



Ministero dell'Istruzione, dell'Università e della Ricerca
I044 – ESAME DI STATO DI ISTRUZIONE SECONDARIA SUPERIORE

Indirizzo: ITIA - INFORMATICA E TELECOMUNICAZIONI

ARTICOLAZIONE INFORMATICA

Tema di: INFORMATICA - *Tipologia C*

Il candidato (che potrà eventualmente avvalersi delle conoscenze e competenze maturate attraverso esperienze di alternanza scuola-lavoro, stage o formazione in azienda) svolga la prima parte della prova e due tra i quesiti proposti nella seconda parte.

PRIMA PARTE

Un'azienda *start-up* vuole costruire una piattaforma Web che consenta il *car pooling* tra viaggiatori sul territorio nazionale, con l'obiettivo di diffondere l'uso di una mobilità flessibile e personalizzata in termini di percorsi e costi.

Gli utenti della piattaforma possono essere di due tipi: utenti-autisti (coloro che offrono un passaggio con la propria macchina) e utenti-passeggeri (coloro che usufruiscono del passaggio).

Gli autisti devono registrarsi sul sito ed inserire i propri dati: generalità, numero e scadenza patente di guida, dati dell'automobile utilizzata, recapito telefonico, email, fotografia.

Per ogni viaggio che intendono condividere, gli autisti devono indicare città di partenza, città di destinazione, data ed ora di partenza, contributo economico richiesto ad ogni passeggero, tempi di percorrenza stimati. È responsabilità dell'autista, mano a mano che accetterà passeggeri per un certo viaggio, dichiarare chiuse le prenotazioni per quel viaggio, utilizzando un'apposita funzione sul portale.

L'utente-passeggero si deve registrare sulla piattaforma, indicando cognome e nome, documento di identità, recapito telefonico ed email. La piattaforma fornisce ai passeggeri la possibilità di indicare città di partenza e di destinazione e data desiderata; presenta quindi un elenco di viaggi (per cui non siano ancora chiuse le prenotazioni), ciascuno con le caratteristiche dell'autista e le modalità del viaggio stesso inserite dall'autista (orario, eventuali soste previste alle stazioni di servizio, possibilità di caricare bagaglio o animali, ...).

Il passeggero sceglie quindi il viaggio desiderato con il corrispondente autista, anche esaminando il voto medio e i giudizi dei *feedback* assegnati tramite la piattaforma dai precedenti passeggeri all'autista stesso, e si prenota. Le informazioni sul passeggero vengono inviate per email dalla piattaforma all'autista scelto, il quale può consultare sul portale il voto medio e i giudizi dei *feedback* ricevuti dal passeggero da parte di precedenti autisti e decidere se accettarlo o meno. Il passeggero di conseguenza riceverà una email di accettazione o di rifiuto della prenotazione effettuata, contenente, in caso di accettazione, un promemoria con città di partenza e destinazione, data e orario del viaggio, dati dell'autista e della sua automobile.

A viaggio effettuato, il passeggero può inserire un *feedback* sull'autista, espresso sia in forma di voto numerico che di giudizio discorsivo. A sua volta, l'autista può inserire un *feedback* sul passeggero, espresso sia in forma di voto numerico che di giudizio discorsivo. Sia i voti medi che i singoli giudizi dei *feedback* ricevuti da ciascun autista sono disponibili ai passeggeri; analogamente, sia i voti medi che i singoli giudizi dei *feedback* ricevuti da ciascun passeggero sono disponibili agli autisti.



Ministero dell'Istruzione, dell'Università e della Ricerca
I044 – ESAME DI STATO DI ISTRUZIONE SECONDARIA SUPERIORE

Indirizzo: ITIA - INFORMATICA E TELECOMUNICAZIONI

ARTICOLAZIONE INFORMATICA

Tema di: INFORMATICA - *Tipologia C*

Il candidato, fatte le opportune ipotesi aggiuntive, sviluppi:

1. un'analisi della realtà di riferimento, giungendo alla definizione di uno schema concettuale della base di dati che, a suo motivato giudizio, sia idoneo a gestire la realtà presentata;
2. il relativo schema logico;
3. le seguenti interrogazioni espresse in linguaggio SQL:
 - a) data una città di partenza, una di arrivo e una data, elencare gli autisti che propongono un viaggio corrispondente con prenotazioni non ancora chiuse, in ordine crescente di orario, riportando i dati dell'auto e il contributo economico richiesto;
 - b) dato il codice di una prenotazione accettata, estrarre i dati necessari per predisporre l'email di promemoria da inviare all'utente passeggero;
 - c) dato un certo viaggio, consentire all'autista di valutare le caratteristiche dei passeggeri visualizzando l'elenco di coloro che lo hanno prenotato, con il voto medio dei feedback ricevuti da ciascun passeggero, presentando solo i passeggeri che hanno voto medio superiore ad un valore indicato dall'autista;
4. il progetto di massima della struttura funzionale dell'applicazione Web, realizzando, con appropriati linguaggi a scelta sia lato client che lato server, un segmento significativo dell'applicazione che consente l'interazione con la base di dati.

