**RELAZIONE SCRITTA REALATIVA AL PROGRAMMA DI GESTIONE DI UNA CATENA DI CINEMA**

**classe V D Mercurio**

**ITE A. Bassi di Lodi**

**(Succursale via Giovanni XXIII - Lodi)**

*Galdi Simona*

Lodi 25 marzo 2013

**INDICE**

|  |  |
| --- | --- |
| Nome | Pagina |
| [Traccia del problema](#traccia) ……………………………………………… | 3 |
| [Analisi di massima](#massima)………………………………………………….. | 4 |
| [Modello E/R](#ER)…………………………………………………………… | 5 |
| [Modello relazionale](#relazionale) ………………………………………………… | 6 |
| [Spiegazione programma](#spiegazione)……...……………………………………. | 7 |
| [Concetto di integrità referenziale](#integrita)………………………………… | 7 |
| [Regole di vincolo](#vincolo) …………………………………………………… | 7 |
| [Tabella delle variabili](#variabili) ………………………………………………… | 8 |
| [Script creazione](#creazione) …….………………………………………………… | 11 |
| [Inserimento e queries](#inserimento) ……..……………………….………………… | 11 |
| [Pagine realizzate](#pagine)………………………………….…………………... | 12 |
| [Manuale d’uso (italiano)](#italiano)……………………….…………………... | 13 |
| [Manuale d’uso (Inglese)](#inglese)……………………………………………... | 14 |

**Traccia del problema**

Si vuole progettare una base di dati per gestire informazioni su Attori, Registi, Film e Proiezioni.
Attori e registi sono identificati dal nome; di essi interessa la nazionalità e l’anno di nascita.
I film sono identificati da un codice; di essi interessa il titolo, l’anno di produzione, il regista, gli attori che vi recitano e, per ogni attore, se è protagonista o non protagonista.
Di una proiezione interessa la città, la sala, la data, l’ora, il numero di spettatori.

**Analisi di massima**

Il programma permette di gestire una catena di cinema presupponendo che in ogni città sia presente un solo cinema. Nella home viene presentata la struttura e i servizi che vengono forniti e nella pagina dei contatti sono presenti il numero di telefono e l’indirizzo e-mail a disposizione dell’utente.

L’area di gestione autorizza l’accesso solo all’amministratore che, inserendo il nome utente e la password nell’apposita pagina riesce gestire (con inserimento, modifica, cancellazione) i dati degli attori, dei film, dei registi e delle proiezioni che vengono effettuate in ogni cinema. Per tutti gli utenti invece, c’è la possibilità di consultare la programmazione dei film di un cinema inserendo il nome della città in questo si trova.

**Modello E-R**

**Nome**

**Cognome**

**Nome**

Nazionalita

AnnoDiNascita

ATTORE

(1,N)

Protagonista

RECITAZIONE

**Nome**

**Cognome**

(0,N)

Titolo

AnnoProduzione

(1,N)

(1,1)

FILM

REGISTA

REGIA

**CodFilm**

(1,N)

AnnoDiNascita

Nazionalita

PROGRAMMAZIONE

(1,1)

Sala

CodProiezione

Ora

Data

Citta

PROIEZIONE

n\_spettatori

**Modello Relazionale**

ATTORI(nome, cognome, nazionalita, annodinascita)

REGISTI(nome, cognome, nazionalita, annodinascita)

FILM(codice, nomeregista, cognomeregista, titolo, annoproduzione)

FK(nomeregista, cognomeregista) REF REGISTI(nome, cognome)

RECITAZIONE(nomeattore, cognomeattore, codfilm, protagonista)

FK(nomeattore, cognomeattore) REF ATTORI(nome, cognome)

FK(codfilm) REF FILM(codice)

PROIEZIONI(codProiezione, codFilm, data, ora, sala, citta, n\_spettatori)

FK(codFilm) REF FILM(codice)

**Spiegazione del programma**

La grafica del sito è stata realizzata utilizzando un template.

Nell’area GESTIONE è possibile inserire, modificare, eliminare dati che riguardano attori, registi, film proiezioni e di inserire o eliminare il ruolo che ogni attore ha in un particolare film; però per poter accede è necessario essere amministratori, infatti occorre inserire l’username(simona) e la password(admin) nel modulo form della pagina.

In ogni tabella l’inserimento dei dati avviene compilando un modulo form che si trova nella pagina HTML(o PHP qual ora sia necessaria una visualizzazione di una tabella). I dati, attraverso il vettore $\_POST, passano nella pagina PHP e vengono sottoposti a una serie di controlli. Uno di questi prevede che l’utente abbia correttamente compilato tutti i campi: se il risultato è positivo i dati vengono registrati nel Database, altrimenti viene visualizzato un messaggio di errore. In ogni pagina php sono presenti, inoltre, una sequenza di istruzioni che permettono di effettuare la connessione al Database e una serie di controlli che verificano che la connessione sia avvenuta correttamente e che il Database selezionato sia già esistente.

