

Cognome e nome: _____ Matricola: _____ Turno: _____

Riportare sui fogli i seguenti dati: cognome, nome, matricola e turno di laboratorio.

Esame di SQL

Punteggi massimi:

- Domande 1 e 2 svolte perfettamente: 23;
- Domande 1 e 3 svolte perfettamente: 25;
- Domande 2 e 3 svolte perfettamente: 28;
- Domande 1, 2 e 3 svolte perfettamente: 33.

Lo svolgimento corretto di una sola domanda non permette il raggiungimento della sufficienza.

Le seguenti relazioni definiscono una base di dati “**NerdFlix**” per gestire una piattaforma di streaming online di serie TV. Gli attributi sottolineati sono le chiavi primarie delle relazioni.

SERIETV(NomeSerie, Produttore, Nazione, Genere)

EPISODIO(NomeSerie, Stagione, Episodio, TitoloEpisodio, Anno)

AUDIO(NomeSerie, Stagione, Episodio, Lingua, AdattatoreDialoghi*, Originale)

SOTTOTITOLI(NomeSerie, Stagione, Episodio, Lingua)

Vincoli di integrità referenziale:

EPISODIO(NomeSerie) referencia SERIETV(NomeSerie),

AUDIO(NomeSerie,Stagione,Episodio) referencia EPISODIO(NomeSerie,Stagione,Episodio),

SOTTOTITOLI(NomeSerie,Stagione,Episodio) referencia EPISODIO(NomeSerie,Stagione,Episodio).

Genere può assumere i valori “Animazione”, “Dramma”, “Fantascienza” e “Commedia”. Originale può assumere i valori True (se l’audio è quello della lingua originale) e False (se si tratta di un doppiaggio). I rimanenti attributi sono autoesplicativi. AdattatoreDialoghi può essere NULL (indicato con il simbolo *).

A titolo di esempio, la seguente tupla della relazione EPISODIO

<'Stranger Things', 3, 2, 'Capitolo due - Incubi', '2019'>

si riferisce al secondo episodio della terza stagione della serie TV “Stranger Things”. **Non è detto che tutti gli episodi (o che tutte le stagioni) di una serie siano presenti nella base di dati.**

Con riferimento alla base di dati "NerdFlix" esprimere in SQL le seguenti interrogazioni. Commentare ogni soluzione proposta spiegando le varie parti.

Domanda 1 (bassa complessità).

Trovare le serie TV tedesche di genere “Fantascienza” con almeno un episodio sottotitolato in inglese. Mostrare le serie TV in ordine alfabetico crescente del nome della serie. NON usare sottoquery.

Soluzione 1.

```
SELECT DISTINCT se.NomeSerie
FROM serietv se JOIN sottotitoli so ON (se.NomeSerie=so.NomeSerie)
WHERE se.Genere='Fantascienza' AND se.Nazione='Germania' AND so.Lingua='Inglese'
ORDER BY se.NomeSerie;
```

Domanda 2 (media complessità).

Mostrare le serie TV di cui sono presenti nel database almeno 5 stagioni aventi ciascuna almeno un episodio con

audio originale in spagnolo. NON usare sottoquery.

Soluzione 2.

```
SELECT e.NomeSerie
FROM episodio e JOIN audio a ON (e.NomeSerie=a.NomeSerie
                                AND e.Stagione=a.Stagione AND e.Episodio=a.Episodio)
WHERE a.Lingua='Spagnolo' AND a.Originale='true'
GROUP BY e.NomeSerie
HAVING COUNT(DISTINCT e.Stagione)>=5;
```

Domanda 3 (alta complessità).

Tra le serie TV che **non hanno** il numero massimo di stagioni presenti nel database, trovare quella (o quelle) con più episodi. Non usare costrutti come LIMIT, ROWNUM, TOP.

Soluzione 3.

```
WITH NumStagioniSerieTV AS (
    SELECT e.NomeSerie, COUNT(distinct e.Stagione) NumStagioni
    FROM episodio e
    GROUP BY e.NomeSerie),
NumEpisodiSerieTVEscluseMaxNumStagioni AS (
    SELECT e.NomeSerie, COUNT(*) NumEpisodiSTV
    FROM episodio e
    WHERE e.NomeSerie IN (SELECT NomeSerie
                          FROM NumStagioniSerieTV
                          WHERE NumStagioni < ANY (SELECT NumStagioni
                                                    FROM NumStagioniSerieTV))
    GROUP BY e.NomeSerie)
SELECT NomeSerie
FROM NumEpisodiSerieTVEscluseMaxNumStagioni
WHERE NumEpisodiSTV = (SELECT MAX(NumEpisodiSTV)
                      FROM NumEpisodiSerieTVEscluseMaxNumStagioni);
```

Esame di Teoria

Domanda 1 (14 punti).

Con riferimento alla base di dati “NerdFlix”:

- A. Esprimere in algebra relazionale l'interrogazione
Elencare le serie TV sottotitolate^(*) in tutte le lingue^().**
- B. Esprimere, nel calcolo relazione su tuple con dichiarazione di range, la seguente domanda:
Elencare nome e produttore delle serie TV drammatiche per cui esistono sia i sottotitoli^(*) in spagnolo che in inglese^().**

(*) Per considerare una serie sottotitolata, è sufficiente che esistano i sottotitoli di un solo episodio della serie.

