

# Prova di Laboratorio di Basi di Dati I

23 Luglio 2003

**Connessione a Windows.** Connettersi a Windows con Login:1995s140, Password:biologi2.

**Connessione a SQL Server.** Utilizzare il DBMS su Winchester. La connessione a SQL Server deve avvenire utilizzando la login e la password che vi verranno comunicate all'inizio della prova. Sia `cognomeNumN` la vostra login.

**Si consiglia di leggere tutto il testo prima di iniziare a svolgere la prova.**

**Esercizio.** Si consideri la base di dati `cognomeNumN`, (dove `cognomeNumN` è la vostra login), per gestire alcuni dati di un club di vela. Tale base di dati contiene le seguenti tabelle:

- Tabella **Velisti**(Codice,Cognome,Nome,Eta,Brevetto,Bonus), in cui gli attributi **Cognome**, **Nome** ed **Eta** hanno il significato ovvio, mentre l'attributo **Brevetto** indica se il velista possiede, o meno, il brevetto nautico e l'attributo **Bonus** indica la quota anticipata dal velista per i suoi futuri noleggi. Tale tabella contiene informazioni su tutti gli iscritti al club di vela.
- Tabella **Barche**(Targa,Nome,Colore,Categoria), in cui gli attributi **Targa**, **Nome**, **Colore** hanno il significato ovvio, mentre l'attributo **Categoria** indica se la barca può essere utilizzata per regate d'*altura* o di *deriva*.

Si richiede di eseguire le seguenti operazioni (si noti che i punti 2 e 3 sono tra loro indipendenti).

1. *Estensione progettazione base di dati* (Indicativamente punti 7).

Estendere lo schema della base di dati (quindi creare con Query Analyzer o con Enterprise Manager le opportune tabelle e i vincoli associati) nell'ipotesi che vengano considerati i seguenti nuovi requisiti.

Si vogliono gestire i noleggi effettuati dai velisti per le uscite in mare con le barche del club. Per ogni uscita in mare, si vuole tenere traccia del ruolo ricoperto dal velista nel governare la barca. Il ruolo può essere **prodiero**, **timoniere**, **tattico**, oppure semplice **mozzo** e quello più comune è **mozzo**. Ogni barca esce in mare una sola volta al giorno con a bordo un team composto da tutti i velisti che l'hanno noleggiata per quel giorno (ad esempio: il giorno *21 luglio 2003* i velisti *Mario Rosso*, *Sara Caruso* e *Antonio Velli* hanno noleggiato il *Nautilus* per un'uscita in mare). A seconda del ruolo ricoperto da un velista su una barca, il velista deve corrispondere un importo di noleggio diverso, ma mai inferiore a 20 euro.

Non si vogliono eliminare i velisti del club se questi hanno noleggiato almeno una barca, mentre se una barca viene dismessa tutti i noleggi che la riguardano devono essere rimossi. Tutte le modifiche che riguardano le barche e/o i velisti devono essere propagate ai noleggi.

Inserire, inoltre, almeno 5 tuple in ogni nuova tabella generata.

2. *Creazione stored procedure* (Indicativamente punti 10).

Definire una stored procedure `NoleggioBarca` che, preso in input la targa di una barca, verifica se ci sono stati noleggi per quella barca da parte di velisti con il brevetto nautico. In caso affermativo, produce in output un report di sintesi relativo ai noleggi di tale barca. Le informazioni devono essere ordinate in base alla data di noleggio e devono essere formattate come segue: “In data: ” data di noleggio, “il velista:” nome e cognome del velista “ha noleggiato la barca pagando la quota di:” l’importo di noleggio pagato. Inoltre, a tutti i velisti maggiorenni con brevetto che hanno noleggiato tale barca viene aumentato il bonus del 5%. Se invece la barca non è mai stata noleggiata da parte di velisti con il brevetto nautico, la barca viene rimossa.

3. *Applicazione JDBC* (Indicativamente punti 11).

Il template Java `templateJava.java` utilizzabile per sviluppare l’applicazione è contenuto in `\\Murphy\DB1\`.

Il file creato deve **obbligatoriamente** avere un’intestazione contenente i seguenti dati:

- Nome, Cognome
- Numero di Matricola
- Data di Nascita
- Login SQL Server (quella assegnata per la prova di laboratorio)
- Password SQL Server (quella assegnata per la prova di laboratorio)

Si richiede di:

- (a) Associare un identificatore ODBC alla base di dati chiamato `laboDSN`.
- (b) Sviluppare un programma JDBC `cognomeNumN1` (dove `cognomeNumN` è la vostra login) che preso in input l’identificativo di una barca verifica, per ogni sua uscita in mare (ovvero per ogni data in cui la barca sia stata noleggiata) che il numero di velisti con ruoli distinti sia maggiore di 2. In caso affermativo, stampa per ogni uscita con tali caratteristiche, la data in cui è avvenuta ed il numero di velisti con ruoli distinti che facevano parte del team della barca, in accordo col seguente formato di output: “In data ”, la relativa data, “la barca aveva a bordo un team di velisti che ricoprivano ” il numero di ruoli distinti, “ruoli distinti”. Altrimenti, stampa la scritta “In data ”, la relativa data, “la barca ”, la targa della barca, “non ha un equipaggio sufficiente”. I dati devono essere formattati opportunamente.

Per sveltire le modalità di consegna, l’applicazione JDBC sviluppata deve essere salvata come testo commentato (quindi compreso tra i caratteri `/*` e `*/`) di una stored procedure di nome `JDBC1`, nella vostra base di dati.

4. *Estensione applicazione JDBC* (Indicativamente punti 5).

Sviluppare un programma JDBC `cognomeNumN2` che estende l’applicazione `cognomeNumN1`. La nuova applicazione, prima di chiudere la connessione con la banca dati, rimuove dalla base di dati tutte le barche, della stessa categoria di quella presa in input, che non sono mai state noleggiate. Inoltre, diminuisce del 10% il valore del bonus a disposizione di ogni velista che non ha effettuato noleggi.

Per sveltire le modalità di consegna, l’applicazione JDBC sviluppata deve essere salvata come testo commentato (quindi compreso tra i caratteri `/*` e `*/`) di una stored procedure di nome `JDBC2`, nella vostra base di dati.