

Practice 14 Solutions

1. Create the tables based on the following table instance charts. Choose the appropriate data types and be sure to add integrity constraints.

a. Table name: MEMBER

Column_ Name	MEMBER_ ID	LAST_ NAME	FIRST_NAM E	ADDRESS	CITY	PHONE	JOIN _ DATE
Key Type	PK						
Null/ Unique	NN,U	NN					NN
Default Value							System Date
Data Type	NUMBER	VARCHAR2	VARCHAR2	VARCHAR2	VARCHAR2	VARCHAR2	DATE
Length	10	25	25	100	30	15	

```
CREATE TABLE member
```

```
(member_id      NUMBER(10)
                        CONSTRAINT member_member_id_pk PRIMARY KEY,
last_name       VARCHAR2(25)
                        CONSTRAINT member_last_name_nn NOT NULL,
first_name      VARCHAR2(25),
address         VARCHAR2(100),
city VARCHAR2(30),
phone          VARCHAR2(15),
join_date      DATE DEFAULT SYSDATE
                        CONSTRAINT member_join_date_nn NOT NULL);
```

Practice 14 Solutions (continued)

b. Table name: TITLE

Column_ Name	TITLE_ID	TITLE	DESCRIPTION	RATING	CATEGORY	RELEASE_ DATE
Key Type	PK					
Null/ Unique	NN,U	NN	NN			
Check				G, PG, R, NC17, NR	DRAMA, COMEDY, ACTION, CHILD, SCIFI, DOCUMENTARY	
Data Type	NUMBER	VARCHAR2	VARCHAR2	VARCHAR2	VARCHAR2	DATE
Length	10	60	400	4	20	

```
CREATE TABLE title
(title_id NUMBER(10)
    CONSTRAINT title_title_id_pk PRIMARY KEY,
title          VARCHAR2(60)
    CONSTRAINT title_title_nn NOT NULL,
description    VARCHAR2(400)
    CONSTRAINT title_description_nn NOT NULL,
rating         VARCHAR2(4)
    CONSTRAINT title_rating_ck CHECK
        (rating IN ('G', 'PG', 'R', 'NC17', 'NR')),
category VARCHAR2(20),
    CONSTRAINT title_category_ck CHECK
        (category IN ('DRAMA', 'COMEDY', 'ACTION',
            'CHILD', 'SCIFI', 'DOCUMENTARY')),
release_date  DATE);
```

Practice 14 Solutions (continued)

c. Table name: TITLE_COPY

Column Name	COPY_ID	TITLE_ID	STATUS
Key Type	PK	PK,FK	
Null/Unique	NN,U	NN,U	NN
Check			AVAILABLE, DESTROYED, RENTED, RESERVED
FK Ref Table		TITLE	
FK Ref Column		TITLE_ID	
Data Type	NUMBER	NUMBER	VARCHAR2
Length	10	10	15

```
CREATE TABLE title_copy
(copy_id          NUMBER(10) ,
 title_id         NUMBER(10)

CONSTRAINT title_copy_title_id_fk REFERENCES title(title_id) ,
 status          VARCHAR2(15)
CONSTRAINT title_copy_status_nn NOT NULL
CONSTRAINT title_copy_status_ck CHECK (status IN
('AVAILABLE', 'DESTROYED', 'RENTED', 'RESERVED')),
CONSTRAINT title_copy_copy_id_title_id_pk
PRIMARY KEY (copy_id, title_id));
```

Practice 14 Solutions (continued)

d. Table name: RENTAL

Column Name	BOOK_DATE	MEMBER_ID	COPY_ID	ACT_RET_DATE	EXP_RET_DATE	TITLE_ID
Key Type	PK	PK,FK1	PK,FK2			PK,FK2
Default Value	System Date				System Date + 2 days	
FK Ref Table		MEMBER	TITLE_COPY			TITLE_COPY
FK Ref Column		MEMBER_ID	COPY_ID			TITLE_ID
Data Type	DATE	NUMBER	NUMBER	DATE	DATE	NUMBER
Length		10	10			10

```
CREATE TABLE rental
(book_date      DATE DEFAULT SYSDATE,
 member_id      NUMBER(10)
                CONSTRAINT rental_member_id_fk
                REFERENCES member(member_id),
 copy_id        NUMBER(10),
 act_ret_date   DATE,
 exp_ret_date   DATE DEFAULT SYSDATE + 2,
 title_id       NUMBER(10),
                CONSTRAINT rental_book_date_copy_title_pk
                PRIMARY KEY (book_date, member_id,
                             copy_id,title_id),
                CONSTRAINT rental_copy_id_title_id_fk
                FOREIGN KEY (copy_id, title_id)
                REFERENCES title_copy(copy_id, title_id));
```

