

# Basi di Dati 19/20: Progetto

---

## Studenti:

- **Salvatore Cavallaro** - Matricola : 858351

## INDICE

---

- **Progettazione Concettuale**
  - Requisiti iniziali
  - Glossario dei termini
  - Requisiti rivisti in gruppi di frasi omogenee
  - Schema ER
  - Business Rules
    - Dizionario dei Dati (entità)
    - Dizionario dei Dati (associazioni)
    - Vincoli di Integrità
    - Vincoli di Derivazione
- **Progettazione Logica**
  - Tavola dei volumi
  - Tavola delle Operazioni
  - Ristrutturazione Schema ER
    - Analisi delle ridondanze
    - Partizionamento/Accorpamento di entità e associazioni
    - Scelta degli identificatori principali
    - Schema ER ristrutturato + Business Rules
  - Schema Relazionale
- **Implementazione**
  - DDL di creazione del database
  - DML di popolamento di tutte le tabelle del database
  - Operazioni di cancellamento e modifica dei dati

# Progettazione Concettuale

---

## Requisiti iniziali

Si vuole realizzare una base di dati per la gestione di una piattaforma che fornisce informazioni su film in uscita nei cinema e su serie e programmi TV in onda, liberamente ispirato a piattaforme come [ComingSoon](#).

Un numero limitato di utenti della redazione si occupa di aggiornare le informazioni sui contenuti disponibili inserendo i dati di film e serie, comprese le date di uscita e programmazione in sala e in TV. Degli utenti si memorizzano nome utente, password e indirizzo email. Per i redattori si tiene traccia anche della data di inizio collaborazione.

I contenuti hanno una serie di caratteristiche come evidenziato in Figura 1. In particolare, a ogni film e serie TV è associata una lista di attori, con l'indicazione del personaggio interpretato. Notare che alcune informazioni tipiche dei film, ad esempio la regia, per le serie TV sono specifiche dei singoli episodi. Attori e registi hanno una propria scheda (esempio in Figura 2) dove vengono visualizzati nome, foto, e alcune informazioni anagrafiche. Qui può essere anche mostrata una lista dei contenuti più recenti a cui la persona ha partecipato.

I film vengono proiettati in svariati cinema in tutto il paese. Per ogni cinema, si tiene traccia del suo nome, contatti, e localizzazione (regione, provincia e indirizzo). La proiezione avviene a degli orari specifici in diverse date, in una delle sale disponibili al cinema, e comporta il pagamento di un biglietto il cui prezzo è indicato sul sito.

Programmi e serie TV vanno invece in onda su diverse piattaforme (es. Netflix, Rai Gulp, ...), che possono essere canali TV tradizionali o piattaforme di streaming video. Le serie TV sono organizzate per genere, come i film, es. commedia, drammatico, etc. In più, i serial sono composti da episodi, racchiusi in una o più stagioni, dove ogni episodio ha uno specifico titolo, durata, regista, e sceneggiatore. Anche per i programmi sono memorizzate alcune informazioni, come l'anno di messa in onda, genere, paese, durata e una descrizione testuale.

I visitatori del sito, previa registrazione tramite email e password, possono lasciare un voto (da 1 a 5 stelline) su qualunque contenuto. La media dei voti ricevuti da film e programmi viene mostrata nelle rispettive schede. Inoltre, gli utenti possono cliccare su un link "inserisci tra i preferiti" presente nelle schede dei contenuti, in modo da poter visualizzare in seguito i contenuti salvati. Nel momento della cancellazione di un utente dal sito, i suoi preferiti possono essere rimossi dal database, mentre i voti espressi dal visitatore rimangono in memoria.

## Glossario dei termini

Termine	Descrizione	Sinonimi	Collegamenti
Utente	Persona registrata	Lettore, visitatore, redazione	Contenuto, Film, Serie TV, Programmi Tv, preferiti, voto
Contenuto	Include il concetto di Film, Serie Tv o Programmi tv		Attore, Utente
Film	Pellicola cinematografica o anche fotografica	Cortometraggio, documentario, opera, kolossal, lungometraggio o pellicola	Attore, Cinema, Sala, Contenuto, Regista, Proiezione, Utente
Programma Tv	Produzione audiovisiva trasmessa tramite televisione o internet	Programma, produzione, trasmissione,	Contenuto, Attore, Utente
Serie Tv	Opera composta da episodi ed eventualmete anche in stagioni	Serial, fiction, opera	Attore, Contenuto, Utente, Stagione, Episodio, Regista
Redazione	Sottoinsieme di utenti che hanno la possibilità esclusiva di aggiornare e modificare i contenuti disponibili	Redattore del sito	Contenuto
Preferiti	Contenuti preferiti dagli utenti		Utente, Contenuto
Cinema	Luogo pubblico destinato alla visione di opere cinematografiche	Sala cinematografica, cinematografo, multisala, arena	Film, Proiezione, Sala
Proiezione	Proiezione cinematografica specifica per ogni cinema		Cinema, Sala
Attore	Persona che rappresenta o interpreta una parte o un ruolo in uno spettacolo cinematografico, televisivo.	Personaggio, star, interprete, protagonista	Contenuto
Regista	Persona che ha la responsabilità artistica e operativa di una rappresentazione (cinematografica, televisiva)	Direttore artistico, realizzatore	Film, Episodio

## Requisiti rivisti e strutturati in gruppi di frasi omogenee

Si vuole realizzare una base di dati per una piattaforma digitale che da la possibilità di visionare e consultare informazioni su film in uscita nei cinema e su serie e programmi TV in onda.

### **Utente :**

E' una generalizzazione con una sola entità figlia "Redazione", la quale sarà sottoinsieme per Utente. Gli utenti diventano tali dopo essersi correttamente iscritti fornendo un valido indirizzo email, una password e un nome utente.

### **Redazione :**

Sono alcuni utenti che possono modificare e aggiornare i dati sui contenuti disponibili, per i redattori oltre a memorizzare una email una password e un nome utente, si memorizza anche una data di inizio collaborazione.

