



**BUAN 6320.001 - Database Foundations for Business  
Analytics - F21**

---

**PROJECT I**

---

**Submitted by**

**Kevin Solomon (kxs200057)**

**Riddhi Solanki (rxs210059)**

**Instructor**

**Prof. Uri Smashnov**

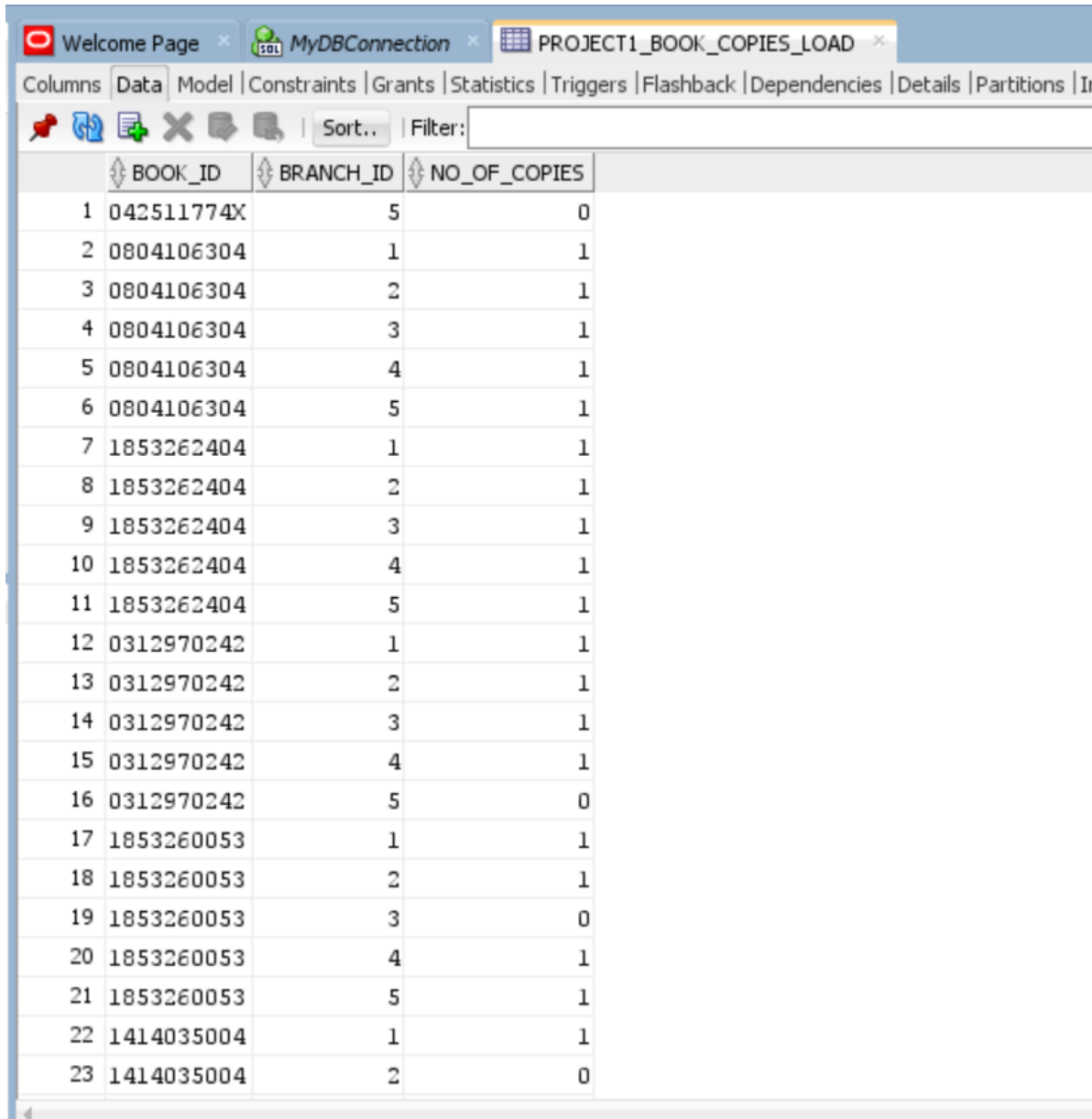
**Date:10/28/2021**

# Design and DB Architecture

## Initial load tables:

Initial load table 1:

```
select * from project1_book_copies_load;
```



The screenshot shows the SQL Developer interface with the 'PROJECT1\_BOOK\_COPIES\_LOAD' table selected. The table has three columns: BOOK\_ID, BRANCH\_ID, and NO\_OF\_COPIES. The data is displayed in a grid with 23 rows. The first column contains an index from 1 to 23. The second column contains book IDs, and the third column contains the number of copies for each book at each branch.

	BOOK_ID	BRANCH_ID	NO_OF_COPIES
1	042511774X	5	0
2	0804106304	1	1
3	0804106304	2	1
4	0804106304	3	1
5	0804106304	4	1
6	0804106304	5	1
7	1853262404	1	1
8	1853262404	2	1
9	1853262404	3	1
10	1853262404	4	1
11	1853262404	5	1
12	0312970242	1	1
13	0312970242	2	1
14	0312970242	3	1
15	0312970242	4	1
16	0312970242	5	0
17	1853260053	1	1
18	1853260053	2	1
19	1853260053	3	0
20	1853260053	4	1
21	1853260053	5	1
22	1414035004	1	1
23	1414035004	2	0

Initial load table 2:

```
select * from project1_books_load;
```

For this load table, the column "authro" is changed to "author" dynamically with the use of SQL Developer UI tool

