CS252 Fundamentals of Relational Databases — Solutions to Worksheet 2
Getting Started with Oracle

1. Practical question.

2. Create table as in the lecture notes using the CREATE TABLE statement.

   CREATE TABLE Collection
   (artist CHAR(16),
    album CHAR(40),
    tracks INTEGER,
    company CHAR(16),
    year INTEGER);

   Insert values as shown in the notes. It is not possible to rectify mistakes after pressing return using the INSERT INTO statement, so take care. Using a second INSERT INTO statement will add and extra CD to the table Collection.

   INSERT INTO Collection VALUES ('U2', 'The Unforgettable Fire', 10, 'Island', 1984);
   INSERT INTO Collection VALUES ('U2', 'Rattle and Hum', 17, 'Island', 1988);
   INSERT INTO Collection VALUES ('U2', 'Achtung Baby', 12, 'Island', 1991);
   INSERT INTO Collection VALUES ('Underworld', 'Second Toughest in the Infants', 8, 'Junior', 1995);
   INSERT INTO Collection VALUES ('The Verve', 'Urban Hymns', 13, 'Virgin', 1997);
   INSERT INTO Collection VALUES ('Foo Fighters', 'The Colour and the Shape', 13, 'Capital', 1997);

3. To show the whole table, type “SELECT * FROM Collection;”.

   To show just artist, album and year, type “SELECT artist, album, year FROM Collection;”.

4. The following SQL will fail because the tuple is not long enough:

   INSERT INTO Collection VALUES ('Leftfield', 'Leftism', 11);

   Instead, you can replace the missing values with NULL:

   INSERT INTO Collection VALUES ('Leftfield', 'Leftism', 11, NULL, NULL);

   The missing NULL values appear as blank spaces when using “SELECT * FROM Collection;”.

5. Possible selections that satisfy the query:

   (a) “SELECT artist, album FROM Collection WHERE year <= 1991;”.

   The Leftfield album is not listed as its year value is set to NULL. Equals (=) does not test against NULL values.

   (b) “SELECT artist FROM Collection WHERE artist NOT LIKE 'U2';”.

   (c) “SELECT DISTINCT artist FROM Collection WHERE tracks BETWEEN 10 AND 13;”.

6. This is a trick question. You do not know how to add these additional details yet using the INSERT statement. You need the UPDATE statement that is covered in the second seminar.
7. There are two possibilities. The first is to add another column to record the quantity for a particular CD. The second is to add another table. This you can do with:

```sql
CREATE TABLE Quantity ( artist CHAR(16),
   album CHAR(40), number INTEGER);
```

8. INSERT INTO Collection VALUES (NULL,NULL,NULL,NULL,NULL); SELECT * FROM Collection WHERE artist IS NULL;

9. DELETE FROM Collection WHERE artist IS NULL AND album IS NULL AND tracks IS NULL AND company IS NULL AND year IS NULL;

10. (a) SELECT album FROM Collection WHERE album LIKE 'U%' OR album LIKE 'u%';
    (b) SELECT artist FROM Collection WHERE year < 1990 AND year > 1996;
    (c) SELECT album,year FROM Collection WHERE year < 1997 AND album NOT LIKE 'a%'
        AND album NOT LIKE 'A%';
    (d) SELECT artist,album,year FROM Collection WHERE year IN (1984,1991,1997);
    (e) SELECT artist,album,tracks FROM Collection WHERE year < (SELECT year FROM Collection WHERE album = 'Second Toughest in the Infants');

11. (a) SELECT artist,album FROM Collection WHERE album LIKE 'The%' AND year > 1990;
    (b) SELECT artist,album FROM Collection WHERE year > 1990 AND album IN (SELECT album FROM Collection WHERE album LIKE 'The%');
    (c) SELECT artist,album FROM Collection WHERE year > 1990 AND artist IN (SELECT artist FROM Collection WHERE album LIKE 'The%');
    (d) SELECT artist,album FROM Collection WHERE year > 1990 AND artist IN (SELECT artist,album FROM Collection WHERE album LIKE 'The%');

The last of these will give an error since the sub-select must be a single column of values.

12. UPDATE Collection SET company = 'Columbia', year = 1994 WHERE artist = 'Leftfield' AND album = 'Leftism';

1. Some advantages:

   (a) The queries are very simple to write.
   (b) The table is easy to represent on paper.

Some disadvantages:

   (a) Information is duplicated (a waste, possibility of inconsistency).
   (b) A lot of NULL's have to be used.
   (c) Additional information about each CD has to be added as columns of that table.
   (d) Information that does not relate directly to a CD does not make much sense when placed into the same table.
2. The 'CD_Quantity' table would need to have the following columns (barcode,quantity).
   Using the barcode avoids any issues of duplicate album names.

