hanno confermato le prenotazioni

```
SELECT IdSede, COUNT(*), AVG(PartAtt), COUNT(DISTINCT ConfDa)
FROM Sede natural join Aula natural join Prenotazione
GROUP BY IdSede
```



- b) Determinare le aule la cui capienza è superiore alla capienza media delle aule dello stesso tipo nella stessa sede

```
SELECT IdAula
FROM Aula A
WHERE Capienza >
```

```
( SELECT AVG(Capienza)
  FROM Aula
 WHERE IdSede = A.IdSede
   AND TipoAula = A.TipoAula)
```



15420765101 11454232046 27401706422 75721021601 24307131652 43135317123 23411544745

COGNOME	NOME	MATRICOLA
---------	------	-----------

COGNOME

NOME

MATRICOLA

PARTE III. DOMANDE, SOLO PER 12 CFU

- a) Descrivere il livello di isolamento READ COMMITTED, specificando quando devono essere acquisiti e rilasciati i lock condivisi ed esclusivi secondo lo standard, quali anomalie permette di evitare e perché. Usare un esempio se lo si ritiene opportuno.

Il livello read committed permette solo anomalie di irrepeatable reads e phantom rows, presto per determinare una minor rigidità rispetto allo strong phasing. I lock esclusivi vengono ~~acquisiti immediatamente~~ e rilasciati al termine della transazione; quelli condivisi vengono acquisiti e rilasciati appena possibile.



- b) Descrivere le politiche steal e no-steal utilizzate nell'implementazione del protocollo Write-Ahead-Logging.

Fissato uno stato intermedio della transazione (ancora non ancora committed) la scrittura su disco di una pagina modificata nel buffer segue una di due politiche: steal e no steal. ~~steal determina una scrittura immediata dopo la modifica sul buffer, di contro ha molte scritture intermedie.~~ ^{quando più conviene,} ~~no steal determina una scrittura ritardata al momento più opportuno,~~ ^{del commit} di contro questo tiene in memoria parecchie pagine portando potenzialmente all'esaurimento della memoria centrale e dunque scrittura necessaria su disco.

COGNOME	NOME	MATRICOLA
---------	------	-----------

c) Descrivere la fase di ottimizzazione fisica delle interrogazioni, illustrandone i vari passi.

La fase di ottimizzazione fisica segue i seguenti passi:

- si parte da un LAP, ovvero un piano logico già ottimizzato
 \rightarrow si trova il risultato mediante algoritmi non catalogati
- si determinano ed enumerano tutti i piani fisici ammissibili per quel LAP
- si filtrano tali piani sulla base di costi ed euristiche (si trova il costo)
- viene scelto un LAP, ovvero un piano fisico, definitivo e ottimizzato
- viene processato il piano fisico, elaborando l'interrogazione e restituendo un query execution plan