

Basi di Dati, 3/11/2003, primo compito, compito A

- a) Una compagnia che gestisce un piccolo motore di ricerca sul web vuole usare una base di dati per tenere traccia della struttura delle pagine indicizzate, e delle sessioni che gli utenti hanno con il motore di ricerca. Per quanto riguarda la struttura delle pagine indicizzate, interessa conoscere, per ogni pagina, l'URI (URL), il titolo, l'insieme delle pagine puntate dai riferimenti (anchor – link) che appaiono nella pagina, l'insieme dei termini che vi appaiono e, per ogni termine, al numero di volte in cui il termine appare nella pagina. Le sessioni sono così strutturate: l'utente presenta un termine di ricerca, il sistema segnala tutte le pagine correlate, e l'utente legge alcune di queste pagine; specifichiamo ora a quali informazioni la compagnia è interessata, per quanto riguarda le sessioni. Per ogni sessione, l'organizzazione è interessata al giorno in cui si è svolta ed al termine di ricerca presentato. La sessione può essere eseguita da un utente anonimo, sul quale il sistema non ha informazioni, oppure da un utente registrato, del quale il sistema conosce username, password, indirizzo e-mail, e l'elenco di tutti gli indirizzi IP dai quali l'utente si è collegato nel passato. Per le sessioni con utenti registrati interessa memorizzare anche l'ora di inizio e di fine, l'utente, e la liste delle pagine effettivamente lette dall'utente.
- 1) Si disegni lo schema concettuale che descrive la situazione, indicando in modo testuale i vincoli non catturati graficamente.
 - 2) Si traduca lo schema concettuale in uno schema relazionale grafico e testuale, usando la notazione $R(\underline{A}, B^*, \dots)$
- b) Si consideri la base di dati:

Studenti(Matricola: integer, SNome: string, CorsoLaurea: string, tipoLaurea: string, età: integer)

Corsi(SiglaC: string, oraRicevimento: time, Aula: string, DocenteId*: integer)

Iscrizioni(Matricola*: integer, SiglaC*: string)

Docenti(DocenteId: integer, DNome: string, Dipartimento: string)

1. Si dia lo schema relazionale in formato grafico.

Scrivere le seguenti interrogazioni in SQL per produrre risultati privi di duplicati:

2. Trovare i nomi degli studenti di un corso di laurea (tipoLaurea = 'L') che seguono un corso del docente Tizio.
3. Trovare la sigla dei corsi frequentati da più di 5 studenti e tenuti da docenti del Dipartimento di Informatica.
4. Trovare i nomi degli studenti iscritti a qualche corso.
5. Per ogni tipo di laurea, restituire il tipoLaurea e l'età media degli studenti.
6. Trovare il nome e l'età dello studente più anziano (o degli studenti più anziani) tra quelli iscritti ad una qualunque laurea specialistica (tipoLaurea = 'LS').
7. (Opzionale) Per ogni coppia di corsi che hanno almeno dieci studenti in comune, trovare la sigla e l'aula dei due corsi ed il numero di studenti in comune.
8. Disegnare l'albero di sintassi astratta di un'espressione algebrica ("albero logico") per l'interrogazione 4.

Basi di Dati, 3/11/2003, primo compitino, compito B

- a) Una compagnia che gestisce un piccolo motore di ricerca sul web vuole usare una base di dati per tenere traccia della struttura delle pagine indicizzate, e delle sessioni che gli utenti hanno con il motore di ricerca. Per quanto riguarda la struttura delle pagine indicizzate, interessa conoscere, per ogni pagina, l'URI (URL), il titolo, l'insieme dei termini che vi appaiono, e l'insieme delle pagine puntate dai riferimenti (anchor – link) che appaiono nella pagina. Alcune pagine sono sponsorizzate; per tali pagine interessa conoscere anche lo Sponsor e la tariffa che lo sponsor riconosce per ogni volta che la pagina è letta durante una qualche sessione. Di ogni sponsor interessa il nome ed un insieme di recapiti. Le sessioni sono così strutturate: l'utente presenta un termine di ricerca, il sistema segnala tutte le pagine correlate, e l'utente legge alcune di queste pagine; specifichiamo ora a quali informazioni la compagnia è interessata, per quanto riguarda le sessioni. Per ogni sessione, l'organizzazione è interessata al giorno in cui si è svolta, al termine di ricerca presentato, all'insieme delle pagine sponsorizzate lette, e, per ogni pagina sponsorizzata, al numero di volte in cui tale pagina è stata letta durante la sessione.
- 1) Si disegni lo schema concettuale che descrive la situazione, indicando in modo testuale i vincoli non catturati graficamente.
 - 2) Si traduca lo schema concettuale in uno schema relazionale grafico e testuale, usando la notazione $R(\underline{A}, B^*, \dots)$
- b) Si consideri la base di dati:

Studenti(Matricola: integer, SNome: string, CorsoLaurea: string, tipoLaurea: string, età: integer)

Corsi(SiglaC: string, oraRicevimento: time, Aula: string, DocenteId*: integer)

Iscrizioni(Matricola*: integer, SiglaC*: string)

Docenti(DocenteId: integer, DNome: string, Dipartimento: string)

1. Si dia lo schema relazionale in formato grafico.

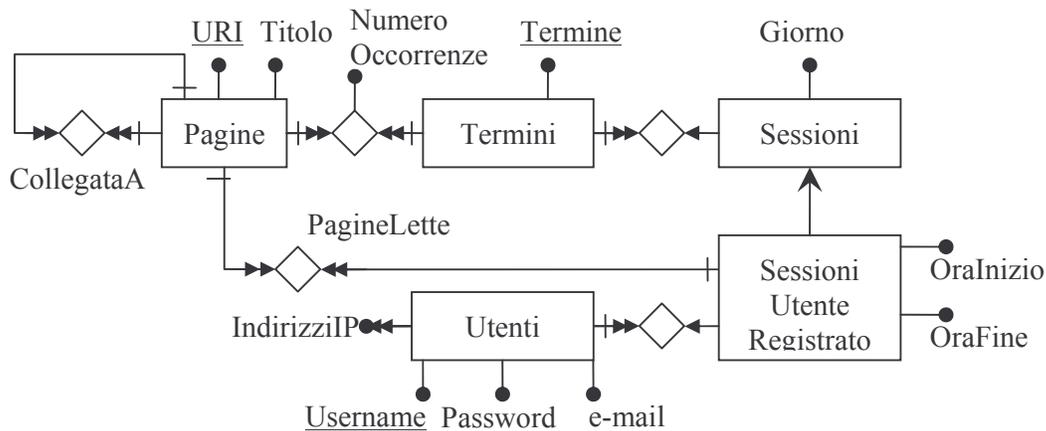
Scrivere le seguenti interrogazioni in SQL per produrre risultati privi di duplicati:

2. Trovare i nomi dei corsi a cui è iscritto qualche studente.
3. Per ogni corso di laurea, trovare il nome e l'età media degli studenti.
4. Trovare i nomi dei corsi del docente Tizio frequentati da uno studente di un corso di Laurea (tipoLaurea = 'L').
5. Trovare il nome e l'età dello studente più anziano (o degli studenti più anziani) tra quelli iscritti ad una qualunque laurea specialistica (tipoLaurea = 'LS').
6. Trovare la sigla dei corsi frequentati da più di 10 studenti e tenuti da docenti del Dipartimento di Informatica.
7. (Opzionale) Per ogni coppia di studenti che hanno almeno tre corsi in comune tra quelli a cui sono iscritti, trovare la matricola ed il nome dei due studenti ed il numero di corsi in comune.
8. Disegnare l'albero di sintassi astratta di un'espressione algebrica ("albero logico") per l'interrogazione 2.

Basi di Dati, 3/11/2003, primo compitino, compito A: Soluzioni (V 1.3)

a) Una compagnia che gestisce un piccolo motore di ricerca sul web vuole usare una base di dati per tenere traccia della struttura delle pagine indicizzate, e delle sessioni che gli utenti hanno con il motore di ricerca. Per quanto riguarda la struttura delle pagine indicizzate, interessa conoscere, per ogni pagina, l'URI (URL), il titolo, l'insieme delle pagine puntate dai riferimenti (anchor – link) che appaiono nella pagina, l'insieme dei termini che vi appaiono e, per ogni termine, al numero di volte in cui il termine appare nella pagina. Le sessioni sono così strutturate: l'utente presenta un termine di ricerca, il sistema segnala tutte le pagine correlate, e l'utente legge alcune di queste pagine; specifichiamo ora a quali informazioni la compagnia è interessata, per quanto riguarda le sessioni. Per ogni sessione, l'organizzazione è interessata al giorno in cui si è svolta ed al termine di ricerca presentato. La sessione può essere eseguita da un utente anonimo, sul quale il sistema non ha informazioni, oppure da un utente registrato, del quale il sistema conosce username, password, indirizzo e-mail, e l'elenco di tutti gli indirizzi IP dai quali l'utente si è collegato nel passato. Per le sessioni con utenti registrati interessa memorizzare anche l'ora di inizio e di fine, l'utente, e la liste delle pagine effettivamente lette dall'utente.