	ISBN10	ISBN13	TITLE	AUTHOR	COVER
1	042518630X	9780425186305	Purity In Death	J.D. Robb	<a href="http://www.openisbnn.com/cover/042518630X_7">http://www.openisbnn.com/cover/042518630X_7</a>
2	0440223571	9780440223573	This Year It Wi...	Maevie Binchy	<a href="http://www.openisbnn.com/cover/0440223571_7">http://www.openisbnn.com/cover/0440223571_7</a>
3	0812523873	9780812523874	Proxies	Laura J. Mixon	<a href="http://www.openisbnn.com/cover/0812523873_7">http://www.openisbnn.com/cover/0812523873_7</a>
4	0842342702	9780842342704	Left Behind: A ...	Tim LaHaye, Jerry B. Jenkins	<a href="http://www.openisbnn.com/cover/0842342702_7">http://www.openisbnn.com/cover/0842342702_7</a>
5	0440225701	9780440225706	The Street Lawyer	John Grisham	<a href="http://www.openisbnn.com/cover/0440225701_7">http://www.openisbnn.com/cover/0440225701_7</a>
6	0060914068	9780060914066	Love, Medicine ...	Bernie S. Siegel M.D.	<a href="http://www.openisbnn.com/cover/0060914068_7">http://www.openisbnn.com/cover/0060914068_7</a>
7	0156047624	9780156047623	All The King's Men	Robert Penn Warren	<a href="http://www.openisbnn.com/cover/0156047624_7">http://www.openisbnn.com/cover/0156047624_7</a>
8	0245542957	9780245542954	Pacific Northwest	Hans Johannes Hoefer	<a href="http://www.openisbnn.com/cover/0245542957_7">http://www.openisbnn.com/cover/0245542957_7</a>
9	0380715899	9780380715893	Soldier Of The ...	Mark Helprin	<a href="http://www.openisbnn.com/cover/0380715899_7">http://www.openisbnn.com/cover/0380715899_7</a>
10	0553280333	9780553280333	Getting Well Ag... 0.	Carl Simonton M.D., James Creighton Ph.D., Stepha	<a href="http://www.openisbnn.com/cover/0553280333_7">http://www.openisbnn.com/cover/0553280333_7</a>
11	0961769947	9780961769949	Northwest Wines...	Chuck Hill	<a href="http://www.openisbnn.com/cover/0961769947_7">http://www.openisbnn.com/cover/0961769947_7</a>
12	0964778319	9780964778313	An Atmosphere 0...	David Iglehart	<a href="http://www.openisbnn.com/cover/0964778319_7">http://www.openisbnn.com/cover/0964778319_7</a>
13	0671623249	9780671623241	Lonesome Dove	Larry McMurtry	<a href="http://www.openisbnn.com/cover/0671623249_7">http://www.openisbnn.com/cover/0671623249_7</a>
14	0679810307	9780679810308	Shabanu: Daught...	Suzanne Fisher Staples	<a href="http://www.openisbnn.com/cover/0679810307_7">http://www.openisbnn.com/cover/0679810307_7</a>
15	0679865691	9780679865698	Havell	Suzanne Fisher Staples	<a href="http://www.openisbnn.com/cover/0679865691_7">http://www.openisbnn.com/cover/0679865691_7</a>
16	2070423204	9782070423200	Lieux Dits	Michel Tournier	<a href="http://www.openisbnn.com/cover/2070423204_7">http://www.openisbnn.com/cover/2070423204_7</a>
17	0345260317	9780345260314	The Dragons Of ...	Carl Sagan	<a href="http://www.openisbnn.com/cover/0345260317_7">http://www.openisbnn.com/cover/0345260317_7</a>
18	0394743741	9780394743745	The Yawning Hei...	Aleksandr Zinoviev	<a href="http://www.openisbnn.com/cover/0394743741_7">http://www.openisbnn.com/cover/0394743741_7</a>
19	042511774X	9780425117743	Breathing Lessons	Anne Tyler	<a href="http://www.openisbnn.com/cover/042511774X_7">http://www.openisbnn.com/cover/042511774X_7</a>
20	0804106304	9780804106306	The Joy Luck Club	Amy Tan	<a href="http://www.openisbnn.com/cover/0804106304_7">http://www.openisbnn.com/cover/0804106304_7</a>
21	1853262404	9781853262401	Heart Of Darkne...	Joseph Conrad	<a href="http://www.openisbnn.com/cover/1853262404_7">http://www.openisbnn.com/cover/1853262404_7</a>
22	0312970242	9780312970246	Deepak Chopra's...	Deepak Chopra, Martin Greenberg	<a href="http://www.openisbnn.com/cover/0312970242_7">http://www.openisbnn.com/cover/0312970242_7</a>
23	1853260053	9781853260056	Tess Of The D'U...	Thomas Hardy	<a href="http://www.openisbnn.com/cover/1853260053_7">http://www.openisbnn.com/cover/1853260053_7</a>

Initial load table 3:

```
select * from project1_library_branch_load;
```

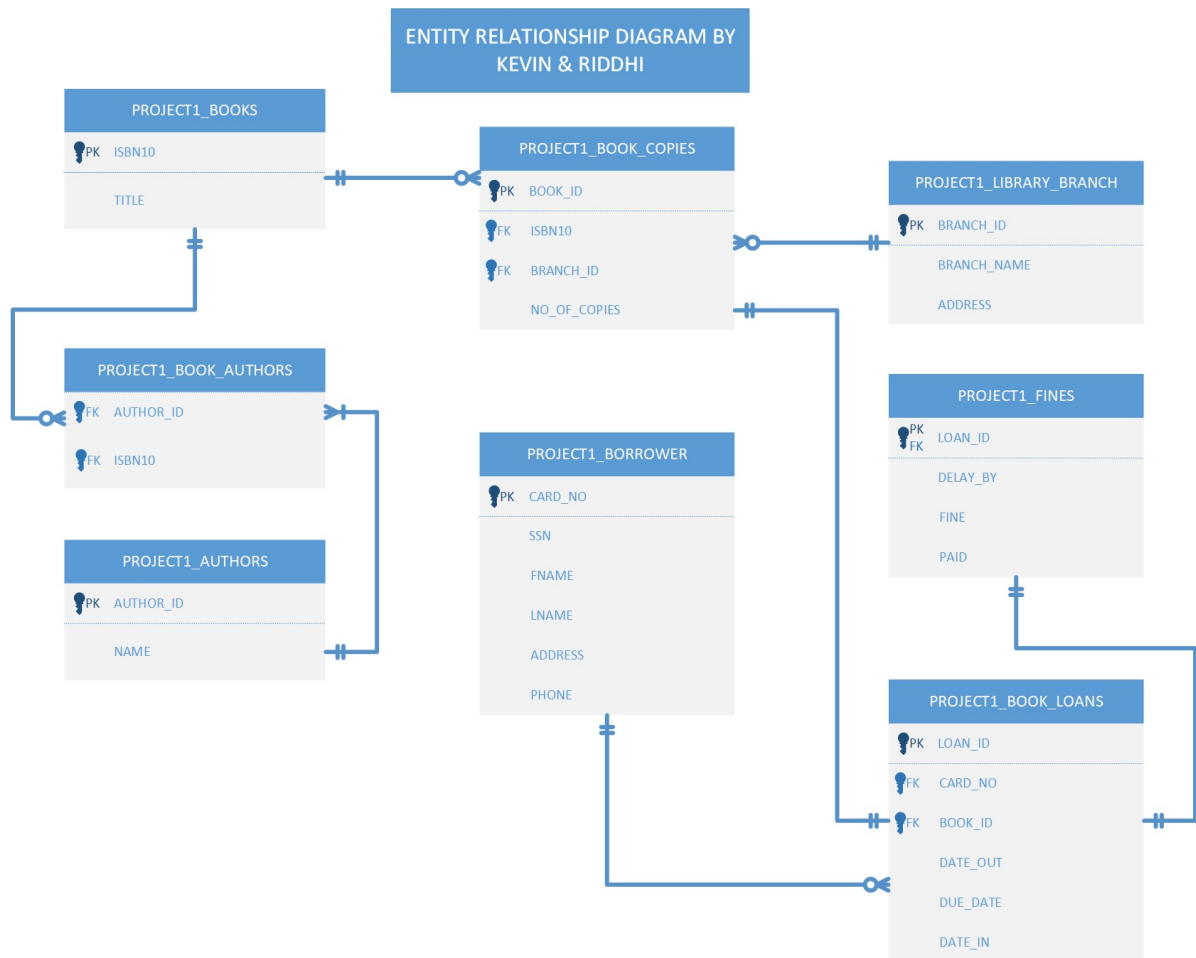
	BRANCH_ID	BRANCH_NAME	ADDRESS
1	1	Oak Lawn	4100 Cedar Springs Road, 75219
2	2	Lakewood	6121 Worth Street, 75214
3	3	Grauwlyer Park	2146 Gilford Street, 75235
4	4	Highland Hills	3624 Simpson Stuart Road, 75241
5	5	Audelia Road	10045 Audelia Road, 75238