### **Contenuto :**

Generalizzazione parziale ed esclusiva di tre specifiche sottocategorie : Film, Serie TV e Programmi TV. Tutti gli attributi presenti in Contenuto saranno ereditati dalle tre sottocategorie. Tra questi, ho un'attributo denominato "id" che identificherà univocamente ogni istanza presente in esso.

### **Attore :**

Gli attori sono descritti da : un nome, un cognome, dati anagrafici come data di nascita e luogo di nascita, un'età, una breve biografia ed un eventuale foto. Gli attori saranno identificati dal nome, dal cognome e dalla data di nascita, in quanto posso esistere attori che presentano stesso nome e cognome.

### **Regista :**

I registi sono identificati dall'attributo chiave "id" il quale identificherà univocamente un regista, in quanto possono esistere più registi omonimi.

### **Film :**

Specializzazione per Contenuto.

Saranno identificati dall' "id" ereditato da Contenuto, non uso il titolo come identificatore poichè posso coesistere più film con lo stesso titolo. Essendo una specializzazione per Contenuto, avrò diversi attributi specifici per l'istanza di tipo Film, come la sceneggiatura, la durata e le musiche.

### **Cinema :**

Specifica i cinema in cui andranno proiettati i film. Un cinema è identificato dal "Nome" e dall' "Indirizzo".

**Proiezione :**

Specifica oggettivamente una proiezione cinematografica, viene identificata da un identificatore esterno che lega gli attributi “id\_film”, “data” , ”ora” con l’associazione a Cinema. Questo perchè possono esistere proiezioni dello stesso film che avvengono nello stesso giorno e alla stessa ora in cinema diversi. Ma l’identificatore esterno legato con Cinema solleva il problema.

A una specifica istanza di Proiezione è anche associato un’attributo “prezzo” che identifica il prezzo relativo a quella determinata proiezione.

**Sala :**

Specifica in un cinema quale sala cinematografica proietterà un certo film. La identifico per mezzo di un identificatore esterno. Quindi una sala può essere identificata dall’attributo “Numero” e dall’associazione con Cinema, in quanto un cinema non può avere due sale con uguale numero.

**Programmi TV :**

Specializzazione per Contenuto. Li identifico, così come i Film, tramite l’ “id” ereditato da Contenuto. Essendo una specializzazione, in più andrò a specificare attributi come “Piattaforma” , “Durata” e “Descrizione”.

**Serie TV :**

Specializzazione per Contenuto. Identificati anche loro attraverso l’attributo “id” di Contenuto, oltre ad aggiungere attributi come “sceneggiatura”, “musiche” e “Piattaforma”, andro’ a suddividere le Serie TV in Stagioni e poi in Episodi, in quanto diversi attributi caratterizzano gli specifici episodi.

**Stagione :**

Specifica una determinata stagione per una Serie Tv. Uso un identificatore esterno. L’identificazione è resa attraverso l’attributo “Numero” di Stagione e l’associazione “Possiede” con Serie TV, in quanto non posso usare solo l’attributo “Numero” di Stagione perchè lo stesso numero potrei ritrovarlo per una stessa istanza di Serie TV.

