



Ministero dell'Istruzione, dell'Università e della Ricerca
M963 – ESAME DI STATO DI ISTRUZIONE SECONDARIA SUPERIORE

Indirizzo: ITIA - INFORMATICA E TELECOMUNICAZIONI
ARTICOLAZIONE INFORMATICA

Tema di: INFORMATICA

Il candidato svolga la prima parte della prova e risponda a due tra i quesiti proposti nella seconda parte.

PRIMA PARTE

Si vuole realizzare una web community per condividere dati e commenti relativi a eventi dal vivo di diverse categorie, ad esempio concerti, spettacoli teatrali, balletti, ecc. che si svolgono in Italia.

Gli eventi vengono inseriti sul sistema direttamente dai membri stessi della community, che si registrano sul sito fornendo un nickname, nome, cognome, indirizzo di e-mail e scegliendo una o più categorie di eventi a cui sono interessati.

Ogni membro iscritto riceve periodicamente per posta elettronica una newsletter, emessa automaticamente dal sistema, che riporta gli eventi delle categorie da lui scelte, che si svolgeranno nella settimana seguente nel territorio provinciale dell'utente.

I membri registrati possono interagire con la community sia inserendo i dati di un nuovo evento, per il quale occorre specificare categoria, luogo di svolgimento, data, titolo dell'evento e artisti coinvolti, sia scrivendo un post con un commento ed un voto (da 1 a 5) su un evento.

Il sito della community offre a tutti, sia membri registrati sia utenti anonimi, la consultazione dei dati on line, tra cui:

- visualizzazione degli eventi di un certo tipo in ordine cronologico, con possibilità di filtro per territorio di una specifica provincia
- visualizzazione di tutti i commenti e voti relativi ad un evento.

Il candidato, fatte le opportune ipotesi aggiuntive, sviluppi

1. un'analisi della realtà di riferimento individuando le possibili soluzioni e scelga quella che a suo motivato giudizio è la più idonea a rispondere alle specifiche indicate
2. uno schema concettuale della base di dati
3. uno schema logico della base di dati
4. la definizione in linguaggio SQL di un sottoinsieme delle relazioni della base di dati in cui siano presenti alcune di quelle che contengono vincoli di integrità referenziale e/o vincoli di dominio, laddove presenti



Ministero dell'Istruzione, dell'Università e della Ricerca
M963 – ESAME DI STATO DI ISTRUZIONE SECONDARIA SUPERIORE

Indirizzo: ITIA - INFORMATICA E TELECOMUNICAZIONI
ARTICOLAZIONE INFORMATICA

Tema di: INFORMATICA

5. le seguenti interrogazioni espresse in linguaggio SQL:
 - a. elenco degli eventi già svolti, in ordine alfabetico di provincia
 - b. elenco dei membri che non hanno mai inserito un commento
 - c. per ogni evento il voto medio ottenuto in ordine di categoria e titolo
 - d. i dati dell'utente che ha registrato il maggior numero di eventi
6. il progetto della pagina dell'interfaccia WEB che permetta ad un utente registrato di svolgere le operazioni specificate
7. la codifica in un linguaggio a scelta di un segmento significativo dell'applicazione Web che consente l'interazione con la base di dati.

SECONDA PARTE

Il candidato (che potrà eventualmente avvalersi delle conoscenze e competenze maturate attraverso esperienze di alternanza scuola-lavoro, stage o formazione in azienda) risponda a due quesiti a scelta tra quelli sotto riportati:

- I In relazione al tema proposto nella prima parte, descriva in che modo è possibile integrare lo schema concettuale sopra sviluppato in modo da poter gestire anche inserzioni pubblicitarie. Ogni inserzione è costituita da un testo e un link e può essere correlata a una o più categorie di eventi in modo da essere visualizzata in funzione dei contenuti visitati e delle preferenze degli utenti.
- II In relazione al tema proposto nella prima parte, progetti un layout di pagina idoneo a garantire un aspetto grafico comune a tutte le pagine dell'applicazione e ne codifichi alcuni elementi in un linguaggio per la formattazione di pagine Web a sua scelta.



Ministero dell' Istruzione, dell' Università e della Ricerca
M963 – ESAME DI STATO DI ISTRUZIONE SECONDARIA SUPERIORE

Indirizzo: ITIA - INFORMATICA E TELECOMUNICAZIONI
 ARTICOLAZIONE INFORMATICA

Tema di: INFORMATICA

III Si consideri la seguente tabella:

Cognome	Nome	Telefono	Livello	Tutor	Tel-tutor	Anticipo versato
Verdi	Luisa	345698741	avanzato	Bianca	334563215	100
Neri	Enrico	348523698	avanzato	Carlo	369852147	150
Rosi	Rosa	347532159	base	Alessio	333214569	120
Bianchi	Paolo	341236547	base	Carlo	369852147	150
Rossi	Mario	349567890	base	Carlo	369852147	90
Neri	Enrico	348523698	complementi	Dina	373564987	100

Il candidato verifichi le proprietà di normalizzazione e proponga uno schema equivalente che rispetti la 3^a Forma Normale, motivando le scelte effettuate.