Practice 14 Solutions (continued)

e. Table name: RESERVATION

Column Name	RES_DATE	MEMBER_ID	TITLE_ID
Key Type	PK	PK,FK1	PK,FK2
Null/Unique	NN,U	NN,U	NN
FK Ref Table		MEMBER	TITLE
FK Ref Column		MEMBER_ID	TITLE_ID
Data Type	DATE	NUMBER	NUMBER
Length		10	10

```
CREATE TABLE reservation
(res_date      DATE,
 member_id     NUMBER(10)
              CONSTRAINT reservation_member_id
              REFERENCES member(member_id),
 title_id      NUMBER(10)
              CONSTRAINT reservation_title_id
              REFERENCES title(title_id),
 CONSTRAINT reservation_resdate_mem_tit_pk PRIMARY KEY
 (res_date, member_id, title_id));
```

Practice 14 Solutions (continued)

2. Verify that the tables and constraints were created properly by checking the data dictionary.

```
SELECT    table_name
FROM      user_tables
WHERE     table_name IN ('MEMBER', 'TITLE', 'TITLE_COPY',
                        'RENTAL', 'RESERVATION');
```

```
SELECT    constraint_name, constraint_type, table_name
FROM      user_constraints
WHERE     table_name IN ('MEMBER', 'TITLE', 'TITLE_COPY',
                        'RENTAL', 'RESERVATION');
```

3. Create sequences to uniquely identify each row in the MEMBER table and the TITLE table.
 - a. Member number for the MEMBER table: start with 101; do not allow caching of the values. Name the sequence MEMBER_ID_SEQ.

```
CREATE SEQUENCE member_id_seq
START WITH 101
NOCACHE;
```

- b. Title number for the TITLE table: start with 92; no caching. Name the sequence TITLE_ID_SEQ.

```
CREATE SEQUENCE title_id_seq
START WITH 92
NOCACHE;
```

- c. Verify the existence of the sequences in the data dictionary.

```
SELECT    sequence_name, increment_by, last_number
FROM      user_sequences
WHERE     sequence_name IN ('MEMBER_ID_SEQ', 'TITLE_ID_SEQ');
```

Practice 14 Solutions (continued)

4. Add data to the tables. Create a script for each set of data to add.
 - a. Add movie titles to the `TITLE` table. Write a script to enter the movie information. Save the statements in a script named `lab14_4a.sql`. Use the sequences to uniquely identify each title. Enter the release dates in the `DD-MON-YYYY` format. Remember that single quotation marks in a character field must be specially handled. Verify your additions.

```
SET ECHO OFF
INSERT INTO title(title_id, title, description, rating,
                 category, release_date)
VALUES (title_id_seq.NEXTVAL, 'Willie and Christmas Too',
       'All of Willie's friends make a Christmas list for
       Santa, but Willie has yet to add his own wish list.',
       'G', 'CHILD', TO_DATE('05-OCT-1995','DD-MON-YYYY'))
/
INSERT INTO title(title_id, title, description, rating,
                 category, release_date)
VALUES (title_id_seq.NEXTVAL, 'Alien Again', 'Yet another
       installment of science fiction history. Can the
       heroine save the planet from the alien life form?',
       'R', 'SCIFI', TO_DATE('19-MAY-1995','DD-MON-YYYY'))
/
INSERT INTO title(title_id, title, description, rating,
                 category, release_date)
VALUES (title_id_seq.NEXTVAL, 'The Glob', 'A meteor crashes
       near a small American town and unleashes carnivorous
       goo in this classic.', 'NR', 'SCIFI',
       TO_DATE('12-AUG-1995','DD-MON-YYYY'))
/
INSERT INTO title(title_id, title, description, rating,
                 category, release_date)
VALUES (title_id_seq.NEXTVAL, 'My Day Off', 'With a little
       luck and a lot ingenuity, a teenager skips school for
       a day in New York.', 'PG', 'COMEDY',
       TO_DATE('12-JUL-1995','DD-MON-YYYY'))
/
...
COMMIT
/
SET ECHO ON

SELECT title
FROM title;
```

Practice 14 Solutions (continued)