SECONDA PARTE

- I. In relazione al tema proposto nella prima parte, il candidato integri il modello già realizzato al fine di gestire in automatico il numero di posti disponibili nei vari viaggi, evitando che sia responsabilità dell'autista dichiarare chiuse le prenotazioni sul portale. Nel momento in cui inserisce un viaggio, l'autista dichiara il numero massimo di posti disponibili. Mano a mano che gli autisti accettano le prenotazioni, il sistema visualizzerà solo i viaggi con posti ancora disponibili: a tal fine, una prenotazione non ancora accettata dall'autista non comporta alcun impegno del posto, che resta così ancora disponibile per prenotazioni di altri passeggeri. Per ciascun viaggio, la piattaforma mostrerà il numero dei posti disponibili e il numero delle prenotazioni non ancora accettate. Il candidato sviluppi poi la pagina web, sia lato client che lato server, per fornire ai passeggeri tali informazioni.



Ministero dell'Istruzione, dell'Università e della Ricerca
I044 – ESAME DI STATO DI ISTRUZIONE SECONDARIA SUPERIORE

Indirizzo: ITIA - INFORMATICA E TELECOMUNICAZIONI

ARTICOLAZIONE INFORMATICA

Tema di: INFORMATICA - Tipologia C

- II. In relazione al tema proposto nella prima parte, il candidato immagini di volere documentare al committente l'operatività della piattaforma proposta. A tal fine, imposti una relazione tecnica che presenti le principali caratteristiche dell'applicazione Web in termini di organizzazione e funzionalità. In particolare, imposti la struttura di tale relazione, motivando le scelte e scrivendo un esempio significativo dei relativi contenuti.
- III. Dato il seguente schema relazionale:
- film (id, titolo, durata, anno di produzione, genere);
attore (id, nome, cognome, data_nascita, fotografia);
recita (id_film, id_attore, ruolo);
- il candidato:
- determini la modalità di gestione del campo 'fotografia' che prevede la memorizzazione di una immagine dell'attore in un formato grafico (es. JPG);
 - formalizzi in linguaggio SQL lo schema fisico corrispondente allo schema relazionale, sapendo che:
 - a. il campo 'genere' ammette solo i seguenti valori: fantasy, giallo, commedia, horror, drammatico, fantascienza, azione;
 - b. per la relazione 'recita', i campi 'id_film' e 'id_attore' referenziano rispettivamente la chiave primaria delle relazioni 'film' e 'attore';
 - discuta l'uso degli indici nel modello fisico di una base di dati e suggerisca con motivato giudizio indici appropriati per questo schema relazionale, definendoli in linguaggio SQL.
- IV. Un'azienda desidera sviluppare un'applicazione Web per la prenotazione on-line di eventi culturali, fruibile sia da computer desktop che da dispositivi mobili come tablet e smartphone. Il candidato esponga i punti critici da affrontare relativamente alle differenti proprietà di visualizzazione delle varie tipologie di dispositivi e alla rispettiva fruizione dei contenuti. Illustri possibili misure risolutive, con esempi relativi all'applicazione in questione.

Durata massima della prova: 6 ore.

È consentito l'uso di manuali tecnici e di calcolatrici tascabili non programmabili.

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.

① PRIMA PARTE - PREMESSA

PER POTER SODDISFARE QUESTE RICHIESTE L'APPLICAZIONE DA REALIZZARE È UNA WEB APPLICATION CHE PERMETTA AGLI UTENTI DI VISIONARE I VIAGGI PENSABILI E DI SEGUIRNE BASANDOSI SU FEEDBACK DEI VARI UTENTI, TRAPIC POSSIBILI.

L'APPLICAZIONE DOVrà ESSERE ACCESSIBILE DA VARI DISPOSITIVI (PC, TABLET, SMARTPHONE) PER QUESTIONI DI FACILE ACCESSIBILITÀ NEL CASO AD ESEMPIO NON SIA DISPONIBILE UN PC QUANDO DURANTE IL VIAGGIO CI SI È QUANTOPIÙ DA CASA.