IV Nella formalizzazione di uno schema concettuale, le associazioni tra entità sono caratterizzate da una cardinalità: esponga il significato e la casistica che si può presentare.

Durata massima della prova: 6 ore.

È consentito l'uso di manuali tecnici (references riportanti solo la sintassi, non guide) dei linguaggi utilizzati.

È consentito l'uso del dizionario bilingue (italiano-lingua del paese di provenienza) per i candidati di madrelingua non italiana.

Non è consentito lasciare l'Istituto prima che siano trascorse 3 ore dalla dettatura del tema.

3) UNO SCHEMA LOGICO DATA BD

(a) MAPPING RELAZIONE ASSOCIAZIONE "SEGNALE" TRA LE ENTITA' "MEMBRO" E "EVENTO" DI POS. N:1

MEMBRO (NICKNAME, COGNOME, NOME, PRIN, INDIRIZZO, CUIA, CAP, EMAIL)

EVENTO (CODE, TIPO, DATA, DURATA, CODL1, CODC1)

- ON "CODL1" CHE E' FK SU "CODL" DELLA RELAZIONE "LUOGO"
- ON "CODC1" CHE E' FK SU "CODC" DELLA RELAZIONE "CATEGORIA"

SEGNALE (NICKNAME1, CODE1, DATA, SEGNALE)

- ON "NICKNAME1" CHE E' FK SU "NICKNAME" DELLA RELAZIONE "MEMBRO"
- ON "CODE1" CHE E' FK SU "CODE" DELLA RELAZIONE "EVENTO"

$$\left\{ \begin{array}{l} VR_{NICKNAME1} (SEGNALE) \subseteq VR_{NICKNAME} (MEMBRO) \\ VR_{CODE1} (SEGNALE) \subseteq VR_{CODE} (EVENTO) \end{array} \right.$$
 VR DOVUTA AL RAPPORTO RELAZIONALE DI UNA RELAZIONE ASSOCIATA N:1

$$VR_{CODE} (EVENTO) \subseteq VR_{CODE1} (SEGNALE)$$
 VR DOVUTA AL TIPO ASS. INV. "ESEGNALE"

(b) MAPPING RELAZIONE ASSOCIAZIONE "SERIE" TRA LE ENTITA' "MEMBRO" E "POST" DI POS. 1:1

MEMBRO: RELAZ. CUI MAPPA

POST (CODP, COGNOME, VOTO, DATA, NICKNAME2, CODE2)

- ON "NICKNAME2" CHE E' FK SU "NICKNAME" DELLA RELAZIONE "MEMBRO"
- ON "CODE2" CHE E' FK SU "CODE" DELLA RELAZIONE "EVENTO"

$$VR_{NICKNAME2} (POST) \subseteq VR_{NICKNAME} (MEMBRO)$$
 VR DOVUTA AL TIPO ASS. INV. "E SERIE"

(c) MAPPING RELAZIONE ASSOCIAZIONE "RUEVE" TRA LE ENTITA' "EVENTO" E "POST" DI POS. 1:1

EVENTO: RELAZ. CUI MAPPA

POST: RELAZ. CUI MAPPA

$$VR_{CODE2} (POST) \subseteq VR_{CODE} (EVENTO)$$
 VR DOVUTA AL TIPO ASS. INV. "E RUEVE"

(d) MAPPING RELAZIONE ASSOCIAZIONE "POSSEDE" TRA LE ENTITA' "EVENTO" E "CATEGORIA" DI POS. N:1

EVENTO: RELAZ. CUI MAPPA

CATEGORIA (CODC, DESCRIZIONE)

$$VR_{CODC1} (EVENTO) \subseteq VR_{CODC} (CATEGORIA)$$
 VR DOVUTA AL TIPO ASS. DI R. "POSSEDE"

(e) MAPPING RAZIONALE ALGEBRE "CARATTERI" TRA LE ENTRA "LUOGO" ed "EVENTO" di \mathbb{N}

LUOGO (CodL , DEDICAZIONE, LATITUDINE, LONGITUDINE, REGIONE, PROVINCIA)

EVENTO: RUMORE CUI MAPPA

$VR_{\text{CodL}}(\text{LUOGO}) \subseteq VR_{\text{CodL}}(\text{EVENTO})$ VR DALLA TOT-ASS. DIR "CARATTERI"

$VR_{\text{CodL}}(\text{EVENTO}) \subseteq VR_{\text{CodL}}(\text{LUOGO})$ VR DALLA TOT-ASS. INV "E CARATTERI"