1) Si disegni lo schema concettuale che descrive la situazione, indicando in modo testuale i vincoli non catturati graficamente.

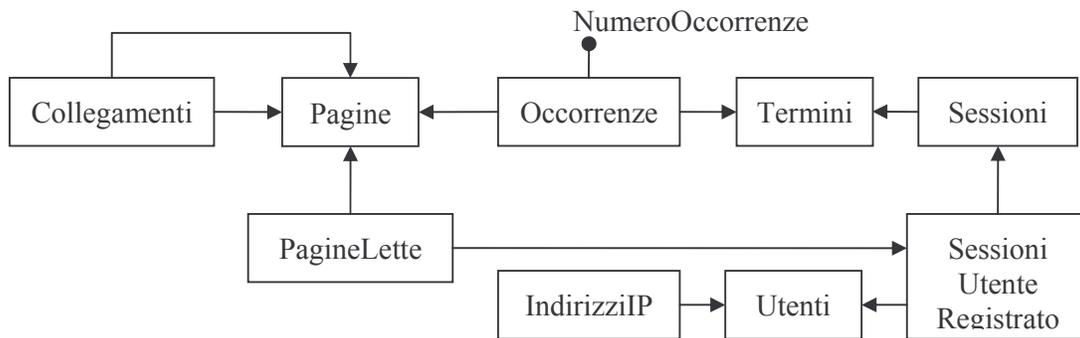


Vincoli: Ogni pagina letta durante una sessione è una delle pagine in cui appare il termine cercato nella sessione

Osservazione: Non c'è motivo di inserire classi per gli utenti non registrati

2) Si traduca lo schema concettuale in uno schema relazionale grafico e testuale, usando la notazione $R(A, B^*, \dots)$

Notazione grafica



Notazione testuale:

Collegamenti(Da*, A*)

Pagine(URI, Titolo)

Occorrenze(URI*, Termine*, NumeroOccorrenze)

Termini(Termine)

Sessioni(SessionId, Giorno, Termine*)

PagineLette(SessionId*, URI*)

SessioniUtenteRegistrato(SessionId*, OraInizio, OraFine, Username*)

Utenti(Username, Password, E-mail)

IndirizziIP(Username*, IndirizzoIP)

b) Si consideri la base di dati:

Studenti(Matricola: integer, SNome: string, CorsoLaurea: string, tipoLaurea: string, eta: integer)

Corsi(SiglaC: string, oraRicevimento: time, Aula: string, DocenteId*: integer)

Iscrizioni(Matricola*: integer, SiglaC*: string)

Docenti(DocenteId: integer, DNome: string, Dipartimento: string)

1. Si dia lo schema relazionale in formato grafico.

Scrivere le seguenti interrogazioni in SQL per produrre risultati privi di duplicati:

2. Trovare i nomi degli studenti di un corso di laurea (tipoLaurea = 'L') che seguono un corso del docente Tizio.

```
select distinct s.SNome
from Studenti s, Iscrizioni i, Corsi c, Docenti d
where s.Matricola = i.Matricola and i.SiglaC = c.SiglaS and c.DocenteId = d.DocenteId
and s.tipoLaurea = 'L' and d.DNome = 'Tizio';
```

3. Trovare la sigla dei corsi frequentati da più di 5 studenti e tenuti da docenti del Dipartimento di Informatica.

```
select c.SiglaC
from Iscrizioni i, Corsi c, Docenti d
```

```

where i.SiglaC = c.SiglaS and c.DocenteId = d.DocenteId
      and d.Dipartimento = 'Informatica'
group by c.SiglaC
having count(*) > 5;

```

4. *Trovare i nomi degli studenti iscritti a qualche corso.*

```

select distinct s.SNome
from   Studenti s, Iscrizioni i
where  s.Matricola = i.Matricola;

```

5. *Per ogni tipo di laurea, restituire il tipoLaurea e l'età media degli studenti.*

```

select  s.tipoLaurea, avg(s.eta)
from    Studenti s
group by s.tipoLaurea;

```

6. *Trovare il nome e l'età dello studente più anziano (o degli studenti più anziani) tra quelli iscritti ad una qualunque laurea specialistica (tipoLaurea = 'LS').*

```

select distinct s.SNome, s.eta
from   Studenti s
where  s.tipoLaurea = 'LS' and s.eta = (
      select max(s2.eta) from Studenti s2
      where s2.tipoLaurea = 'LS');

oppure

select distinct s.SNome, s.eta
from   Studenti s
where  s.tipoLaurea = 'LS' and not exists (
      select * from Studenti s2
      where s2.tipoLaurea = 'LS' and s2.eta > s.eta);

