

# Laboratorio di Basi di dati 1

## IV esercitazione 1 aprile 2003

Sia data la base di dati *Esercitazione3*, che rappresenta i dati relativi all'Università di Genova, con schema:

- *Professori*(id, *Cognome*, *Nome*, *Stipendio*)
- *CorsiDiLaurea*(id, *Facolta*, *Denominazione*, *Attivazione*)
- *Corsi*(id, *CorsoDiLaurea*, *Denominazione*, *Professore*, *Attivato*)
- *Studenti*(Matricola, *Cognome*, *Nome*, *Residenza*, *DataNascita*, *LuogoNascita*, *CorsoDiLaurea*, *Iscrizione*, *Relatore*, *Laurea*)
- *Esami*(Studente, Corso, Data, *Voto*)
- *PianiDiStudio*(Studente, AnnoAccademico, Anno)

dove:

- *Professori* rappresenta i professori dell'Università di Genova; *id* è chiave primaria della relazione.
- *CorsiDiLaurea* rappresenta i corsi di laurea attivati presso l'Università di Genova; *id* è chiave primaria della relazione; *Attivazione* rappresenta l'anno accademico in cui il corso di laurea è stato attivato. Si noti, inoltre, che la coppia di attributi *Facolta* e *Denominazione* costituiscono una chiave secondaria della relazione (utilizzo di vincoli UNIQUE nella definizione della tabella).
- *Corsi* rappresenta i Corsi tenuti nei vari corsi di laurea attivati presso l'Università di Genova; *id* è chiave primaria della relazione, *Professore* e *CorsoDiLaurea* sono chiavi esterne per la relazione e rappresentano rispettivamente il professore titolare del corso e il corso di laurea in cui il corso è tenuto; *Attivato* è un valore booleano che specifica se il corso è attivo durante l'anno accademico '2001/2002'. Si noti, inoltre, che la coppia di attributi *CorsoDiLaurea* e *Denominazione* costituiscono una chiave secondaria della relazione (utilizzo di vincoli UNIQUE nella definizione della tabella).
- *Studenti* rappresenta gli studenti dell'Università di Genova; *Matricola* è chiave primaria della relazione; *Relatore* e *CorsoDiLaurea* sono chiavi esterne per la relazione e rappresentano rispettivamente il professore relatore della tesi di laurea dello studente e il corso di laurea a cui lo studente è iscritto; *DataNascita* e *LuogoNascita* rappresentano la data e il luogo di nascita dello Studente; *Laurea* rappresenta la data in cui lo studente ha conseguito la laurea.

- **Esami** rappresenta gli esami registrati presso l'Università di Genova. **Studente**, **Corso** e **Data** sono chiave primaria della relazione e rappresentano, rispettivamente, lo studente che ha dato l'esame, il corso al quale l'esame si riferisce e la data in cui l'esame è stato dato; **Studente** e **Corso** sono anche chiavi esterne della relazione. Si noti che uno studente può avere più voti registrati per lo stesso corso.
- **PianiDiStudio** rappresenta i piani di studio presentati dagli studenti dell'Università di Genova. Si supponga che gli studenti di tutte le facoltà debbano presentare, per ogni anno accademico a cui si iscrivono, il piano di studio corrispondente. **Studente**, **AnnoAccademico** e **Anno** sono chiave primaria della relazione e rappresentano, rispettivamente, lo studente che presenta il piano di studio, l'anno accademico a cui il piano di studio si riferisce e l'anno, relativamente al corso di laurea (es. primo anno, secondo, etc), al quale il piano di studio si riferisce. Si noti che uno studente può presentare, in anni accademici diversi, più piani di studio per lo stesso anno di corso (studenti ripetenti).

Utilizzando il Query Analyzer di SQL Server, connettersi al server **MURPHY** e selezionare la base di dati **Esercitazione3**, che contiene un campione di dati che idealmente potrebbero essere a disposizione dell'Università di Genova. Dopodichè, eseguire le interrogazioni elencate di seguito.

### Osservazioni

- Si utilizzi l'Enterprise Manager o l'Object Browser di SQL Server per visualizzare i dettagli implementativi della base di dati, che possono essere utili per l'esecuzione corretta delle interrogazioni (tipicamente, i tipi di dati utilizzati come domini per gli attributi, le relazioni e i vincoli impostati).
- Si utilizzino, dove si ritiene utile o necessario, gli alias di colonna.
- Esprimere i predicati di join in modi diversi (join esplicito o predicati di join nella clausola where)
- Svolgere almeno tre interrogazioni per ogni gruppo di interrogazioni. In questo modo, se ci sono problemi, potete chiedere assistenza durante le ore di lezione.