(f) MAPPING RAZIONALE ALGEBRE "PARTELLA" TRA LE ENTRA "ARABIA" e "EVENTO" di \mathbb{N}

ARABIA (CodA , PENINISOLE, I/BOCCA)

EVENTO: RUMORE CUI MAPPA

PARTELLA (CodA3 , CodE3)

- GN "CodA3" CHE E' PK SU "CodA" DALLA MAPP. "ARABIA"

- GN "CodE3" CHE E' PK SU "CodE" DALLA MAPP. "EVENTO"

$VR_{\text{CodA3}}(\text{PARTELLA}) \subseteq VR_{\text{CodA}}(\text{ARABIA})$ VR DALLA ALGEBRA DI UN ALGEBRE \mathbb{N}

$VR_{\text{CodE3}}(\text{PARTELLA}) \subseteq VR_{\text{CodE}}(\text{EVENTO})$ RAZIONALE DI UN ALGEBRE \mathbb{N}

$VR_{\text{CodA}}(\text{ARABIA}) \subseteq VR_{\text{CodA3}}(\text{PARTELLA})$ VR DALLA TOT-ASS. DIR "PARTELLA"

$VR_{\text{CodE}}(\text{EVENTO}) \subseteq VR_{\text{CodE3}}(\text{PARTELLA})$ VR DALLA TOT-ASS. INV "PARTELLA"

(g) MAPPING RAZIONALE DI UNO

VINGLI DI INTERAZIONE IMPLICI di PK

\Rightarrow V. INTERAZIONE \rightarrow INTERM SU PK \mathbb{N} -pk

VINGLI DI INTERAZIONE IMPLICI DALLA ASS. TOTALE DALLA ASS. DIR

\Rightarrow V. INTERRELAZIONE \rightarrow ESTER REFERENZIALE

$V1: (\dots)$

$\Rightarrow V1 / \text{POST} : (V1 \text{ BETWEEN } 1 \text{ AND } 5)$

VINGLI INTRA RAZIONALE \rightarrow INTERAZIONE SINGOLA \mathbb{N} -pk SU BILIBO DI 1 ASS. DIR

(N.B. CHE TOTI GU ALM)

V2(...) ⇒ V2(Categoria): (DEJIMPRE IN ("GOLLENZO", "BAMBO", "SPECIALI TRATTI"))

V3(..) ⇒ V3(Luogo): (REGIONE IN ("VALLE D'AOSTA", "CAMPANIA", ..))

V4(⊙) ⇒ V4(Luogo): (PROVINCIA IN ("NA", "AV", "CE", "SA", "BN", ..))

V5(..) ⇒ V5(Artista): (TIPOLOGIA IN ("SOLISTA", "BANDO", "CORO", "ALTRA", "BAMBO"))

A questi vanno aggiunti VR dove per natura di una certa associazione di
 "ALTERAZIONE" N°N che vengono realizzati in singoli interventi →
 ESTERNE REFERENZIALI

Lo schema globale così ottenuto è in **3FN**

SQL

```

4 CREATE DATABASE WebCommunity;
USE WebCommunity;
CREATE DOMAIN MioVoto AS INT(1) NOT NULL
CHECK (VALUE BETWEEN 1 AND 5); // V1
CREATE DOMAIN MioDejimpres AS VARCHAR(20) NOT NULL
CHECK (VALUE IN ("GOLLENZO", "BAMBO", "SPECIALI TRATTI")); // V2
CREATE DOMAIN MioRegioni AS VARCHAR(20) NOT NULL
CHECK (VALUE IN ("VALLE D'AOSTA", "CAMPANIA", ..)); // V3
CREATE DOMAIN MioProvince AS VARCHAR(20) NOT NULL
CHECK (VALUE IN ("NA", "AV", "CE", "SA", "BN", ..)); // V4
CREATE DOMAIN MioTipologie AS VARCHAR(10) NOT NULL
CHECK (VALUE IN ("SOLISTA", "BANDO", "CORO", "ALTRA", "BAMBO")); // V5

CREATE TABLE Membri (
  NICKNAME VARCHAR(10) NOT NULL,
  COGNOME VARCHAR(30) NOT NULL,
  NOME VARCHAR(30) NOT NULL,
  
```

```

DATAIN DATE
INDIRIA VARCHAR(100)
CUNA VARCHAR(30),
CAP VARCHAR(5)
EMILIA VARCHAR(50) NOT NULL,
PRIMARY KEY (MICKNAME)
);

```

```

CREATE TABLE CATEGORIA
(
  CODC VARCHAR(10) NOT NULL,
  DESCRIZIONE MIADESCRIZIONE,
  PRIMARY KEY (CODC)
);

