

# Basi di dati

*ale-cci*

## 1 Definizioni Generali

Una **chiave** è un gruppo di attributi che identifica univocamente una relazione, essa è legata allo schema di una relazione, non ai valori in esso contenuti. Una **superchiave minimale** è l'insieme di attributi di cui nessun sottogruppo di esso può essere considerato come **chiave**.

Ogni relazione per definizione ha una chiave.

Una chiave viene detta **primaria** se non può assumere valori nulli.

Un **Vincolo di integrità relazionale** fra  $X$  attributi di  $R_1$  e  $R_2$  è soddisfatto se tutte le tuple "istanze" di  $X$  compaiono come chiave primaria su  $R_2$

## 2 Algebra relazionale

- *Union* ( $r_1 \cup r_2$ ), *Intersezione*, *Differenza* tra insiemi
- *Ridenominazione* (AS), *Selezione* (Where), *Proiezione* (Select)
- Join Naturale, Theta Join

## 3 Selezioni ( $\sigma$ )

Data una relazione  $r$  ed una condizione da soddisfare  $F$ , la selezione produce una nuova relazione, formata dal sottinsieme delle tuple di  $r$  che soddisfano  $F$ .

Si indica con  $\sigma_F(r)$  o  $SEL_F(r)$ , dove  $F$  è una condizione da verificare ed  $r$  la relazione a cui è applicata la selezione.

In MySql la selezione è l'operatore **WHERE**

## 4 Proiezioni ( $\Pi$ )

Data una relazione  $r(X)$  e  $Y \subseteq X$ , la proiezione di  $r$  su  $Y$  si indica con:

$$\Pi_Y(r) \text{ oppure } PROJ_Y(r)$$

Ed è l'insieme di tuple su  $Y$  ottenute dalle tuple di valori di  $r$ .

Una proiezione ha un numero di tuple **minore o uguale** rispetto alla relazione a cui è applicata.

Il numero tuple è **uguale** se e solo se  $Y$  è superchiave per  $r$

In mysql le proiezioni sono rappresentate attraverso i parametri specificati a select

In MySql le proiezioni sono le colonne selezionate attraverso l'operatore **SELECT**

## 5 Join ( $\Join$ )

$$r_1 \Join r_2 = \{t \text{ su } X_1 X_2 \mid t[X_1] \in r_1 \wedge t[X_2] \text{ in } r_2\}$$

Ottenuto concatenando ogni tupla della prima relazione con ogni tupla della seconda, in cui gli attributi comuni hanno lo stesso valore.

$$\prod_{X_1}(R_1 \diamond R_2) \subseteq R_1$$

$$\prod_{X_2}(R_1 \diamond R_2) \subseteq R_2$$

Proiettando i risultati del Join su una delle due relazioni, ne ottengo un sottinsieme, quindi se tento di ricostruire la relazione iniziale ho una perdita di informazione.

Scomponendo lo schema di una relazione  $R$  in due sottoschemi tale che  $R(X), X = X_1 \cup X_2$ , si ha che  $R$  è contenuta nella relazione, risultato del *join* fra le proiezioni di  $R$ .

Il **Theta-Join** viene usato per correlare attributi con nome diverso (i.e.  $X_1 \cap X_2 = \emptyset$ )

$r_1 \diamond_F r_2 = \sigma_F(r_1 \diamond r_2)$  con  $F$  Attributo da selezionare, se  $F$  è una condizione di uguaglianza fra attributi, allora viene detto **Equi-Join**

## 6 Interrogazioni (Query)

Una interrogazione è una funzione  $E(r)$  che applicata ad istanze  $r$  di una base di dati produce una relazione su un dato insieme di attributi  $X$ .

In altre parole L'applicazione di una Selezione insieme a Proiezione e/o Join.

## 7 Viste

Diverse rappresentazioni per gli stessi dati, si dividono in:

- Relazioni di base: hanno un contenuto autonomo, originariamente contenuto nella base di dati
- Relazioni derivate: Relazioni il cui contenuto è funzione del contenuto di altre relazioni
- Relazioni Virtuali: Definite attraverso query scritte, non realmente presenti all'interno del database, ma utilizzabili come se lo fossero. Devono essere rivalutate ogni volta.
- viste materializzate: Relazioni virtuali inserite dentro il database. Sono immediatamente disponibili, ma devono essere aggiornate insieme al cambiamento dei dati all'interno del database.

Vantaggi delle viste sono:

- Permettono di mostrare solo le parti del database interessate agli utenti
- Query molto complesse possono essere definite come viste, rendendole più leggibili
- È possibile definire diritti d'accesso anche per una singola vista
- Dato che forniscono un livello d'astrazione, in caso di modifiche allo schema del database, le vecchie relazioni possono essere ancora ricalcolate, consentendo l'uso di applicazioni che fanno riferimento al vecchio schema

**Dominio:** Tipo di dato

**Vincoli interrelazionali:** Vincoli che coinvolgono più tabelle foreign key, on delete (cascade, set null, no action)

**Vincoli intrarelazionali:** Vincoli applicati su campi, (es: not null, unique...)

## 8 Interrogazioni Nidificate (SubQuery)

Guarda dogmario SQL nella sezione SubQuery