Title	Description	Rating	Category	Release_date
Willie and Christmas Too	All of Willie's friends make a Christmas list for Santa, but Willie has yet to add his own wish list.	G	CHILD	05-OCT-1995
Alien Again	Yet another installation of science fiction history. Can the heroine save the planet from the alien life form?	R	SCIFI	19-MAY-1995
The Glob	A meteor crashes near a small American town and unleashes carnivorous goo in this classic.	NR	SCIFI	12-AUG-1995
My Day Off	With a little luck and a lot of ingenuity, a teenager skips school for a day in New York	PG	COMEDY	12-JUL-1995
Miracles on Ice	A six-year-old has doubts about Santa Claus, but she discovers that miracles really do exist.	PG	DRAMA	12-SEP-1995
Soda Gang	After discovering a cache of drugs, a young couple find themselves pitted against a vicious gang.	NR	ACTION	01-JUN-1995

Practice 14 Solutions (continued)

- b. Add data to the `MEMBER` table. Place the insert statements in a script named `lab14_4b.sql`. Execute commands in the script. Be sure to use the sequence to add the member numbers.

First_Name	Last_Name	Address	City	Phone	Join_Date
Carmen	Velasquez	283 King Street	Seattle	206-899-6666	08-MAR-1990
LaDoris	Ngao	5 Modrany	Bratislava	586-355-8882	08-MAR-1990
Midori	Nagayama	68 Via Centrale	Sao Paolo	254-852-5764	17-JUN-1991
Mark	Quick-to-See	6921 King Way	Lagos	63-559-7777	07-APR-1990
Audry	Ropeburn	86 Chu Street	Hong Kong	41-559-87	18-JAN-1991
Molly	Urguhart	3035 Laurier	Quebec	418-542-9988	18-JAN-1991

```
SET ECHO OFF
SET VERIFY OFF
INSERT INTO member(member_id, first_name, last_name, address,
                    city, phone, join_date)
VALUES (member_id_seq.NEXTVAL, '&first_name', '&last_name',
        '&address', '&city', '&phone', TO_DATE('&join_date',
        'DD-MM-YYYY')) ;
COMMIT;
SET VERIFY ON
SET ECHO ON
```

Practice 14 Solutions (continued)

c. Add the following movie copies in the `TITLE_COPY` table:

Note: Have the `TITLE_ID` numbers available for this exercise.

Title	Copy_Id	Status
Willie and Christmas Too	1	AVAILABLE
Alien Again	1	AVAILABLE
	2	RENTED
The Glob	1	AVAILABLE
My Day Off	1	AVAILABLE
	2	AVAILABLE
	3	RENTED
Miracles on Ice	1	AVAILABLE
Soda Gang	1	AVAILABLE

```
INSERT INTO title_copy(copy_id, title_id, status)
VALUES (1, 92, 'AVAILABLE');
```

```
INSERT INTO title_copy(copy_id, title_id, status)
VALUES (1, 93, 'AVAILABLE');
```

```
INSERT INTO title_copy(copy_id, title_id, status)
VALUES (2, 93, 'RENTED');
```

```
INSERT INTO title_copy(copy_id, title_id, status)
VALUES (1, 94, 'AVAILABLE');
```

```
INSERT INTO title_copy(copy_id, title_id, status)
VALUES (1, 95, 'AVAILABLE');
```

```
INSERT INTO title_copy(copy_id, title_id, status)
VALUES (2, 95, 'AVAILABLE');
```

```
INSERT INTO title_copy(copy_id, title_id, status)
VALUES (3, 95, 'RENTED');
```

```
INSERT INTO title_copy(copy_id, title_id, status)
VALUES (1, 96, 'AVAILABLE');
```

```
INSERT INTO title_copy(copy_id, title_id, status)
VALUES (1, 97, 'AVAILABLE');
```

Practice 14 Solutions (continued)

d. Add the following rentals to the RENTAL table:

Note: Title number may be different depending on sequence number.

Title_ Id	Copy_ Id	Member_ Id	Book_date	Exp_Ret_Date	Act_Ret_Date
92	1	101	3 days ago	1 day ago	2 days ago
93	2	101	1 day ago	1 day from now	
95	3	102	2 days ago	Today	
97	1	106	4 days ago	2 days ago	2 days ago

```
INSERT INTO rental(title_id, copy_id, member_id,  
                  book_date, exp_ret_date, act_ret_date)  
VALUES (92, 1, 101, sysdate-3, sysdate-1, sysdate-2);  
INSERT INTO rental(title_id, copy_id, member_id,  
                  book_date, exp_ret_date, act_ret_date)  
VALUES (93, 2, 101, sysdate-1, sysdate-1, NULL);  
INSERT INTO rental(title_id, copy_id, member_id,  
                  book_date, exp_ret_date, act_ret_date)  
VALUES (95, 3, 102, sysdate-2, sysdate, NULL);  
INSERT INTO rental(title_id, copy_id, member_id,  
                  book_date, exp_ret_date, act_ret_date)  
VALUES (97, 1, 106, sysdate-4, sysdate-2, sysdate-2);  
COMMIT;
```