(**) I sottotitoli in lingue diverse possono riguardare episodi diversi della stessa serie (cioè, non è necessario che un singolo episodio sia sottotitolato in tutte le lingue).

Soluzione 1.

A. Una possibile soluzione in algebra relazionale è la seguente:

$$\pi_{NomeSerie, Lingua}(sottotitoli) \div \pi_{Lingua}(sottotitoli)$$

B. Una possibile soluzione è la seguente:

$$\{s.NomeSerie, s.Produuttore \mid s(SERIETV) \mid s.Genere='Dramma' \wedge \\ \exists st(SOTTOTITOLI)(st.NomeSerie=s.NomeSerie \wedge st.Lingua='Spagnolo') \wedge \\ \exists st'(SOTTOTITOLI)(st'.NomeSerie=s.NomeSerie \wedge st'.Lingua='Inglese')\}$$

Domanda 2 (9 punti).

Dati $R(\text{Matricola}, \text{CorsoLaurea}, \text{CodCorso}, \text{Crediti})$ e $F = \{\text{Matricola} \rightarrow \text{CorsoLaurea}, \text{CodCorso} \rightarrow \text{Crediti}\}$ dire se R è in 3FN motivando la risposta e, se non lo è, decomporla mostrando tutti i passaggi. Dire motivando se l'eventuale risultato è BCNF.

Soluzione 2.

Per prima cosa occorre individuare la chiave della relazione R . Si nota immediatamente che una possibile superchiave è $\{\text{Matricola}, \text{CodCorso}\}$, che è anche l'unica chiave candidata.

Nessuna delle due d.f. dell'insieme F è quindi in 3FN (Matricola e CodCorso da soli non sono chiave e i due attributi CorsoLaurea e Crediti non sono parte della chiave).

Si cerca quindi l'insieme di copertura minimale di F , che in questo caso coincide con F stesso (non ci sono né attributi estranei né d.f. ridondanti).

Si procede quindi alla normalizzazione, ottenendo le due relazioni:

$R1(\underline{\text{Matricola}}, \text{CorsoLaurea})$

$R2(\underline{\text{CodCorso}}, \text{Crediti})$

a cui si deve aggiungere la relazione contenente la chiave:

$R3(\underline{\text{Matricola}}, \underline{\text{CodCorso}})$.

La decomposizione è anche BCNF perché tutte le d.f. sono di tipo superchiave.

Domanda 3 (10 punti).

Con riferimento alla base di dati “NerdFlix” e i seguenti dati quantitativi:

$$\text{CARD}(\text{SERIETV}) = 1000$$

$$\text{CARD}(\text{EPISODIO}) = 20000$$

$$\text{VAL}(\text{Anno}, \text{EPISODIO}) = 50$$

$$\text{VAL}(\text{Stagione}, \text{EPISODIO}) = 20$$

disegnare gli alberi sintattici prima e dopo l’ottimizzazione logica e calcolare il numero di tuple “mosse” prima e dopo l’ottimizzazione logica della seguente query:

$\sigma_{\text{Anno}='2021' \wedge \text{Genere}='Dramma' \wedge \text{Stagione}='1'}(\text{episodio} \bowtie_{\text{episodio.NomeSerie}=\text{serietv.NomeSerie}} \text{serietv})$

Soluzione 3.

La query ottimizzata dividendo la selezione e portandola verso le foglie è:

$(\sigma_{\text{Anno}='2021' \wedge \text{Stagione}='1'}(\text{episodio})) \bowtie_{\text{episodio.NomeSerie}=\text{serietv.NomeSerie}} (\sigma_{\text{Genere}='Dramma'}(\text{serietv}))$

Prima dell’ottimizzazione:

- Costo $r_1 = (\text{episodio} \bowtie_{\text{episodio.NomeSerie}=\text{serietv.NomeSerie}} \text{serietv})$: $1000 \cdot 20000 = 20\,000\,000$.
- Cardinalità $|r_1| = \text{CARD}(\text{EPISODIO}) = 20\,000$ (equijoin attraverso la chiave esterna)
- Costo della selezione: $|r_1|$
- Costo totale = $20\,000\,000 + 20\,000 \sim 2 \cdot 10^7$.

Dopo l’ottimizzazione:

- Costo $\sigma_1 = \sigma_{\text{Anno}='2021' \wedge \text{Stagione}='1'}(\text{episodio}) = \text{CARD}(\text{episodio}) = 20\,000$
- Costo $\sigma_2 = \sigma_{\text{Genere}='Dramma'}(\text{serietv}) = 1\,000$
- Tuple prodotte dalla selezione $|\sigma_1| = 1/\text{VAL}(\text{Anno}, \text{EPISODIO}) \cdot 1/\text{VAL}(\text{Stagione}, \text{EPISODIO}) \cdot \text{CARD}(\text{EPISODIO}) = 1/50 \cdot 1/20 \cdot 20\,000 = 20$
- Tuple prodotte dalla selezione $|\sigma_2| = 1/\text{VAL}(\text{Genere}, \text{SERIETV}) \cdot \text{CARD}(\text{SERIETV}) = 1/4 \cdot 1000 = 250$
- Costo join $r = \sigma_1 \bowtie_{\text{episodio.NomeSerie}=\text{serietv.NomeSerie}} \sigma_2 = 20 \cdot 250 = 5\,000$.
- Costo totale = $20\,000 + 1\,000 + 5\,000 = 26\,000 = 2,6 \cdot 10^4$.