```

```

CREATE TABLE LUOGO
(
  CODL VARCHAR(10) NOT NULL,
  DESCRIZIONE VARCHAR(100) NOT NULL,
  LATITUDINE VARCHAR(10) NOT NULL,
  LONGITUDINE VARCHAR(10) NOT NULL,
  REGIONE MIAREGIONE,
  PROVINTA MIAPROVINTA,
  PRIMARY KEY (CODL)
);

```

```

CREATE TABLE EVENTO
(
  CODE VARCHAR(10) NOT NULL,
  TITOLO VARCHAR(50) NOT NULL,
  DATA SUBMISSIONE DATE NOT NULL,

```

```

CodL1    VARCHAR (10) NOT NULL,
CodC1    VARCHAR (10) NOT NULL,
PRIMARY KEY (CodE),
FOREIGN KEY (CodL1) REFERENCES Luoghi (CodL)
ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY (CodC1) REFERENCES Categorie (CodC)
ON DELETE CASCADE
ON UPDATE CASCADE
);

```

CREATE TABLE **Segnali**

```

(
NickName1 VARCHAR (10) NOT NULL,
Code1     VARCHAR (10) NOT NULL,
DataSegnali DATE NOT NULL,
PRIMARY KEY (NickName1, Code1),
FOREIGN KEY (NickName1) REFERENCES Membre (NickName)
ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY (Code1) REFERENCES Eventi (Code)
ON DELETE CASCADE
ON UPDATE CASCADE
);

```

CREATE TABLE **Post**

```

(
CodP     VARCHAR (10) NOT NULL,
Contenuto VARCHAR (255) NOT NULL,
Visto    Mlo Visto,
DataPubb DATE NOT NULL,

```

L. 10/10 7

```

Nickname2 VARCHAR(10) NOT NULL,
CodE2 VARCHAR(10) NOT NULL,
PRIMARY KEY (CodE2),
FOREIGN KEY (Nickname2) REFERENCES Member (Nickname)
ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY (CodE2) REFERENCES Evento (CodE)
ON DELETE CASCADE
ON UPDATE CASCADE
);

```

CREATE TABLE **Arrotta**

```

(
CodA VARCHAR(10) NOT NULL,
DENormalzone VARCHAR(100) NOT NULL,
Tipologia Multiploca,
PRIMARY KEY (CodA)
);

```

CREATE TABLE **Parabucca**

```

(
CodA3 VARCHAR(10) NOT NULL,
CodE3 VARCHAR(10) NOT NULL,
PRIMARY KEY (CodA3, CodE3),
FOREIGN KEY (CodA3) REFERENCES Arrotta (CodA)
ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY (CodE3) REFERENCES Evento (CodE)
ON DELETE CASCADE
ON UPDATE CASCADE
);

```

query

5 a) ELVIO DEGLI EVENTI CHE SIA IN OLTRE ALFABETICI DI PROVINTA

Gu' SUBO Para Solumis < NOW()

```

SELECT *
FROM EVENTO LUOGO
WHERE (Cod L1 = Cod L) AND (Para Solumis < NOW())
ORDER BY PROVINTA ASC;

```

Annotations: Red circles around 'Para Solumis < NOW()' and 'CURDATE()' with arrows pointing to 'Gu' SUBO' above.

b) ELVIO DEI MEMBRI CHE NON HANNO MAI INSERITO IN GAZZINO

```

SELECT *
FROM MEMBRO
WHERE NICKNAME NOT IN (
  SELECT NICKNAME
  FROM MEMBRO, POST
  WHERE (NICKNAME = NICKNAME)
)

```

Annotations: Red cloud-like shape around the subquery. Text below: 'oppure <> ALL'.

query che DA TE O DA TE MEMBRI CHE HANNO INSERITO POST

c) PER OGNI EVENTO IL VOTO MEDIO OTTENUTO IN ORDINE DI CATEGORIA E TITOLO

```

SELECT CATEGORIA, DESCRIZIONE, EVENTO, TITOLO, AVG(VOTO)
FROM EVENTO, POST, CATEGORIA
WHERE (Cod E = Cod E2) AND (Cod C1 = Cod C)
GROUP BY CATEGORIA, DESCRIZIONE, EVENTO, TITOLO;

```

d) I DATI DELL'UTENTE CHE HA REGISTRATO IL MAGGIOR NUMERO DI UTENTI

1) CREATE VIEW t1 AS (SELECT NICKNAME, COUNT(*) AS NumEventi
FROM MEMBR, SEGWAA
WHERE (NICKNAME = NICKNAME1)
GROUP BY NICKNAME);

2) SELECT *
FROM MEMBR

WHERE NICKNAME = (SELECT NICKNAME
FROM t1

WHERE NumEventi = (SELECT MAX(NumEventi)
FROM t1));

