

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);

CF → NomePaz, CodASL, IDReg (3)

CT → CF, Data, Esito (1+2)

CodASL, IDReg → ASL, SedeASL (4)

IDReg → NomeReg (5)

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

CT

Infatti, CT+ = {CT, CF, Data, Esito, NomePaz, CodASL, IDReg, ASL, SedeASL, NomeReg}

- c) spiegare perché essa non soddisfa la BCNF;

Non soddisfa BCNF perché alcune dipendenze non contengono una chiave nella parte sinistra (ad esempio IDReg → NomeReg)

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);

Applichiamo algoritmo e otteniamo:

Paziente(CF, NomePaz, CodASL<sup>ASL</sup>, IDReg<sup>ASL</sup>)

CF → NomePaz, CodASL, IDReg

Chiave CF

Tampone(CT, CF<sup>Paziente</sup>, Data, Esito)

CT → CF, Data, Esito

Chiave CT

ASL(CodASL, IDReg<sup>Regione</sup>, ASL, SedeASL)

CodASL, IDReg → ASL, SedeASL

Chiave CodASL, IDReg

Regione(IDReg, NomeReg)

IDReg → NomeReg

Chiave IDReg

Esiste già una relazione che contiene la chiave (CT) quindi non dobbiamo aggiungere altro.

La scomposizione è in BCNF: ogni dipendenza contiene a sinistra la chiave della relazione a cui si riferisce

<u>CF</u>	NomePaz	CodASL	IDReg
RSSMRI...	M. Rossi	1	LAZ
RSSMNL...	M. Rossi	1	LAZ
BRNLGU...	L. Bruni	3	LAZ
BNILCA ...	L. Bini	1	MAR
VRDPRI ...	P. Verdi	1	MOL

CF	<u>CT</u>	Data	Esito
RSSMRI...	1	12/12/20	Pos
RSSMNL...	2	12/01/21	Neg
RSSMNL...	3	15/12/20	Dub
RSSMNL...	4	15/12/20	Neg
BRNLGU...	5	12/01/21	Pos
BNILCA ...	6	12/12/20	Neg
VRDPRI ...	7	15/12/20	Pos

<u>CodASL</u>	ASL	SedeASL	IDReg
1	RM 1	Via Po	LAZ
3	LT	Via Italia	LAZ
1	Ma 1	Via Roma	MAR
1	Mo 1	Corso ...	MOL

IDReg	NomeReg
LAZ	Lazio
MAR	Marche
MOL	Molise

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

La decomposizione è senza perdita: per ogni coppia di relazioni in cui abbiamo decomposto la relazione iniziale, gli attributi comuni, se esistono, contengono la chiave di almeno una delle relazioni decomposte.

Come conseguenza, osservando l'istanza, si osserva che il join di tutte le relazioni decomposte genera l'istanza della relazione iniziale [ATTENZIONE: questo non basta per dire che la decomposizione è senza perdita perché per altre istanze della relazione si potrebbe ottenere un comportamento differente].

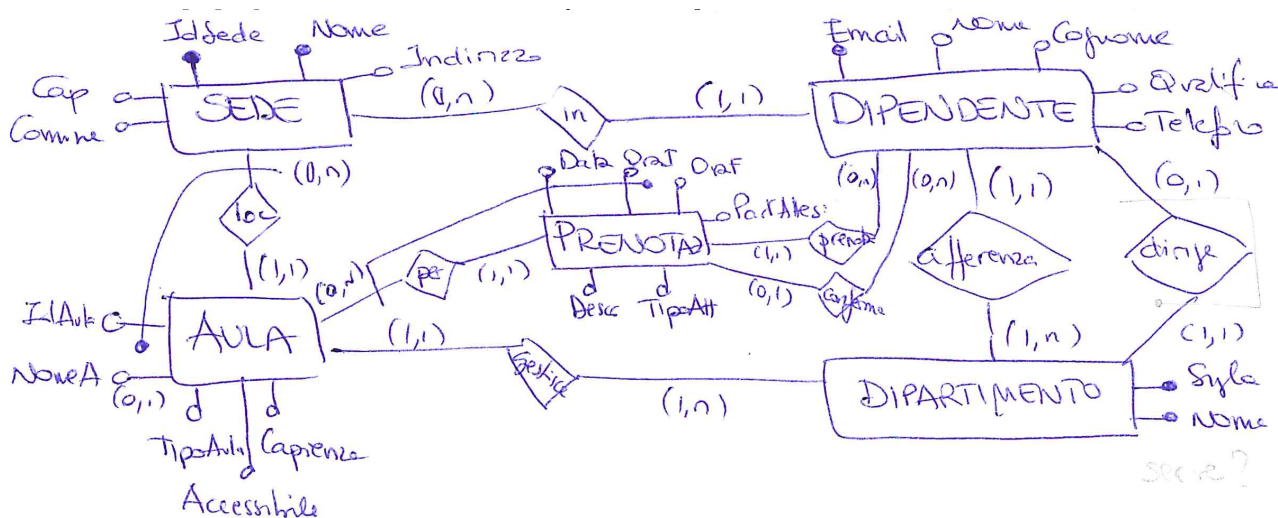
La decomposizione preserva le dipendenze: gli attributi di ogni dipendenza individuata inizialmente sono presenti in uno schema generato dalla scomposizione (quindi ogni dipendenza iniziale diventa una dipendenza per uno degli schemi ottenuti dalla scomposizione).

