

Cognome:

Nome:

Matricola:

Basi di Dati – Corso di Laurea in Informatica Umanistica

Verifica del 24.03.2006

Parte 1 – modello relazionale, SQL (A)

Docente: Dino Pedreschi

Si consideri il seguente schema di base di dati universitaria:

TABLE Studenti
{*Matricola* integer **PRIMARY KEY**,
Cognome char(30) **NOT NULL**,
Nome char(20) **NOT NULL**,
Sesso char(1),
CorsoStudi char(20),
AnnoCorso integer,
Relatore char(4) **REFERENCES**
Professori(Codice) }

TABLE Esami
{*Studente* integer **REFERENCES**
Studenti(*Matricola*),
Corso char(3) **REFERENCES** *Corsi*(*Codice*),
Data date **NOT NULL**,
Voto integer **NOT NULL**,
Lode bool **NOT NULL**,
CHECK (*voto* >= 18 AND *voto* <= 30),
CHECK (*not lode* OR *voto* = 30),
PRIMARY KEY(*Studente*, *Corso*) }

TABLE Professori
{*Codice* char(4) **PRIMARY KEY**,
Cognome char(30) **NOT NULL**,
Nome char(20) **NOT NULL**,
Qualifica char(15),
Facoltà char(20) }

TABLE Corsi
{*Codice* char(3) **PRIMARY KEY**,
Titolo varchar(20) **NOT NULL**,
CorsoStudi char(20),
Docente char(4) **REFERENCES** *Professori*(*Codice*) }

1) Elencare i corsi che sono tenuti da un professore ordinario della facoltà di giurisprudenza (algebra oppure SQL, punti 5)

2) Elencare ogni professore che è relatore di uno studente che ha sostenuto almeno un esame con il professore medesimo (algebra oppure SQL, punti 5)

Cognome:

Nome:

Matricola:

3) Elencare i professori che tengono un corso alla laurea in Informatica Umanistica ed uno anche alla laurea in Lettere oppure a quella in Informatica (algebra o SQL, punti 7)

4) Elencare gli studenti che hanno sostenuto l'esame di "basi di dati" e, successivamente, quello di "progettazione siti web"(algebra o SQL, punti 8)

5) Elencare gli studenti che hanno preso almeno un trenta e lode e nessun voto al di sotto del 27 (algebra o SQL, punti 7)

Cognome:

Nome:

Matricola:

Basi di Dati – Corso di Laurea in Informatica Umanistica

Verifica del 24.03.2006

Parte 1 – modello relazionale, SQL (B)

Docente: Dino Pedreschi

Si consideri il seguente schema di base di dati universitaria:

TABLE Studenti
{*Matricola* integer **PRIMARY KEY**,
Cognome char(30) **NOT NULL**,
Nome char(20) **NOT NULL**,
Sesso char(1),
CorsoStudi char(20),
AnnoCorso integer,
Relatore char(4) **REFERENCES**
Professori(Codice) }

TABLE Esami
{*Studente* integer **REFERENCES**
Studenti(Matricola),
Corso char(3) **REFERENCES** *Corsi(Codice)*,
Data date **NOT NULL**,
Voto integer **NOT NULL**,
Lode bool **NOT NULL**,
CHECK (*voto* >= 18 AND *voto* <= 30),
CHECK (*not lode* OR *voto* = 30),
PRIMARY KEY(*Studente*, *Corso*) }

TABLE Professori
{*Codice* char(4) **PRIMARY KEY**,
Cognome char(30) **NOT NULL**,
Nome char(20) **NOT NULL**,
Qualifica char(15),
Facoltà char(20) }

TABLE Corsi
{*Codice* char(3) **PRIMARY KEY**,
Titolo varchar(20) **NOT NULL**,
CorsoStudi char(20),
Docente char(4) **REFERENCES** *Professori(Codice)* }

1) Elencare i professori che sono relatori di almeno una studentessa (algebra oppure SQL, punti 5)

2) Elencare gli studenti che hanno come relatore un professore docente di un corso di cui hanno sostenuto l'esame (algebra oppure SQL, punti 5)

Cognome:

Nome:

Matricola:

3) Elencare i professori che tengono un corso alla laurea in Informatica Umanistica ed uno alla laurea in Lettere (algebra o SQL, punti 7)

4) Elencare gli studenti che hanno sostenuto sia l'esame di "basi di dati" che quello di "progettazione siti web" prendendo nel secondo un voto superiore al primo (algebra o SQL, punti 8)

5) Elencare i professori ordinari che non hanno mai dato un trenta e lode (algebra o SQL, punti 7)