PER PRIMO RIGUARDA GLI STRUMENTI SW DA UTILIZZARE SI PUÒ CONSIDERARE UNA CLASSICA CONFIGURAZIONE COMPLESSA DA:

- MySQL COME DBMS RELAZIONALE PER LA GESTIONE DEL DB
 - Apache COME WEB SERVER
 - PHP COME LINGUAGGIO DI SCRIPTING lato SERVER PER L'INTERAZIONE CON IL DB
 - HTML5, CSS e JAVASCRIPT PER IMPLEMENTARE L'INTERFACCIA WEB (FRAMMENTI BOOTSTRAP)
- VEDI RIEPIGLO SECONDA PARTE - IV

REGOLE DI LOGICA

- (*) UN PASSEGGERO PUO' PRENDERE NESSUNO O PIU' VIAGGI E VIVERE IN VIAGGI PUO' ESSERE PRENDUTO DA NESSUNO O PIU' PASSEGGERI
- (*) UN ARTISTA PUO' EFFETTUARE NESSUNO O PIU' VIAGGI E VIVERE IN UN VIAGGIO DEVE ESSERE EFFETTUATO DA UN UNO ARTISTA
- (*) UN PASSEGGERO PUO' INSERIRE NESSUNO O PIU' FEEDBACK E VIVERE IN UN FEEDBACK PUO' ESSERE INSERITO DA NESSUNO O UN PASSEGGERO
- (*) UN ARTISTA PUO' INSERIRE NESSUNO O PIU' FEEDBACK E VIVERE IN UN FEEDBACK PUO' ESSERE INSERITO DA NESSUNO O UN ARTISTA

VIAGGI DI INTEGRAZIONE

- IMPLICAZIONI : DOWTA CHINESE PRIMA

- AFFRANTO "COST" E PK SUI "ENTRATA" "PASSEGGERO"
- AFFRANTO "COST" E PK SUI "ENTRATA" "VIAGGI"
- AFFRANTO "COST" E PK SUI "ENTRATA" "ARTISTA"
- AFFRANTO "COST" E PK SUI "ENTRATA" "FEEDBACK"

- DOWTA CHINESE DUE ASSOCIAZIONI (ARTISTI / VIAGGI)

- DOWTA ASSOCIAZIONE INVERSA "EFFETTUATO TRA LE KINIS" "VIAGGI" "ARTISTA"

- ESPRESSIONI :

V1: (VIAGGI, PRENDITORI IN ("SI", "NO"))

SI PRENDITORI POSSIBILI
NO PRENDITORI CHINA

V2: (VIAGGI, NPOSTI ≥ 0)

V3: (VIAGGI, COSTO > 0)

V4: (VIAGGI, TEMPO > 0)

V5: (VIAGGI, SOSTA > 0)

V6: (VIAGGI, BACCHI IN ("SI", "NO"))

V7: (VIAGGI, ANNI IN ("SI", "NO"))

- NUMERI DI PREVISIONE

SI PAGAMENTO ANNI
NO PAGAMENTO NON ANNI
SI ANNI ANNI A BORDO
NO ANNI NON ANNI A BORDO

V8: (PRENDITORI, ACCETTAZIONE IN ("SI", "NO"))

SI PRENDITORI ACCETTAZIONE
NO PRENDITORI NON ACCETTAZIONE

V9: (FEEDBACK, VOTO BASTA 1 A 10)

V10: (FEEDBACK, TIPO DI IN ("ARTISTA", "PASSEGGERO"))

DETERMINAZIONE DEL FEEDBACK

V11: (SE (TIPO DI = "ARTISTA")

ANON

COSTO COSTO IN COSTO ANNI VOTO PERMANENTE DEL FEEDBACK

ANON

FINALE) COSTO COSTO IN COSTO PASSEGGERO VOTO PERMANENTE DEL FEEDBACK

② Schema Logico Relazionale

a) MAPPING RELAZIONE ASSOCIAZIONE BINARIA "PRENOTA" TRA LE ENTITÀ "PASSEGGERO" E "VIAGGIO" DI TIPOLOGIA "1:N"

PASSEGERO (CodP, Cognome, Nome, DataN, Indirizzo, Città, Cap, Documento, Telefono, Email, User, Pswd)

VIAGGIO (CodV, Prenotabile, N.Gest, CanP, CanA, DataP, DataV, Corso, TerrP, Seta, Bagagli, Amani, CodA)