LA QUERY 1) CREA UNA VISTA (TABLE) COSI ORGANIZZATA

NICKNAME	NUMEVENTI
rijo	42
pippe	22
jamy	82

+ 1 ESTERNA

LA QUERY 2) HA 2 SOTTOQUERY CHE FUNZIONANO INTERNE

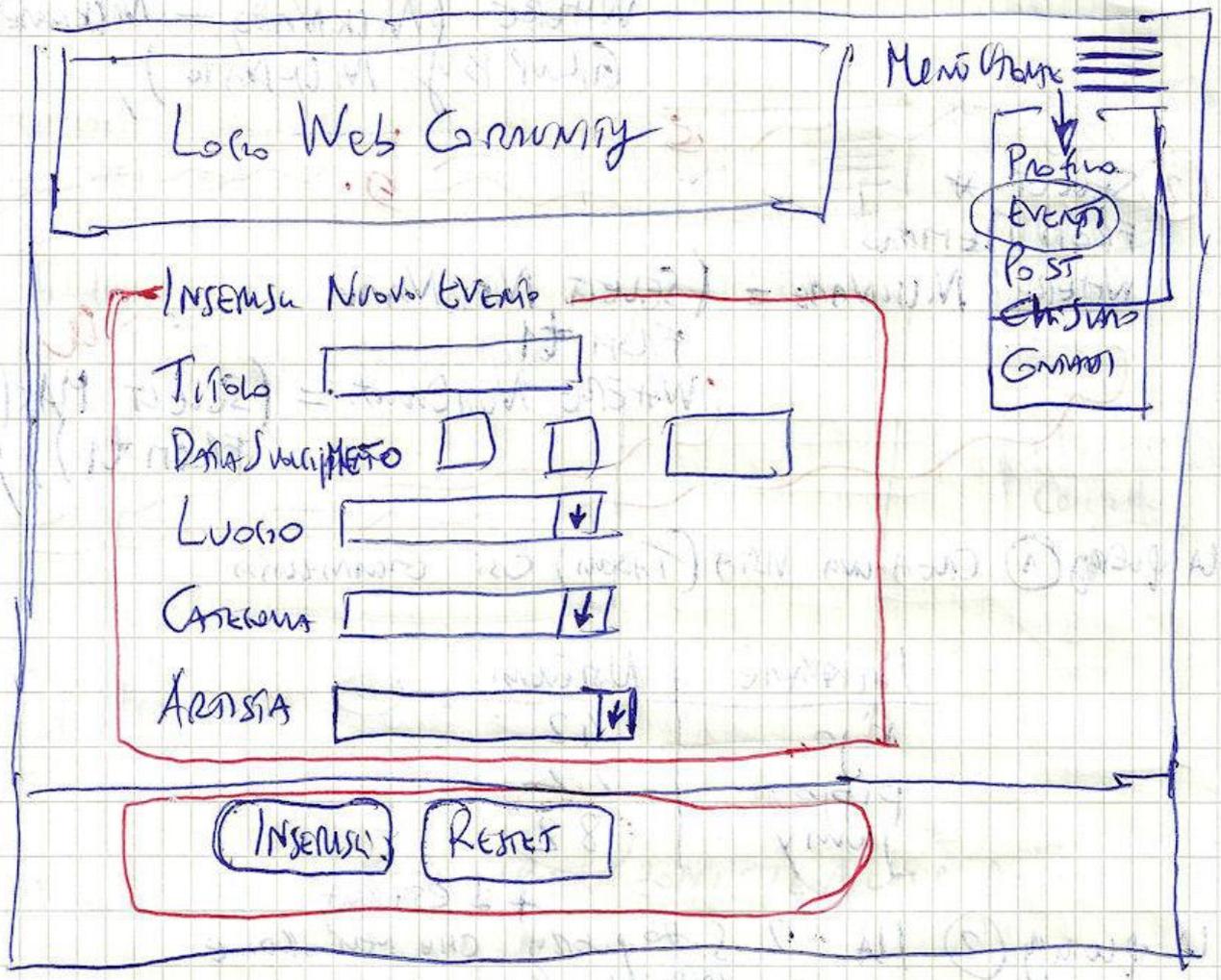
a) preleva il valore MAX della colonna NUMEVENTI di t1 (es. 82)

b) preleva il NICKNAME della tabella t1 la cui colonna NUMEVENTI ha valore pari al MAX prelevato (es. "jamy")

c) la query complessiva si basa e ~~preleva~~ tutti i stringhe della tabella MEMBR, ma solo di quelle "jamy"

6) INTERFACCIA WEB

PIRENEJA: L'UTENTE REGISTRATO (MEMBER) È STATO RICONOSCIUTO ED HA AVUTO ACCESSO ALL'INTERFACCIA WEB



QUINDI SONO NOTI il suo NickNAME, il suo Nome, il suo Cognome e le sue mail

Possono essere inseriti I DATI NEL DATABASE

7

Codifica un campo A scelta

11
8
(I)