Initial load table 4:

```
select * from project1_borrowers_load;
```

ID0000ID	SSN	FIRST_NAME	LAST_NAME	EMAIL	ADDRESS	CITY	STATE	PHONE
1 ID000070	999-23-0361	Annie	Welch	awelchlx@ftc.gov	80 Stuart Road	Dallas	TX	(972) 516-1320
2 ID000071	999-44-1761	Lois	Hunter	lhunterly@vimeo.com	49 Blaine Lane	Plano	TX	(972) 809-1141
3 ID000072	999-79-5907	Lisa	Cook	lcCooklz@netscape.com	4412 Lakewood Gardens Plaza	Richardson	TX	(972) 653-9871
4 ID000073	999-30-8402	Kathleen	Harvey	kharvey20@foxnews.com	13 Grayhawk Hill	Dallas	TX	(469) 547-7753
5 ID000074	999-57-6305	Diane	Stevens	dsteven21@home.pl	0091 American Circle	Dallas	TX	(469) 942-1364
6 ID000075	999-63-2676	Samuel	Young	syoung22@pcworld.com	0157 Holmberg Street	Richardson	TX	(469) 918-5729
7 ID000076	999-25-3782	Mark	Riley	mriley23@theglobeandmail.com	4900 Thompson Court	Dallas	TX	(972) 767-2036
8 ID000077	999-32-5935	Bonnie	Ross	brOSS24@discovery.com	0 Walton Court	Richardson	TX	(972) 766-6908
9 ID000078	999-35-0549	Elizabeth	Arnold	earnold25@bloglovin.com	952 Sage Pass	Dallas	TX	(214) 967-2424
10 ID000079	999-86-6950	John	Walker	jwalker26@fastcompany.com	2 Packers Crossing	Dallas	TX	(469) 796-5094
11 ID000080	999-11-9636	William	Simmons	wsimmons27@vebeden.co.uk	1232 Hayes Crossing	Dallas	TX	(972) 903-7706
12 ID000081	999-38-5276	Lisa	Kelley	lkelley28@storify.com	3718 Twin Pines Way	Dallas	TX	(972) 900-0487
13 ID000082	999-24-2372	Carl	Larson	clarson29@sohu.com	096 Goodland Parkway	Dallas	TX	(469) 550-2228
14 ID000083	999-73-4007	Brandon	Anderson	banderson2a@cargocollective.com	8 Leroy Circle	Dallas	TX	(469) 993-9270
15 ID000084	999-33-4174	Gloria	Myers	gmyers2b@wufoo.com	03 Elmside Center	Plano	TX	(469) 853-3967
16 ID000085	999-25-9640	Daniel	Freeman	dfreeman2c@dell.com	9584 Elka Place	Dallas	TX	(972) 769-9270
17 ID000086	999-98-4647	Judith	Mitchell	jmitche112d@vimeo.com	2506 Merchant Terrace	Dallas	TX	(972) 902-9669
18 ID000087	999-38-5159	Alan	King	aking2e@meetup.com	958 Burrows Street	Dallas	TX	(214) 583-5636
19 ID000088	999-80-3166	Steve	Phillips	sphillips2f@reverbnation.com	45557 Jana Court	Dallas	TX	(972) 937-9620
20 ID000089	999-92-7423	Brenda	Butler	bbutler2g@google.com.br	5 Calypso Avenue	Dallas	TX	(972) 892-6177
21 ID000090	999-37-8900	Patricia	Spencer	pspencer2h@psu.edu	65143 Union Lane	Dallas	TX	(214) 909-9264
22 ID000091	999-64-6193	Larry	Olson	lolson2i@cargocollective.com	50299 Lien Parkway	Dallas	TX	(972) 965-7073
23 ID000092	999-41-6460	Victor	Sullivan	vsullivan2j@yahoo.com	272 Milwaukee Point	Plano	TX	(972) 759-4852

