

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, Regista, Anno)

AUDIO(NomeSerie, Stagione, Episodio, Lingua, 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”, “Commedia” e “Documentario”. “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.

A titolo di esempio, la tupla della relazione EPISODIO <'Mercoledì', 1, 6, 'Un triste compleanno', '2022'> si riferisce al sesto episodio della prima stagione della serie TV “Mercoledì”. **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.

Domanda 1 (bassa complessità).

Elencare le serie TV prodotte da J.J. Abrams con almeno un episodio uscito tra il 2004 e il 2010 e doppiato in una lingua diversa da quella originale. Ordinare i risultati per nome della serie, in ordine alfabetico decrescente.

Soluzione 1.

```
SELECT DISTINCT s.NomeSerie
FROM   serietv s JOIN episodio e ON (s.NomeSerie=e.NomeSerie) JOIN
      audio a ON (e.NomeSerie=a.NomeSerie AND e.Stagione=a.Stagione AND
      e.Episodio=a.Episodio)
WHERE  s.Produttore='J.J. Abrams' AND e.Anno>=2004 AND e.Anno<=2010 AND
      a.Originale=False
ORDER BY s.NomeSerie DESC;
```

Domanda 2 (media complessità).

Senza usare sottointerrogazioni o operatori insiemistici, mostrare le serie TV di cui è stata prodotta una sola

stagione e in quell'unica stagione gli episodi sono diretti tutti dallo stesso regista e sottotitolati in almeno tre lingue (anche per episodi diversi). Ordinare le serie TV per anno crescente dell'ultimo episodio prodotto (si può ipotizzare che l'ultimo episodio sia sempre il più recente della serie).

Soluzione 2.

```
SELECT e.NomeSerie, COUNT(*) AS N_Episodi
FROM episodio e JOIN sottotitoli s ON (e.NomeSerie=s.NomeSerie
                                AND e.Stagione=s.Stagione AND e.Episodio=s.Episodio)
GROUP BY e.NomeSerie
HAVING COUNT(DISTINCT e.Stagione)=1 AND COUNT(DISTINCT e.Regista)=1 AND
COUNT(DISTINCT s.Lingua)>=3
ORDER BY MAX(e.Anno);
```

Domanda 3 (alta complessità).

Tra i registi/le registe che hanno diretto solo episodi di serie TV prodotte in Germania, trovare quelli o quelle con il maggior numero di episodi diretti in Serie TV di cui sono anche produttori/produttrici (non utilizzare costrutti come LIMIT, ROWNUM, TOP).

Soluzione 3.

```
WITH EpisodiRegistiGermania AS (
    SELECT e.Regista, e.NomeSerie, e.Stagione, e.Episodio
    FROM episodio e JOIN serietv s ON (e.NomeSerie=s.NomeSerie)
    WHERE e.Regista NOT IN (
        SELECT Regista
        FROM episodio e JOIN serietv s ON (e.NomeSerie=s.NomeSerie)
        WHERE s.Nazione<>'Germania'
    )),
NumEpisodiRegProd AS (
    SELECT e.Regista, e.NomeSerie, COUNT(*) NumEpisodi
    FROM serietv s JOIN EpisodiRegistiGermania e ON (s.NomeSerie=e.NomeSerie)
    WHERE s.Produttore=e.Regista
    GROUP BY e.Regista, e.NomeSerie)
SELECT Regista
FROM NumEpisodiRegProd
WHERE NumEpisodi = (SELECT MAX(NumEpisodi)
                    FROM NumEpisodiRegProd);
```

Esame di Teoria (rispondere su fogli separati rispetto alla parte di SQL)

Domanda 1 (9 punti).

Con riferimento alla base di dati “NerdFlix”:

- A. (5 punti) Esprimere in Algebra Relazionale l'interrogazione
Elencare le serie TV doppiate^(*) in tutte le lingue^()(^{***}) tranne che in inglese.**
- B. (4 punti) Esprimere, nel calcolo dei predicati su tuple con dichiarazione di range, la seguente domanda:
Elencare nome e nazione delle serie TV drammatiche dirette interamente dalla stessa persona che le ha prodotte.

(*) è sufficiente che esista il doppiaggio di un solo episodio della serie

(**) il doppiaggio può riguardare anche episodi diversi della stessa serie

(***) si può assumere che per ogni lingua ci sia almeno una tupla nella relazione AUDIO

Soluzione 1.

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

$$(\pi_{NomeSerie, Lingua}(audio) \div \pi_{Lingua}(\sigma_{Lingua <> 'Inglese'}(audio))) \\ - \pi_{NomeSerie}(\sigma_{Lingua = 'Inglese'}(audio))$$

B. Una possibile soluzione è la seguente:

$$\{s.NomeSerie, s.Nazione \mid s(SERIETV) \mid s.Genere = 'Dramma' \wedge \forall e(EPISODIO)(e.NomeSerie = s.NomeSerie \Rightarrow (e.Regista = s.Produttore))\}$$

Domanda 2 (9 punti).

