Laboratorio di Basi di dati 1
VII Esercitazione 6 maggio 2003

Si consideri la base di dati `GestioneAziendale`N (dove N è il numero del gruppo a cui appartenete) ed in particolare le tabelle create durante l'esercitazione V mediante l'esecuzione dello script `creadb_es5.sql`.

SOLO per chi non aveva eseguito tale script:

1. Selezionare la base di dati creata durante la prima esercitazione `GestioneAziendale`N, (dove N è il numero del gruppo a cui appartenete) che rappresenta i dati relativi all'Università di Genova.
2. Cancellare le tabelle create in precedenza, oppure rinominarle.
3. Scaricare lo script `creadb_es5.sql` dalla pagina Web del laboratorio di basi di dati `http://www.disi.unige.it/person/MesitiM/teach/BD1-02-03/lab.html`
4. Dal Query Analizer aprire lo script in locale ed eseguirlo. Siate sicuri di aver selezionato il database corretto, prima di eseguire lo script, altrimenti modificate di conseguenza la prima riga dello script e decommentatela.

Si richiede di eseguire le seguenti operazioni, che coinvolgono l'utilizzo del linguaggio T-SQL, incluse le estensioni imperative (si ricorda che è opportuno utilizzare la visualizzazione testo del Query Analizer):

1. *Uso di costrutti imperativi.* Si scriva e si esegua da Query Analyzer il codice T-SQL per eseguire la seguente operazione: determinare lo stipendio medio dei professori, titolari di almeno un corso attivato; se tale stipendio medio è inferiore a 15000 euro, aumentare del 10% gli stipendi di tutti i professori titolari di almeno un corso attivato, altrimenti aumentare tale stipendio del 5%. Restituire infine l'elenco di tali professori, indicando per ciascuno il cognome, il nome e lo stipendio, ordinati in modo decrescente rispetto allo stipendio, senza utilizzare il comando PRINT.
2. *Uso di cursori.* Si scriva e si esegua da Query Analyzer il codice T-SQL per eseguire l'operazione precedente ma con una diversa visualizzazione. In particolare, le informazioni di ciascun professore devono essere visualizzate secondo il seguente formato:

   `Cognome: <cognome>, Nome: <nome>, Stipendio: <stipendio>`
3. *Esecuzione di script T-SQL da linea di comando.* Scrivere in un file di testo `script`N`-1.txt`, dove N è il numero assegnato al vostro gruppo, i comandi T-SQL relativi alla seguente operazione: determinare lo stipendio massimo dei professori che sono titolari di al più un corso (indipendentemente dal fatto che il corso sia attivato o meno); se tale stipendio massimo è superiore a 15000 euro, ridurre del 15% tale stipendio, altrimenti ridurlo del 5%. Restituire infine l'elenco di tutti professori, indicando per ciascuno il cognome, il nome, lo stipendio e lo scarto dallo stipendio massimo calcolato prima della modifica, ordinati in modo decrescente rispetto allo stipendio. Le informazioni di ciascun professore devono essere visualizzate secondo il seguente formato:

   `Cognome: <cognome>, Nome: <nome>, Stipendio: <stipendio>, Scarto da stipendio max: <stipendio massimo calcolato> - <stipendio>`

   Nota: si vuole eseguire una sola volta l'interrogazione per determinare lo stipendio massimo.

   Eseguire lo script `script`N`-1.txt` da linea di comando utilizzando il comando `osql`.
4. *Creazione stored procedure senza parametri da Query Analyzer ed esecuzione.* Da Query Analyzer, creare una procedura `AggiornaStip1`, per eseguire i comandi inseriti nello script `script`N`-1.txt` generato al punto precedente.

   Verificare da Enterprise Manager che tale procedura è stata inserita tra gli oggetti della base di dati.

   Eseguire quindi tale procedura da Query Analyzer.

5. *Creazione stored procedure con parametri di input ed esecuzione.*

   (a) Da Query Analyzer, creare una procedura `AggiornaStip2`, con due parametri di input di tipo intero, `Perc1` e `Perc2`. Se tali valori sono compresi tra 0 e 100, la procedura visualizza il messaggio *Parametri corretti* ed esegue esattamente quanto descritto nello script `script`N`-1.txt`, sostituendo 15 con `Perc1` e 5 con `Perc2`. Se i parametri non sono corretti, la procedura scrive a video il messaggio *Parametri non corretti.*

   Eseguire quindi varie volte tale procedura da Query Analyzer, passando parametri diversi.

   (b) Da Query Analyzer, creare una tabella `ErroriCorsi(DenominazioneCorso)`, dove `DenominazioneCorso` rappresenta il nome di un corso. Creare quindi una procedura `CambiaDenCorso` che prende in input la denominazione di un corso `Den-old` e una nuova denominazione `Den-new`. Se non esiste alcun corso chiamato `Den-old`, la procedura deve inserire `Den-old` nella tabella `ErroriCorsi`. Altrimenti, deve sostituire `Den-old` con `Den-new`, in ogni tupla in cui `Den-old` appare.

   Eseguire tale procedura da Query Analyzer con parametri di input a vostra scelta.

6. *Creazione stored procedure con parametri di input e output ed esecuzione.*

   (a) Da Query Analyzer, creare una procedura `AggiornaStip3`, definita come `AggiornaStip2` ma con un ulteriore parametro di output. Tale parametro dovrà restituire il valore corrispondente allo stipendio massimo calcolato.

   Eseguire quindi tale procedura da Query Analyzer, passando parametri a vostra scelta e visualizzando il valore del parametro di output restituito dalla procedura, insieme ad un opportuno messaggio.

   (b) Da Query Analyzer, creare una procedura `ListaLaureati` che prende in input una data e visualizza, ordinati rispetto alla data di laurea, il nome, il cognome e la data di laurea degli studenti che si sono laureati in data successiva a quella fornita in input. La visualizzazione di ogni laureato deve seguire il seguente formato:

   `Nome :  <nome> Cognome:  <cognome> Data Laurea <datalaurea>`

   La procedura deve restituire in output il numero dei laureati che soddisfano la condizione precedente.

   Eseguire tale procedura da Query Analyzer con parametri di input a vostra scelta, visualizzando successivamente il numero dei laureati visualizzati, insieme ad un opportuno messaggio.

7. *Creazione stored procedure da Enterprise Manager ed esecuzione.* Da Enterprise Manager creare una procedura `InserisciEsame1` che prende in input un valore per ogni campo della tabella `Esame` ed inserisce la tupla corrispondente nella tabella. Se l'inserimento genera un errore, visualizza a video il codice dell'errore generato associato ad un opportuno messaggio.

   Eseguire quindi tale procedura da Query Analyzer, passando parametri a vostra scelta.

8. *Chiamate tra stored procedure.* Da Query Analyzer o da Enterprise Manager creare una procedura `InserisciEsame2` che prende in input la matricola di uno studente, il codice di un corso e un voto, quindi esegue la procedura `InserisciEsame1` passandogli, oltre ai parametri precedenti, la data corrente.

   Eseguire quindi tale procedura da Query Analyzer, passando parametri a vostra scelta.

9. *Esecuzione di stored procedure da script.* Scrivere in un file di testo `script`N`-2.txt`, dove N è il numero assegnato al vostro gruppo, i comandi T-SQL relativi a 10 chiamate alla procedura `InserisciEsame2`, con parametri a vostra scelta.

   Eseguire lo script `script`N`-2.txt` da linea di comando utilizzando il comando `osql`.