## Entity Relationship Diagram:

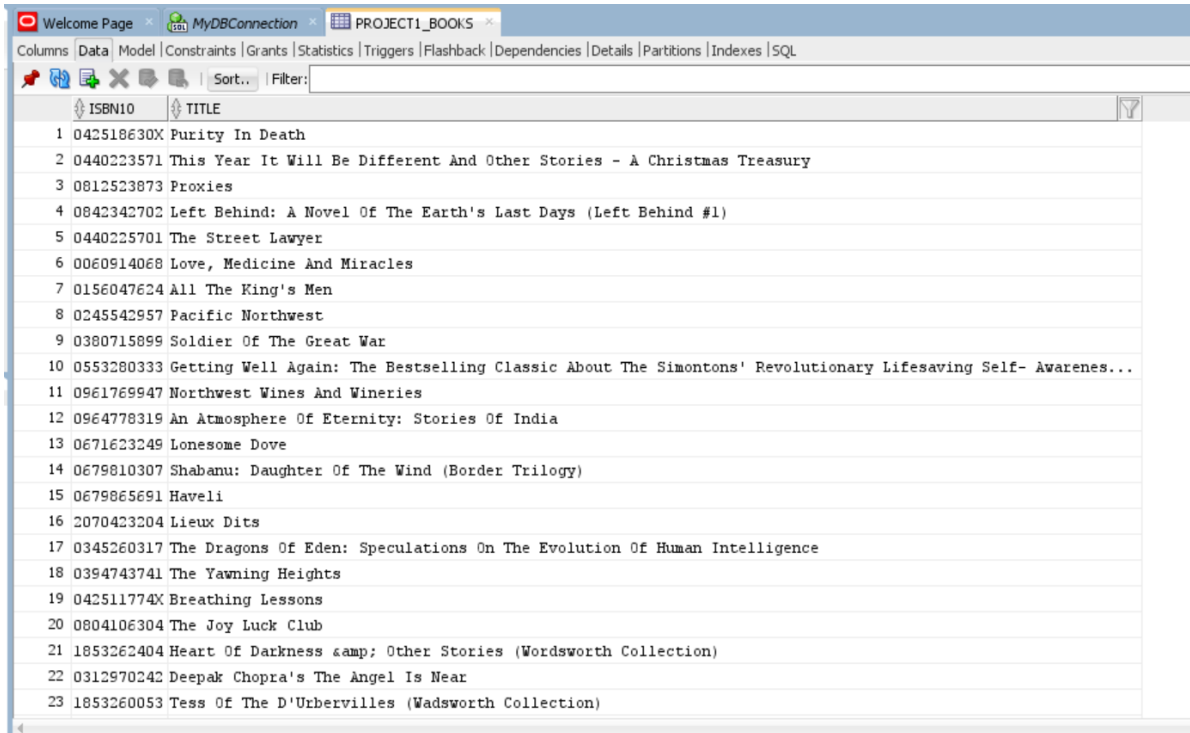


# Database load, Normalization and data generation

## 1- PROJECT1\_BOOKS

```
CREATE TABLE PROJECT1_BOOKS AS (select ISBN10, TITLE from project1_books_load);  
COMMIT;
```

```
ALTER TABLE PROJECT1_BOOKS  
ADD PRIMARY KEY (ISBN10);  
COMMIT;
```



ISBN10	TITLE
042518630X	Purity In Death
0440223571	This Year It Will Be Different And Other Stories - A Christmas Treasury
0812523873	Proxies
0842342702	Left Behind: A Novel Of The Earth's Last Days (Left Behind #1)
0440225701	The Street Lawyer
0060914068	Love, Medicine And Miracles
0156047624	All The King's Men
0245542957	Pacific Northwest
0380715899	Soldier Of The Great War
0553280333	Getting Well Again: The Bestselling Classic About The Simontons' Revolutionary Lifesaving Self- Awareness...
0961769947	Northwest Wines And Wineries
0964778319	An Atmosphere Of Eternity: Stories Of India
0671623249	Lonesome Dove
0679810307	Shabanu: Daughter Of The Wind (Border Trilogy)
0679865691	Haveli
2070423204	Lieux Dits
0345260317	The Dragons Of Eden: Speculations On The Evolution Of Human Intelligence
0394743741	The Yawning Heights
042511774X	Breathing Lessons
0804106304	The Joy Luck Club
1853262404	Heart Of Darkness & Other Stories (Wordsworth Collection)
0312970242	Deepak Chopra's The Angel Is Near
1853260053	Tess Of The D'Urbervilles (Wadsworth Collection)

## 2- PROJECT1\_AUTHORS

separating Author names by comma and assigning an unique id

```
CREATE TABLE PROJECT1_AUTHORS AS  
(  
select row_number() over (order by NAME) as AUTHOR_ID,C.* from (  
select NAME from (  
SELECT m.author  
,REGEXP_SUBSTR (author, '^[,]+', 1, 1) AS NAME  
, REGEXP_SUBSTR (author, '^[,]+', 1, 2) AS part_2  
, REGEXP_SUBSTR (author, '^[,]+', 1, 3) AS part_3  
, REGEXP_SUBSTR (author, '^[,]+', 1, 4) AS part_4  
, REGEXP_SUBSTR (author, '^[,]+', 1, 5) AS part_5  
FROM project1_books_load m where REGEXP_SUBSTR (author, '^[,]+', 1, 1) is not  
null)  
union  
select part_2 from (  
SELECT m.author  
,REGEXP_SUBSTR (author, '^[,]+', 1, 1) AS NAME  
, REGEXP_SUBSTR (author, '^[,]+', 1, 2) AS part_2  
, REGEXP_SUBSTR (author, '^[,]+', 1, 3) AS part_3  
, REGEXP_SUBSTR (author, '^[,]+', 1, 4) AS part_4
```

```

, REGEXP_SUBSTR (author, '[^,]+', 1, 5) AS part_5
FROM project1_books_load m where REGEXP_SUBSTR (author, '[^,]+', 1, 2) is not
null)
union
select part_3 from (
SELECT m.author
, REGEXP_SUBSTR (author, '[^,]+', 1, 1) AS NAME
, REGEXP_SUBSTR (author, '[^,]+', 1, 2) AS part_2
, REGEXP_SUBSTR (author, '[^,]+', 1, 3) AS part_3
, REGEXP_SUBSTR (author, '[^,]+', 1, 4) AS part_4
, REGEXP_SUBSTR (author, '[^,]+', 1, 5) AS part_5
FROM project1_books_load m where REGEXP_SUBSTR (author, '[^,]+', 1, 3) is not
null)
UNION
select part_4 from (
SELECT m.author
, REGEXP_SUBSTR (author, '[^,]+', 1, 1) AS NAME
, REGEXP_SUBSTR (author, '[^,]+', 1, 2) AS part_2
, REGEXP_SUBSTR (author, '[^,]+', 1, 3) AS part_3
, REGEXP_SUBSTR (author, '[^,]+', 1, 4) AS part_4
, REGEXP_SUBSTR (author, '[^,]+', 1, 5) AS part_5
FROM project1_books_load m where REGEXP_SUBSTR (author, '[^,]+', 1, 4) is not
null)
union
select part_5 from (
SELECT m.author
, REGEXP_SUBSTR (author, '[^,]+', 1, 1) AS NAME
, REGEXP_SUBSTR (author, '[^,]+', 1, 2) AS part_2
, REGEXP_SUBSTR (author, '[^,]+', 1, 3) AS part_3
, REGEXP_SUBSTR (author, '[^,]+', 1, 4) AS part_4
, REGEXP_SUBSTR (author, '[^,]+', 1, 5) AS part_5
FROM project1_books_load m where REGEXP_SUBSTR (author, '[^,]+', 1, 5) is not
null
))C);
COMMIT;

```

```

ALTER TABLE PROJECT1_AUTHORS ADD PRIMARY KEY (AUTHOR_ID);
COMMIT;

