

Basi di Dati

Oracle SQLPLUS - Esercitazione n. 3

Questa esercitazione è composta da due parti. La parte I ha come obiettivo quello di creare, dato lo schema logico di una base di dati, appositi script di creazione e popolamento della base di dati, e di scrivere ed eseguire alcuni comandi di aggiornamento e cancellazione utilizzando il linguaggio SQL. La parte II contiene alcune interrogazioni da risolvere usando il linguaggio SQL su una base di dati già esistente.

PARTE I

Passi preliminari per lo svolgimento dell'esercitazione

La finalità di questa prima parte esercitazione è quella di creare, dato lo schema logico di una base di dati, appositi script di creazione e popolamento della base di dati, e di scrivere ed eseguire alcune comandi di aggiornamento e cancellazione utilizzando il linguaggio SQL.

Connessione al server oracle

Le interrogazioni SQL e gli script sono eseguiti tramite un'interfaccia web. L'URL al quale connettersi è il seguente: **<http://cclix4.polito.it:8080/apex>**

Connettersi usando i seguenti dati:

User: bdatiXY

Password: oracXY

Dove XY indica le ultime due cifre del numero del pc utilizzato

Ad esempio user: bdati12, password: orac12 se il numero del pc finisce con il numero 12

Scrittura ed esecuzione degli script

- Gli script sono semplici file di testo e possono essere scritti con un qualsiasi editor (WordPad, Notepad, Vim, Notepad++, Gedit, ecc.)
- Il caricamento e l'esecuzione degli script avviene tramite l'uso dell'interfaccia web di oracle. Selezionare la sezione **Home->SQL->SQL scripts** e cliccare il tasto **Upload (Carica script)** per caricare uno script e poi il tasto **Run (Esegui)** per eseguirlo.

1. Descrizione della base di dati *Palestra*

La base di dati da realizzare è relativa alle attività di una palestra ed è caratterizzata dal seguente schema logico (le chiavi primarie sono sottolineate e le chiavi esterne sono in corsivo):

ISTRUTTORE (CodFisc, Nome, Cognome, DataNascita, Email, Telefono*)

CORSO (CodC, Nome, Tipo, Livello)

PROGRAMMA (CodFisc, Giorno, OraInizio, Durata, *CodC*, Sala)

Per ogni istruttore è noto il codice fiscale, il nome, il cognome, la data di nascita, l'indirizzo e-mail e il numero di telefono. Per ogni corso è noto il codice, il nome, il tipo (es. attività musicale) e il livello (un numero compreso tra 1 e 4). Il programma dei corsi riporta il giorno della settimana (ad esempio lunedì, martedì, ecc.) e l'ora di inizio in cui ogni istruttore svolge una lezione di un determinato corso e la durata in minuti della lezione. Per ogni lezione programmata è noto il numero della sala in cui si svolge.

Tabella ISTRUTTORE

<u>CodFisc</u>	Nome	Cognome	DataNascita	Email	Telefono
SMTPLA80N31B791Z	Paul	Smith	31/12/1980	p.smith@email.it	NULL
KHNJHN81E30C455Y	John	Johnson	30/5/1981	j.johnson@email.it	+2300110303444
AAAGGG83E30C445A	Peter	Johnson	30/5/1981	p.johnson@email.it	+2300110303444

Tabella CORSO

<u>CodC</u>	Nome	Tipo	Livello
CT100	Spinning principianti	Spinning	1
CT101	Ginnastica e musica	Attività musicale	2
CT104	Spinning professionisti	Spinning	4

Tabella PROGRAMMA

<u>CodFisc</u>	<u>Giorno</u>	<u>OrarioInizio</u>	Durata	<i>CodC</i>	Sala
SMTPLA80N31B791Z	Lunedì	10:00	45	CT100	S1
SMTPLA80N31B791Z	Martedì	11:00	45	CT100	S1
SMTPLA80N31B791Z	Martedì	15:00	45	CT100	S2
KHNJHN81E30C455Y	Lunedì	10:00	30	CT101	S2
KHNJHN81E30C455Y	Lunedì	11:30	30	CT104	S2
KHNJHN81E30C455Y	Mercoledì	9:00	60	CT104	S1

Figura 1. Contenuto che deve avere la base di dati dopo l'esecuzione dello script *popolaDB.sql* creato durante l'esercitazione

2. Script

1. Usando un qualunque editor di testo, creare uno script SQL (*creaDB.sql*) con le istruzioni (CREATE TABLE) per la creazione della base di dati corrispondente allo schema logico riportato nella sezione 1 del testo.

