

# Basi di dati, secondo modulo

6 febbraio 2003

Tempo a disposizione: un'ora e trenta minuti. Libri aperti.

## Domanda 1 (15%)

Discutere il ruolo dei linguaggi di interrogazione nei sistemi per basi di dati a oggetti e le relative difficoltà.

## Domanda 2 (25%)

In al massimo due pagine (circa 300-500 parole), sintetizzare i principali difetti del sito esaminato nel progetto dal punto di vista della qualità del contenuto, formulando anche proposte di miglioramento. La risposta a questa domanda deve essere tecnicamente corretta, ma deve essere comprensibile anche per un "committente" non esperto.

Poiché sarà valutata anche la chiarezza espositiva, si consiglia fortemente di procedere preliminarmente attraverso una brutta copia.

## Domanda 3 (60%)

Si vuole realizzare una applicazione Web per la gestione di semplici sondaggi di opinione on line. Un sondaggio corrisponde ad una domanda alla quale sono associate tre possibili risposte (ad esempio: Domanda: "Sei d'accordo con la proposta di riforma del campionato provinciale di tiro con l'arco?" risposte: "SI", "NO", "NON SO"). I dati relativi a *tutti* i sondaggi vengono memorizzati in una base di dati che ha il seguente schema:

```
SONDAGGI(codice, domanda, risposta1, risposta2, risposta3)
CONTEGGI(codice, numeroRisposte1, numeroRisposte2, numeroRisposte3)
```

La tabella SONDAGGI memorizza il testo di ciascun sondaggio. La tabella RISULTATI memorizza i risultati (anche parziali) di ciascun sondaggio.

L'applicazione deve essere realizzata come segue:

- la domanda e le tre risposte vengono presentate attraverso la pagina server `PresentaSondaggio.jsp`. Questa pagina server compie le seguenti operazioni:
  1. legge il parametro `codiceSondaggio` associato alla domanda HTTP
  2. legge dalla base di dati la domanda e le risposte del sondaggio con codice `codiceSondaggio`. L'interazione con la base di dati avviene attraverso una chiamata al metodo `static Sondaggio getSondaggio(int codiceSondaggio)` della classe `Interrogazioni`. Questo metodo restituisce, in un opportuno bean `Sondaggio`, la domanda e le tre risposte del sondaggio.
  3. costruisce una pagina che mostra la domanda e le tre risposte, e contiene un campo di input attraverso il quale l'utente può specificare la propria risposta (per semplicità si può supporre che questa consiste in un intero 1,2 o 3, che indica ad una delle tre possibili opzioni)
  4. mostra un prospetto che illustra i risultati correnti del sondaggio. I risultati del sondaggio sono letti dalla base di dati e memorizzati in un opportuno bean attraverso il metodo `static Risultati getRisultato(int codiceSondaggio)` della classe `Interrogazioni`.
- il voto espresso attraverso la pagina `PresentaSondaggio.jsp` viene processato dalla servlet `ProcessaVoto` come segue:
  1. si verifica che l'utente non abbia già votato a questo sondaggio. Questa verifica viene fatta attraverso un cookie: ad ogni utente che esprime il proprio voto, il sistema risponde con una pagina di ringraziamento (`Ringraziamento.jsp`) alla quale viene associato un opportuno cookie. Quando un utente prova a votare il sistema verifica o meno la presenza e il contenuto del cookie.
  2. se l'utente ha già votato, la servlet reindirizza il controllo alla pagina `NonSiVotaDueVolte.jsp`, che mostra un opportuno messaggio; altrimenti viene incrementato il contatore nella tabella `RISULTATI`, e la servlet reindirizza il controllo alla pagina `Ringraziamento.jsp`, che, oltre a presentare un opportuno messaggio di ringraziamento associa alla risposta un cookie. Per incrementare il contatore, si fa ricorso al metodo `static int incrementaRisposta(int codiceSondaggio, String risposta)` della classe `Interrogazioni`.

Scrivere il codice della servlet `ProcessaVoto` e della pagina `jsp NonSiVotaDueVolte.jsp`.