- con l'attributo "CodA" che risulta FK sull'attributo "CodA" della relazione "PRENOTA"

PRENOTA - (CodP, CodV, Accensione)

- con l'attributo "CodP" che risulta FK sull'attributo "CodP" della relazione "PASSEGGERO"

- con l'attributo "CodV" che risulta FK sull'attributo "CodV" della relazione "VIAGGIO"

$\{VR_{CodP(PRENOTA)} \subseteq VR_{CodP(PASSEGERO)} \quad VR_{DataP(PASSEGERO)} \subseteq VR_{DataP(VIAGGIO)}$
 $\{VR_{CodV(PRENOTA)} \subseteq VR_{CodV(VIAGGIO)} \quad VR_{DataV(VIAGGIO)} \subseteq VR_{DataV(PASSEGERO)}$

b) MAPPING RELAZIONE ASSOCIAZIONE BINARIA "EFFETTUA" TRA LE ENTITÀ "AUSITA" E "VIAGGIO" DI TIPOLOGIA "1:N"

AUSITA (CodA, Cognome, Nome, DataN, Indirizzo, Città, Cap, Telefono, NumP, User, Pswd, DataP, Mura, Perno, CuiMura, Partenza, Moneta, Effetto)

VIAGGIO: RELAZIONE TRIPPA IN ESISTENZA

c) MAPPING RELAZIONE ASSOCIAZIONE BINARIA "INVIATO" TRA LE ENTITÀ "PASSEGGERO" E "FEEDBACK" DI TIPOLOGIA "1:N"

PASSEGERO: RELAZIONE CON TRIPPA IN ESISTENZA

FEEDBACK (CodF, Voto, Giorno, TipPerc, CodDest, DataInvi, CodP, CodA)

- con l'attributo "CodP" che risulta FK sull'attributo "CodP" della relazione "PASSEGGERO"

- con l'attributo "CodA" che risulta FK sull'attributo "CodA" della relazione "AUSITA"

d) MAPPING RELAZIONE ASSOCIAZIONE BINARIA "INVIATO" TRA LE ENTITÀ "AUSITA" E "FEEDBACK" DI TIPOLOGIA "1:N"

AUSITA: RELAZIONE CON TRIPPA IN ESISTENZA

FEEDBACK: RELAZIONE CON TRIPPA IN ESISTENZA

② MAPPING FUNZIONANTE DEI VINCOLI DI INTEGRAZIONE

VINCOLI DI INTEGRAZIONE IMPLICATI
DALLA PK



VINCOLI DI INTEGRAZIONE INTRARELAZIONALI O INTER-
SU PIÙ N-PLE

VINCOLI DI INTEGRAZIONE IMPLICATI
DALLA NON FUNZIONANTE
ASSUNZIONI DFE/INV



VINCOLI DI INTEGRAZIONE INTERRELAZIONALI O
ESTERNI REFERENZIALI

V1(...)	⇒	V1(VIAGGIO): (Acquazione IN("SI", "NO"))
V2(...)	⇒	V2(VIAGGIO): ($N_{PSS} \geq \phi$)
V3(...)	⇒	V3(VIAGGIO): ($C_{SS} > \phi$)
V4(...)	⇒	V4(VIAGGIO): ($T_{ETRO} > \phi$)
V5(...)	⇒	V5(VIAGGIO): ($S_{SSA} > \phi$)
V6(...)	⇒	V6(VIAGGIO): (BAGNO IN("SI", "NO"))
V7(...)	⇒	V7(VIAGGIO): (AMMAGLI IN("SI", "NO"))

VINCOLI DI INTEGRAZIONE INTRARELAZIONALI O INTER-
SU SIMILI N-PLE SU SIMILI AGGIUNTI

V8(...)	⇒	V8(PROMIA): (Acquazione IN("SI", "NO"))
---------	---	---

VINCOLI DI INTEGRAZIONE INTRARELAZIONALI O INTER-
SU SIMILI N-PLE SUL DING DI UN AGGIUNTO

V9(...)	⇒	V9(FEDERAZIONE): (VIA BETWEEN 1 AND 17)
V10(...)	⇒	V10(FEDERAZIONE): (TIPUS IN("AMMAGLI", "INCHIESTA"))

VINCOLI DI INTEGRAZIONE INTRARELAZIONALI O INTER-
SU SIMILI N-PLE SUL DING DI UN AGGIUNTO

V11(...) NB: STRADA DI UN'ORA DA REGIMATO VIA SOFTWARE

IN AGLIATI V12(FEDERAZIONE): (SE (CODA < NULL)