Per quanto riguarda la modifica, questa è stata gestita con 3 pagine php per ogni tabella presente del Database. Nella prima pagina, utilizzando una query di visualizzazione, viene mandato a video il contenuto della tabella che si sta prendendo in considerazione e, in un modulo form vengono fatti inserire i valori di chiave primaria che identificano la tupla che si vuole modificare. I dati inseriti vengono recuperati nella seconda pagina php attraverso il vettore $\_POST e vengono effettuati i controlli sui campi: se ci sono uno o più campi che sono stati lasciati vuoti o se il valore inserito non corrisponde a nessun dato tra quelli presenti nella tabella, viene visualizzato un messaggio di errore; se invece sono corretti, viene mandata a video l’intera tupla da modificare. Questa seconda pagina contiene un modulo form che permette di inserire i nuovi dati della tupla selezionata che andranno a sovrascrivere quelli già esistenti. I dati passano all’ultima pagina php e, dopo aver effettuato i controlli necessari viene eseguita un’operazione di UPDATE che aggiorna la tabella con i nuovi dati.

L’eliminazione di una tupla in una tabella avviene inserendo nel modulo form i valori di chiave primaria che identifica la tupla che si vuole eliminare e, una volta eseguiti i controlli sulla validità dei valori, attraverso l’operazione di DELETE viene eliminato il record desiderato.

Per quanto riguarda la pagina nella quale posso accedere anche gli utenti non amministratori, non è necessario una registrazione. In questa pagina è presente un modulo form in cui l’utente deve inserire il nome della citta in cui si trova il cinema di cui vuole visualizzare la programmazione. I dati vengono controllati e, nella pagina ricerca.php viene effettuata una query che permette la visualizzazione di una parte della tabella PROIEZIONI.

Oltre al controllo sul valore dei campi, in fase di modifica o di aggiornamento viene anche eseguito un controllo che riguarda le relazioni tra le tabelle. Infatti il programma si basa sul vincolo di integrità referenziale che verrà spiegato successivamente, per cui: data una tabella A e una tabella B, dove B contiene una chiave esterna sul campo di chiave primaria della tabella A:

- non è possibile eliminare o modificare una tupla della tabella A se una delle tuple della tabella B contiene nel campo di chiave esterna, il valore della chiave primaria che si sta modificando/eliminando

-non è possibile inserire nella tabella B, nei campi che sono chiave esterna, dei valori che non sono già presenti della tabella A.

**Integrità referenziale**

Nei database relazionali, perché sia rispettata l'integrità referenziale, ogni campo in una tabella che sia stato dichiarato come foreign key può contenere solo valori della chiave primaria di una tabella "madre" e, inoltre, il campo, deve avere lo stesso tipo.

Alcuni RDBMS possono garantire l'integrità relazionale, o cancellando le rispettive righe della tabella che contiene la foreign key, oppure interrompendo l'operazione e non effettuando la cancellazione.

Nel nostro programma per la memorizzazione dei dati viene utilizzato un Database con il linguaggio SQL e per la creazione delle tabelle viene utilizzato il motore InnoDB che supporta l’integrità referenziale.

**Regole di vincolo**

1. Ogni attore può recitare più film

2. Ogni film può essere recitato da più attori redatto da un unico regista

3. Ogni regista può redigere più film

4. Ogni proiezione si riferisce ad un unico film

**Tabella delle variabili**

Nelle pagine seguenti si trova una tabella che per ogni variabile indica il nome, la pagina in cui la variabile si trova e la funzione della variabile.

