Compito Basi di Dati. Tempo concesso: 2 ore **23 Giugno 2022**

Nome: _____ Cognome: _____ Matricola: _____

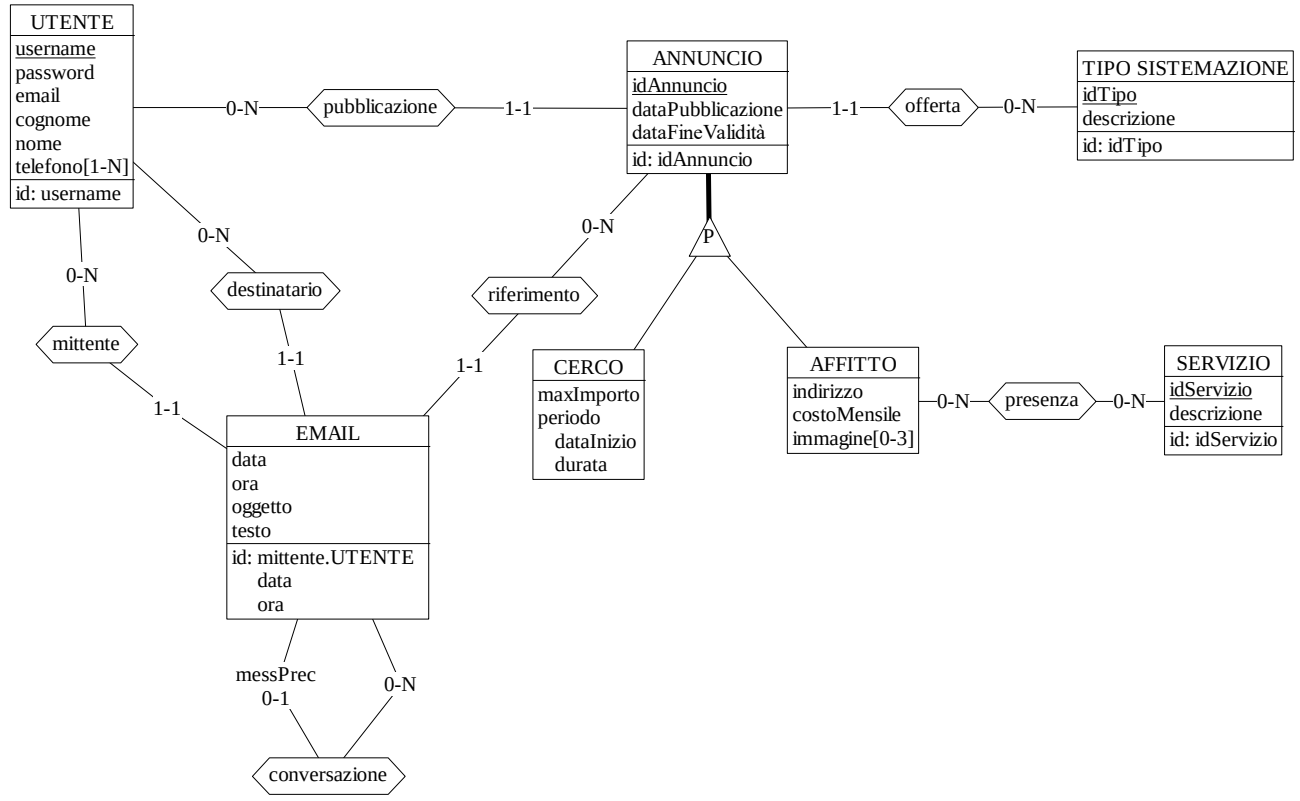
Esercizio 1

Si considerino le seguenti specifiche relative alla realizzazione della base dati di un sito web per la gestione annunci.

