

Basi di Dati e Sistemi Informativi I, 17 febbraio 2016

Adriano Peron

DIETI, Corso di Laurea in Informatica, Università di Napoli 'Federico II', Italy

E-mail: adrperon@unina.it

Si consideri il seguente schema relazionale che descrive un frammento della base di dati per gestire progetti Java e le successive versioni dei progetti. La tabella *CLASSE* descrive le classi ed in particolare il progetto a cui appartiene e il path del file che contiene il codice e la riga iniziale e finale delle definizioni della classe nel file. Ad una classe sono associati un insieme di metodi descritti nella tabella *METODO* come per le classi viene dato il path del file che contiene il codice del metodo e la riga iniziale e finale delle istruzioni del metodo. Classi e metodi sono identificati da un codice e da un numero di versione (un intero). L'attributo stato della versione ha possibili valori 'Updated' (il metodo viene modificato rispetto alla versione precedente), 'Deleted' (il metodo viene cancellato rispetto alla versione precedente), 'Added' (il metodo viene aggiunto rispetto alla versione precedente), 'Unchanged' (rimane invariato rispetto alla versione precedente). Per i metodi di ciascuna versione la tabella *SCRITTURA* indica gli autori dei metodi (associazione molti a molti). I metodi possono avere delle segnalazioni (malfunzionamenti, modifiche etc.) che hanno una data di apertura e una data di chiusura (parziale). La data di chiusura se presente coincide con la data di risoluzione del problema legata alla segnalazione.

CLASSE(*CodC*, *Versione*, *Progetto*, *Nome*, *Path*, *RigaI*, *RigaF*, *Stato*)
METODO(*CodM*, *Versione*, *Progetto*, *Nome*, *Path*, *RigaI*, *RigaF*, *Segnatura*, *CodC*, *Stato*)
SCRITTURA(*CodM*, *Versione*, *Data*, *CodA*)
SEGNALAZIONE(*CodS*, *DataApertura*, *CodM*, *Versione*, *DataChiusura*, *Tipo*, *Descrizione*)
ASSEGNAZIONE(*CodS*, *CodA*, *DataApertura*, *DataChiusura*, *Descrizione*)
AUTORE(*CodA*, *Nome*, *Cognome*, *Ruolo*)

Esercizio 01 (Punti 8) Si scriva una espressione in algebra relazionale che se valutata fornisce il nome delle classi che hanno mantenuti invariati (unchanged) tutti i loro metodi in tutte le versioni del progetto.

Esercizio 02 (Punti 8) Si implementino nel modo più adeguato i seguenti vincoli

1. Se per una classe esiste una istanza con $Versione > 0$, devono esistere istanze della classe per tutti i valori di versione compresi tra 0 e $Versione$.
2. Se la segnalazione riguardante un metodo è stata assegnata a un autore, questi deve essere anche l'autore del metodo.
3. La data di chiusura di una segnalazione se presente deve essere successiva alla data di apertura e la descrizione non deve essere nulla;
4. Non ci può essere più di una segnalazione aperta (*DataChiusura* a NULL) per un metodo.

Esercizio 03 (Punti 8) Si scriva un trigger che viene attivato all'inserimento di una segnalazione per un metodo in *SEGNALAZIONE*. Il trigger aggiunge nella tabella *ASSEGNAZIONE*, l'assegnazione della gestione della segnalazione all'autore del metodo (un metodo è in genere scritto da più autori) che ha meno assegnazioni di segnalazioni aperte da gestire al momento.

Esercizio 04 (Punti 8) Si scriva una funzione SQL che riceve in ingresso una lista di codici di classi separate dal carattere separatore '@' e utilizzando SQL DINAMICO restituisce una lista di codici e segnature di metodi delle classi passate come parametro che sono stati cancellati nell'ultima versione disponibile.