|  |  |  |
| --- | --- | --- |
| NOME | PAGINA/E | FUNZIONE |
| $nome | Inserimento\_attori.phpInserimento\_registi.phpmodifica\_attori1.phpmodifica\_registi1.phpelimina\_attore.phpelimina\_regista.php | Riportato il valore inserito nel campo nome |
| $cognome | Inserimento\_attori.phpInserimento\_registi.phpmodifica\_attori1.phpmodifica\_registi1.phpelimina\_attore.phpelimina\_regista.php | Riportato il valore inserito nel campo cognome |
| $annodinascita | Inserimento\_attori.phpInserimento\_registi.phpmodifica\_attori1.phpmodifica\_registi1.php | Riportato il valore inserito nel campo annodinascita |
| $nazionalita | Inserimento\_attori.phpInserimento\_registi.phpmodifica\_attori1.phpmodifica\_registi1.php | Riportato il valore inserito nel campo nazionalità |
| $ruolo | Inserimento\_attori.phpInserimento\_ruolo.php | Riportato il valore inserito nel campo ruolo |
| $titolo | Inserimento\_film.php | Riportato il valore inserito nel campo titolo |
| $annoproduzione | Inserimento\_film.php | Riportato il valore inserito nel campo anno di produzione |
| $codice | Inserimento\_film.phpModifica\_film1.phpElimina\_film.php | Riportato il valore inserito nel campo codice |
| $citta | Inserimento\_proiezioni.phpModifica\_proiezioni1.phpElimina\_proiezione.php | Riportato il valore inserito nel campo ccittà |
| $sala | Inserimento\_proiezioni.phpModifica\_proiezioni1.phpElimina\_proiezione.php | Estrazione di una parte di riga delimitata sala |
| $data | Inserimento\_proiezioni.phpModifica\_proiezioni1.phpElimina\_proiezione.php | Riportato il valore inserito del campo data |
| $ora | Inserimento\_proiezioni.phpModifica\_proiezioni1.phpElimina\_proiezione.php | Riportato il valore inserito del campo ora |
| $spettatori | Inserimento\_proiezioni.phpModifica\_proiezioni1.phpElimina\_proiezione.php | Riportato il valore inserito del campo spettatori |
| $conn | In tutte le pagine | Contiene dati per la connessione al dababase  |
| $ nome\_db | In tutte le pagine | Identifica il nome del databse  |
| $db | In tutte le pagine | Connessione al database  |
| $rec\_ … | In tutte le pagine di modifica e eliminazione | Riportato il valore inserito nel campo titolo |
| $x | In tutte le pagine di modifica e eliminazione | Contatore utilizzato nel vettore |
| $query3 | In tutte le pagine di modifica e eliminazione | Contiene istruzioni query |
| $result3 | In tutte le pagine di modifica e eliminazione | Contiene risultato di query |

**Istruzioni di inserimento dati**

VALUES ('01032450072','Luca','Rossi','Milano','03714589752');

**Queries presenti nel programma**

1. Eliminare l’attore Mario Rossi

 "DELETE FROM attori WHERE nome='Mario’ and cognome='Rossi ";

1. Visualizzare il filtutti i dati del film m che ha per codice ‘0000124578’

 SELECT \* FROM film WHERE codice='0000124578’;

1. Visualizzare I dati di tutti I registi

SELECT \* FROM registi;

1. Visualizzare i dati delle proiezioni e i titoli dei film in programma nella citta di Milano ordinando il risultato dalla data più recente.

SELECT codiceproiezione, codicefilm, titolo, data, ora, sala, citta FROM `proiezioni`,`film` WHERE film.codice = proiezioni.codicefilm and`citta` like 'Milano' ORDER BY `data` DESC;

1. Visualizzare l’elenco dei film con i rispettivi attori e l’indicazione del ruolo.

SELECT codicefilm,titolo,nomeattore,cognomeattore,protagonista FROM recitazioni,film WHERE film.codice = recitazioni.codicefilm

**Script di creazione del Database**

CREATE TABLE IF NOT EXISTS `attori` (

 `nome` varchar(20) NOT NULL,

 `cognome` varchar(20) NOT NULL,

 `nazionalita` varchar(20) NOT NULL,

 `annonascita` int(4) NOT NULL,

 PRIMARY KEY (`nome`,`cognome`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `registi` (

 `nome` varchar(20) NOT NULL,

 `cognome` varchar(20) NOT NULL,

 `nazionalita` varchar(20) NOT NULL,

 `annodinascita` int(4) NOT NULL,

 PRIMARY KEY (`nome`,`cognome`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `film` (

 `codice` varchar(10) NOT NULL,

 `nomeregista` varchar(20) NOT NULL,

 `cognomeregista` varchar(20) NOT NULL,

 `titolo` varchar(30) NOT NULL,

 `annoproduzione` int(4) NOT NULL,

 PRIMARY KEY (`codice`),

FOREIGN KEY (`nomeregista`, `cognomeregista`) REFERENCES `registi` (`nome`, `cognome`) ON UPDATE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `proiezioni` (

 `codiceproiezione` varchar(10) NOT NULL,

 `codicefilm` varchar(10) NOT NULL,

 `data` date NOT NULL,

 `ora` varchar(5) NOT NULL,

 `sala` int(11) NOT NULL,

 `citta` varchar(10) NOT NULL,

 `n\_spettatori` int(11) NOT NULL,

 PRIMARY KEY (`codiceproiezione`),

FOREIGN KEY (`codicefilm`) REFERENCES `film` (`codice`) ON UPDATE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `recitazioni` (

 `codicefilm` varchar(10) NOT NULL,

 `nomeattore` varchar(20) NOT NULL,

 `cognomeattore` varchar(20) NOT NULL,

 `ptotagonista` varchar(2) NOT NULL,

 PRIMARY KEY (`codicefilm`,`nomeattore`,`cognomeattore`),

 FOREIGN KEY (`nomeattore`, `cognomeattore`) REFERENCES `attori` (`nome`, `cognome`) ON UPDATE CASCADE