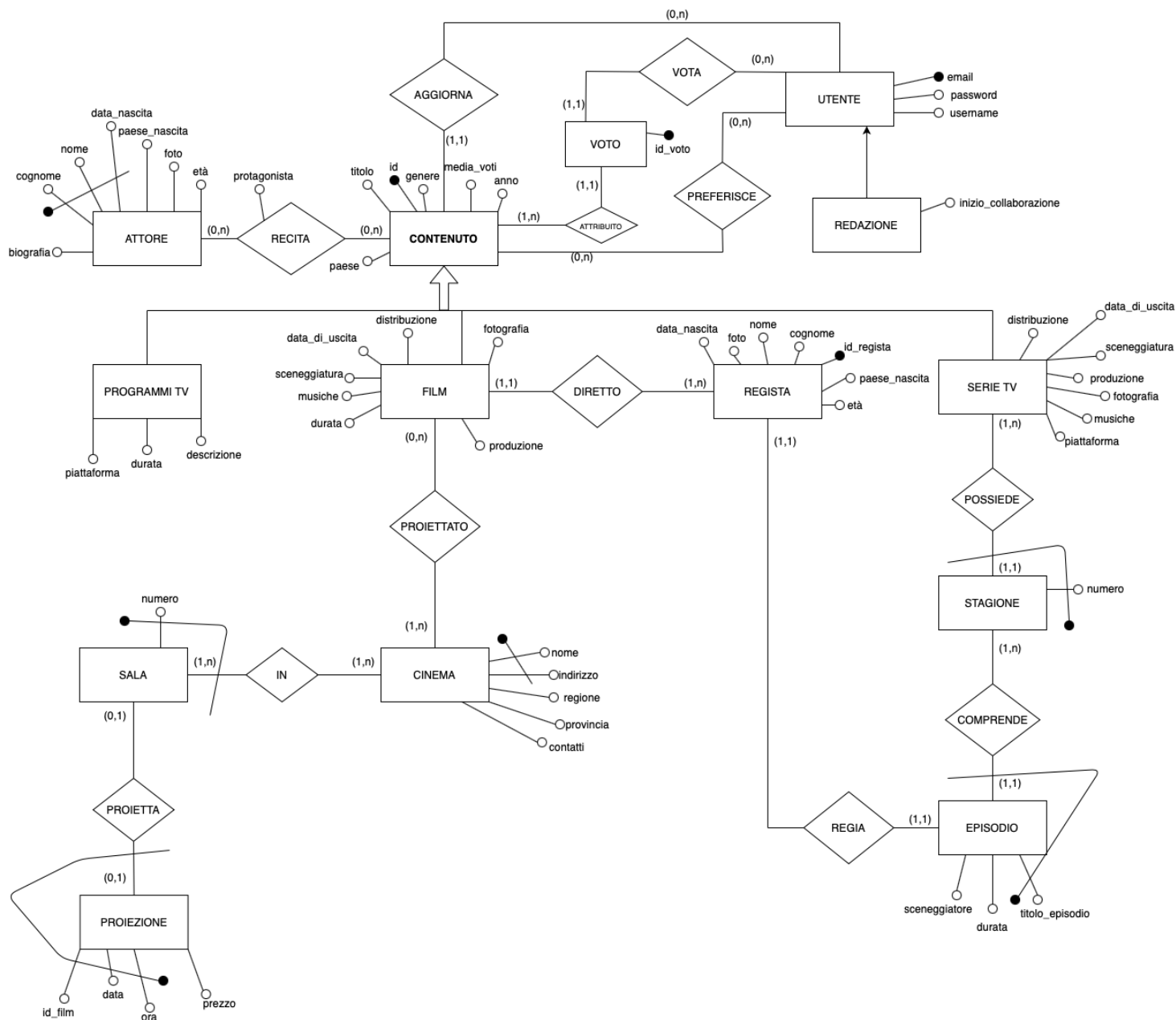
**Episodio :**

Come per la Stagione, specifica un determinato episodio appartenente a una specifica stagione di una certa Serie TV. La identifico tramite un identificatore esterno, il quale coinvolge Stagione che era già stata identificata esternamente. Così posso ottenere l’identificazione di un determinato episodio a partire dalla combinazione di “id” (specifica Serie TV) e “Numero” (specifica Stagione).

**Voto :**

Specifica i voti riferiti a un determinato Contenuto. Uso una chiave “id\_voto” per identificarla.

Schema ER



## Business Rules

- Dizionario dei dati (entità)

Entità	Descrizione	Attributi	Identificatore
Utente	Utente che può consultare i contenuti	email, password, username	email
Redazione	Piccola parte di utenti che posso aggiornare i contenuti	inizio_collaborazione	email
Contenuto	Insieme di tutti i contenuti consultabili dagli utenti	id, titolo, genere, paese, anno, media_voti	id
Attore	Persona che interpreta un personaggio con un ruolo in un contenuto.	nome, cognome, data_nascita, paese_nascita, foto, età, biografia	nome, cognome, data_nascita
Programmi TV	Trasmissioni audiovisive messe in onda su diverse piattaforme	piattaforma, durata, descrizione	id
Serie TV	Serial trasmessi su diverse piattaforme, possono comprendere diverse stagioni con più episodi	distribuzione, data_di_uscita, sceneggiatura, produzione, fotografia, musiche, piattaforma	id
Film	Pellicola cinematografica proiettata presso diversi cinema	distribuzione, data_di_uscita, sceneggiatura, produzione, fotografia, musiche, durata	id
Regista	Colui che produce e dirige un contenuto	nome, cognome, id_regista, foto, data_nascita, età, paese_nascita	id_regista
Cinema	Ove vengono proiettate le pellicole cinematografiche	nome, indirizzo, regione, provincia, contatti	nome, indirizzo
Sala	Sala presente presso i cinema	numero	numero, Cinema
Proiezione	Informazioni dirette a descrivere univocamente una determinata proiezione cinematografica	id_film, data, ora, prezzo	id_film, data, ora, Cinema
Stagione	Stagioni di un serial	Numero	numero, Serie TV
Episodio	Episodi appartenenti a una determinata stagione	sceneggiatore, durata, titolo_episodio	titolo_episodio, Stagione
Voto	Voti che appartengono a un dato Contenuto		id_voto

- **Dizionario dei dati (associazioni)**

Relazioni	Descrizione	Componenti	Attributi
Aggiorna	Aggiornare un contenuto	Utente, Contenuto	
Vota	Votare un contenuto	Utente, Voto	
Preferisce	Preferire un contenuto	Utente, Contenuto	
Recita	Appartenenza di un attore ad un contenuto	Attore, Contenuto	protagonista
Diretto	Indica quale regista dirige artisticamente un film o serie tv	Film, Regista	
Possiede	Appartenenza di una stagione ad una serie tv	Serie TV, Stagione	
Comprende	Appartenenza di un episodio ad una stagione	Stagione, Episodio	
Regia	Indica quale regista dirige uno specifico episodio di una serie tv	Episodio, Regista	
Proiettato	Proiezione di un film presso un cinema	Film, Cinema	
Proietta	Quale cinema proietta una specifica proiezione	Cinema, Proiezione	
in	Appartenenza di una sala cinematografica a un cinema	Sala, Cinema	
Attribuito	Indica quale voto si riferisce a quale contenuto	Voto, Contenuto	

### Vincoli di integrità

1. L'attributo durata delle specializzazioni non è inserito nell'entità genitore Contenuto, in quanto la durata per la specializzazione Serie TV è relativa ai singoli episodi che la costituiscono, a differenza dei Film e dei Programmi TV.
3. La redazione è un'istanza particolare di Utente, ecco perchè la rappresento come subset.
4. I preferiti di ciascun Utente verranno eliminati qualora l'utente venisse eliminato dal database.
5. I voti di ciascun Utente sui Contenuti non vengono eliminati se l'utente viene rimosso dal database

### Vincoli di derivazione

1. La lista dei contenuti preferiti di ciascun utente è considerata come l'insieme di tutti i contenuti preferiti dall'utente stesso.



2. Ogni volta che un redattore aggiunge un nuovo contenuto l'attributo id, dell'entità Contenuto, verrà auto incrementato di uno.
3. L'attributo tipo di Utente può assumere solo due valori : “user” , “admin”.
4. Solo ed esclusivamente gli utenti di tipo “admin” possono aggiornare un contenuto.
5. L'attributo media\_voti di Contenuto rappresenta la media dei voti espressi da ciascun Utente su ogni istanza presente in Contenuto. Ogni qual volta un'istanza di Contenuto ricevesse un nuovo voto da parte di un Utente, verrà aggiornato l'attributo media\_voti, dopo averne calcolato la nuova media.