```

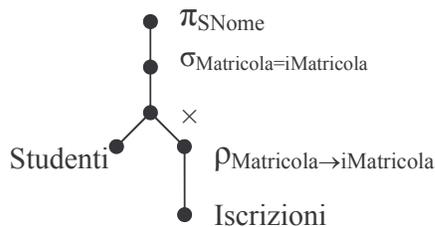
7. *(Opzionale) Per ogni coppia di corsi che hanno almeno dieci studenti in comune, trovare la sigla e l'aula dei due corsi ed il numero di studenti in comune.*

```

select  c1.SiglaC, c1.Aula, c2.SiglaC, c2.Aula, count(*)
from    Corsi c1, Iscrizioni i1, Iscrizioni i2, Corsi c2
where   c1.SiglaC = i1.SiglaC and i1.Matricola = i2.Matricola
      and i2.siglaC = c2.SiglaC and c1.SiglaC <> c2.SiglaC
group by c1.SiglaC, c1.Aula, c2.SiglaC, c2.Aula
having count(*) >= 10;

```

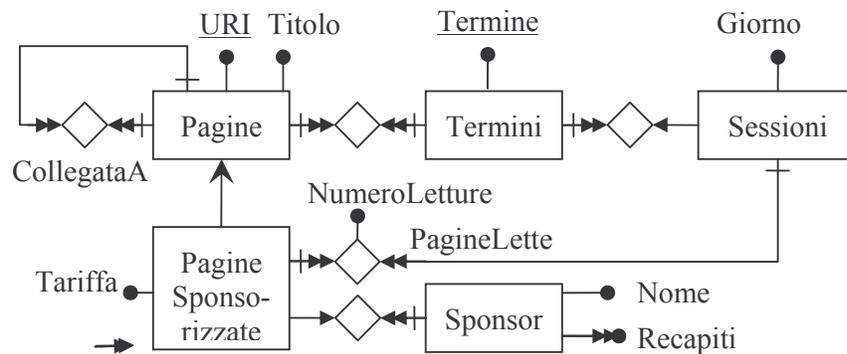
8. *Disegnare l'albero di sintassi astratta di un'espressione algebrica ("albero logico") per l'interrogazione 4.*



Basi di Dati, 3/11/2003, primo compito, compito B: Soluzioni (V 1.2)

a) Una compagnia che gestisce un piccolo motore di ricerca sul web vuole usare una base di dati per tenere traccia della struttura delle pagine indicizzate, e delle sessioni che gli utenti hanno con il motore di ricerca. Per quanto riguarda la struttura delle pagine indicizzate, interessa conoscere, per ogni pagina, l'URI (URL), il titolo, l'insieme dei termini che vi appaiono, e l'insieme delle pagine puntate dai riferimenti (anchor – link) che appaiono nella pagina. Alcune pagine sono sponsorizzate; per tali pagine interessa conoscere anche lo Sponsor e la tariffa che lo sponsor riconosce per ogni volta che la pagina è letta durante una qualche sessione. Di ogni sponsor interessa il nome ed un insieme di recapiti. Le sessioni sono così strutturate: l'utente presenta un termine di ricerca, il sistema segnala tutte le pagine correlate, e l'utente legge alcune di queste pagine; specifichiamo ora a quali informazioni la compagnia è interessata, per quanto riguarda le sessioni. Per ogni sessione, l'organizzazione è interessata al giorno in cui si è svolta, al termine di ricerca presentato, all'insieme delle pagine sponsorizzate lette, e, per ogni pagina sponsorizzata, al numero di volte in cui tale pagina è stata letta durante la sessione.

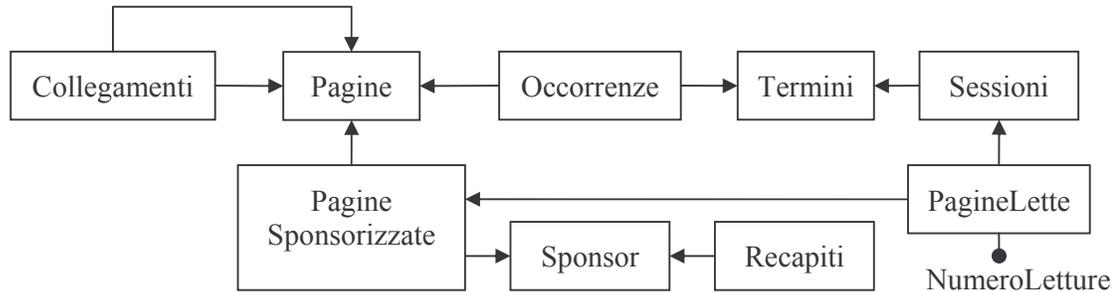
1) Si disegni lo schema concettuale che descrive la situazione, indicando in modo testuale i vincoli non catturati graficamente.



