

## Verifica 14/10

### Modello logico

Utente(ID, nome, cognome, data\_inizio\_abb, data\_fine\_abb, sesso, nazione\_di\_residenza, data\_di\_nascita)  
Episodio( ID, titolo, durata, \*\*IDserie)  
Attore( ID, nome, cognome)  
Visualizza( \*\*IDutente, \*\*IDepisodio, data, valutazione)  
Valuta( \*\*IDutente, \*\*IDserie, valutazione)  
Recita( \*\*IDepisodio, \*\*IDattore, nome\_personaggio, cognome\_personaggio)  
Serie( ID, nome, trama, stagioni)  
Genere( ID, nome )  
Appartiene( \*\*IDgenere, \*\*IDserie)

SQL:

1 - Gradimento medio episodio 'La scatola del bravo ragazzo' della serie tv Dahmer  
Select avg(visualizza.valutazione) as gradimento\_medio  
from episodio inner join serie\_tv  
on episodio.idserie = serie\_tv.id  
inner join visualizza  
on visualizza.idepisodio = episodio.id  
where serie\_tv.nome = 'Dahmer'  
and episodio.titolo = 'La scatola del bravo ragazzo'

select avg(valutazione) from visualizza  
where idepisodio = (select id from episodio where nome = 'La scatola del bravo ragazzo' and idserie = (select id from serie where nome = 'Dahmer'))

2 - elenco serie tv con il numero degli episodi  
select serie.nome, count(\*) as numero  
from serie inner join episodio  
on serie.id = episodio.idserie  
group by serie.id, serie.nome, numero

3 - utenti che hanno guardato più di 100 episodi  
select utente.nome, utente.cognome, count(\*) as numero\_episodi  
from utente inner join visualizza  
on utente.id = visualizza.idutente  
group by utente.id, utente.nome, utente.cognome  
having count(\*) > 100  
order by numero\_episodi desc

4 - utenti che hanno guardato episodi di più di 10 serie diverse

```
select utente.id, utente.nome, utente.cognome, count(serie.id) as serie_viste
from utente inner join visualizza
on utente.id = visualizza.idutente
inner join episodio
on episodio.id = visualizza.idepisodio
inner join serie
on serie.id = episodio.idserie
group by utente.id, utente.nome, utente.cognome
having count(serie.id) >= 10
```

5 - elenco utenti che nell'ultimo mese non hanno guardato la serie tv Dahmer

```
select * from utente
where utente.id not in (select utente.id from utente
inner join visualizza
on utente.id = visualizza.idutente
inner join episodio
on episodio.id = visualizza.idepisodio
inner join serie
on serie.id = episodio.idserie
where serie.nome = 'Dahmer'
and datediff (day, visualizza.data, getdate()) < 30)
```

6 - avere l'elenco delle serie di genere drammatico e fantascientifico ordinare in ordine alfabetico visualizzando prima le serie di fantascienza e poi quelle drammatiche

```
select * from serie
inner join appartiene
on serie.id = appartiene.idserie
inner join genere
on genere.id = appartiene.idgenere
where genere.nome = 'drammatico' or genere.nome = 'fantascienza'
– where genere.nome in ('drammatico', 'fantascienza')
order by genere.nome desc, serie.nome
```

7 - avere l'elenco degli episodi(titolo e serie tv) in cui ha recitato Flavio Nedina,

```
select episodio.titolo, serie.nome from attore
inner join recita
on attore.id = recita.idattore
inner join episodio
on episodio.id = recita.idepisodio
inner join serie
on serie.id = episodio.idserie
```

where attore.nome = 'Flavio' and attore.cognome = 'Nedina'

avere l'elenco delle serie in cui compare la parola 'the' all'inizio del nome della serie  
tv

```
select * from serie  
where nome like 'the %'
```

8 - avere il numero di utenti che hanno guardato episodi nel settembre 2022

```
select count(*) as numero_utenti from utente  
inner join visualizza  
on utente.id = visualizza.idutente  
where visualizza.data between '2022-09-01' and '2022-09-30'
```

9 - creare una tabella che rappresenta un'entità

```
create table Genere(  
id int identity(0,1) primary key,  
nome varchar(50)  
)
```

10 - creare una tabella che rappresenta una relazione collegata all'entità precedente

```
create table appartiene(  
idserie int foreign key references serie(id),  
idgenere int foreign key references genere(id)  
)
```

11 - aggiungere il campo numero di telefono all'utente

```
alter table utente  
add telefono varchar(20)
```

12 - eliminare l'attributo sesso dall'utente

```
alter table utente  
drop column sesso
```

13 - eliminare tutti gli utenti a cui è scaduto l'abbonamento

```
delete from utente  
where data_fine_abb < getdate()
```

14 - modificare la valutazione degli episodi passando da un voto da 1 a 10 a un voto da 1 a 5 stelle

```
alter table visualizza  
alter column visualizza decimal(2,1)  
update visualizza  
set valutazione = valutazione / 2  
update visualizza
```

set valutazione = 1  
where valutazione = 0.5

1	2	3	4	5	6	7	8	9	10
1	1	1.5	2	2.5	3	3.5	4	4.5	5