

# Basi di Dati

Soluzione della prova scritta del 22 Settembre 2005

## Esercizio 1 (3 punti)

Illustrare sinteticamente le diverse fasi in cui si suddivide la progettazione *logica* di una base di dati, a partire dallo schema E-R prodotto in fase di progettazione concettuale.

### Schema Relazionale per gli Esercizi 2 e 3

Considerare lo schema di base di dati contenente le relazioni:

DOCENTE (CodiceFiscale, Nome, Cognome)

STUDENTE (CodiceFiscale, Nome, Cognome)

ARGOMENTO(Codice, Descrizione)

LEZIONE(Argomento, Data, Docente, NumeroStudenti)

Foreign key Argomento References ARGOMENTO(Codice)

Foreign key Docente References DOCENTE(CodiceFiscale)

INTERROGAZIONE(Argomento, Data, Studente, Voto)

Foreign key Argomento, Data References LEZIONE(Argomento, Data)

Foreign key Studente References STUDENTE(CodiceFiscale)

## Esercizio 2 (4 punti)

Formulare in SQL la seguente interrogazione: *mostrare per ogni argomento la media dei voti riportati dagli studenti interrogati su quell'argomento, considerando solo gli studenti che sono stati interrogati almeno tre volte sull'argomento in questione.*

### Soluzione

```
SELECT I.Argomento, AVG(I.Voto)
FROM Interrogazione I,
WHERE I.Studente IN (SELECT I1.Studente
                     FROM Interrogazione I1
                     WHERE I1.Argomento = I.Argomento
                     HAVING COUNT(*) >= 3)
GROUP BY I.Argomento
```

## Esercizio 3 (8 punti)

Formulare in Algebra Relazionale (3 punti) e SQL (5 punti) la seguente interrogazione: *Mostrare il codice fiscale del docente che ha svolto lezioni su tutti gli argomenti con descrizione 'Fisica'*

### Soluzione

#### Algebra Relazionale

$\pi_{\text{CodiceFiscale}}(\text{DOCENTE}) - \pi_{\text{CodiceFiscale}}((\pi_{\text{Codice}, \text{CodiceFiscale}}(\sigma_{\text{Descrizione} = \text{'Fisica'}}(\text{ARGOMENTO} \text{ Join } \text{DOCENTE}))) - \pi_{\text{Argomento}, \text{Docente}}(\text{LEZIONE}))$

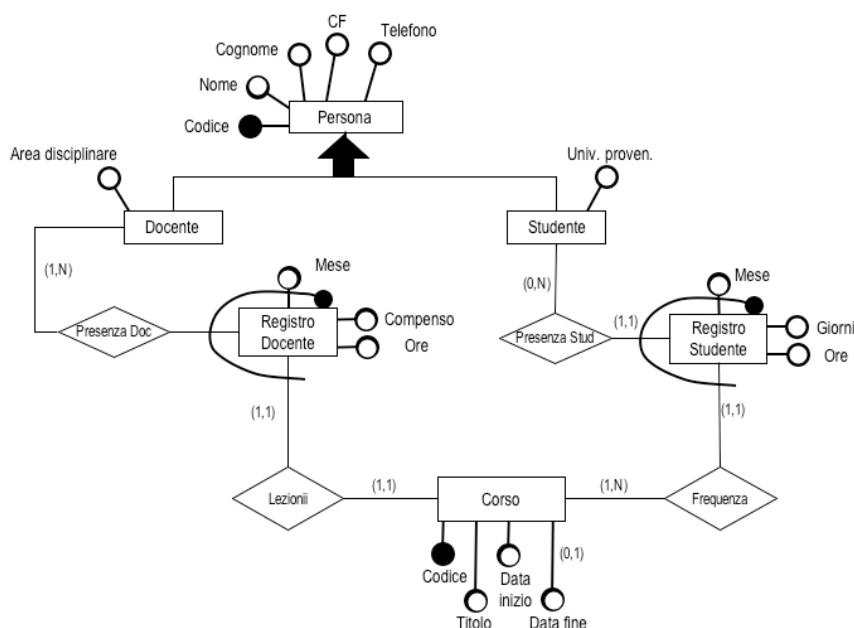
#### SQL

```
SELECT DISTINCT CodiceFiscale
FROM Lezione L1
WHERE NOT EXISTS (SELECT *
                  FROM Argomento A
                  WHERE Descrizione = 'Fisica'
                  AND NOT EXISTS (SELECT *
                                FROM Lezione L2
                                WHERE L1.Docente = L2.Docente
                                AND A.Codice = L2.Argomento))
```

#### Esercizio 4 (6 punti)

Progettare uno schema E-R per la gestione delle ore di lezione di corsi di specializzazione post-laurea.

- Per i docenti, identificati con un codice, si richiede di registrare il nome, cognome, luogo e data di nascita, la residenza, l'area disciplinare, il codice fiscale, il titolo di studio, il numero di telefono ed eventuali note.
- Ogni corso è identificato da un codice, ha un titolo, una data di inizio e una data di fine (quest'ultima facoltativa)
- Per gli studenti, identificati con un codice, memorizziamo il nome, cognome, luogo e data di nascita, residenza, codice fiscale, titolo di studio e numero di telefono, l'Università di provenienza.
- Ogni studente può frequentare più corsi (anche in anni diversi) e le relative presenze, raccolte mensilmente, vengono registrate sia in numero di giorni e sia in numero di ore effettive.
- Anche le ore svolte dai docenti sono raccolte mensilmente. Un docente può svolgere le ore di lezione per diversi corsi, per più mesi
- Ogni docente riceve un compenso per le ore di lezione effettuate. Il compenso per le ore di lezione può cambiare rispetto al mese, al corso e al docente.



