

Basi di Dati
Allievi Informatici e Telecomunicazionisti
Proff. Stefano Ceri e Maristella Matera
PRIMA PROVA IN ITINERE – 17 NOVEMBRE 2003

UTENTE (IdUsr, Nome, Password, Categoria)
PAGINA (URL, Titolo, DimInByte, Tipo, Proprietario)
ACCESSO (IdUsr, Data, Ora, UrlPag, PaginaProvenienza)
COLLEGAMENTO (PagFrom, PagTo)

Lo schema si riferisce agli accessi alle pagine di un sito Web da parte di utenti registrati. Memorizza inoltre le informazioni relative al collegamento tra le pagine (attraverso i link) e per ogni accesso registra la pagina di provenienza. Ogni pagina ha un utente – rappresentato tramite il suo identificatore - come “proprietario”. Per la pagina iniziale di ogni navigazione, la pagina di provenienza è posta a **Null**.

A. DDL (2 punti)

Scrivere i comandi SQL per creare le tabelle PAGINA e COLLEGAMENTO, effettuando opportune e ragionevoli ipotesi sui domini, sui vincoli e sulle reazioni ai cambiamenti.

B. LINGUAGGI FORMALI (7 punti)

1. Esprimere in Algebra Relazionale ottimizzata, Calcolo Relazionale e Datalog la seguente interrogazione **(3 punti)**:
Trovare le pagine “flop”, cioè quelle che sono state visitate soltanto dai loro proprietari.
2. Esprimere in algebra relazionale ottimizzata la seguente interrogazione **(1.5 punti)**:
Trovare le pagine per cui l'ultimo utente che le ha consultate è il loro proprietario.
[Si presti attenzione al fatto che l'istante di accesso è contraddistinto da data e ora].
3. Usando la ricorsione, esprimere la seguente interrogazione **(1.5 punti)**:
Sapendo che l'utente “Alex” ha visitato la pagina “p001” in data “03-09-03” alle ore “13:20:05”, estrarre tutte le pagine cui l'utente ha fatto accesso successivamente.

Facoltativo (1 punto): Discutere come realizzare un insieme di regole che traccino il percorso seguito da Alex dopo l'accesso alla pagina “p001” in data “03-09-03” alle ore “13:20:05”, cioè che estraggano il numero progressivo di accesso, l'ora e la pagina acceduta, limitandosi ad osservare gli accessi del giorno corrente (cioè in data “03-09-03”).

C. Interrogazioni in SQL (8 PUNTI)

1. Estrarre la dimensione totale in KByte delle pagine scaricate dagli “studenti” nel luglio 2003 **(1 punto)**
2. Trovare i titoli delle pagine che non sono state visitate dai loro proprietari nell'ottobre 2003 **(1.5 punti)**
3. Determinare la categoria di utenti (diversa da “studenti”) che ha effettuato il maggior numero di accessi a pagine di tipo “didattica” **(2.5 punti)**.
4. Trovare gli utenti che hanno visitato almeno 10 pagine diverse e non hanno mai iniziato le loro navigazioni prima delle 13:00 [l'inizio di navigazione viene caratterizzato da un accesso privo di pagina di provenienza] **(3 punti)**.

BASI DI DATI - PROFF. CERI, MATERA
RECUPERO I PROVA - 13 FEBBRAIO 2004

PROPRIETARIOAUTO (IdProp, Targa, DataAcq, DataVendita, Nome, ComuneResid, ProvResid)
AUTOIMM (Targa, ComImm, ProvImm, DataImm, Marca, Modello, Colore, PotenzaKW)
DEMOLIZIONE (Targa, Data, Luogo, NomeDemolitore)
PRA(Comune, Provincia, Regione)

Lo schema si riferisce alle immatricolazioni, ai passaggi di proprietà e alle demolizioni di auto in Italia. Si ipotizzi che ogni auto possa essere immatricolata una sola volta, che più persone possano essere comproprietarie di una auto (nel caso, le date di acquisto devono coincidere) e che l'attributo DataVendita valga **Null** fino all'avvenuta trascrizione degli atti di vendita.

A. DDL (2 punti)

Scrivere i comandi SQL per

- 1) Creare le tabelle AUTOIMM e PRA, effettuando opportune ragionevoli ipotesi su domini, vincoli e reazioni ai cambiamenti.
- 2) Aggiungere alla tabella PRA l'attributo TipoSede, che assuma il valore "Normale" in assenza di altre indicazioni.

