

Esercizio 1 (punti 8)

Si consideri il seguente schema relazionale, relativo alle gestione delle tesine da assegnare per la nuova laurea in Informatica:

TESINE(CodT, Titolo, Descrizione, CodD, Num_Tot_Stud)
DOCENTI(CodD, Nome, Cognome, Num_Ufficio, Tel, Email)
LAUREANDI(MatrS, Nome, Cognome, Email, CodT)
CONOSCENZE(CodCon, Descrizione)
RICHIEDE(CodT, CodCon)

Nello schema precedente, la tabella CONOSCENZE contiene la descrizione delle varie conoscenze che lo svolgimento delle tesine può richiedere. Il campo Num_Tot_Stud in TESINE indica il numero totale di studenti che svolgono la tesina in questione. Campi con lo stesso nome rappresentano chiavi e chiavi esterne. Il campo Descrizione in CONOSCENZE è chiave secondaria.

Si richiede di:

1. Progettare uno schema concettuale ad oggetti equivalente allo schema relazionale precedente, motivando le scelte effettuate.
2. Tradurre lo schema concettuale in uno schema logico relazionale ad oggetti per Oracle, motivando le scelte effettuate.
3. Presentare gli statement di creazione della parte di schema logico relativa all'entità TESINE utilizzando il sistema dei tipi di Oracle.
4. Supponendo che le tabelle contengano un certo numero di tuple, presentare gli statement SQL per eseguire le seguenti operazioni:
 - o Inserire tra le conoscenze della tesina identificata da CodT=14, la conoscenza "basi di dati testuali".
 - o Determinare nome e cognome dei docenti le cui tesine non sono state scelte da alcun studente.
 - o Determinare il numero medio di studenti che svolgono le tesine assegnate da Marina Ribaudò che richiedono come conoscenze "basi di dati".

Tipi in Oracle

```
CREATE TYPE Nome AS OBJECT  
( Dichiarazioni_attributi)  
[FINAL | NOT FINAL];
```

```
CREATE TYPE Nome UNDER Nome_superclasse  
( Dichiarazioni_attributi)  
[FINAL | NOT FINAL];
```

Default: NOT FINAL

```
CREATE TYPE Nome AS TABLE OF Tipo;
```

```
CREATE TYPE Nome AS VARRAY(Numero) OF Tipo;
```

```
CREATE TABLE Nome OF Nome_Tipo  
[({Vincoli chiave primaria, chiave esterna, not null, su campi nel tipo |  
SCOPE FOR NomeCampoREF is NomeTabella})]  
[NESTED TABLE NomeCampoNested STORE AS NomeTabella];
```

```
CREATE TABLE Nome  
(Dichiarazioni_attributi)  
[NESTED TABLE NomeCampoNested STORE AS NomeTabellaStorage];
```

con

- NomeCampoNested: nome del campo della tabella di tipo TABLE.
- NomeTabellaStorage: nome tabella utilizzata per la memorizzazione della tabella nested.
- I tipi utilizzati in Dichiarazioni_attributi possono essere:
 - Tipi atomici
 - Tipi definiti dall'utente
 - REF Nome [SCOPE IS Nome_tabella]
La clausola SCOPE IS puo' essere inserita solo negli statement di CREATE TABLE.
La clausola puo' anche essere aggiunta ai campi delle tabelle tipate (SCOPE FOR).

Funzioni OR in Oracle

- ref(): dato un oggetto di un certo tipo, restituisce l'identificatore per quell'oggetto
- deref(): dato un identificatore, restituisce l'oggetto puntato
- value(): prende un alias di relazione e restituisce l'oggetto tupla associato (utilizzando il costruttore opportuno)
- TABLE(): data una valori di tipo collezione, restituisce la tabella corrispondente al contenuto di tale valore.

Trigger SQL-99

```
CREATE TRIGGER Nome  
{BEFORE|AFTER} Evento ON Relazione  
[REFERENCING {OLD AS Var | NEW AS Var | OLD TABLE AS Var | NEW TABLE AS Var}]  
[FOR EACH {ROW | STATEMENT}]  
[WHEN Condizione ]  
Comandi SQL
```

con

- Evento: INSERT, DELETE, UPDATE [OF Lista Attributi]
- Condizione: clausola WHERE SQL