Si vuole realizzare la base dati per un sito web che gestisce annunci relativi a stanze e appartamenti in affitto. Per l'accesso al sito è necessaria una registrazione; per ogni utente registrato, caratterizzato da uno username univoco, si memorizzano una password, un indirizzo e-mail, il nome, il cognome e uno o più recapiti telefonici. Gli utenti registrati possono consultare e inserire annunci nel sito web. Ogni utente può inserire più annunci. Gli annunci si dividono in due categorie: "cerco casa" e "affitto casa". Per tutti gli annunci, identificati da un codice univoco, si memorizzano la data di pubblicazione, la data di fine validità e il tipo di sistemazione (camera singola, camera doppia o appartamento). Per gli annunci di tipo "cerco casa" è necessario inoltre indicare l'importo mensile massimo che l'utente è disposto a pagare e il periodo per cui la sistemazione è richiesta (data di inizio e durata). Per gli annunci del tipo "affitto casa" bisogna specificare invece l'indirizzo, e il costo mensile; è possibile inoltre inserire fino a 3 immagini della sistemazione e indicare la presenza di alcuni servizi accessori (connessione internet, aria condizionata, ascensore, lavastoviglie, lavatrice).
Gli utenti possono interagire scambiandosi delle e-mail in riferimento agli annunci pubblicati sul sito e il sistema deve tenere traccia di tutte le e-mail inviate dagli utenti. Per ciascuna e-mail si devono memorizzare il mittente, il destinatario, il riferimento all'annuncio, la data e ora, l'oggetto e il testo. Alcune e-mail sono inviate in risposta ad un'altra e-mail; in tal caso il sistema deve tenerne traccia.

*Si definisca il relativo **schema E/R** (nella metodologia proposta a lezione) e si indichino esplicitamente eventuali **attributi derivati** presenti nelle specifiche, commentandoli adeguatamente. Si evidenzino inoltre eventuali **vincoli inespressi**.*

Svolgimento



Compito Basi di Dati. Tempo concesso: 2 ore **23 Giugno 2022 Matricola: _____**

Esercizio 2

1. Nell'ambito della normalizzazione di schemi relazionali si definiscano i concetti di "dipendenza funzionale", "decomposizione senza perdita" e "decomposizione che preserva le dipendenze".

2. È dato il seguente schema relazionale:

   LISTINIPRODOTTI (codProdotto, descrizioneProdotto, codMagazzino, indirizzoMagazzino, giacenzaProdotto, codCategoria, nomeCategoria, codCliente, nomeCliente, nazionalitàCliente, quantità, prezzo, codFornitore, nomeFornitore, indirizzoFornitore).

   Sapendo che:

   - ciascun prodotto appartiene a una sola categoria ed è identificato da un codice univoco all'interno della categoria di appartenenza;
   - ogni prodotto ha una descrizione e una giacenza in magazzino;
   - ogni prodotto è stoccato in un unico magazzino;
   - ogni categoria di prodotti è fornita da un solo fornitore;
   - i prezzi dei prodotti variano in base al cliente e alla quantità acquistata;

   Evidenziare tutte le dipendenze funzionali non banali presenti nello schema e decomporre lo schema in terza forma normale verificando che la decomposizione ottenuta sia senza perdita e preservi le dipendenze.

Svolgimento

codCategoria, codProdotto → descrizioneProdotto, codMagazzino, giacenzaMagazzino
codMagazzino → indirizzoMagazzino
codCategoria → nomeCategoria, codFornitore
codFornitore → nomeFornitore, indirizzoFornitore
codCliente → nomeCliente, nazionalitàCliente
codCategoria, codProdotto, codCliente, quantità → prezzo

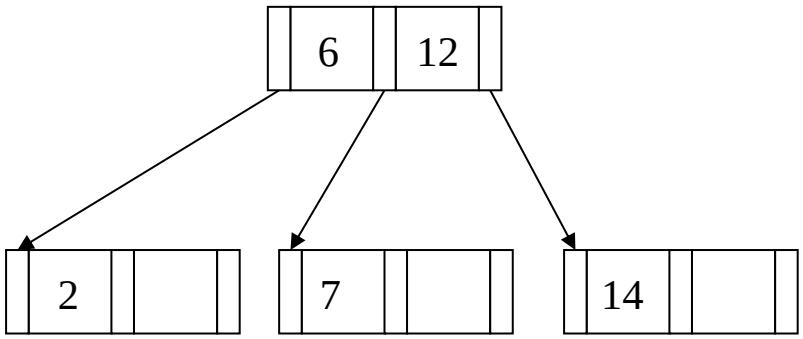
MAGAZZINI (codMagazzino, indirizzoMagazzino)
CLIENTI (codCliente, nomeCliente, nazionalitàCliente)
FORNITORI (codFornitore, nomeFornitore, indirizzoFornitore)
CATEGORIE (codCategoria, nomeCategoria, codFornitore: FORNITORI)
PRODOTTI (codCategoria: CATEGORIE, codProdotto, descrizioneProdotto, codMagazzino: MAGAZZINI, giacenzaMagazzino)
LISTINIPRODOTTI ((codCategoria, codProdotto): PRODOTTI, codCliente: CLIENTI, quantità, prezzo)