## Progettazione Logica

### Tavola dei volumi

Concetto	Tipo	Volume
Utente	E	10k
Redazione	E	100
Attore	E	5k
Programmi TV	E	1k
Film	E	5k
Serie TV	E	1k
Regista	E	500
Stagione	E	3k
Episodio	E	40k
Cinema	E	1200
Sala	E	3500
Proiezione	E	12000
Aggiorna	A	100 * 50
Vota	A	10k * 3
Preferisce	A	10k * 3
Recita	A	5k * 10
Diretto	A	500 * 4
Proiettato	A	10 * 1200
Proietta	A	1200 * 3
In	A	1200 * 3
Regia	A	40k
Possiede	A	1k * 3
Comprende	A	3k * 8
Voto	E	30k
Attribuito	A	30k*7k

## Tavola delle operazioni

Media Giornaliera :

- Utenti attivi : 2k

Operazione	Tipo	Frequenza giornaliera
1. Inserire un utente	I	30
2. Aggiungere contenuto	I	10
3. Aggiornare un contenuto	I	30
4. Preferire un contenuto	I	500
5. Votare un contenuto	I	1k
6. Assegnare un attore a un contenuto	I	100
7. Per ogni film, trova il regista che gli appartiene	I	1k
8. Per ogni serie tv, elenca le stagioni che le appartengono	I	1k
9. Per ogni stagione, trova i relativi episodi	I	1k
10. Per ogni episodio trova il regista corrispondente	I	1k
11. Per ogni film, trova i cinema che lo proiettano	I	1k
12. Per ogni cinema, trova le sale che gli appartengono	I	1k
13. Per ogni cinema trova tutte le proiezioni disponibili	I	1k

## Ristrutturazione dello schema E-R

---

### Analisi delle ridondanze

#### Ridondanze presenti :

Non sono presenti ridondanze. Sono, invece, presenti delle generalizzazioni che occorre eliminare/sostituire con entità e/o associazioni.

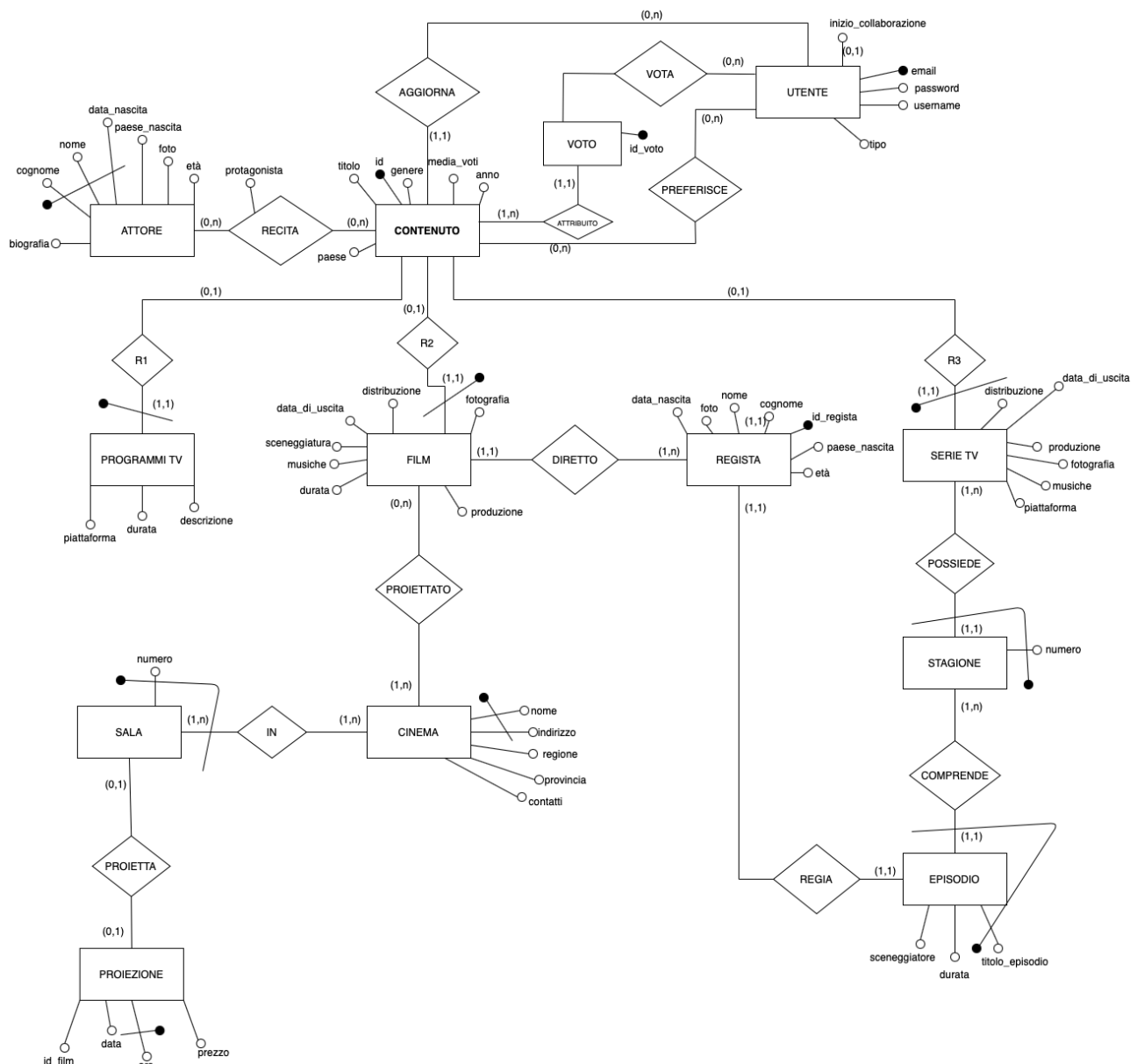
## Eliminazione delle generalizzazioni :

Poichè occorre fare distinzione negli accessi alle occorrenze delle entità figlie ( Programmi TV, Film, Serie TV), procedo ad eliminare la generalizzazione, in questo caso di tipo parziale ed esclusiva, tramite la sostituzione con associazioni. Quindi la generalizzazione Contenuto andrà modificata e inserita nel modello E-R ristrutturato.

Mentre la seconda generalizzazione riguarda Utente, che generalizza l'entità Redazione, in questo caso si tratta di un sottoinsieme per Utente. Procederò ad eliminare la generalizzazione accorpendo l'entità figlia, Redazione, in Utente.