```

Welcome Page x MyDBConnection x PROJECT1_AUTHORS x		
Columns   Data   Model   Constraints   Grants   Statistics   Triggers   Flashback   Dependencies		
Sort..   Filter:		
	AUTHOR_ID	NAME
1	1	Ail
2	2	Alan (translator) Bergman
3	3	And Tada
4	4	And Vogel
5	5	Baroness Orczy
6	6	Bill; Lee
7	7	D.D.
8	8	Dan Gardiner
9	9	Donald; Bear
10	10	Dr. Hessayon
11	11	Earl Spencer
12	12	Emilie Dailey
13	13	Etal. Lademan
14	14	Etc. Bronte
15	15	Florence Atwater
16	16	Gary Rubin
17	17	Greg; Stirling

### 3- PROJECT1\_BOOK\_AUTHORS

creating the table object by Assigning the correct authors for the right book

```
CREATE TABLE PROJECT1_BOOK_AUTHORS AS (
SELECT DENSE_RANK() over (order by NAME) AS author_id ,C.isbn10 FROM(
select isbn10, NAME from (
SELECT m.isbn10, m.AUTHOR
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 1) AS NAME
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 2) AS part_2
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 3) AS part_3
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 4) AS part_4
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 5) AS part_5
FROM PROJECT1_BOOKS_LOAD m where REGEXP_SUBSTR (author, '[^,]+', 1, 1) is not
null)
union
select isbn10, part_2 from (
SELECT m.isbn10, m.AUTHOR
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 1) AS NAME
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 2) AS part_2
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 3) AS part_3
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 4) AS part_4
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 5) AS part_5
```

```

FROM PROJECT1_BOOKS_LOAD m where REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 2) is not
null)
union
select isbn10, part_3 from (
SELECT m.isbn10, m.AUTHOR
,REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 1) AS NAME
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 2) AS part_2
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 3) AS part_3
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 4) AS part_4
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 5) AS part_5
FROM PROJECT1_BOOKS_LOAD m where REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 3) is not
null)
union
select isbn10, part_4 from (
SELECT m.isbn10, m.AUTHOR
,REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 1) AS NAME
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 2) AS part_2
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 3) AS part_3
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 4) AS part_4
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 5) AS part_5
FROM PROJECT1_BOOKS_LOAD m where REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 4) is not
null)
union
select isbn10, part_5 from (
SELECT m.isbn10, m.AUTHOR
,REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 1) AS NAME
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 2) AS part_2
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 3) AS part_3
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 4) AS part_4
, REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 5) AS part_5
FROM PROJECT1_BOOKS_LOAD m where REGEXP_SUBSTR (AUTHOR, '[^,]+', 1, 5) is not
null) )C);
COMMIT;

ALTER TABLE PROJECT1_BOOK_AUTHORS
ADD CONSTRAINT AUTHOR_ID_FK FOREIGN KEY (AUTHOR_ID) REFERENCES
PROJECT1_AUTHORS(AUTHOR_ID);
COMMIT;
ALTER TABLE PROJECT1_BOOK_AUTHORS
ADD CONSTRAINT isbn10_FK FOREIGN KEY (ISBN10) REFERENCES PROJECT1_BOOKS(ISBN10);
COMMIT;






```



Welcome Page × MyDBConnection × PROJECT1_BOOK_AUTHORS ×		
Columns   Data   Model   Constraints   Grants   Statistics   Triggers   Flashback   Dependencies		
Sort..   Filter:		
	AUTHOR_ID	ISBN10
1	1	0373264011
2	2	0553087258
3	3	0310239729
4	4	0866119531
5	5	0816704589
6	6	0373264011
7	7	0140187642
8	8	1559723475
9	9	0671720791
10	10	0903505193
11	11	0312266499
12	12	0373151896
13	13	1889733008
14	14	1857995457

#### 4- PROJECT1\_LIBRARY\_BRANCH

```
CREATE TABLE PROJECT1_LIBRARY_BRANCH as (select * from
project1_library_branch_load);
COMMIT;
alter table PROJECT1_LIBRARY_BRANCH add constraint branchid_pk primary key
(branch_id);
COMMIT;
```

Welcome Page	MyDBConnection	PROJECT1_LIBRARY_BRANCH
Columns	Data	Model   Constraints   Grants   Statistics   Triggers   Flashback   Dependencies   Details   P
    	Sort..	Filter:

	BRANCH_ID	BRANCH_NAME	ADDRESS
1	1	Oak Lawn	4100 Cedar Springs Road, 75219
2	2	Lakewood	6121 Worth Street, 75214
3	3	Grauwylers Park	2146 Gilford Street, 75235
4	4	Highland Hills	3624 Simpson Stuart Road, 75241
5	5	Audelia Road	10045 Audelia Road, 75238

#### 5- PROJECT1\_BOOK\_COPIES

For this table, we create a temporary table bookload\_tmp

```
create table BOOKLOAD_TMP
(book_id varchar2(10),
 branch_id number,
 no_of_copies NUMBER);

commit;
```

Then loaded the temp tables with the project book copies load data. Placed some logical conditions to divide and populate records based on the count on no\_of\_copies column.

```
DECLARE
    rep_cnt NUMBER;
    CURSOR c_trx
    IS
        SELECT
            book_id,branch_id, no_of_copies
        FROM project1_book_copies_load;
BEGIN
    FOR r_trx IN c_trx
    LOOP

        rep_cnt := r_trx.no_of_copies;
        LOOP
            IF rep_cnt <= 0
            THEN
                EXIT;
            END IF;
            INSERT INTO BOOKLOAD_TMP VALUES
(r_trx.book_ID,r_trx.branch_id,r_trx.no_of_copies);
            rep_cnt := rep_cnt-1;
        END LOOP;

    END LOOP;
COMMIT;
END;
/
```