Compito Basi di Dati. Tempo concesso: 2 ore **23 Giugno 2022 Matricola: _____**

Esercizio 3

1. Si descrivano brevemente le principali differenze tra le strutture B-tree e B+-tree.

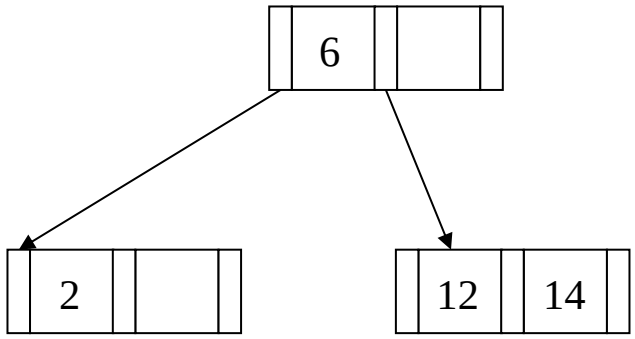
2. Si consideri il B-tree sotto riportato (**ordine g=1**), nell'ipotesi di assenza di gestione di overflow:



Riportare la struttura dopo l'eliminazione della chiave 7, motivando la risposta.

Svolgimento

In questo caso viene attivata la procedura di catenation.



Compito Basi di Dati. Tempo concesso: 2 ore **23 Giugno 2022 Matricola: _____**

Esercizio 4

Dato il seguente schema relazionale:

ATTORI (codAttore, cognome, nome, dataNascita)
SPETTACOLI (codSpettacolo, nomeSpettacolo, descrizione, durata, genere, regista, anno)
PARTECIPAZIONI (codAttore: ATTORI, codSpettacolo: SPETTACOLI, ruolo)
RAPPRESENTAZIONI (codSpettacolo: SPETTACOLI, dataRappresentazione, numeroSpettatori)

1. Scrivere un'espressione di algebra relazionale che visualizzi gli spettacoli (codSpettacolo, nomeSpettacolo, descrizione) che in nessuna delle rappresentazioni effettuate hanno avuto più di 50 spettatori.
2. Scrivere una query SQL che visualizzi, per ogni attore, il numero di spettacoli ai quali ha partecipato (codAttore, cognome, nome, numeroSpettacoli).
3. Scrivere una query SQL che visualizzi le coppie di attori che hanno recitato insieme in uno stesso spettacolo (codAttore1, cognome1, nome1, codAttore2, cognome2, nome2).
4. Scrivere una query SQL che visualizzi il numero di spettacoli (numeroSpettacoli) ai quali hanno partecipato più di 10 attori.

Svolgimento

1. $\pi_{codSpettacolo,nomeSpettacolo,descrizione}(SPETTACOLI) \rhd\lhd \left(\pi_{codSpettacolo}(SPETTACOLI) - \pi_{codSpettacolo}\left(\sigma_{numeroSpettatori>50}(SPETTACOLI)\right)\right)$

2. SELECT A.codAttore, cognome, nome, COUNT(*) AS numeroSpettacoli
   FROM ATTORI A, PARTECIPAZIONI P
   WHERE A.codAttore = P.codAttore
   GROUP BY A.codAttore, cognome, nome

3. SELECT A.codAttore, A.cognome, A.nome, A1.codAttore, A1.cognome, A1.nome
   FROM ATTORI A, ATTORI A1, PARTECIPAZIONI P, PARTECIPAZIONI P1
   WHERE A.codAttore = P.codAttore
   AND A1.codAttore = P1.codAttore
   AND P.codSpettacolo = P1.codSpettacolo
   AND A.codAttore < A1.codAttore

4. SELECT COUNT(*) AS numeroSpettacoli
   FROM SPETTACOLI
   WHERE codSpettacolo IN (SELECT P.codSpettacolo
   FROM PARTECIPAZIONI P
   GROUP BY P.codSpettacolo
   HAVING COUNT(*) >10)