**Entrambe le modifiche sono riportate nel modello E-R ristrutturato.**

## Schema E-R ristrutturato



## Business Rules

---

- Dizionario dei dati (entità)

Entità	Descrizione	Attributi	Identificatore
Utente	Utente che può consultare i contenuti	email, password, username	email
Contenuto	Insieme di tutti i contenuti consultabili dagli utenti	id, titolo, genere, paese, anno, media_voti	id
Attore	Persona che interpreta un personaggio con un ruolo in un contenuto.	nome, cognome, data_nascita, paese_nascita, foto, età, biografia	nome, cognome, data_nascita
Programmi TV	Trasmissioni audiovisive messe in onda su diverse piattaforme	piattaforma, durata, descrizione	id
Serie TV	Serial trasmessi su diverse piattaforme, possono comprendere diverse stagioni con più episodi	distribuzione, data_di_uscita, sceneggiatura, produzione, fotografia, musiche, piattaforma	id
Film	Pellicola cinematografica proiettata presso diversi cinema	distribuzione, data_di_uscita, sceneggiatura, produzione, fotografia, musiche, durata	id
Regista	Colui che produce e dirige un contenuto	nome, cognome, id_regista, foto, data_nascita, età, paese_nascita	id_regista
Cinema	Ove vengono proiettate le pellicole cinematografiche	nome, indirizzo, regione, provincia, contatti	nome, indirizzo
Sala	Sala presente presso i cinema	numero	numero, Cinema
Proiezione	Informazioni dirette a descrivere univocamente una determinata proiezione cinematografica	id_film, data, ora, prezzo	id_film, data, ora, Cinema
Stagione	Stagioni di un serial	Numero	numero, Serie TV
Episodio	Episodi appartenenti a una determinata stagione	sceneggiatore, durata, titolo_episodio	titolo_episodio, Stagione
Voto	Voti espressi dagli utenti verso i Contenuti	id_voto, voto	id_voto

• **Dizionario dei dati (associazioni)**

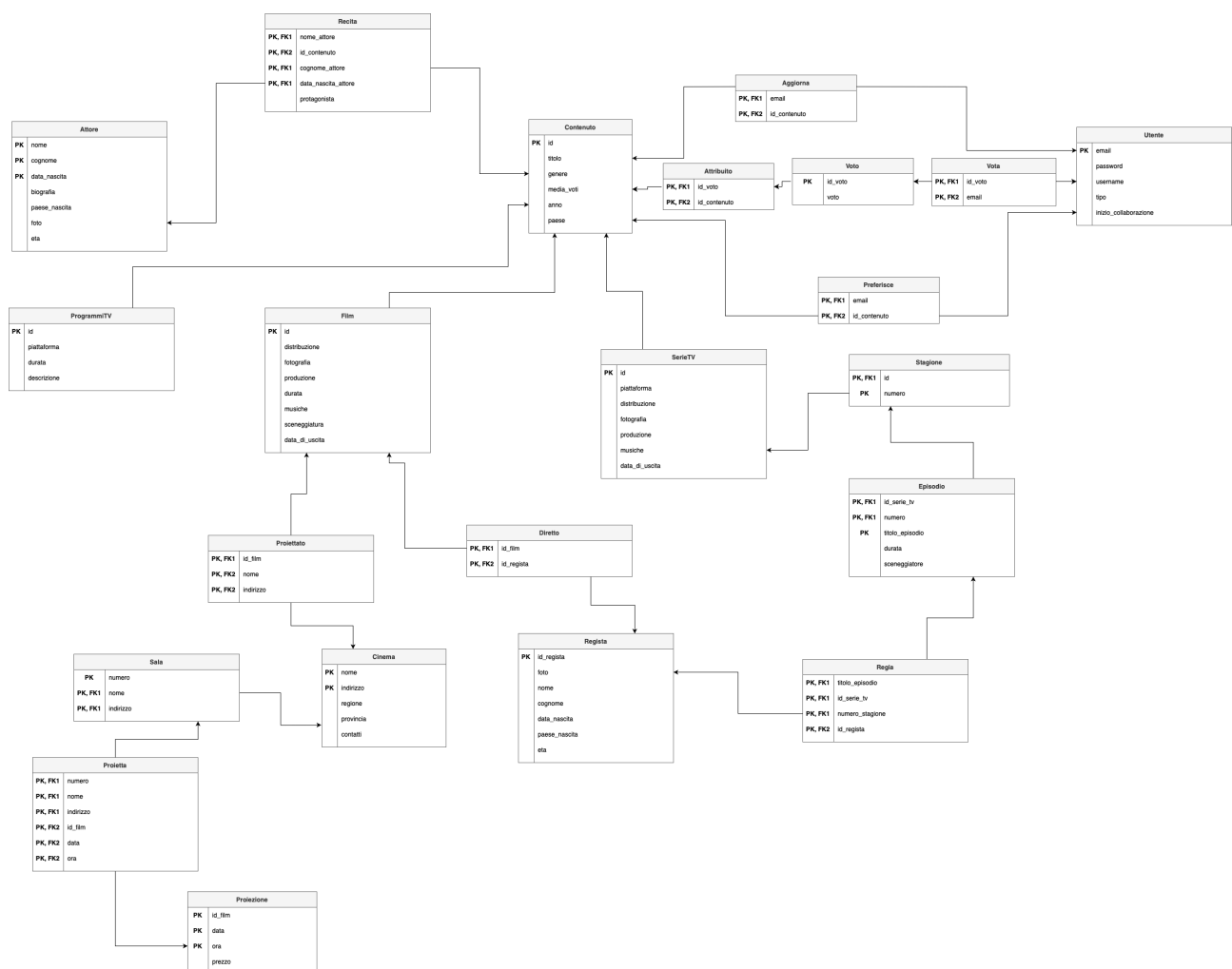
Relazioni	Descrizione	Componenti	Attributi
Aggiorna	Aggiornare un contenuto	Utente, Contenuto	
Vota	Votare un contenuto	Utente, Voto	
Preferisce	Preferire un contenuto	Utente, Contenuto	
Recita	Appartenenza di un attore ad un contenuto	Attore, Contenuto	protagonista
Diretto	Indica quale regista dirige artisticamente un film o serie tv	Film, Regista	
Possiede	Appartenenza di una stagione ad una serie tv	Serie TV, Stagione	
Comprende	Appartenenza di un episodio ad una stagione	Stagione, Episodio	
Regia	Indica quale regista dirige uno specifico episodio di una serie tv	Episodio, Regista	
Proiettato	Proiezione di un film presso un cinema	Film, Cinema	
Proietta	Quale cinema proietta una specifica proiezione	Cinema, Proiezione	
in	Appartenenza di una sala cinematografica a un cinema	Sala, Cinema	
Attribuito	Riferimento a quale Contenuto si riferisce un certo Voto	Voto, Contenuto	