AND

CODA = NULL

DATA12 = NULL

DATA11 < NULL

FINE SE

NB: V13(FEDERAZIONE): ANALOGO PER CODA < NULL

③ Query Expression Language SQL

① SELECT Cognome, Nome, Marca, Perno, Cuneo, Csto
FROM Anzisa, Viaggio
WHERE (Anzisa.GoA = Viaggio.GoA) AND (Cuneo = [X])
AND (Cuneo = [Y]) AND (DataP = [Z])
AND (Perno = "SI")
ORDER BY DataP ASC;

② SELECT Cuneo, GoA, DataP, Perno, Anzisa.Cognome, Anzisa.Nome,
Marca, Perno, Cuneo
FROM ^{Perno}Passenger, Viaggio, Anzisa
WHERE (Passenger.GoP = Perno.GoP) AND (Perno.GoV = Viaggio.GoV)
AND (Viaggio.GoA = Anzisa.GoA)
AND (Perno.GoP = [X]) AND (Perno.GoV = [Y]) AND
(Accurato = "SI")

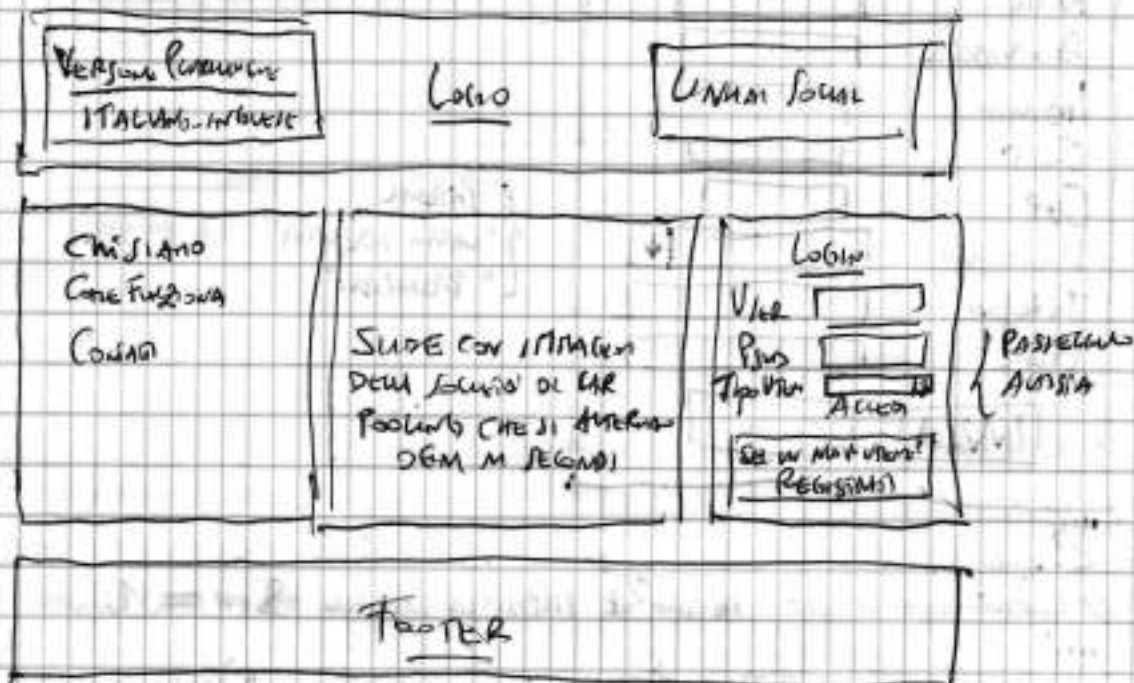
③ SELECT Passenger.GoP, AVG(Voto) AS MeanVoto
FROM Passenger, Perno, Viaggio, Feedback
WHERE (Passenger.GoP = Perno.GoP) AND (Perno.GoV = Viaggio.GoV)
AND (Feedback.GoP = Passenger.GoP)
AND (Viaggio.GoV = [X])
GROUP BY Passenger.GoP
HAVING MeanVoto > [Y];

4

IN SINTESI

A

partec. php



a) SE CILSI LOGIN COME "PASSEGGERO" SI ACCDE ALLA HOME PAGE UTENTE COME SEGUENTI FUNZIONALITA' (PRINCIPALI)

- PRENOTA VIAGGIO
- CANCELLA PRENOTAZIONE VIAGGIO
- INTERNA FEEDBACK IN ANZISA

SE CILSI LOGIN COME "ANZISA" SI ACCDE ALLA HOME PAGE UTENTE COME SEGUENTI FUNZIONALITA' (PRINCIPALI)

- INTERNA VIAGGIO
- CANCELLA VIAGGIO
- INTERNA FEEDBACK IN PASSEGGERO
- CANCELLA PRENOTAZIONE VIAGGIO (SE NON AUTORIZZATA)

b) CILSI IL "REGISTRATI" SI ACCDE IN BASE AL VALORE DELLA LIST BOX "TIPO UTENTE" AD UN FORM DI REGISTRAZIONE RELATIVO.