PENSATO AL FORM DI REGISTRAZIONE DI UN MEMBRO

1

REGISTRAZIONE. HTML

Registrazione Membro

Nickname

Cognome Nome

Prima Seconda Terza

Indirizzo

Città Cap

Email

form HTML con metodo POST delle variabili

Utente clicca qui

2

REGISTRAZIONE. PHP

Vengono passati i dati del form al file

`$connessione =`

```
(1) mysql_connect ("host", "user", "password");
```

```
(2) mysql_select_db ("WebCommunity");
```

```
(3) $query = "INSERT INTO Membro
VALUES ('$nickname', '$cognome', '$nome', '$dataN',
'$indirizzo', '$citta', '$cap', '$email');"
```

```
(4) $risultato = mysql_query ($query);
```

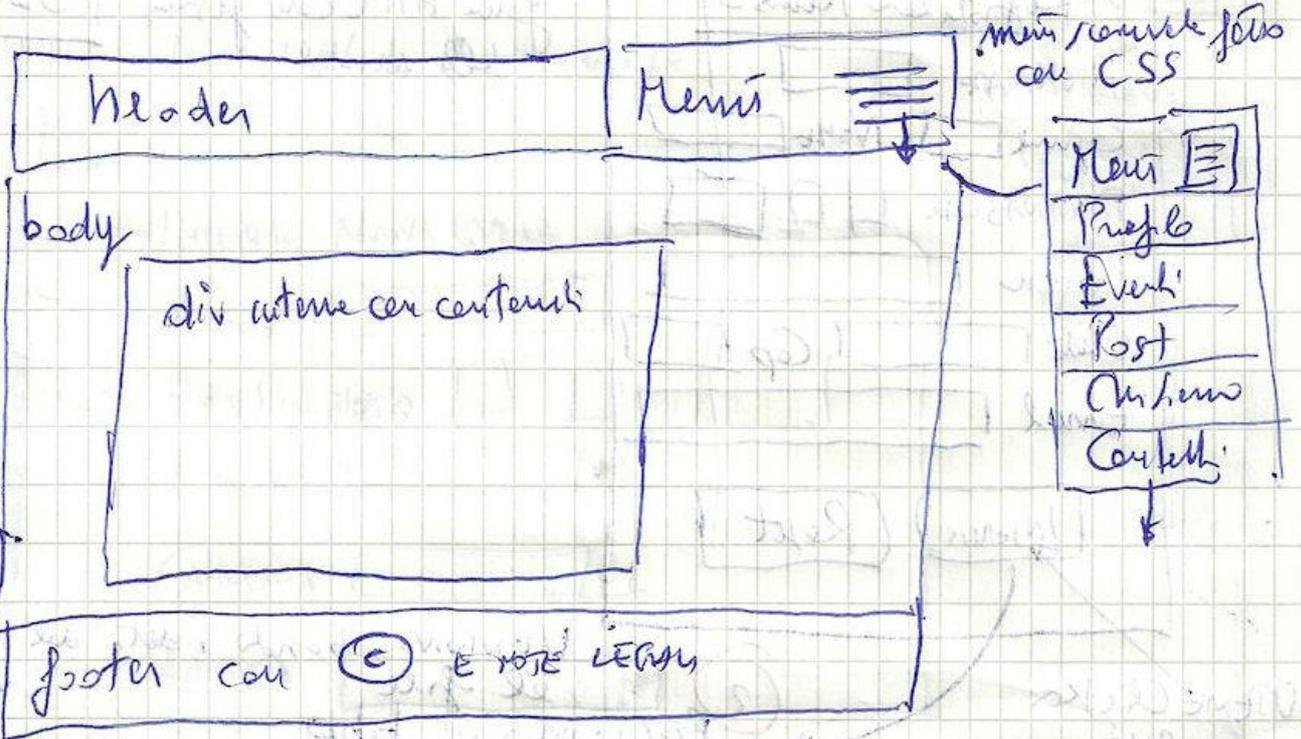
```
(5) $mysql_close ($connessione)
```

SECONDA PARTE

12

(I) SI VEDA L'ASSICURAZIONE "SIPRIENSIS" TRA LE ENTRA "INTELLIGENTE PUBBLICITÀ" e "CATEGORIA" DI PUBBLICITÀ N: N

(II) LAYOUT BASE DI OGNI PAGINA



III e IV CONTE DA SPIEGAZIONI TECNICHE per allegato

DEF: Una **forma normale** è una proprietà di uno *schema relazionale* che ne garantisce la qualità misurata in assenza di determinati difetti.

DEF: La **normalizzazione** è un procedimento che serve a trasformare uno schema che presenta **anomalie** (*schema non normalizzato*) in uno **equivalente** (con lo stesso contenuto informativo) in cui tali anomalie sono state eliminate (*schema normalizzato*).

