

- Dato lo schema:

Escursione(id, titolo, descrizione, durata, difficoltà, costo)

DataEscursione(id, data, idescursione, id guida)

Partecipante(idpartecipante, idescursione)

Persona(id, nome, cognome)

- Indicare le chiavi primarie ed esterne dello schema e le relazioni esistenti tra le tabelle .

Escursione(id, titolo, descrizione, durata, difficoltà, costo)

DataEscursione(id, data, idescursione, idguida)

Partecipante(idpartecipante, idescursione)

Persona(id, nome, cognome)

- Trovare le escursioni (indicando titolo, descrizione e difficoltà) che hanno un costo massimo

Escursione(id, titolo, descrizione, durata, difficoltà, costo)

DataEscursione(id, data, idescursione, id guida)

Partecipante(idpartecipante, idescursione)

Persona(id, nome, cognome)

$\pi_{titolo, descrizione, durata, difficoltà} (Escursione)$

—

$$\pi_{titolo, descrizione, durata, difficoltà} \left(\sigma_{costo1 > costo} \left(\delta_{\substack{id1 \leftarrow id, \\ titolo1 \leftarrow titolo, \\ descrizione1 \leftarrow descrizione, \\ durata1 \leftarrow durata, \\ difficoltà1 \leftarrow difficoltà, \\ costo1 \leftarrow costo}} (Escursione) \times Escursione \right) \right)$$

- Trovare i partecipanti (dando nome e cognome in output) che hanno partecipato a tutte le escursioni

Escursione(id, titolo, descrizione, durata, difficoltà, costo)

Partecipante(idpartecipante, idescursione)

Persona(id, nome, cognome)

DataEscursione(id, data, idescursione, id guida)

Algebra

- Trovare le guide che non hanno mai partecipato ad escursioni di difficoltà massima

-

Escursione(id, titolo, descrizione, durata, difficoltà, costo)

Partecipante(idpartecipante, idescursione)

Persona(id, nome, cognome)

DataEscursione(id, data, idescursione, id guida)

ALGEBRA

- Trovare le coppie di persone che hanno partecipato sempre alle stesse escursioni

$R1 = PARTECIPANTE$

$R2 = PARTECIPANTE$

$$R3 = \pi_{R1.idp, R2.idp, R1.ide} \left(R1 \bowtie_{\substack{R1.ide=R2.ide \\ \wedge R1.idp > R2.idp}} R2 \right)$$

$$R5 = \pi_{R1.idp, R2.idp, R1.ide} \left(R1 \bowtie_{\substack{R1.ide \neq R2.ide \\ \wedge R1.idp > R2.idp}} R2 \right) - R3$$

$$R6 = \pi_{R1.idp, R2.idp, R2.ide} \left(R1 \bowtie_{\substack{R1.ide \neq R2.ide \\ \wedge R1.idp > R2.idp}} R2 \right) - R3$$

$$\pi_{R1.idp, R2.idp}(R3) - (\pi_{R1.idp, R2.idp}(R5) \cup \pi_{R1.idp, R2.idp}(R6))$$

- Dire ogni accompagnatore quante escursioni ha guidato;

- Si consideri lo schema di base di dati sulle relazioni:
 - MATERIE (Codice, Facoltà, Denominazione, Professore)
 - STUDENTI (Matricola, Cognome, Nome, Facoltà)
 - PROFESSORI (Matricola, Cognome, Nome)
 - ESAMI (Studente, Materia, Voto, Data)
 - PIANIDISTUDIO (Studente, Materia, Anno)

- Formulare in algebra relazionale ed in SQL le seguenti query:
 1. gli studenti che hanno riportato in almeno un esame una votazione pari a 30, mostrando , per ciascuno di essi, nome e cognome e data della prima di tali occasioni;
 2. per ogni insegnamento della facoltà di ingegneria, gli studenti che hanno superato l' esame nell'ultima seduta svolta;
 3. gli studenti che hanno superato tutti gli esami previsti dal rispettivo piano di studio;
 4. per ogni insegnamento della facoltà di lettere, lo studente (o gli studenti) che hanno superato l'esame con il voto più alto;
 5. gli studenti che hanno in piano di studio solo gli insegnamenti della propria facoltà;
 6. nome e cognome degli studenti che hanno sostenuto almeno un esame con un professore che ha il loro stesso nome proprio.

- Si consideri lo schema relazionale composto dalle seguenti relazioni:
 - PROFESSORI (Codice, Cognome, Nome)
 - CORSI (Codice, Denominazione, Professore)
 - STUDENTI (Matricola, Cognome, Nome)
 - ESAMI (Studente, Corso, Data, Voto)
- Formulare le espressioni dell'algebra che producano:
 1. Gli esami superati dallo studente Pico della Mirandola (supposto unico), con indicazione, per ciascuno, della denominazione del corso, del voto e del cognome del professore;
 2. i professori che tengono due corsi (e non più di due), con indicazione di cognome e nome del professore e denominazione dei due corsi.