**Vincoli di integrità**

1. L'attributo durata delle specializzazioni non è inserito nell'entità genitore Contenuto, in quanto la durata per la specializzazione Serie TV è relativa ai singoli episodi che la costituiscono, a differenza dei Film e dei Programmi TV.
2. Ogni istanza di Contenuto partecipa al più a R1 o R2 o R3.
3. La redazione è un'istanza particolare di Utente, ecco perchè la rappresento come subset.
4. I preferiti di ciascun Utente verranno eliminati qualora l'utente venisse eliminato dal database.
5. I voti di ciascun Utente sui Contenuti non vengono eliminati se l'utente viene rimosso dal database
6. Ogni volta che un utente viene aggiunto al database occorre specificare l'attributo tipo di Utente. Tale scelta viene determinata dal momento in cui l'utente inserirà in fase di registrazione

## Vincoli di derivazione

- ## Schema Relazionale



---

# Implementazione

---

## DDL di creazione del database

```
CREATE TABLE Utente (  
    email VARCHAR(15) NOT NULL,  
    password VARCHAR(20) NOT NULL,  
    username VARCHAR(20) NOT NULL UNIQUE,  
    tipo VARCHAR(20) NOT NULL,  
    inizio_collaborazione DATE DEFAULT NULL,  
    PRIMARY KEY (email  
)  
  
CREATE TABLE Aggiorna (  
    email VARCHAR(15) NOT NULL,  
    id_contenuto INT NOT NULL,  
    PRIMARY KEY (email, id_contenuto),  
    FOREIGN KEY (email) REFERENCES Utente (email)  
        ON DELETE CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (id_contenuto) REFERENCES Contenuto (id)  
        ON DELETE CASCADE ON UPDATE CASCADE  
)  
  
CREATE TABLE Vota (  
    email VARCHAR(15) NOT NULL,  
    id_voto INT NOT NULL,  
    PRIMARY KEY (email, id_voto),  
    FOREIGN KEY (email) REFERENCES Utente (email)  
        ON UPDATE CASCADE,  
    FOREIGN KEY (id_voto) REFERENCES Voto (id_voto)  
        ON UPDATE CASCADE  
)  
  
CREATE TABLE Preferisce (  
    email VARCHAR(15) NOT NULL,  
    id_contenuto INT NOT NULL,
```

```
PRIMARY KEY (email, id_contenuto),  
FOREIGN KEY (email) REFERENCES Utente (email)  
    ON DELETE CASCADE ON UPDATE CASCADE,  
FOREIGN KEY (id_contenuto) REFERENCES Contenuto (id)  
    ON DELETE CASCADE ON UPDATE CASCADE
```

)

```
CREATE TABLE TipoContenuto (  
    tipo_contenuto VARCHAR(1) NOT NULL,  
    nome_contenuto VARCHAR(20) NOT NULL,  
    PRIMARY KEY(tipo_contenuto, nome_contenuto),  
    UNIQUE (nome_contenuto)
```

)

```
CREATE TABLE Contenuto (  
    titolo VARCHAR(15) NOT NULL,  
    id INT NOT NULL AUTO_INCREMENT,  
    genere VARCHAR(20) NOT NULL,  
    media_voti INT DEFAULT NULL,  
    anno INT DEFAULT NULL,  
    paese VARCHAR(20) NOT NULL,  
    PRIMARY KEY (id)
```

)

```
CREATE TABLE Attore (  
    nome VARCHAR(15) NOT NULL,  
    cognome VARCHAR(20) NOT NULL,  
    biografia VARCHAR(20) DEFAULT NULL,  
    data_nascita VARCHAR(20) NOT NULL,  
    paese_nascita VARCHAR(20) DEFAULT NULL,  
    foto BLOB,  
    eta INT,  
    PRIMARY KEY (nome, cognome, data_nascita)
```

)

```
CREATE TABLE Recita (  
    Protagonista VARCHAR(30) NOT NULL,  
    nome_attore VARCHAR(15) NOT NULL,  
    cognome_attore VARCHAR(20) NOT NULL,  
    data_nascita_attore VARCHAR(20) NOT NULL,  
    id_contenuto INT NOT NULL,  
    PRIMARY KEY (nome_attore, cognome_attore, data_nascita_attore, id_contenuto),  
    FOREIGN KEY (nome_attore, cognome_attore, data_nascita_attore) REFERENCES Attore(nome,  
    cognome, data_nascita)  
    ON DELETE CASCADE ON UPDATE CASCADE,
```



FOREIGN KEY (id\_contenuto) REFERENCES Contenuto(id) ON DELETE CASCADE  
ON UPDATE CASCADE

)

CREATE TABLE ProgrammiTV (  
id INT NOT NULL,  
piattaforma VARCHAR(20),  
durata INT NOT NULL,  
descrizione VARCHAR(400),  
PRIMARY KEY (id),  
FOREIGN KEY (id) REFERENCES Contenuto (id)  
ON DELETE CASCADE ON UPDATE CASCADE

)

CREATE TABLE Film (  
id INT NOT NULL,  
distribuzione VARCHAR(30) NOT NULL,  
fotografia VARCHAR(30) NOT NULL,  
produzione VARCHAR(150) DEFAULT NULL,  
durata INT(11) NOT NULL,  
musiche VARCHAR(30) DEFAULT NULL,  
sceneggiatura VARCHAR(150) DEFAULT NULL,  
data\_di\_uscita DATE,  
PRIMARY KEY (id),  
FOREIGN KEY (id) REFERENCES Contenuto (id)  
ON DELETE CASCADE ON UPDATE CASCADE

)