3. CREATE TABLE Band_members (  
   artist VARCHAR(100) PRIMARY KEY,  
   mbr1 VARCHAR(50),  
   mbr2 VARCHAR(50),  
   mbr3 VARCHAR(50),  
   mbr4 VARCHAR(50)  
);  

CREATE TABLE Pop_albums (  
   barcode NUMBER(12) PRIMARY KEY,  
   artist VARCHAR(100) NOT NULL REFERENCES Band_members(artist),  
   album VARCHAR(150) NOT NULL,  
   UNIQUE (artist, album)  
);  

CREATE TABLE Classification (  
   barcode NUMBER(12) PRIMARY KEY,  
   classification VARCHAR(15) NOT NULL  
);  

CREATE TABLE CD_company (  
   barcode NUMBER(12) PRIMARY KEY,  
   company VARCHAR(15) NOT NULL  
);  

CREATE TABLE CD_year (  
   barcode NUMBER(12) PRIMARY KEY,  
   year NUMBER(4)  
);  

CREATE TABLE CD_tracks (  
   barcode NUMBER(12) PRIMARY KEY,  
   tracks NUMBER(2) CHECK (tracks > 0)  
);  

INSERT INTO Band_members VALUES ('U2', 'Bono', 'Edge', 'Clayton', 'Mullen Jr.');  
INSERT INTO Band_members VALUES ('Foo Fighters', 'Grohl', 'Smear', 'Mendel', 'Goldsmith');  
INSERT INTO Band_members VALUES ('Underworld', NULL, NULL, NULL, NULL);  
INSERT INTO Band_members VALUES ('The Verve', 'Ashcroft', NULL, NULL, NULL);
INSERT INTO Pop_albums VALUES ('042282289827', 'U2', 'The Unforgettable Fire');
INSERT INTO Pop_albums VALUES ('042284229920', 'U2', 'Rattle and Hum');
INSERT INTO Pop_albums VALUES ('731451034725', 'U2', 'Achtung Baby');
INSERT INTO Pop_albums VALUES ('026734000524', 'Underworld', 'Second Toughest in the Infants');
INSERT INTO Pop_albums VALUES ('724384491321', 'The Verve', 'Urban Hymns');
INSERT INTO Pop_albums VALUES ('724385583223', 'Foo Fighters', 'The Colour and the Shape');

INSERT INTO Classification VALUES ('042282289827', 'Pop Album');
INSERT INTO Classification VALUES ('042284229920', 'Pop Album');
INSERT INTO Classification VALUES ('731451034725', 'Pop Album');
INSERT INTO Classification VALUES ('026734000524', 'Pop Album');
INSERT INTO Classification VALUES ('724384491321', 'Pop Album');
INSERT INTO Classification VALUES ('724385583223', 'Pop Album');
INSERT INTO Classification VALUES ('724383719020', 'Compilation');
INSERT INTO Classification VALUES ('891030505032', 'Classical');

INSERT INTO CD_Company VALUES ('042282289827', 'Island');
INSERT INTO CD_Company VALUES ('042284229920', 'Island');
INSERT INTO CD_Company VALUES ('731451034725', 'Island');
INSERT INTO CD_Company VALUES ('026734000524', 'Junior');
INSERT INTO CD_Company VALUES ('724384491321', 'Virgin');
INSERT INTO CD_Company VALUES ('724385583223', 'Capital');
INSERT INTO CD_Company VALUES ('724383719020', 'EMI');
INSERT INTO CD_Company VALUES ('891030505032', 'Naxos');

INSERT INTO CD_Year VALUES ('042282289827', 1984);
INSERT INTO CD_Year VALUES ('042284229920', 1988);
INSERT INTO CD_Year VALUES ('731451034725', 1991);
INSERT INTO CD_Year VALUES ('026734000524', 1996);
INSERT INTO CD_Year VALUES ('724384491321', 1997);
INSERT INTO CD_Year VALUES ('724385583223', 1997);
INSERT INTO CD_Year VALUES ('724383719020', 1996);
INSERT INTO CD_Year VALUES ('891030505032', 1992);

INSERT INTO CD_tracks VALUES ('042282289827', 10);
INSERT INTO CD_tracks VALUES ('042284229920', 17);
INSERT INTO CD_tracks VALUES ('731451034725', 12);
INSERT INTO CD_tracks VALUES ('026734000524', 8);
INSERT INTO CD_tracks VALUES ('724384491321', 13);
INSERT INTO CD_tracks VALUES ('724385583223', 13);
INSERT INTO CD_tracks VALUES ('724383719020', 14);
INSERT INTO CD_tracks VALUES ('891030505032', 7);
4. Since the barcode is the primary key of the Pop_albums table then only a single row can exist with that barcode. The other fields in the Pop_albums table do not have any constraints on them and duplicate artist and album names are allowed.

5. (a) SELECT artist,album FROM Pop_albums,CD_Year WHERE
    Pop_albums.barcode = CD_Year.barcode AND
    CD_Year.year < 1991;

(b) SELECT artist,album FROM Pop_albums,CD_Year WHERE
    Pop_albums.barcode = CD_Year.barcode AND
    CD_Year.year < 1997 AND
    Pop_albums.album NOT LIKE '%a%' AND
    Pop_albums.album NOT LIKE '%A%';

(c) SELECT artist,album,tracks FROM Pop_albums,CD_Year,CD_tracks WHERE
    Pop_albums.barcode = CD_Year.barcode AND
    Pop_albums.barcode = CD_tracks.barcode AND
    CD_Year.year <
    (SELECT year FROM Pop_albums,CD_Year WHERE
    Pop_albums.barcode = CD_Year.barcode AND
    Pop_albums.album = 'Second Toughest in the Infants');