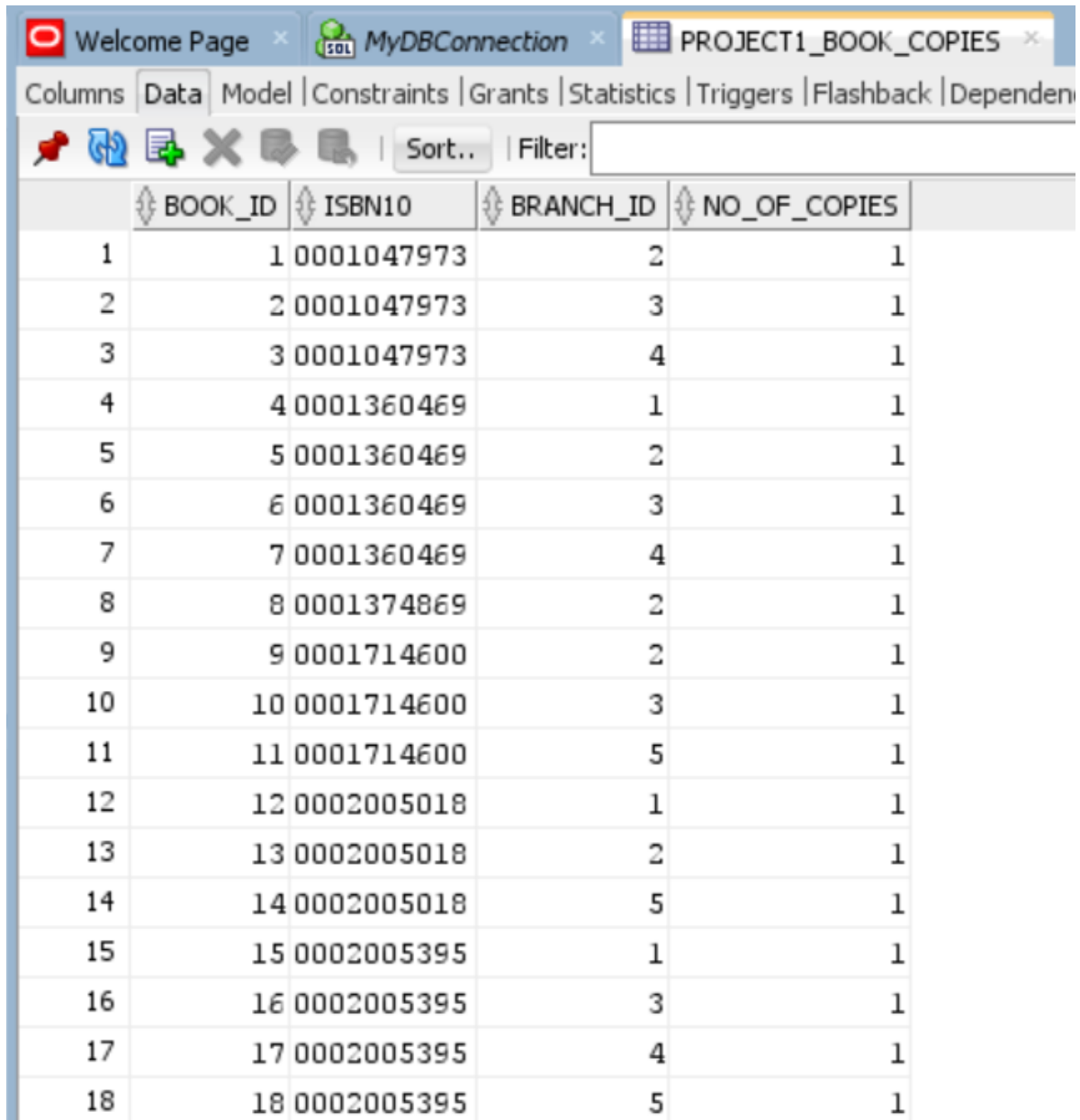
created the main book copies table taken from the bookload\_tmp data and assigned each row with a unique id called book\_id

```
CREATE TABLE PROJECT1_BOOK_COPIES AS (SELECT row_number() over (order by
book_id, branch_id) as book_id,book_id as isbn10, branch_id, no_of_copies FROM
BOOKLOAD_TMP);
COMMIT;
```

```

alter table PROJECT1_BOOK_COPIES add constraint bookid_pk primary key (BOOK_ID);
COMMIT;
ALTER TABLE PROJECT1_BOOK_COPIES
ADD CONSTRAINT isbn10bc_FK FOREIGN KEY (ISBN10) REFERENCES
PROJECT1_BOOKS(ISBN10);
COMMIT;
ALTER TABLE PROJECT1_BOOK_COPIES
ADD CONSTRAINT branch_FK FOREIGN KEY (BRANCH_ID) REFERENCES
project1_library_branch(branch_id);
COMMIT;

```



	BOOK_ID	ISBN10	BRANCH_ID	NO_OF_COPIES
1	1	0001047973	2	1
2	2	0001047973	3	1
3	3	0001047973	4	1
4	4	0001360469	1	1
5	5	0001360469	2	1
6	6	0001360469	3	1
7	7	0001360469	4	1
8	8	0001374869	2	1
9	9	0001714600	2	1
10	10	0001714600	3	1
11	11	0001714600	5	1
12	12	0002005018	1	1
13	13	0002005018	2	1
14	14	0002005018	5	1
15	15	0002005395	1	1
16	16	0002005395	3	1
17	17	0002005395	4	1
18	18	0002005395	5	1

## 6- PROJECT1\_BORROWER

```

CREATE TABLE PROJECT1_BORROWER as (select ID0000ID AS CARD_NO, SSN, FIRST_NAME
AS FNAME,
LAST_NAME AS LNAME, ADDRESS, PHONE from project1_borrowers_load);
COMMIT;
alter table PROJECT1_BORROWER add constraint cardno_pk primary key (CARD_NO);
COMMIT;

```

Welcome Page

MyDBConnection

PROJECT1\_BORROWER

Columns

Data

Model

Constraints

Grants

Statistics

Triggers

Flashback

Dependencies

Details

Partitions

Indexes

SQL

</

## 7- PROJECT1\_BOOK\_LOANS

For this table, we have taken 200 borrowers and 100 different book ids and assigned a random loan id for each entry

```
create table PROJECT1_book_loans as (
(select c.* from (
select ROW_NUMBER() OVER(order by b1.card_no) as loan_id, b1.card_no,bc.BOOK_ID,
SYSDATE - ROUND(DBMS_RANDOM.value(1,60),0) as DATE_OUT,
(SYSDATE - ROUND(DBMS_RANDOM.value(1,60),0)) + ROUND(DBMS_RANDOM.value(1,90),0)
AS DUE_DATE,
((SYSDATE - ROUND(DBMS_RANDOM.value(1,60),0)) +
ROUND(DBMS_RANDOM.value(1,90),0)) + ROUND(DBMS_RANDOM.value(1,120),0) AS DATE_IN
from (SELECT *
FROM (
select *
from PROJECT1_BORROWER
ORDER BY DBMS_RANDOM.RANDOM )
where rownum<200) b1,
(SELECT *
FROM (
select *
from PROJECT1_BOOK_COPIES
ORDER BY DBMS_RANDOM.RANDOM )
where rownum < 100) bc
ORDER BY DBMS_RANDOM.RANDOM DESC
FETCH FIRST 400 ROWS ONLY) c));
COMMIT;
```

then, using sequence corrected the loan\_id

```

create sequence loan_id increment by 1;
COMMIT;

UPDATE PROJECT1_book_loans SET
loan_id = loan_id.nextval;
COMMIT;

ALTER table PROJECT1_book_loans add constraint PROJECTloanid_pk primary key
(loan_id);
COMMIT;
ALTER TABLE PROJECT1_book_loans
ADD CONSTRAINT cardnolloan_FK FOREIGN KEY (card_no) REFERENCES
PROJECT1_BORROWER(CARD_NO);
COMMIT;
ALTER TABLE PROJECT1_book_loans
ADD CONSTRAINT BOOKIDLOAN_FK FOREIGN KEY (BOOK_ID) REFERENCES
PROJECT1_BOOK_COPIES(BOOK_ID);
COMMIT;