CREATE TABLE Diretto (  
id\_film INT NOT NULL,  
id\_regista INT NOT NULL,  
PRIMARY KEY (id\_film, id\_regista),  
FOREIGN KEY (id\_film) REFERENCES Contenuto (id)  
ON DELETE CASCADE ON UPDATE CASCADE,  
FOREIGN KEY (id\_regista) REFERENCES Regista (id\_regista)  
ON DELETE CASCADE ON UPDATE CASCADE

)

CREATE TABLE Regista (  
data\_nascita DATE,  
foto BLOB,  
nome VARCHAR(20) NOT NULL,  
cognome VARCHAR(20) NOT NULL,  
id\_regista INT NOT NULL AUTO\_INCREMENT,  
paese\_nascita VARCHAR(20),

```
    eta INT,  
    PRIMARY KEY (id_regista)  
)
```

```
CREATE TABLE Proiettato (  
    id_film INT NOT NULL,  
    nome VARCHAR(20) NOT NULL,  
    indirizzo VARCHAR(50) NOT NULL,  
    PRIMARY KEY (id_film, nome, indirizzo),  
    FOREIGN KEY (id_film) REFERENCES Film (id)  
        ON DELETE CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (nome, indirizzo) REFERENCES Cinema (nome, indirizzo)  
        ON DELETE CASCADE ON UPDATE CASCADE  
)
```

```
CREATE TABLE Voto(  
    id_voto INT NOT NULL AUTO_INCREMENT,  
    voto INT NOT NULL,  
    PRIMARY_KEY(id_voto)  
)
```

```
CREATE TABLE Attribuito(  
    id_voto INT NOT NULL,  
    id_contenuto INT NOT NULL,  
    PRIMARY KEY(id_voto,id_contenuto)  
    FOREIGN KEY (id_voto) REFERENCES Voto(id_voto)  
        ON DELETE CASCADE ON UPDATE CASCADE  
    FOREIGN KEY(id_contenuto) REFERENCES Contenuto(id)  
        ON DELETE CASCADE ON UPDATE CASCADE  
)
```

```
CREATE TABLE Cinema (  
    nome VARCHAR(20) NOT NULL,  
    indirizzo VARCHAR(50) NOT NULL,  
    regione VARCHAR(20) NOT NULL,  
    provincia VARCHAR(20) NOT NULL,  
    contatti VARCHAR(15),  
    PRIMARY KEY (nome, indirizzo)  
)
```

```
CREATE TABLE Sala (  
    numero_sala INT NOT NULL,  
    nome VARCHAR(20) NOT NULL,  
    indirizzo VARCHAR(50) NOT NULL,  
    PRIMARY KEY (numero, nome, indirizzo),
```

FOREIGN KEY (nome, indirizzo) REFERENCES Cinema(nome, indirizzo)  
ON DELETE CASCADE ON UPDATE CASCADE

)

```
CREATE TABLE Proietta (  
    numero_sala INT NOT NULL,  
    nome VARCHAR(20) NOT NULL,  
    indirizzo VARCHAR(50) NOT NULL,  
    id_film INT NOT NULL,  
    data DATE NOT NULL,  
    ora TIME NOT NULL,  
    PRIMARY KEY (numero, nome, indirizzo, id_film, data, ora),  
    FOREIGN KEY (numero, nome, indirizzo) REFERENCES Sala(numero, nome, indirizzo)  
        ON DELETE CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (id_film, data, ora) REFERENCES Proiezione(id_film, data, ora)  
        ON DELETE CASCADE ON UPDATE CASCADE  
)
```

```
CREATE TABLE Proiezione (  
    id_film INT NOT NULL,  
    data DATE NOT NULL,  
    ora TIME NOT NULL,  
    prezzo VARCHAR(6) DEFAULT NULL,  
    PRIMARY KEY (id_film, data, ora)  
)
```

```
CREATE TABLE SerieTV (  
    id INT NOT NULL,  
    piattaforma VARCHAR(20) NOT NULL,  
    distribuzione VARCHAR(30) NOT NULL,  
    fotografia VARCHAR(30) NOT NULL,  
    produzione VARCHAR(150),  
    musiche VARCHAR(30),  
    data_di_uscita DATE,  
    PRIMARY KEY (id),  
    FOREIGN KEY (id) REFERENCES Contenuto(id)  
        ON DELETE CASCADE ON UPDATE CASCADE  
)
```

```
CREATE TABLE Stagione (  
    id INT NOT NULL,  
    numero_stagione INT NOT NULL,  
    PRIMARY KEY (id, numero),  
    FOREIGN KEY (id) REFERENCES SerieTV(id)  
        ON DELETE CASCADE ON UPDATE CASCADE
```

)

```
CREATE TABLE Episodio (  
    id_serie_tv INT NOT NULL,  
    numero_stagione INT NOT NULL,  
    titolo_episodio VARCHAR(30) NOT NULL,  
    durata INT NOT NULL,  
    sceneggiatore VARCHAR(150),  
    PRIMARY KEY ( id_serie_tv, numero, titolo_episodio),  
    FOREIGN KEY (id_serie_tv, numero) REFERENCES Stagione(id, numero)  
        ON DELETE CASCADE ON UPDATE CASCADE  
)
```

)

```
CREATE TABLE Regia (  
    titolo_episodio VARCHAR(30) NOT NULL,  
    id_serie_tv INT NOT NULL,  
    numero_stagione INT NOT NULL,  
    id_regista INT NOT NULL,  
    PRIMARY KEY (titolo_episodio, id_serie_tv, numero_stagione, id_regista),  
    FOREIGN KEY (id_regista) REFERENCES Regista(id_regista)  
        ON DELETE CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (id_serie_tv, numero_stagione, titolo_episodio) REFERENCES Episodio(  
        id_serie_tv, numero, titolo_episodio)  
        ON DELETE CASCADE ON UPDATE CASCADE  
)
```

)