B. LINGUAGGI FORMALI (7 punti)

1. Esprimere in Algebra Relazionale ottimizzata, Calcolo Relazionale e Datalog la seguente interrogazione **(4 punti)**:
Trovare i proprietari che hanno acquistato sempre e solo auto nuove
2. Esprimere in algebra relazionale ottimizzata la seguente interrogazione **(3 punti)**:
Trovare i proprietari che hanno posseduto auto tutte dello stesso colore

C. Interrogazioni in SQL (8 PUNTI)

5. Calcolare, per ogni regione, la potenza media in KW delle auto immatricolate nel 2003 **(1,5 punti)**
6. Trovare le targhe delle auto demolite senza mai subire alcun passaggio di proprietà **(2 punti)**
7. Determinare il modello di auto più "gradito" agli automobilisti della provincia di Savona (si conti come una "preferenza" ogni acquisto, sia del nuovo sia dell'usato) **(2 punti)**
8. Costruire il registro delle auto in circolazione, nella forma (Targa, Modello, Proprietario, ComuneResPr, ProvinciaResPr), considerando solo l'ultimo proprietario di ogni auto non ancora demolita. **(2,5 punti)**

Basi di Dati
Prof. Stefano Ceri
PARTE 1
APPELLO DEL 28 GIUGNO 2004

A. SQL (P. 9)

Si considerino le tabelle:

DEPLIANT(ID, Titolo, Compositore, NumPag, Formato, Data, Durata)

CONTIENE(D-ID, P-ID, Foto, Descrizione, Sconto)

PRODOTTO(P-ID, Nome, Categoria, CostoListino)

Il depliant descrive un insieme di prodotti e ha un titolo, un compositore, un numero di pagine, un formato, la data di pubblicazione e una durata di validità. Lo sconto è espresso in percentuale. Esprimere in SQL le seguenti interrogazioni:

1. Estrarre il depliant valido (rispetto alla data corrente SYSDATE) che presenta lo sconto più alto relativo al prodotto MSOffice (P.3).
2. Estrarre i depliant che non contengono prodotti il cui costo scontato è inferiore a 1000 Euro e il cui numero di pagine supera il triplo dei prodotti presenti. (P.3)
3. Estrarre il nome dei prodotti per i quali esistano depliant in cui la differenza fra lo sconto più alto e lo sconto più basso praticato sia superiore al 10 per cento e la cui categoria sia "PC" oppure "Mouse". (P.3)

B. LINGUAGGI FORMALI DI INTERROGAZIONE (P. 5)

1. Esprimere in Calcolo e Datalog la prima interrogazione dell'esercizio precedente.

B. DDL (P. 2)

Scrivere i comandi SQL per

1. Creare la tabella CONTIENE, effettuando opportune ragionevoli ipotesi su domini e vincoli e reazioni ai cambiamenti.
2. Eliminare dalla tabella CONTIENE l'attributo Foto.

I PROVA

Si consideri il seguente schema, relativo a un sistema di prenotazione e noleggio di biciclette, in cui le indicazioni orarie possono assumersi piene (dalle 8 alle 19):

CLIENTE (Num, Nome, TipoDoc, NumeroDoc)

PRENOTA (Num-Cli, TipoBici, Giorno, OraInizio, OraFine)

USA (Num-Cli, Num-Bici, Giorno, OraInizio, OraFine, Costo)

BICICLETTA (Num-Bici, TipoBici, CostoOra, CostoGiorno)

A. DDL (2 punti)

Scrivere i comandi SQL per creare le tabelle PRENOTA e BICICLETTA, effettuando opportune ragionevoli ipotesi su domini, vincoli e reazioni ai cambiamenti.

B. LINGUAGGI FORMALI (7 punti)

1. Esprimere in Algebra Relazionale ottimizzata, Calcolo Relazionale e Datalog la seguente interrogazione **(4.5 punti)**:

Trovare il nome dei clienti che hanno prenotato bici di tipo "da corsa" senza mai usarle

2. Esprimere in algebra relazionale ottimizzata la seguente interrogazione **(2.5 punti)**:

Trovare il tipo dell'ultima bicicletta usata da Mario Rossi

C. Interrogazioni in SQL (7 PUNTI)

9. *Elencare il nome dei clienti che hanno usufruito di una bicicletta di tipo diverso da quello da loro prenotato[nota: far corrispondere prenotazioni e relativi usi] (2 punti)*
10. *Elencare il nome dei clienti che hanno prenotato una bicicletta senza usarla per piu' di una volta, e che non hanno mai usato una bicicletta (2 punti)*
11. *Per ogni cliente, indicare la bicicletta usata per il maggior numero di ore ed il corrispondente numero totale di ore (3 punti)*