Quando uno schema relazionale non è normalizzato può comportare **comportamenti non desiderati** che possono compromettere le operazioni di congruenza durante le operazioni di:

- *inserimento* dei dati;
- *aggiornamento* dei dati;
- *cancellazione* dei dati;

Vediamo quali anomalie si possono avere considerando la traccia fornita.

Consideriamo la seguente tabella

<u>Cognome</u>	<u>Nome</u>	<u>Telefono</u>	<u>Livello</u>	<u>Tutor</u>	<u>Tel-Tutor</u>	<u>AnticipoVersato</u>
Verdi	Luisa	345698741	Avanzato	Bianca	334563215	100
Neri	Enrico	348523698	Avanzato	Carlo	369852147	150
Rosi	Rosa	347532159	Base	Alessio	333214569	120
Bianchi	Paolo	341236547	Base	Carlo	369852147	150
Rossi	Mario	349567890	Base	Carlo	369852147	90
Neri	Enrico	348523698	Complementi	Dina	373564987	100

Una possibile chiave primaria della relazione è rappresentata dall'insieme di attributi (Cognome, Nome, Tutor)

a) anomalie di inserimento:

- non è possibile inserire un nuovo *corsista* senza inserire i dati relativi al tutor ed ai dettagli del corso svolto
- non è possibile inserire un nuovo *tutor* senza inserire i dati relativi al corsista ed ai dettagli del corso svolto

b) anomalie di aggiornamento:

- per modificare il telefono di un *corsista* occorre modificare tutte le ennuple in cui compare
 - per modificare il telefono del *tutor* occorre modificare tutte le ennuple in cui compare
- Se le modifiche fossero parziali si lascerebbe la base dei dati in uno stato detto **inconsistente**

c) anomalie di cancellazione:

- cancellando la ennupla (“Rosi”, “Rosa”, “Alessio”) si perdono tutte le informazioni relative al tutor
- cancellando la ennupla (“Verdi”, “Luisa”, “Bianca”) si perdono tutte le informazioni relative al corsista

Queste anomalie si verificano perché abbiamo rappresentato informazioni eterogenee tra loro con un'unica relazione.

Abbiamo infatti raggruppato in un'unica relazione informazioni relative ad:

- i **tutor** che effettuano un corso;
- i **corsisti** che vi partecipano;
- i **corsi** svolti dai tutor ai corsisti.

Il **processo di normalizzazione** elimina tali **anomalie** effettuando una serie di trasformazioni successive delle relazioni di partenza di uno schema relazionale ottenendo altre relazioni che a seconda del tipo di trasformazione applicata, possono rispondere a diversi livelli di "bontà" dette **forme normali**.

Esistono molte forme normali per uno schema relazionale:

- prima forma normale o **1FN**;
- seconda forma normale o **2FN**;
- terza forma normale o **3FN** con definizione alternativa di di Boyce-Codd o **BCFN**;
- quarta forma normale o **4FN**;
- quinta forma normale o **5FN**.

Per i nostri scopi è sufficiente applicare il processo di normalizzazione per ottenere uno schema relazionale in **terza forma normale o 3FN** e vedremo come ottenerlo attraverso lo studio delle dipendenze funzionali.

PRIMA FORMA NORMALE o 1FN (o forma ATOMICA)

DEF: Diremo che una relazione **R** è in **prima forma normale o 1FN** quando rispetta i **requisiti fondamentali** del modello relazionale che sono:

- i **valori di un attributo** (di una colonna) sono dello stesso tipo ovvero appartengono allo stesso dominio;
- i **valori di una ennupla** (di una riga) sono diversi da quelli delle altre ennuple ovvero non possono esistere due ennuple uguali;
- l'**ordine delle ennuple** è irrilevante;
- gli **attributi** sono di tipo **elementare** ossia:
 - non possono essere ulteriormente *scomposti* in attributi più semplici (no attributi multipli);
 - non possono essere *composti* da gruppi di attributi ripetuti (no attributi composti o aggregati).

Dipendenze funzionali

DEF: Data una relazione **R** ed un insieme $X = \{ X_1, X_2, \dots, X_N \}$ di **R** si dice che un attributo **Y** di **R** **dipende funzionalmente da X** e si scrive:

$$X_1, X_2, \dots, X_N \rightarrow Y$$

se e solo se i valori degli attributi di **X** determinano univocamente il valore dell'attributo **Y** per ogni istanza della relazione **R**.

Si dice anche che **X determina Y**.