## DDL di popolamento di tutte le tabelle del database

```
INSERT INTO Utente (email, password, username, tipo, inizio_collaborazione) VALUES  
( 'salvo@gmail.com', 'root', 'salvo_96', 'user', null),  
( 'andrea_boetto@gmail.com', 'root', 'andreaBoetto', 'user', null)  
( 'paolo_incastrone@gmail.com', 'ciao420', 'poseidonon', 'admin', '2019-04-20' );
```

```
INSERT INTO TipoContenuto (tipo_contenuto, nome_contenuto) VALUES ( 's', 'SerieTv' ), ( 'f', 'Film' ),  
( 'p', 'ProgrammaTv' );
```

```
INSERT INTO Contenuto (titolo, genere, media_voti, anno, paese, tipo) VALUES  
( 'Stranger Things', 'Fantascienza - Horror', null, '2017', 'USA', 's' ),  
( 'A Beautiful Mind', 'Biografico - Drammatico', null, '2001', 'USA', 'f' )  
( 'SuperQuark', 'Divulgazione Scientifica', null, '1995', 'Italia', 'p' );
```

```
INSERT INTO aggiorna (email, id_contenuto) VALUES ( 'paolo_incastrone@gmail.com', 1 );
```

INSERT INTO vota (email, id\_contenuto) VALUES ('salvo@gmail.com', 2);

INSERT INTO preferisce (email, id\_contenuto) VALUES ('paolo\_incastrone@gmail.com', 1);

INSERT INTO Attore (nome, cognome, biografia, data\_nascita, paese\_nascita, foto, eta) VALUES ('Russell', 'Crowe', 'È noto soprattutto per avere interpretato il ruolo del generale Massimo Decimo Meridione *Il gladiatore* di Ridley Scott, ruolo che gli ha dato molta popolarità a livello internazionale e gli ha consentito di aggiudicarsi il Premio Oscar come miglior attore protagonista nel 2001.', '1964-04-07', 'Wellington - USA', null, '54');

INSERT INTO Recita (protagonista, nome\_attore, cognome\_attore, data\_nascita\_attore, id\_contenuto) VALUES ('John Nash', 'Russell', 'Crowe', '1964-04-07', 2);

INSERT INTO ProgrammiTV (id, piattaforma, durata, descrizione) VALUES (3, 'Rai - TV', 120, '*Superquark* è un programma televisivo di divulgazione culturale, nato nel gennaio 1995, ideato e condotto da Piero Angela, evoluzione della precedente trasmissione *Quark*.');

INSERT INTO Film (id, distribuzione, fotografia, produzione, durata, musiche, sceneggiatura, data\_di\_uscita) VALUES (2, 'Universal Pictures', 'Roger Deakins', 'Universal Pictures - DreamWorks', 135, 'James Horner', 'Akiva Goldsman', '2002-02-15');

INSERT INTO Regista (data\_nascita, foto, nome, cognome, paese\_nascita, eta) VALUES ('1954-03-01', null, 'Ron', 'Howard', 'Duncan - USA', 66);

INSERT INTO Regista (data\_nascita, foto, nome, cognome, paese\_nascita, eta) VALUES ('1984-02-15', null, 'Matt', 'Duffer', 'Durham - USA', 36);

INSERT INTO Regista (data\_nascita, foto, nome, cognome, paese\_nascita, eta) VALUES ('1984-02-15', null, 'Ross', 'Duffer', 'Durham - USA', 36);

INSERT INTO Diretto (id\_film, id\_regista) VALUES (2, 1);

INSERT INTO Cinema (nome, indirizzo, regione, provincia, contatti) VALUES ('Lux', 'Torino, Galleria S. Federico 33', 'Piemonte', 'Torino', '0115628907');

INSERT INTO Proiettato (id\_film, nome, indirizzo) VALUES (2, 'Lux', 'Galleria S. Federico 33');

INSERT INTO Sala (numero\_sala, nome, indirizzo) VALUES (1, 'Lux', 'Galleria S. Federico 33');

INSERT INTO Proietta (numero\_sala, nome, indirizzo, id\_film, data, ora) VALUES (1, 'Lux', 'Galleria S. Federico 33', 2, '2020-07-13', '19:00');

INSERT INTO Proiezione (id\_film, data, ora, prezzo) VALUES (2, '2020-07-13', '19:00', '7,50');

INSERT INTO SerieTV (id, piattaforma, distribuzione, fotografia, produzione, musiche, data\_di\_uscita)

```
VALUES (1, 'Netflix', 'Netflix', 'Matt Duffer - Ross Duffer', 'CampHero Productions - 21 Laps Entertainment', 'Kyle Dixon - Michael Stein', '2016-07-15');
```

```
INSERT INTO Stagione (id, numero_stagione) VALUES (1, 1);
```

```
INSERT INTO Episodio (id_serie_tv, numero_stagione, titolo_episodio, durata, sceneggiatore) VALUES (1, 1, 'La scomparsa di Will Byers', 47, 'Matt Duffer - Ross Duffer');
```

```
INSERT INTO Regia (titolo_episodio, id_serie_tv, numero_stagione, id_regista) VALUES ('La scomparsa di Will Byers', 1, 1, 2), ('La scomparsa di Will Byers', 1, 1, 3);
```

## Operazioni di eliminazione e modifica dei dati

```
DELETE
FROM utente
WHERE username = 'salvo_96';
```

Eliminando l'username 'salvo\_96' dalla tabella Utente, e di conseguenza la tupla ad esso associata, le relazioni Preferisce, Vota verranno svuotate delle tuple che contengono come una delle due FK l'username 'salvo\_96'. Nonostante ciò, i voti espressi da tale 'salvo\_96' saranno presenti all'interno del DB.

Altro esempio di DELETE di un utente:

```
DELETE
FROM utente
WHERE username = 'andreaBoetto';
```

Alcune operazioni di UPDATE

```
UPDATE Contenuto
SET titolo = 'Stranger Things - Prima Stagione' WHERE titolo = 'Stranger Things';
```

Grazie a ON UPDATE CASCADE, le tuple delle relazioni che coinvolgono il contenuto con titolo ad esempio Recita, farà sì che il titolo originario 'Stranger Things' venga modificato: il titolo infatti verrà cambiato in 'Stranger Things - Prima Stagione'.

Altro esempio di UPDATE

```
UPDATE Cinema
SET nome = 'Cinema The Lux'
WHERE nome = 'Lux' AND indirizzo = 'Torino, Galleria S.Federico 33';
```