Practice 14 Solutions (continued)

5. Create a view named `TITLE_AVAIL` to show the movie titles and the availability of each copy and its expected return date if rented. Query all rows from the view. Order the results by title.

```
CREATE VIEW title_avail AS
SELECT  t.title, c.copy_id, c.status, r.exp_ret_date
FROM    title t, title_copy c, rental r
WHERE   t.title_id = c.title_id
AND     c.copy_id = r.copy_id(+)
AND     c.title_id = r.title_id(+);

SELECT  *
FROM    title_avail
ORDER BY title, copy_id;
```

6. Make changes to data in the tables.
- a. Add a new title. The movie is “Interstellar Wars,” which is rated PG and classified as a scifi movie. The release date is 07-JUL-77. The description is “Futuristic interstellar action movie. Can the rebels save the humans from the evil empire?” Be sure to add a title copy record for two copies.

```
INSERT INTO title(title_id, title, description, rating,
                  category, release_date)
VALUES (title_id_seq.NEXTVAL, 'Interstellar Wars',
        'Futuristic interstellar action movie. Can the
        rebels save the humans from the evil Empire?',
        'PG', 'SCIFI', '07-JUL-77');

INSERT INTO title_copy (copy_id, title_id, status)
VALUES (1, 98, 'AVAILABLE');

INSERT INTO title_copy (copy_id, title_id, status)
VALUES (2, 98, 'AVAILABLE');
```

- b. Enter two reservations. One reservation is for Carmen Velasquez, who wants to rent “Interstellar Wars.” The other is for Mark Quick-to-See, who wants to rent “Soda Gang.”

```
INSERT INTO reservation (res_date, member_id, title_id)
VALUES (SYSDATE, 101, 98);
INSERT INTO reservation (res_date, member_id, title_id)
VALUES (SYSDATE, 104, 97);
```

Practice 14 Solutions (continued)

- c. Customer Carmen Velasquez rents the movie “Interstellar Wars,” copy 1. Remove her reservation for the movie. Record the information about the rental. Allow the default value for the expected return date to be used. Verify that the rental was recorded by using the view you created.

```
INSERT INTO rental(title_id, copy_id, member_id)
VALUES (98,1,101);
UPDATE title_copy
SET     status= 'RENTED'
WHERE   title_id = 98
AND     copy_id = 1;
DELETE
FROM    reservation
WHERE   member_id = 101;
SELECT  *
FROM    title_avail
ORDER BY title, copy_id;
```

7. Make a modification to one of the tables.
- a. Add a PRICE column to the TITLE table to record the purchase price of the video. The column should have a total length of eight digits and two decimal places. Verify your modifications.

```
ALTER TABLE title
ADD (price NUMBER(8,2));
DESCRIBE title
```

Practice 14 Solutions (continued)

- b. Create a script named lab14_7b.sql that contains update statements that update each video with a price according to the following list. Run the commands in the script.

Note: Have the TITLE_ID numbers available for this exercise.

Title	Price
Willie and Christmas Too	25
Alien Again	35
The Glob	35
My Day Off	35
Miracles on Ice	30
Soda Gang	35
Interstellar Wars	29

```
SET ECHO OFF
SET VERIFY OFF
DEFINE price=
DEFINE title_id=
UPDATE title
SET price = &price
WHERE title_id = &title_id;
SET VERIFY OFF
SET ECHO OFF
```

- c. Ensure that in the future all titles contain a price value. Verify the constraint.

```
ALTER TABLE title
MODIFY (price CONSTRAINT title_price_nn NOT NULL);
SELECT constraint_name, constraint_type,
       search_condition
FROM   user_constraints
WHERE  table_name = 'TITLE';
```

Practice 14 Solutions (continued)

8. Create a report titled Customer History Report. This report contains each customer's history of renting videos. Be sure to include the customer name, movie rented, dates of the rental, and duration of rentals. Total the number of rentals for all customers for the reporting period. Save the commands that generate the report in a script file named lab14_8.sql.

```
SET ECHO OFF
SET VERIFY OFF
TTITLE 'Customer History Report'
BREAK ON member SKIP 1 ON REPORT
SELECT      m.first_name||' '||m.last_name MEMBER, t.title,
            r.book_date, r.act_ret_date - r.book_date DURATION
FROM        member m, title t, rental r
WHERE       r.member_id = m.member_id
AND         r.title_id = t.title_id
ORDER BY member;

CLEAR BREAK
TTITLE OFF
SET VERIFY ON
SET ECHO ON
```