**Uso di colonne virtuali e controlli su valori nulli.** Scrivere i comandi SQL che permettano di ottenere e visualizzare i seguenti risultati.

1. Restituire, per ogni corso di laurea, il nome del corso di laurea e il numero di iscritti;
2. per ogni studente, restituire nome, cognome ed età (utilizzare a tal fine le funzioni `getDate()` e `year()` di SQL Server<sup>1</sup>);

---

<sup>1</sup>Consultare il manuale in linea per capire come utilizzarle.

3. restituire la matricola degli studenti, iscritti nell'Anno Accademico 2000/2001, che non hanno un relatore;
4. restituire nome e cognome (in un'unica colonna) degli studenti che hanno conseguito la laurea ordinati in base al cognome e preceduti dal titolo 'Dott.' (es. 'Dott. Anna Maddalena');
5. per ogni studente, restituire la sua matricola e la media dei voti ottenuti negli esami che ha passato (dare alla colonna con la media il nome 'mediaMatematica');
6. per ogni studente che ha sostenuto almeno 3 esami, restituire la sua matricola e la media dei voti ottenuti negli esami che ha passato scartando il voto più alto e quello più basso (dare alla colonna con la media il nome 'mediaPesata');
7. determinare la votazione media riportata per ogni corso attivo per cui esiste un professore titolare.
8. Determinare la matricola e la media dei voti registrati degli studenti di informatica che hanno registrato almeno due voti nel mese di giugno 2000.

**Uso di sottointerrogazioni** Scrivere i comandi SQL che permettano di ottenere (visualizzando i risultati):

1. elenco dei nominativi dei professori che non sono titolari di corsi attivati nell'A.A. '2001/2002';
2. selezionare i professori che hanno un omonimo tra gli studenti dell'Università di Genova o sono stati studenti dell'università di Genova (cioè usare il cognome e nome del professore per fare questa interrogazione);
3. l'elenco dei corsi di laurea che hanno più professori titolari di corsi rispetto a quelli che insegnano in corsi, anche non attivati, a chimica;
4. l'elenco dei corsi di laurea che nell'A.A. 2000/2001 hanno meno iscritti di quelli, per lo stesso A.A., che si sono avuti ad informatica;
5. l'elenco alfabetico, senza duplicati e in ordine decrescente, degli studenti che hanno presentato il piano di studi per il quinto anno del corso di laurea di informatica nell'A.A. '2001/2002' e sono in tesi (hanno assegnato un relatore) con professori che sono titolari di corsi, attivati nell'A.A. '2001/2002', presso il corso di laurea di Matematica;
6. Determinare la matricola degli studenti di informatica che nel mese di giugno 2000 hanno registrato voti per i corsi di basi di dati 1 e interfacce grafiche.

**Uso di sottointerrogazioni correlate** Scrivere i comandi SQL che permettano di ottenere e visualizzare i seguenti risultati.

1. Per ogni corso, indicare la matricola degli studenti che hanno ottenuto un voto sotto la media di ogni corso, indicando corso e voto.
2. Per ogni corso, indicare il cognome e il nome degli studenti che hanno ottenuto un voto sotto la media di ogni corso, indicando denominazione del corso e voto.
3. Determinare la matricola degli studenti di informatica che hanno registrato (almeno) due voti per corsi diversi nello stesso mese, con la media dei voti riportati.
4. Determinare gli studenti non ancora in tesi che hanno passato tutti gli esami del proprio corso di laurea.
5. Determinare, di ogni sessione d'esame (una al mese, per ogni corso) la matricola e il voto degli studenti che hanno ottenuto il voto massimo.
6. Determinare i professori che hanno registrato i voti piu' alti, in media, per i corsi di cui sono titolari.
7. Determinare i corsi in cui si ha il maggior numero di studenti con voti insufficienti.
8. Determinare i nominativi dei professori che insegnano nel maggior numero di corsi.
9. Determinare la frequenza delle bocciature, suddivisa per sessione (mesi).
10. Determinare le sessioni(mesi) in cui si ha il maggior numero di studenti bocciati, con la frequenza delle bocciature.