FOREIGN KEY (`codicefilm`) REFERENCES `film` (`codice`);

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

**Sono state realizzate 40 pagine cosi suddivise:**

**8 pagine html** (index.html – area\_gestione.html - contact.html –inserimento\_attori.html – inserimento\_registi.html –programmazione.html - shop.html);

**32 pagine php** (elimina\_film0.php - elimina\_attore0.php - elimina\_ruolo0.php – elimina\_proiezione0.php – elimina\_regista0.php – elimina\_film.php - elimina\_attore.php - elimina\_ruolo.php – elimina\_proiezione.php – elimina\_regista.php – inserimento\_attori.php – inserimento\_registi.php – inserimento\_ruolo0.php - inserimento\_ruolo.php - inserimento\_film0.php - inserimento\_film.php - inserimento\_proiezioni0.php - inserimento\_proiezioni.php - modifica\_attori.php - modifica\_attori.php - modifica\_attori.php - modifica\_registi.php - modifica\_registi.php - modifica\_registi.php - modifica\_proiezioni.php - modifica\_proiezioni.php - modifica\_proiezioni.php - modifica\_film.php - modifica\_film.php - modifica\_film.php - ricerca.php – controlla.php – visualizzazione\_ruolo.php);

**MANUALE D’USO**

**Manuale d’uso (ITALIANO)**

Il sito è visualizzabile su qualsiasi computer con qualsiasi sistema operativo poiché è presente in rete.

La pagina iniziale non è altro che la pagina di presentazione del sito che permette di accedere alle varie funzioni del programma:

Il pulsante **Home** consente di ricaricare la pagina inziale;

Il pulsante **Gestione** permette all’amministratore di accedere alle principali funzioni del programma:

* **Inserisci attore** consente di inserire i dati di un attore.
* **Modifica attore** consente di modificare i dati di un attore.
* **Elimina attore** consente eliminare i dati di un attore
* **Inserisci regista** consente di inserire i dati di un regista.
* **Modifica regista** consente di modificare i dati di un regista.
* **Elimina regista** consente eliminare i dati di un regista.
* **Inserisci film** consente di inserire i dati di un nuovo film
* **Modifica film** consente di modificare i dati di un film.
* **Elimina film** consente eliminare i dati di un film.
* **Inserisci proiezione** consente di inserire la proiezione di un nuovo film.
* **Modifica proiezione** consente di modificare i dati di una proiezione.
* **Elimina proiezione** consente eliminare la proiezione di un film
* **Inserisci ruolo** consente di inserire il ruolo di un attore in un film
* **Elimina ruolo** consente eliminare il ruolo di un attore in un film
* **Visualizzazione Film-Attori-Ruolo** consente di visualizzare tutti i film con i rispettivi attori e il loro ruolo.

Il pulsante **Programmazione** permette di consultare la programmazione dei film di un cinema inserendo il nome della città.

Il pulsante **Contact** permette di accedere a una pagina checontiene delle informazioni utili (numero di telefono, e-mail) con la quale è possibile prenotare i biglietti o mettersi in contatto con i dirigenti dei cinema.

**Manuale d’uso (INGLESE)**

The site is viewable on any computer with any operating system since it is present in the network.

The home page is nothing more than the presentation page of the site that allows you to access the various functions of the program :

The button **Home** allows you to recharge your home page ;

The button **Gestione** allows the administrator to use the main functions of the program :

* **Inserisci attore** allows you to put in details of an actor.
* **Modifica attore** allows you to modify details of an actor.
* **Elimina attore** allows you to delete details of an actor
* **Inserisci regista** allows you to put in details of a movie director.
* **Modifica regista** allows you to modify details of a movie director.
* **Elimina regista** allows you to delete details of a movie director .
* **Inserisci film** allows you to put in details of a new movie
* **Modifica film** allows you to modify details of a movie .
* **Elimina film** allows you to delete details of a movie.
* **Inserisci proiezione** allows you to put in details of a showing of a new film.
* **Modifica proiezione** allows you to modify details of a showing.
* **Elimina proiezione** allows you to delete a showing of a movie.
* **Inserisci ruolo** allows you to put in the role of an actor in a movie
* **Elimina ruolo** allows to delete the role of an actor in a movie.
* **Visualizzazione Film-Attori-Ruolo** allows you to view all the movies with their actors and their roles.

The button **Programmazione** allows you to browse the program of the movie in a cinema by putting the name of the city.

The button **Contact** allows you to see the page that contains useful information ( telephone number , e-mail ) with which you can book your tickets or get in touch with the manager of the cinema.