```

Welcome Page x MyDBConnection x PROJECT1_BOOK_LOANS x							
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details							
Sort.. Filter:							
	LOAN_ID	CARD_NO	BOOK_ID	DATE_OUT	DUE_DATE	DATE_IN	
1	1	ID000237	49804	15-OCT-21	22-OCT-21	12-DEC-21	
2	2	ID000485	64002	31-AUG-21	21-OCT-21	31-JAN-22	
3	3	ID000126	75573	20-SEP-21	01-JAN-22	10-JAN-22	
4	4	ID000754	62006	11-SEP-21	30-OCT-21	07-NOV-21	
5	5	ID000522	84179	11-OCT-21	10-OCT-21	21-FEB-22	
6	6	ID000740	35225	01-OCT-21	09-NOV-21	08-MAR-22	
7	7	ID000797	72431	05-SEP-21	03-DEC-21	14-FEB-22	
8	8	ID000074	17343	14-SEP-21	17-OCT-21	09-DEC-21	
9	9	ID000690	23154	25-OCT-21	24-NOV-21	08-JAN-22	
10	10	ID000909	72287	29-SEP-21	02-OCT-21	12-DEC-21	
11	11	ID000838	91345	14-OCT-21	17-NOV-21	29-OCT-21	
12	12	ID000311	73107	02-OCT-21	08-OCT-21	01-MAY-22	
13	13	ID000221	51943	31-AUG-21	12-NOV-21	02-FEB-22	
14	14	ID000497	67156	11-OCT-21	11-OCT-21	29-DEC-21	
15	15	ID000337	3565	19-OCT-21	13-DEC-21	18-DEC-21	
16	16	ID000720	12425	24-SEP-21	05-JAN-22	04-JAN-22	
17	17	ID000267	77161	24-OCT-21	03-OCT-21	03-JAN-22	
18	18	ID000647	77594	06-SEP-21	26-NOV-21	04-DEC-21	
19	19	ID000993	37923	04-OCT-21	22-OCT-21	23-NOV-21	

## 8- PROJECT1\_FINES

taking only the loan ids with penalties and payment status.

```

create table project1_fines as(
(SELECT  LOAN_ID,ROUND(DATE_IN-DUE_DATE)AS DELAY_BY,ROUND(DATE_IN-DUE_DATE)*0.50
AS FINE,
(case
    when DBMS_RANDOM.RANDOM > 0
    then 'Y'
    else
        'N'
    End) AS PAID
FROM project1_book_loans
where ROUND(DATE_IN-DUE_DATE)*5 > 1
GROUP BY LOAN_ID,card_no,ROUND(DATE_IN-DUE_DATE) ,ROUND(DATE_IN-DUE_DATE)*0.50 )
FETCH first 50 rows only);
COMMIT;

```

```

ALTER table PROJECT1_FINES add constraint FINEloanid_pk primary key (loan_id);
COMMIT;

```

```

ALTER TABLE PROJECT1_FINES
ADD CONSTRAINT FINESloan_FK FOREIGN KEY (loan_id) REFERENCES
PROJECT1_book_loans(loan_id);
COMMIT;

```

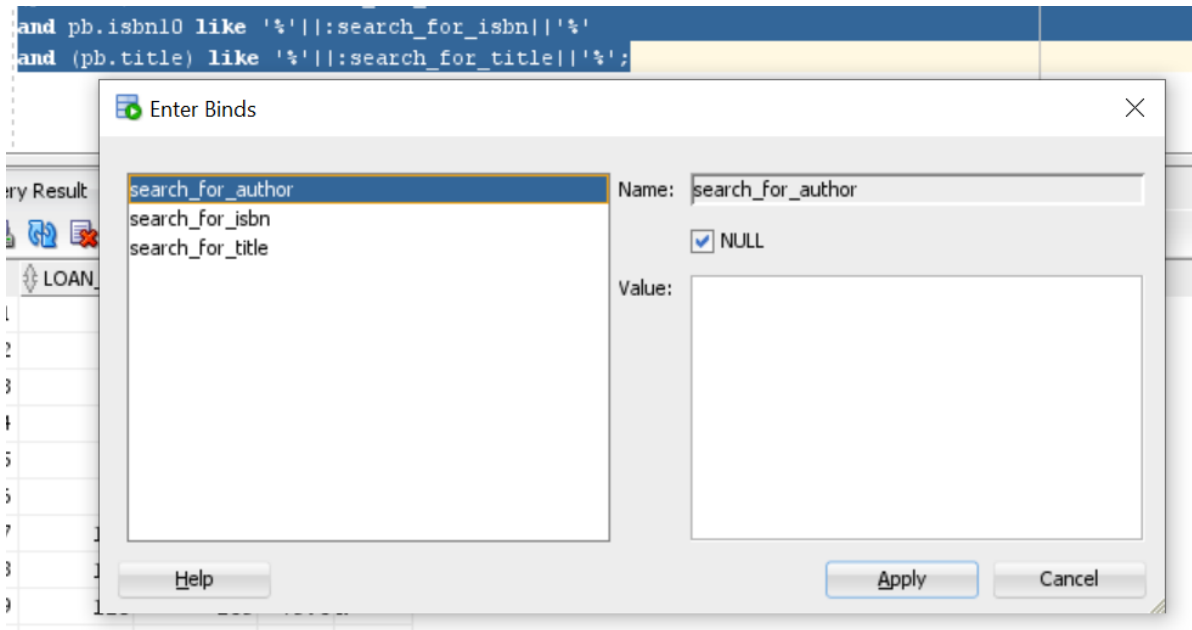
Welcome Page					
MyDBConnection					
PROJECT1_FINES					
Columns   Data   Model   Constraints   Grants   Statistics   Triggers   Flashback					
Sort..   Filter:					
	LOAN_ID	DELAY_BY	FINE	PAID	
1	5	134	67	N	
2	7	73	36.5	Y	
3	18	8	4	N	
4	31	57	28.5	N	
5	40	66	33	Y	
6	48	85	42.5	Y	
7	103	1	0.5	Y	
8	117	173	86.5	Y	
9	120	159	79.5	N	
10	128	118	59	N	
11	134	108	54	N	
12	150	107	53.5	N	
13	166	98	49	Y	
14	175	163	81.5	N	
15	176	56	28	N	
16	178	77	38.5	Y	
17	195	81	40.5	N	
18	202	63	31.5	Y	
19	203	110	55	N	
20	231	98	49	Y	
21	240	65	32.5	N	
22	244	66	36.5	Y	

