

Si consideri il seguente schema relazionale che descrive un frammento della base di dati per gestire depositi di immagini condivise tra utenti. Le foto sono contenute in album. Un album può contenere foto ed altri album (la struttura degli album è analoga a quella di un filesystem con file e cartelle). Ciascuna foto e ciascun album ha un proprietario (owner) espresso da un identificativo di utente. Nella tabella *ALBUM* l'attributo *InAlbum* è l'identificativo dell'album contenete (si assuma che un album alla radice della gerarchia di contenimento è contenuto nell'album di sistema di identificativo *SYSALBUM*). Gli attributi *Aggiunta* e *Rimossa* riportano la data di inserimento della foto e la eventuale data di rimozione (attributo parziale). Foto ed album possono avere dei tag associati. L'elenco dei tag ammissibili si trova nella tabella *HASHTAG*. I tag associati alle foto e agli album si trovano nelle tabelle *TAGFOTO* e *TAGALBUM* rispettivamente. La tabella *VISIBLE* indica invece l'accessibilità dei file ai vari utenti. Il proprietario *CodProp* dell'album *codA* concede il diritto di visualizzazione a *CodUt* su tutte le foto contenute in *codA*. Infine, nella tabella *LOG* vengono registrate tutte le operazioni fatte sulle foto dagli utenti: *codF* è la foto su cui si fa l'operazione, *CodU* l'utente che fa l'operazione ed *Operation* il tipo di operazione (ad es. inserimento, cancellazione, visualizzazione).

*HASHTAG*(*parola*)  
*FOTO*(*CodF*, *uri*, *nome*, *titolo*, *owner*, *CodAlbum*, *Aggiunta*, *Rimossa*)  
*ALBUM*(*CodA*, *nome*, *titolo*, *owner*, *InAlbum*)  
*UTENTE*(*CodU*, *nome*, *cognome*, *email*)  
*TAGFOTO*(*CodF*, *parola*)  
*TAGALBUM*(*CodA*, *parola*)  
*VISIBLE*(*CodProp*, *CodUt*, *CodA*)  
*LOG*(*CodF*, *CodU*, *Time*, *Operation*).

**Esercizio 05** (*Punti 10*) Si scriva una procedura *PLSQL* che riceve in ingresso l'identificativo di un album e che restituisce una stringa contenete tutti i tag associati all'album e agli album in esso contenuti (ad ogni livello di profondità) senza ripetizioni. Si consiglia di avvalersi di una tabella *TMP*(*CodA*) ( che si suppone già definita) dove memorizzare preventivamente l'albero degli album radicato nell'album passato come parametro.

**Esercizio 06** (*Punti 7*) Usando *SQL DINAMICO* si scriva una funzione che riceve in ingresso una lista di tag separati dal carattere @ e che restituisce una stringa degli uri delle foto (separati da @) a cui sono associati tutti i tag passati per parametro.