In particolare:

- definire tutte e tre le tabelle, scegliendo il tipo ritenuto più opportuno per gli attributi presenti nelle tabelle. Porre particolare attenzione alla definizione della chiave primaria e alla definizione dei vincoli di integrità referenziale
- specificare eventuali vincoli di dominio e/o di tupla ritenuti appropriati. **Nota.** In Oracle i vincoli di dominio si definiscono alla fine della create table usando la sintassi:

CONSTRAINT nome_vincolo CHECK (vincolo)

Esempio. Si consideri un'ipotetica tabella che contiene il campo Voto. Per imporre che il dominio di Voto sia tra 0 e 30 si deve scrivere al fondo della create table della tabella che contiene l'attributo Voto il comando

CONSTRAINT vincolo_voti CHECK (Voto>=0 and Voto<=30)

- scegliere opportunamente le politiche di gestione dei vincoli selezionando quelle più idonee al contesto. **Nota.** Oracle non gestisce l'opzione ON UPDATE CASCADE e ON UPDATE SET NULL (l'unica politica, che è anche quella di default, applicata da Oracle per l'update è ON UPDATE NO ACTION). Quindi non specificare nulla per la parte di gestione dei vincoli per la parte relativi agli UPDATE. Gestire invece la parte relativa alle cancellazioni (DELETE) decidendo se usare la politica ON DELETE CASCADE o ON DELETE NO ACTION o ON DELETE SET NULL.

Nota. Fare attenzione all'ordine con il quale vengono create le tabelle. Creare prima le tabelle referenziate e solo in seguito quelle referenzianti.

2. Creare uno script SQL (*popolaDB.sql*) contenente le istruzioni di inserimento (INSERT) necessarie per il popolamento della base di dati creata al punto precedente. Lo script deve contenere le istruzioni di inserimento necessarie per ottenere un'istanza della base di dati contenente gli stessi dati riportati nelle tabelle presenti in figura 1.

Nota. L'ordine di esecuzione delle INSERT è importante. Usare l'ordine corretto al fine di evitare violazioni dei vincoli di integrità referenziale.

3. Testare gli script di creazione e popolamento scritti ai punti precedenti eseguendoli tramite l'interfaccia web di Oracle (sezione **Home** -> **SQL** -> **SQL scripts**).

Nota. Le tre tabelle potrebbero già esistere nella base di dati nel caso in cui qualche altro studente abbia creato le tabelle durante l'orario di laboratorio precedente al vostro. In tal caso, prima di eseguire i vostri script, eseguite i seguenti comandi per cancellare le tabelle create da altri:

- DROP TABLE PROGRAMMA;

- DROP TABLE CORSO;
 - DROP TABLE ISTRUTTORE;
4. Scrivere in SQL e eseguire i seguenti comandi di aggiornamento, uno per volta, e verificare cosa succede nella base di dati. Per eseguire i comandi di aggiornamento usare l'interfaccia web di Oracle, sezione **Home->SQL-SQL commands**, digitando direttamente il comando che si vuole eseguire e poi premendo il tasto **Run**
- 4.1. Aggiornare il numero di telefono dell'istruttore identificato dal codice fiscale 'KHNJHN81E30C455Y' impostandolo al valore '+390112333551'.
- 4.2. Aggiornare la base di dati in modo tale da spostare nella sala 'S4' tutte le lezioni in programma presso la sala 'S2'.
- 4.3. Eliminare dalla tabella CORSI tutti i corsi che sono in programma una sola volta a settimana (ossia che sono presenti una sola volta nella tabella PROGRAMMA). Che effetto ha l'esecuzione di questo comando sulla tabella CORSO? Che effetto ha sul contenuto della tabella PROGRAMMA? L'effetto dell'operazione sulle due tabelle è legato alla politica di gestione delle violazioni scelta durante la creazione delle tabelle?
- 4.4. Inserire un nuovo corso, con le seguenti caratteristiche, nella tabella CORSO: CodC='CT120', Nome='Spinning estremo', Tipo='Spinning', Livello=10. L'esecuzione del comando genera degli errori? Si può inserire un corso di livello 10? **Nota.** Guardare le specifiche relative al dominio ammissibile per l'attributo livello riportato nella sezione 1 del testo.
- 4.5. Eliminare l'istruttore con codice fiscale pari a 'SMTPLA80N31B791Z'. Che effetto ha questa operazione sulle tabelle ISTRUTTORE e PROGRAMMA? L'esito dell'esecuzione dell'istruzione come è legata alla politica di gestione delle violazioni imposta durante la creazione delle tabelle?