Vincoli: Ogni pagina letta durante una sessione è una delle pagine in cui appare il termine cercato nella sessione

2) Si traduca lo schema concettuale in uno schema relazionale grafico e testuale, usando la notazione $R(\underline{A}, B^*, \dots)$

Notazione grafica:



Notazione testuale:

Collegamenti(Da*, A*)
 Pagine(URI, Titolo)
 PagineSponsorizzate(URI*, Tariffa, SponsorId*)
 Sponsor(SponsorId, Nome)
 Recapiti(SponsorId*, Recapito)
 Occorrenze(URI*, Termine*)
 Termini(Termine)
 Sessioni(SessionId, Giorno, Termine*)
 PagineLette(SessionId*, URI*, NumeroLette)

b) Si consideri la base di dati:

Studenti(Matricola: integer, SNome: string, CorsoLaurea: string, tipoLaurea: string, eta: integer)

Corsi(SiglaC: string, oraRicevimento: time, Aula: string, DocenteId*: integer)

Iscrizioni(Matricola*: integer, SiglaC*: string)

Docenti(DocenteId: integer, DNome: string, Dipartimento: string)

1. Si dia lo schema relazionale in formato grafico.

Scrivere le seguenti interrogazioni in SQL per produrre risultati privi di duplicati:

2. Trovare i nomi dei corsi a cui è iscritto qualche studente.

```

select distinct i.SiglaC
from Iscrizioni i
  
```

3. Per ogni corso di laurea, trovare il nome e l'età media degli studenti.

```

select s.corsoLaurea, avg(s.eta)
from Studenti s
group by s.CorsoLaurea
  
```

4. *Trovare i nomi dei corsi del docente Tizio frequentati da uno studente di un corso di Laurea (tipoLaurea = 'L').*

```
select distinct c.SiglaC
from  Studenti s, Iscrizioni i, Corsi c, Docenti d
where s.Matricola = i.Matricola and i.SiglaC = c.SiglaC and c.DocenteId = d.DocenteId
      and s.tipoLaurea = 'L' and d.DNome = 'Tizio';
```

5. *Trovare il nome e l'età dello studente più anziano (o degli studenti più anziani) tra quelli iscritti ad una qualunque laurea specialistica (tipoLaurea = 'LS').*

```
select distinct s.SNome, s.eta
from  Studenti s
where s.tipoLaurea = 'LS' and s.eta = ( select max(s2.età) from Studenti s2
                                       where s2.tipoLaurea = 'LS');

oppure

select distinct s.SNome, s.eta
from  Studenti s
where s.tipoLaurea = 'LS' and not exists ( select * from Studenti s2
                                           where s2.tipoLaurea = 'LS' and s2.eta > s.eta)
```

6. *Trovare la sigla dei corsi frequentati da più di 10 studenti e tenuti da docenti del Dipartimento di Informatica.*

```
select c.SiglaC
from  Iscrizioni i, Corsi c, Docenti d
where i.SiglaC = c.SiglaC and c.DocenteId = d.DocenteId
      and d.Dipartimento = 'Informatica'
group by c.SiglaC
having count(*) > 10;
```

7. *(Opzionale) Per ogni coppia di studenti che hanno almeno tre corsi in comune tra quelli a cui sono iscritti, trovare la matricola ed il nome dei due studenti ed il numero di corsi in comune.*

```
select s1.Matricola, s1.SNome, s2.Matricola, s2.SNome, count(*)
from  Studenti s1, Iscrizioni i1, Iscrizioni i2, Studenti s2
where s1.Matricola = i1.Matricola and i1.SiglaC = i2.SiglaC
      and i2.Matricola = s2.Matricola and s1.Matricola <> s2.Matricola
group by s1.Matricola, s1.SNome, s2.Matricola, s2.SNome
having count(*) >= 3;
```

8. *Disegnare l'albero di sintassi astratta di un'espressione algebrica ("albero logico") per l'interrogazione 2.*

