

Esercitazione 01

Si vuole progettare il sistema informativo per la gestione dei musei del Veneto. Una possibile traccia dello schema relazionale è data nella seguente sezione.

Schema base di dati

Le entità di seguito specificate conterranno le informazioni relative ai musei e alle opere ritenute significative.

<pre> Museo(<u>nome</u>: stringa (30 car.), <u>città</u>: stringa (20 car.), indirizzo: ..., numeroTelefono: ..., giornoChiusura: ..., prezzo: ...,) </pre>	<pre> Opera(<u>nome</u>: stringa (30 car.), <u>cognomeAutore</u>: stringa (20 car.), <u>nomeAutore</u>: stringa (20 car.), museo: ..., città: ..., epoca: ..., anno: ...) </pre>
<pre> Mostra(<u>titolo</u>: stringa (30 car.), <u>inizio</u>: DATE, <u>fine</u>: DATE, museo: ..., città: ..., prezzo: ...,) </pre>	<pre> Orario(<u>progressivo</u>: intero, museo: ..., città: ..., giorno: ..., // proporre un dominio orarioApertura: TIME WITH TIME ZONE, orarioChiusura: TIME WITH TIME ZONE) </pre>

- Gli attributi chiave primaria sono sottolineati.
- Gli attributi museo e città nelle entità Opera, Mostra e Orario sono chiave esportata. La chiave esterna è nella tabella Museo.
- Attributi diversi da null: Museo.giornoChiusura, Museo.prezzo, Mostra.fine, Orario.museo, Orario.città, Orario.giorno.
- Valori di default: Museo.città ha come valore di default 'Verona'. Museo.nome ha come valore di default 'MuseoVeronese'. Museo.prezzo ha come valore di default 10. Orario.orarioApertura ha come valore di default '09:00 CET'. Orario.orarioChiusura ha come valore di default '19:00 CET'.

Per l'implementazione delle chiavi esterne si considerino le seguenti politiche di reazione alla cancellazione e alla modifica.

```
REFERENCES reftable [ ( refcolumn ) ]  
    [ ON DELETE referential_action ]  
    [ ON UPDATE referential_action ] (column constraint)
```

```
FOREIGN KEY ( column_name [, ... ] )  
    REFERENCES reftable [ ( refcolumn [, ... ] ) ]  
    [ ON DELETE referential_action ]  
    [ ON UPDATE referential_action ] (table constraint)
```

La clausola **ON DELETE** specifica l'azione da eseguire quando il valore referenziato nella tabella master viene cancellato. In modo simile, la clausola **ON UPDATE** specifica l'azione da eseguire quando il valore referenziato nella tabella master viene aggiornato con un nuovo valore.

Azioni possibili

NO ACTION

(Default) Produce un errore indicante che l'aggiornamento o la cancellazione violano il vincolo di chiave esportata.

CASCADE

Cancella tutte le righe della tabella slave che fanno riferimento alla riga cancellata nella tabella master, o modifica il valore contenuto nella tabella slave con il nuovo valore inserito nella tabella master.

SET NULL [(**column_name** [, ...])]

Imposta a NULL tutti i valori contenuti nella tabella slave che fanno riferimento ai valori aggiornati nella tabella master. → Viene ritornato un errore se tali colonne devono essere diversi da NULL!

SET DEFAULT [(**column_name** [, ...])]

Imposta al loro valore di default tutti i valori contenuti nella tabella slave che fanno riferimento ai valori aggiornati nella tabella master. → viene ritornato un errore se tali colonne non hanno un valore di default e non possono essere NULL!

Esercizio 1

Scrivere il codice PostgreSQL che generi tutte le tabelle. Per gli attributi di cui non è stato specificato il tipo, scegliere quello opportuno. Specificare tutti i vincoli possibili, sia intra- sia inter-relazionali.

Esercizio 2

Inserire nell'entità Museo le seguenti tuple:

(Arena, Verona, piazza Bra, 045 8003204, martedì, 20);

(CastelVecchio, Verona, Corso Castelveccchio, 045 594734, lunedì, 15);

Esercizio 3

Popolare le tabelle Opera e Mostra con almeno altre tre tuple ciascuna.

Esercizio 4

Provare ad inserire nella relazione Museo tuple che violino i vincoli specificati.

Esercizio 5

Nell'entità Museo, aggiungere l'attributo sitoInternet e inserire gli opportuni valori.

Esercizio 6

Nell'entità Mostra modificare l'attributo prezzo in prezzoIntero ed aggiungere l'attributo prezzoRidotto con valore di default 5. Aggiungere il vincolo (di tabella o di attributo?) che garantisca che Mostra.prezzoRidotto sia minore di Mostra.prezzoIntero.

Esercizio 7

Nell'entità Museo aggiornare il prezzo aggiungendo 1 Euro alle tuple esistenti.

Esercizio 8

Nell'entità Mostra aggiornare il prezzoRidotto aumentandolo di 1 Euro per quelle mostre che hanno prezzoIntero inferiore a 15 Euro.

Esercizio 9

Si assume che in ciascuna tabella della base di dati ci siano almeno 3 righe inserite. Implementare le chiavi esportate per ciascuna delle 4 politiche di reazione presentate nella pagina precedente (usare il comando DROP CONSTRAINTS e ADD CONSTRAINTS per effettuare il cambio di politica). Provare ad eseguire una cancellazione ed un aggiornamento dei valori riferiti (e dei valori non riferiti) per verificare il diverso comportamento del DBMS.