ESEMPIO: IMMAGINANDO DI VOLETE REGISTRARVI NELLA PASSEGGERO SI PRESENTA IL "TIPO UTENTE" = "PASSEGGERO"

8

registrazione.php

NB

Mostrare il KORT GIUSTO
A SECONDA DEL VALORE IMPRESO
NELLO HONOR DEL REGISTRO
NEL CANTO "TIPO UTENTE"

PAG. 8

Cognome	
Nome	
Data NASCITA	
Indirizzo	
Città	
Cap	
Documento	↓
Telefono	
Email	
Invia Annulla	

"PATRONE"
 "CARTA IDENTITÀ"
 "PASSAPORTO"

```

<?php
// Controlla che il tipo di utente sia corretto (impone la variabile $ell = 1;

```

```

if ($ell == 0) && ($tipo_utente == "Passaporto")
{

```

```

echo "
<form id='form1' name='form1' method='POST' action='

```

```

Cognome: <input name='Cognome' type='text' tabindex='1'
Size='50' maxlength='50' />

```

```

Documento: <select name='documento'>
<option> Patrone </option>
<option> Carta Identità </option>
<option> Passaporto </option>
</select>

```

```

<input name='invia' type='submit' value='Invia' />
<input name='annulla' type='reset' value='Annulla' />

```

```

<input type='hidden' name='user' value='$user' />
<input type='hidden' name='pswd' value='$pswd' />

```

3

```

else if ($err == 0) && ($password == "Aurora")
{
    // Gestione Form per la selezione "Aurora"
    ....
}
else
{
    // Gestione Errori
    ....
}
?>

```

reg passeggero.php (C) // file x Registrazione sulla base dati DEL PASSEGGERO

```

<?php
// Connessione alla base dati
....
if ($err == 0)
{
    $link = mysqli_connect ($host, $user, $password, $dbname)
    or die ("Connessione fallita" . mysqli_connect_errno() . " " .
    mysqli_connect_error());

    $sql = "INSERT INTO passeggero VALUES (.....)";
    // NB Con tutti i valori recuperati dal form precedente

```

```

    $result = mysqli_query ($link, $sql) or die ("Insert fallito" .
    mysqli_errno ($link) . " " . mysqli_error ($link));
    // NB in caso di errore opportuna segnalazione

```

```

    mysqli_close ($link) or die ("Chiusura fallita" . mysqli_errno ($link)
    . " " . mysqli_error ($link));
?>

```


I) PER LA GESTIONE AUTOMATICA DEL NUMERO MASSIMO DI PASTECCHI È SUFFICIENTE AGGIUNGERE (COME È STATO FATTO) ALL'ENTRATA "VIAGGI" UN ATTRIBUTO DENOMINATO "NPST".

L'IMPLEMENTAZIONE DEL CODICE USO SERVER (VIA PHP) TRAMITE APPROPRIATE QUERY DEL TIPO UPDATE/SET/WHERE SI OCCUPERÀ OGN'VOLTA CHE UN PASTECCHIO È ACCESATO DA UNA AZIONE DI DETERMINARE IL VALORE DI TALE ATTRIBUTO E DI VERIFICARE TRAMITE INTERROGAZIONE SE IL GIURINAME È ARRIVATO A UNA DETERMINATA ARRETRATO A IL CAMPO "PASTECCHI" VERBA VALORIZZATO AUTOMATICAMENTE A "NO".