Nell'esempio precedente possiamo individuare le seguenti dipendenze funzionali:

- 1) **Cognome, Nome → Telefono**: Il telefono dipende funzionalmente da quel determinato corsista
- 2) **Tutor → Tel-Tutor**: L'attributo Tel-Tutor dipende funzionalmente da quel determinato tutor
- 3) **Cognome, Nome, Tutor → Livello, Anticipo Versato**: il livello e l'anticipo versato da un corsista per un corso dipendono funzionalmente dal cliente e dal tutor

SECONDA FORMA NORMALE o 2FN

DEF: Diremo che una relazione **R** è in **seconda forma normale** o **2FN** se non esistono attributi dipendenti solo da una parte della chiave ossia *dipendenti parzialmente* dalla chiave.

Questa forma normale richiede che tutti gli attributi della relazione siano **omogenei** nel senso che devono essere tutte proprietà associate direttamente alla chiave.

Per trasformare una relazione in 2FN si procede **decomponendola** sulla base delle dipendenze funzionali al fine di separare le proprietà eterogenee.

Le relazioni in 2FN possono ancora essere esposte ad anomalie in quanto possono presentare delle ridondanze

TERZA FORMA NORMALE o 3FN

DEF: Diremo che una relazione **R** è in **terza forma normale** o **3FN** se per ogni possibile chiave di R accade che:

- **R è in 2FN** ossia non esistono attributi non chiave che dipendono solo da una parte della chiave;
- non esistono attributi non chiave che dipendono **transitivamente** dalla chiave ossia non esistono attributi non chiave che dipendono da altri attributi non chiave.

Per trasformare una relazione in **3FN** si crea una nuova relazione per ogni gruppo di *attributi non chiave* coinvolti nella dipendenza funzionale *con attributi non chiave*.

Nel nostro caso formeremo le tre tabelle *tutor*, *corsista* e *corso* e lo schema complessivo ottenuto sarà in 3Fⁿ

Tabella Tutor	
Tutor	Tel-Tutor
Bianca	334563215
Carlo	369852147
Alessio	333214569
Dina	373564987

Tabella Corsista		
Cognome	Nome	Telefono
Verdi	Luisa	345698741
Neri	Enrico	348523698
Rosi	Rosa	347532159
Bianchi	Paolo	341236547
Rossi	Mario	349567890

Tabella Corso (Corsista-Tutor)				
Cognome	Nome	Tutor	Livello	AnticipoVersato
Verdi	Luisa	Bianca	Avanzato	100
Neri	Enrico	Carlo	Avanzato	150
Rosi	Rosa	Alessio	Base	120
Bianchi	Paolo	Carlo	Base	150
Rossi	Mario	Carlo	Base	90
Neri	Enrico	Dina	Complementi	100

Tipi di Associazioni

Definizione: La **molteplicità** di un'associazione fra le entità X ed Y indica quante istanze dell'entità Y possono trovarsi in relazione con una istanza dell'entità X e viceversa.

Definizione: La **cardinalità** di un'associazione fra le entità X ed Y indica il numero delle coppie di elementi di X e di Y che in un certo istante sono collegate dall'associazione.

Un'associazione binaria tra due entità A e B può avere molteplicità:

a) **uno a uno** (ossia **1 : 1**) quando ad un'istanza dell'entità A può/deve corrispondere nessuna/una istanza dell'entità B e viceversa quando ad un'istanza dell'entità B può/deve corrispondere nessuna/una istanza dell'entità A

In modo equivalente si può dire che un'associazione binaria è **uno a uno** quando sia l'associazione diretta sia quella inversa sono totali ed univoche

Graficamente tale tipo di associazione viene rappresentata così (in caso di associazione diretta ed inversa entrambe parziali):



b) **uno a molti** o **semplice** (ossia **1 : N**) quando ad un'istanza dell'entità A può/deve corrispondere nessuna/una o più istanze dell'entità B ed ad ogni istanza dell'entità B può/deve corrispondere nessuna o una sola istanza dell'entità A (la scelta di A o B come entità di partenza essendo del tutto casuale).

In modo equivalente si può dire che un'associazione binaria è **uno a molti** se è multipla l'associazione diretta o l'associazione inversa ma non entrambe.

Graficamente tale tipo di associazione viene rappresentata così (in caso di associazione diretta ed inversa entrambe parziali):

:



N.B. Per un'associazione di molteplicità N:1 il discorso è l'inverso di quello fatto nel caso di molteplicità 1:N

c) **molti a molti** o **complessa** (ossia **N : N**): quando ad un'istanza dell'entità A può/deve corrispondere nessuna/una o più istanze dell'entità B e viceversa ad un'istanza dell'entità B può/deve corrispondere nessuna/una o più istanze dell'entità A.

In modo equivalente si può dire che un'associazione binaria è **molti a molti** quando sia l'associazione diretta sia quella inversa sono totali ed multiple.

Graficamente tale tipo di associazione viene rappresentata così(in caso di associazione diretta ed inversa entrambe parziali) :