- A. Enunciare il teorema/regola del prodotto.
- B. Dimostrare il teorema/regola di decomposizione (ovvero, dato $X \rightarrow YZ$, allora $X \rightarrow Y$ e $X \rightarrow Z$).
- C. 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.

Soluzione 2.

Per i punti A e B, si vedano gli appunti/testo/slide.

- C. Per prima cosa occorre individuare la chiave della relazione R . Si nota immediatamente che una possibile superchiave è $\{\text{Matricola}, \text{CodCorso}\}$. Essa 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 CodCorso 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:

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

Domanda 3 (7 punti).

Con riferimento alla base di dati “NerdFlix” e tenendo conto dei seguenti dati quantitativi:

CARD(serietv) = 1000

CARD(episodio) = 10000

VAL(Regista, episodio) = 2000

VAL(Produttore, serietv) = 200

MIN(Anno,episodio) = 1972

MAX(Anno,episodio) = 2022

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:

$\sigma_{\text{Regista}='Tim\ Burton'\wedge\text{Produttore}='Tim\ Burton'\wedge\text{Anno}>2002}$ (episodio
 \bowtie EPISODIO.NomeSerie=SERIETV.NomeSerie serietv)

Soluzione 3.

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

$(\sigma_{\text{Anno}>2002\wedge\text{Regista}='Tim\ Burton'}(\text{episodio}))\bowtie \text{EPISODIO.NomeSerie}=\text{SERIETV.NomeSerie} (\sigma_{\text{Produttore}='Tim\ Burton'}(\text{serietv}))$

Prima dell'ottimizzazione:

- Costo $r_1 = (\text{episodio} \bowtie \text{EPISODIO.NomeSerie}=\text{SERIETV.NomeSerie serietv})$: $10^3 \cdot 10^4 = 10^7$.
- Cardinalità $|r_1| = \text{CARD}(\text{episodio}) = 10^4$ (equijoin attraverso la chiave esterna)
- Costo della selezione = $|r_1|$
- Costo totale = $10^7 + 10^4 \sim 10^7$.

Dopo l'ottimizzazione:

- Costo $\sigma_1 = \sigma_{\text{Anno}>2002\wedge\text{Regista}='Tim\ Burton'}(\text{episodio}) = \text{CARD}(\text{episodio}) = 10^4$
- Costo $\sigma_2 = \sigma_{\text{Produttore}='Tim\ Burton'}(\text{serietv}) = 10^3$
- Tuple prodotte dalla selezione $|\sigma_1| = (\text{MAX}(\text{Anno,episodio}) - 2002) / (\text{MAX}(\text{Anno,episodio}) - \text{MIN}(\text{Anno,episodio})) \cdot 1/\text{VAL}(\text{Regista,episodio}) \cdot \text{CARD}(\text{episodio}) = 20/50 \cdot 1/2000 \cdot 10^4 = 2$
- Tuple prodotte dalla selezione $|\sigma_2| = 1/\text{VAL}(\text{Produttore,serietv}) \cdot \text{CARD}(\text{serietv}) = 1/200 \cdot 1000 = 5$
- Costo join $r = \sigma_1 \bowtie_{\text{episodio.NomeSerie=serietv.NomeSerie}} \sigma_2 = 2 \cdot 5 = 10$.
- Costo totale = $10^4 + 10^3 + 10 = 10^4$

Domanda 4 (8 punti).

Si consideri un file di log L con il seguente contenuto in seguito ad un crash:

```
<T1, START>;
<T2, START>;
<T1, BS(t1[A], 5), AS(t1[A], 10)>;
<T2, BS(t2[B], 3), AS(t2[B], 5)>;
<T3, START>;
<T2, COMMIT>;
-- checkpoint --;
<T3, BS(t3[C], 3), AS(t3[C], 5)>;
<T1, COMMIT>
```

crash!

- A.** Indicare in modo dettagliato il contenuto del record di checkpoint.
- B.** Descrivere l'algoritmo di ripristino e mostrarne l'esecuzione sul log L

Soluzione 4.

- A.** Il record di checkpoint conterrà la lista delle transazioni non terminate (quindi T1 e T3) con i relativi puntatori ai rispettivi $\langle T_i, \text{START} \rangle$. Contiene inoltre l'OK riguardante la buona riuscita delle operazioni previste dal checkpoint, quindi: $[(T1, p1); (T3, p3); OK]$.
- B.** L'algoritmo di ripristino recupera la lista di transazioni LC in commit dopo l'ultimo checkpoint e prima del crash (solo T1) e la lista delle transazioni LA attive al momento del crash (quindi solo T3). Viene effettuato quindi l'UNDO di T3: viene ripristinato a '3' il valore dell'attributo C della tupla t3 (si effettua contestualmente anche un FORCE della pagina). Successivamente, il gestore del ripristino esegue il REDO di T1: viene reso persistente il valore 10 dell'attributo A della tupla t1 (con conseguente FORCE della pagina).