II) UNA POSIZIONE STRUTTURALE DELLA PRESSIONE TECNICA CONTRO I SEGUENTI PARAGRAFI

PAG 11

a) STUDIO DI FATTIBILITÀ E RACCOLTA DEI REQUISITI

- VALUTAZIONE COSTI E BENEFICI
- PIANIFICAZIONE DELLE ATTIVITÀ E DELLE RISORSE DEL PROGETTO
- RACCOLTA DEI REQUISITI CON PARTICOLARE ATTENZIONE AQUE ESPRESSIONI DEGLI UTENTI

b) ANALISI

- DEFINIZIONE DEL DOMINIO DELL'APPLICAZIONE ESPRESSA DALLE FUNZIONI DELLE VARIE COMPONENTI: LO SCHEMA CONCETTUALE

c) PROGETTO E REALIZZAZIONE

- DEFINIZIONE DELL'ARCHITETTURA DEL PROGRAMMA: CLIENT E SERVER
- RIFERIMENTO AL CODICE SORGENTE DEL PROGRAMMA ED ALLA DOCUMENTAZIONE RELATIVA

d) VERIFICA

- ELENCO DEI TEST DI VERIFICA EFFETTUATI PER GARANTIRE IL SISTEMA DEL PROGRAMMA

e) MANUTENZIONE

- PIANIFICAZIONE DEGLI INTERVENTI E DELLE MODALITÀ DI CORREZIONE ED AGGIORNAMENTO DEL PROGRAMMA

NB PER UNA STESSA DI QUESTA PARTE DEL CORSO SI CONSIDERA LO SCHEMA REALIZZATO IN PRECEDENZA NON PURA PARTE MA LA DEFINIZIONE FORMALE MESSA IN EVIDENZA NEL TESTO DELL'ESAME

III) Film (id, titolo, durata, descrizione, prezzo)
 attore (id, nome, cognome, data nascita, fotografia)
 regista (id_film, id_attore, ruolo)

SEBENSÌ LE FOTO SIANO SALVATE IN UN DB, NON È UN PROBLEMA PER IL CAMPO ^{PIÙ} fotografia CONTENERE LE INFORMAZIONI NECESSARIE PER INDIVIDUARE IL FILE MULTIMEDIALE (IN QUESTO CASO UN'IMMAGINE) SUL SERVER

— **Funzione lettura**: IL CAMPO fotografia CONTIENE IL PATH COMPATTO DEL NOME DEL FILE E DELL'ESTENSIONE

— **Funzione scrittura**: L'ATTORO FORNIRÀ PER OGNI CONSIDERATO CHE AVREMO GIBBI DI CAMPI

Path - foto
 nome - foto
 estensione - foto

PRIMA INIZIARE LE IMMAGINI SI NECESSITA' AVERE DIRECTIONE DEL FILE SYSTEM E NEL DATABASE SI NECESSITA' AVERE I REFERENZIALI IMMAGINE (CODE) IL PERCORSO ED IL NOME FILE

INIZIALIZZAZIONE

- a) LOGGARESI AL DBMS COME DBA
 mysql -u root -p root;
- b) CREARE IL NUOVO UTENTE CHE GESTIRÀ IL NUOVO DATABASE
 CREATE USER pippo@localhost IDENTIFIED BY "pippo";
- c) CREARE IL DATABASE "cinema"
 CREATE DATABASE cinema;
- d) ASSEGNARE AL NUOVO UTENTE TUTTI I DIRITTI ED I PRIVILEGI NECESSARI SUL DB
 GRANT ALL ON cinema.*
 TO pippo@localhost
 IDENTIFIED BY "pippo";
- e) USCIRE COME DBA DAL DBMS
 quit;
- f) LOGGARESI AL DBMS COME UTENTE "pippo"
 mysql -u pippo -p pippo;
- g) SELEZIONARE IL DB "cinema"
 USE cinema;

CREATE DOMAIN tip_genre AS VARCHAR(12) NOT NULL
CHECK (value IN ("fantasy", "drama", "comedy", "horror", "documentary",
"action", "thriller"));

CREATE TABLE film

(
id INT(15) NOT NULL,
title VARCHAR(255) NOT NULL,
duration TIME NOT NULL,
release_year INT(4) NOT NULL,
genre VARCHAR(12) NOT NULL,
PRIMARY KEY(id),
CHECK (genre IN ("fantasy", "drama", "comedy", "horror", "documentary",
"action", "thriller"))
);

CREATE TABLE actors

(
id INT(15) NOT NULL,
column VARCHAR(50) NOT NULL,
name VARCHAR(50) NOT NULL,
date_of_birth DATE NOT NULL,
path_photo VARCHAR(100) NOT NULL,
height VARCHAR(30) NOT NULL,
extension_photo CHAR(3) NOT NULL,
PRIMARY KEY(id),
CHECK (extension_photo IN ("JPG", "BMP", "PNG", "GIF"))
);

CREATE TABLE actor

```
(
  id_film INT(11) NOT NULL,
  id_attor INT(11) NOT NULL,
  ruolo VARCHAR NOT NULL,
  PRIMARY KEY (id_film, id_attor),
  FOREIGN KEY (id_film) REFERENCES film(id_film)
  ON UPDATE CASCADE
  ON DELETE CASCADE,
  FOREIGN KEY (id_attor) REFERENCES attori(id_attor)
  ON UPDATE CASCADE
  ON DELETE CASCADE
);
```

IN SQL è possibile leggere ogni attributo di una tabella alcuni Tabella
specifica delle INDICI.