#### Esercizio 5 (4 punti)

Tradurre il diagramma ER dell'esercizio precedente in uno schema logico. Discutere eventuali ristrutturazioni dello schema. Evidenziare le chiavi delle relazioni e i vincoli di integrità referenziale.

DOCENTE(Codice, CF, Nome, Cognome, Telefono, Abilitazione)

STUDENTE(Codice, CF, Nome, Cognome, Telefono, UniversitàProvenienza)

CORSO(Codice, Titolo, DataIniziale, DataFinale)

REGISTRODOCENTE(Docente, Mese, Corso, Ore, Compenso)

*Foreign Key Docente References* DOCENTE(Codice)

*Foreign Key Corso References* CORSO(Codice)

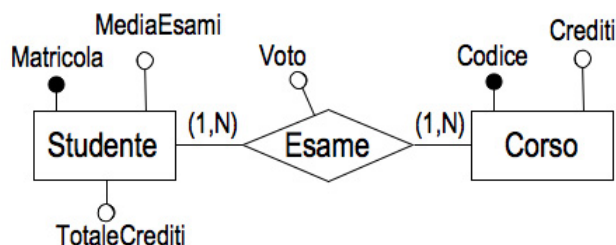
REGISTROSTUDENTE(Studente, Mese, Corso, Giorni, Ore)

*Foreign Key Studente References* STUDENTE(Codice)

*Foreign Key Corso References* CORSO(Codice)

### Esercizio 6 (5 punti)

Dato il seguente frammento di schema E-R, con il seguente volume dei dati e le seguenti operazioni, decidere se è conveniente conservare nello schema gli attributi derivati *MediaEsami* (ottenuta pesando il voto di ciascun esame per il numero dei crediti attribuiti al corso) e *TotaleCrediti*, trascurando l'occupazione di memoria di tale dato. (Per non appesantire lo schema non sono stati riportati altri attributi delle entità, quale Cognome e nome degli studenti, data dell'esame, nome del corso, ecc., che devono essere supposti presenti)



**OP 1:** Dati la matricola di uno studente e il codice di un corso, inserire l'esame sostenuto dallo studente relativamente a quel corso

**OP 2:** Visualizzare i dati di uno studente

**Tavola dei volumi**

Concetto	Tipo	Volume
Studente	E	5.000
Corso	E	400
Esame	R	100.000

**Tavola delle operazioni**

Operazione	Tipo	Frequenza
OP 1	I	1000/Mese
OP 2	B	50/Mese

#### **Soluzione**

Nota: consideriamo il costo di un accesso in lettura pari a 1 e il costo di un accesso in scrittura pari a 2

#### **Con l'attributo ridondante**

**Costo di OP 1:** 1 accesso in scrittura su Esame, 1 accesso in lettura su Corso (lettura del numero di Crediti del corso del quale si è sostenuto l'esame), 1 accesso in lettura su Studente (lettura dei valori correnti di MediaEsami e TotaleCrediti), 1 accesso in scrittura su Studente (aggiornamento MediaEsami e TotaleCrediti) = 6

**Costo di OP 2:** 1 accesso in lettura su Studente (per visualizzare tutti i dati)

**Costo medio totale** al giorno:  $6 \cdot 1.000 + 1 \cdot 50 = 6.050$

#### **Senza l'attributo ridondante**

**Costo di OP 1:** 1 accesso in scrittura su Esame

**Costo di OP 2:** 1 accesso in lettura su Studente, 20 ( $100.000/5.000$ ) accessi in lettura su Esame, 20 accessi in lettura su Corso (lettura del numero di Crediti del corso) = 41

**Costo medio totale** al giorno:  $2 \cdot 1.000 + 41 \cdot 50 = 4.050$

**Conviene eliminare l'attributo ridondante**

**Esercizio 7 (3 punti)**

Spiegare brevemente il ruolo della variabile \$row e delle funzioni mysql\_\*() nel codice di seguito riportato

```
<HTML>
<HEAD>
<TITLE>Test Apache + PHP + MySQL</TITLE>
</HEAD>
<BODY>
<?php
// parametri del database
$db_host = "localhost";
$db_user = "root";
$db_password = "";
$db_name = "testphp";
$db = mysql_connect($db_host, $db_user, $db_password);

if ($db == FALSE) die ("Errore!");

mysql_select_db($db_name, $db) or die ("Errore!");

$query = "SELECT * FROM tabella1 ORDER BY nome";
$result = mysql_query($query);

while($row = mysql_fetch_array($result)) {
    echo 'ID = '.$row['id'].'<br>';
    echo 'Nome = '.$row['nome'].'<br>';
    echo 'Cognome = '.$row['cognome'].'<br>';
    echo '<hr>';
}

mysql_close($db);

?>
</BODY>
</HTML>.
```