**COGNOME****NOME****MATRICOLA****Esercizio 2. (REVERSE ENGINEERING)**

Dato il seguente schema logico relazionale

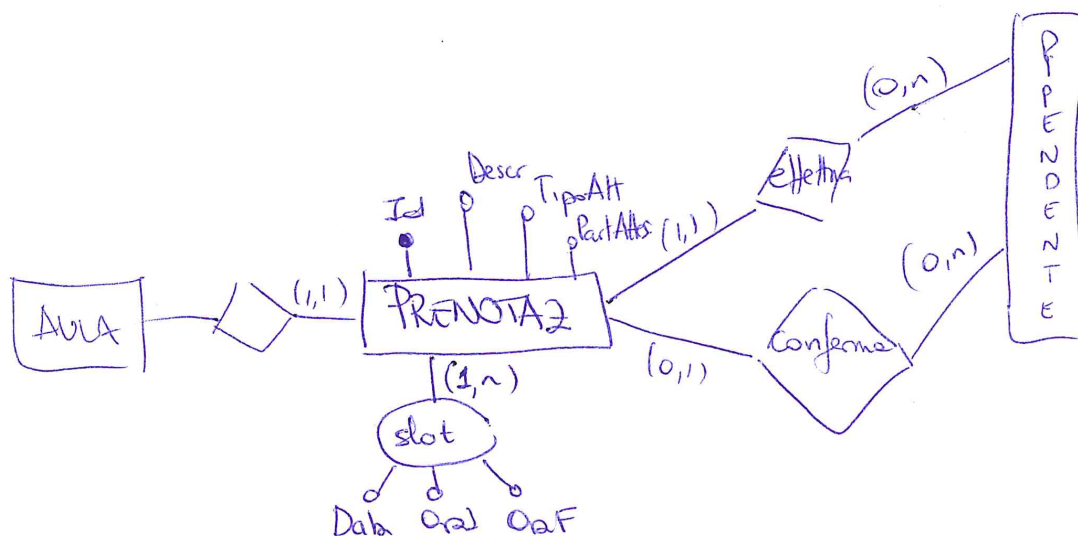
SEDE(IdSede, NomeSede, Indirizzo, CAP, Comune)DIPENDENTE(Email, Nome, Cognome, Qualifica, Telefono, IdSede<sup>SEDE</sup>, Dipartimento<sup>DIPARTIMENTO</sup>)AULA(IdAula, IdSede<sup>SEDE</sup>, NomeAula, TipoAula, Capienza, Accessibile, Dipartimento<sup>DIPARTIMENTO</sup>)PRENOTAZIONE(IdAula<sup>AULA</sup>, IdSede<sup>AULA</sup>, Data, OraI, OraF, Descrizione, TipoAttività, PartecipantiAttesi, PrenotataDa<sup>DIPENDENTE</sup>, ConfermataDa<sup>DIPENDENTE</sup>)DIPARTIMENTO(Sigla, Nome, Direttore<sup>DIPENDENTE</sup>) 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



L'identificazione esterna di dipartimento attraverso dirige può essere indicata come no, non aggiunge informazioni e non cambia la traduzione

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<sup>SEDE</sup>)

AULA(IdAula, IdSede<sup>SEDE</sup>, NomeAula, TipoAula, Capienza, Accessibile)

PRENOTAZIONE(IdAula<sup>AULA</sup>, IdSede<sup>AULA</sup>, Data, OraI, OraF, Descrizione, TipoAttività, PartecipantiAttesi, PrenotataDa<sup>DIPENDENTE</sup>, ConfermataDa<sup>DIPENDENTE</sup>)

**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

$$\pi_{IdAula, IdSede} \left( \sigma_{TipoAula = 'laboratorio'} (AULA) \right. \\ \left. \wedge \sigma_{NomeSede = 'Valletta Puggia'} (SEDE) \right)$$

$$\pi_{IdAula, IdSede} \left( \sigma_{TipoAttività = 'esame'} (PRENOTAZIONE) \right)$$

**NOTA:** IdAula da solo non è chiave per Aula

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

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

÷

$$\pi_{TipoAttività} (PRENOTAZIONE)$$

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

#### Esercizio 4. (SQL)

In riferimento al seguente schema:

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

DIPENDENTE(Email, Nome, Cognome, Qualifica, Telefono, IdSede<sup>SEDE</sup>)

AULA(IdAula, IdSede<sup>SEDE</sup>, NomeAula, TipoAula, Capienza, Accessibile)

PRENOTAZIONE(IdAula<sup>AULA</sup>, IdSede<sup>AULA</sup>, Data, OraI, OraF, Descrizione, TipoAttività, PartecipantiAttesi, PrenotataDa<sup>DIPENDENTE</sup>, ConfermataDa<sup>DIPENDENTE</sup>)

#### 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(PartecipantiAttesi),
       COUNT(DISTINCT ConfermataDa)
FROM 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, IdSede
FROM AULA X
WHERE Capienza > (SELECT AVG(Capienza)
                  FROM AULA
                  WHERE TipoAula = X.TipoAula
                  AND IdSede = X.IdSede)
```