TAN INDICI SONO FILE CONTENENTI LE CHIAVI DEI RECORD DELLE TABELLE
ANCHE QUAN SONO ASSOCIATI CHE FORNISCANO AN' SQL DI ACCEDERE IL PROGR
DI RICERCA DEI DATI ALL'INTERNO DELLA TABELLA CHE SONO IMMAGAZZINATI
ENVIANDO IL TERMO TABLE SCAN

IN PRATICA UN È MAGGIORMENTE CREATO INDICE SU SU "titolo" SU SU "genere"
DALLA TABELLA FILM E SU "cognome" e "nome" della TABELLA ATTORI

CREATE INDEX i1 ON film (titolo); // INDICE SEMPLICE
CREATE INDEX i2 ON film (genere);

CREATE INDEX i3 ON attori (cognome, nome); // INDICE COMBINATO

NP LA CLAVO UNICA VA USATA IN CASO DI VOTO CREATO UN INDICE
SU AMMONTI CHIAVI

IV) BOOTSTRAP È UN INSIEME DI ELEMENTI GRAFICI, STRUTTURE DI IMPIANTAZIONE E JAVASCRIPT PRONTI ALL'USO, NASCIUTI INSIEME A TWITTER AD OPERA DEGLI SVILUPPATORI MARK SUIO E JACOB THORPE. OGGI BOOTSTRAP È UN PROGETTO INDIPENDENTE ED È STATO PRESO A DISPOSIZIONE DEGLI SVILUPPATORI DI TUTTO IL MONDO CHE HANNO LA PERLA DI USARE QUESTO FRAMEWORK, CHE BASATO SULLA PROVA PROVA WEB.

QUESTO FRAMEWORK CI OFFRE IMPOSTAZIONI CON CUI COSTRUIRE PAGINE WEB HTML5 COMPLETAMENTE RESPONSIVE, COERENTI E FUNZIONANTI. LA SUA UTILITÀ È EVIDENTE NELLA SITUAZIONE ATTUALE IN CUI LE PAGINE WEB DI UN'APPLICAZIONE POSSONO ESSERE FRUITE SU UNA SCELTA DI DISPOSITIVI (PC, TABLET, SMARTPHONE) CON CARATTERISTICHE DIVERSE.

SIAMO QUINDI BOOTSTRAP A METTERE A DISPOSIZIONE ELEMENTI DI STILE CHE PERMETTONO ALLA PAGINA DI ADATTARSI AL DISPOSITIVO UTILIZZANDO NEL COMPLETO ELEMENTI DI INTERFACCIA GIÀ PRESENTI NEI MODERNI.

BOOTSTRAP HA ELEMENTI ASSAI IMPORTANTI ELENZIAMO A 4 AREE

- STYLING (o IMPIANTAZIONE): QUESTA AREA CONTIENE TUTTI I PICCOLI ELEMENTI CSS IN GRADO DI DEFINIRE IL LAYOUT DELLA PAGINA
- CSS BASE: CONTIENE GLI STILI PREDEFINITI PER DIVERSI ELEMENTI DELLA PAGINA (TITOLI, TABELLE, PULSANTI, ELEMENTI DI FORM, ETC.).
- COMPONENTI: CONTIENE ELEMENTI PIÙ COMPLESSI, PRIMA GRUPPI DI PULSANTI, BARRA DI NAVIGAZIONE, MENU ADIUNTI, ETC.
- JAVASCRIPT: AREA CHE CONTIENE DIVERSI PLUGIN JQUERY PER REALIZZARE EFFETTI PARTICOLARI COME TRANSIZIONI, FINESTRE MODALI, POPUP, ETC.

Dopo utilizzare Bootstrap per la nostra Web Application avremo un'Applicazione "Responsive" IN GRADO DI RICONOSCERE AUTOMATICAMENTE IL DISPOSITIVO DELL'UTENTE E DI ADATTARSI DI CONSEGUENZA A SECONDA DELLE DIMENSIONI DEL SCHERMO DEL DEVICE SU CUI SI GUARDA.