## Book Search and Availability

```

select pb.isbn10, pb.title, pa.name, plb.branch_name
from project1_books pb
inner join project1_book_authors ba on pb.isbn10=ba.isbn10
inner join project1_authors pa on ba.author_id=pa.author_id
inner join project1_book_copies bc0 on pb.isbn10=bc0.isbn10
inner join project1_library_branch plb on plb.branch_id=bc0.branch_id
where
(pa.name) like '%'||:search_for_author||'% '
and pb.isbn10 like '%'||:search_for_isbn||'% '
and (pb.title) like '%'||:search_for_title||'% '

```



Query Result				
SQL   Fetched 50 rows in 0.102 seconds				
	ISBN10	TITLE	NAME	BRANCH_NAME
1	0002243776	Visible Worlds: A Novel	Marilyn Bowering	Oak Lawn
2	0002243776	Visible Worlds: A Novel	Marilyn Bowering	Lakewood
3	0002243776	Visible Worlds: A Novel	Marilyn Bowering	Grauwlyer Park
4	0006280803	Kundun Pb	Mary Craig	Oak Lawn
5	0006280803	Kundun Pb	Mary Craig	Lakewood
6	0006280803	Kundun Pb	Mary Craig	Highland Hills
7	0006372570	Don't Ask The Price: The Memoirs Of The President Of Marks & Spencer	Marcus Sieff	Oak Lawn
8	0006372570	Don't Ask The Price: The Memoirs Of The President Of Marks & Spencer	Marcus Sieff	Lakewood
9	0006372570	Don't Ask The Price: The Memoirs Of The President Of Marks & Spencer	Marcus Sieff	Grauwlyer Park
10	0006372570	Don't Ask The Price: The Memoirs Of The President Of Marks & Spencer	Marcus Sieff	Highland Hills

## Reports

Report-1: Search functionality for top 10 fines with names which are not paid

```

SELECT * FROM
(SELECT PF.LOAN_ID,PBL.CARD_NO,bor.FNAME,bor.LNAME,PF.PAID,pf.FINE FROM
PROJECT1_FINES PF INNER JOIN PROJECT1_BOOK_LOANS PBL ON PF.LOAN_ID= PBL.LOAN_ID
INNER JOIN PROJECT1_BORROWER bor ON PBL.CARD_NO = bor.CARD_NO
AND pf.paid IN 'N'
ORDER BY pf.fine DESC)
WHERE ROWNUM<= 10;

```



Query Result x						
SQL   All Rows Fetched: 10 in 0.048 seconds						
	LOAN_ID	CARD_NO	FNAME	LNAME	PAID	FINE
1	106	ID000627	Clarence	Morales	N	82
2	175	ID000979	Ryan	Young	N	81.5
3	378	ID000025	Donald	Bowman	N	80
4	120	ID000939	Jimmy	Garrett	N	79.5
5	199	ID000221	Marilyn	Williams	N	76.5
6	5	ID000522	Louise	Garza	N	67
7	61	ID000438	Janice	Garrett	N	63.5
8	128	ID000566	Carol	Ryan	N	59
9	350	ID000720	Helen	Lopez	N	57.5
10	203	ID000108	Steve	Garrett	N	55

Report-2 : Search functionality for top 10 fines with names which are already paid

```

SELECT * FROM
(SELECT PF.LOAN_ID,PBL.CARD_NO,bor.FNAME,bor.LNAME,PF.PAID,pf.FINE FROM
PROJECT1_FINES PF INNER JOIN PROJECT1_BOOK_LOANS PBL ON PF.LOAN_ID= PBL.LOAN_ID
INNER JOIN PROJECT1_BORROWER bor ON PBL.CARD_NO = bor.CARD_NO
AND pf.paid IN 'Y'
ORDER BY pf.fine DESC)
WHERE ROWNUM<= 10;

```

Query Result x						
SQL   All Rows Fetched: 10 in 0.046 seconds						
	LOAN_ID	CARD_NO	FNAME	LNAME	PAID	FINE
1	325	ID000206	Keith	Fisher	Y	97
2	117	ID000830	Samuel	Henry	Y	86.5
3	50	ID000049	Judy	Hamilton	Y	85.5
4	108	ID000496	Christopher	Gonzales	Y	65
5	181	ID000458	Henry	Spencer	Y	54.5
6	55	ID000233	Keith	Perkins	Y	53.5
7	353	ID000824	Scott	Carr	Y	49
8	231	ID000806	Jose	Kelly	Y	49
9	166	ID000779	Clarence	Gray	Y	49

Report-3: Search functionality for top 10 minimum delays with Penalties

```

SELECT * FROM
(SELECT PF.DELAY_BY,PF.LOAN_ID,pf.fine,BL.CARD_NO,BC.ISBN10,B.TITLE FROM
PROJECT1_FINES PF JOIN PROJECT1_BOOK_LOANS BL ON PF.LOAN_ID=BL.LOAN_ID
JOIN PROJECT1_BOOK_COPIES BC ON BL.BOOK_ID = BC.BOOK_ID JOIN PROJECT1_BOOKS B ON
BC.ISBN10=B.ISBN10
ORDER BY DELAY_BY ASC)
WHERE ROWNUM <=10;

```

Query Result x						
All Rows Fetched: 10 in 0.093 seconds						
	DELAY_BY	LOAN_ID	FINE	CARD_NO	ISBN10	TITLE
1	1	103	0.5	ID000978	3466343755	Die Kunst Der Ayurvedischen Massage: Vitalität Und Entspannung Für Körper Und Seele
2	8	4	4	ID000754	0684849690	The Secret Diary Of Anne Boleyn
3	8	18	4	ID000647	0891091165	Celebrate The Seasons
4	16	302	8	ID000839	0515120871	Finding The Dream: The Dream Trilogy #3
5	23	80	11.5	ID000803	0786889608	Middle Of Nowhere
6	25	333	12.5	ID000485	0140092501	Chaos: Making A New Science
7	26	180	13	ID000806	0425070085	Golden Hill
8	33	165	16.5	ID000726	1587670674	The Crossings
9	39	244	19.5	ID000414	1573220922	Freak
10	56	176	28	ID000392	0385413041	Beach Music